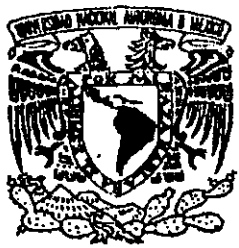
# UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

ESCUELA NACIONAL DE ESTUDIOS PROFESIONALES
ACATLAN



# *"SERVICIOS BANCARIOS EN INTERNET"*

T E S I S
QUE PARA OBTENER EL TITULO DE:
**LICENCIADO EN MATEMÁTICAS APLICADAS A LA COMPUTACIÓN**
P R E S E N T A :
**MIGUEL GILBERTO PÉREZ REYNA**

DIRECTOR DE TESIS:
Ing. Leonardo E. Domínguez Pastrana

MÉXICO, D.F. 1999

# CONTENIDO

# INTRODUCCION

La última década que está precidiendo al tercer milenio, ha sido testigo de cambios exponenciales en lo que a tecnología informática respecta y a su influencia en la vida cotidiana. La década de los 80's estuvo marcada por la aparición de la PC, asumiendo el reto de poner en las manos de personal capacitado en informática las capacidades y facilidades que hasta entonces sólo se habían encontrado en las grandes computadoras. La evolución que estas computadoras personales han sufrido, difícilmente podría imaginarse en sus inicios. Las capacidades que hoy encontramos en una computadora personal sobre cualquier escritorio supera con mucho a los grandes equipos que existían antes de la aparición de las PCs. La comercialización de productos multimedia para este tipo de equipos ha facilitado su penetración no sólo en empresas sino en los hogares, acercando las bondades de la cibernética a gente sin ningún perfil informático. Esta tendencia, aunada al auge que ha tenido la Red Internet está permitiendo que vivamos hoy situaciones que hace dos décadas pertenecían al dominio de la ciencia ficción.

El crecimiento explosivo que ha tenido Internet en el último par de años abre una puerta no sólo a un sinnúmero de oportunidades de intercambio de información a nivel mundial, sino también a una amplia gama de alternativas adicionales, al convertirse en un inmenso escaparate, donde pueden ser ofrecidos todo tipo de servicios. El incremento vertiginoso de usuarios de Internet, ya no sólo en la comunidad universitaria, sino en empresas, corporaciones, organismos públicos y privados, así como en los mismos hogares, representa para muchas empresas la posibilidad, nada despreciable, de llegar a un extenso mercado para ofrecer sus servicios con la comodidad y ventajas que ofrece este medio electrónico aceptando el reto de acercarse a la modernidad en el umbral del nuevo siglo.

El sector financiero no puede permanecer ajeno a esta oleada tecnológica que llegó para quedarse. Muchos bancos están asomándose ya a Internet, aprovechando esta plataforma para promover sus servicios. Sin embargo, los servicios que se empezaron a ofrecer resultaban muy limitados, fundamentalmente por no encontrarse de todo resuelta la seguridad, tema nada trivial en el mundo de Internet. Los adelantos en esta materia se están dando con tal velocidad, que cuando se inició la redacción del presente trabajo, las incursiones de los bancos en Internet se limitaban a publicar páginas estáticas con contenido informativo sobre sus productos y servicios. Al cerrar la edición de esta tésis podemos encontrar entre los sitios mexicanos en Internet, que los principales bancos están ya ofreciendo sus servicios en línea con la comodidad que representa hacer consultas y transacciones desde cualquier computadora conectada a la Red Internet, desde cualquier parte del mundo y a cualquier hora del día.

La explosión demográfica y el crecimiento de las sociedades urbanas ha propiciado un ritmo de vida cada vez más acelerado, las grandes distancias que se tienen que recorrer para llegar a los centros laborales y el tiempo invertido en congestionamientos ocasiona que la gente busque cambiar sus hábitos persiguiendo cada vez más el simplificar algunas de las tareas rutinarias. Esto es lo que ha favorecido la aceptación de medios de entrega de servicios financieros catalogados dentro de lo que se conoce como Banca Electrónica, tales como cajeros permanentes, servicios bancarios por teléfono, kioscos de autoservicio, etc Claro está que estos medios de entrega han tenido mucha mayor aceptación en el segmento de gente joven, mucho más abierto a cambios de estilo de vida y de paradigmas. Pero es precisamente la gente joven la que está impulsando con mayor fuerza a Internet, por lo que no hay por qué dudar que en unos cuantos años más sea quizás el medio de entrega de servicios bancarios más aceptado.

Por otro lado, los costos operativos de las sucursales convencionales están representando un lastre para los bancos haciendo que cada vez les resulte más difícil ser competitivos. Es por esto que los bajísimos costos que representa el ofrecer servicios bancarios a través de un medio de entrega como Internet , estén representando una alternativa muy atractiva para ir, con el tiempo, convenciendo al mercado para adoptar este nuevo medio y paulatinamente ir reduciendo las sucursales en tamaño y número.

La velocidad tan vertiginosa con la que está avanzando todo lo relacionado con Internet en contraste con la gran inversión que los bancos tienen en sistemas y plataformas que soportan los productos y servicios que ofrecen a través de sus medios de entrega tradicionales, obligan a pensar en soluciones pragmáticas, dinámicas y flexibles para ofrecer con toda oportunidad sus productos y servicios a través de Internet. Se debe pensar en arquitecturas tipo puente, que liguen las nuevas tecnologías que circundan a Internet con los sistemas legados que tradicionalmente han ofrecido los productos y servicios financieros, aprovechando de esta manera, toda la inversión en tiempo, dinero y esfuerzo que se ha realizado hasta la fecha, sin incurrir en el costo de oportunidad que representaría el no subirse al tren tecnológico representado por Internet

El propósito del presente trabajo es plantear, después de un recorrido que nos permita entender lo que es Internet, así como los elementos y conceptos circundantes, una arquitectura que ofrezca en armonía, todos los elementos necesarios para poder ofrecer a través de este nuevo medio, los principales servicios a los que hoy se tiene acceso a través de la Banca Electrónica.

# CAPITULO I

# Internet, una puerta a la modernidad

## 1.1 ¿Qué es Internet ?

*Internet* es una colección masiva de redes de computadoras que conectan a millones de computadoras, personas, programas de software, bases de datos y archivos. Todas estas piezas están dispersas por todo el mundo e interactuando constantemente.

*Internet* fue creada en Estados Unidos por *DARPA (Defense Advanced Research Projects Agency)* para asegurar que su sistema de comunicaciones pudiera continuar trabajando en caso de guerra. Al inicio *Internet* fue vista principalmente como una red académica y de investigación. Recientemente, empresas comerciales y un amplio número de consumidores han reconocido el potencial de *Internet*. Hoy en día la gente y los negocios alrededor de todo el mundo usan Internet para obtener información, comunicarse, conducir negocios y accesar una amplia gama de servicios y recursos en línea.

Millones de personas usan *Internet* por sus capacidades de correo electrónico. El correo electrónico sin embargo, es sólo una pequeña parte de lo que *Internet* ofrece. Los usuarios se pueden unir a cualquiera de los miles de grupos de discusión que existen en *Internet*, buscar información específica en enormes bibliotecas electrónicas, o transferir una variedad de archivos a su computadora. También pueden explorar la *World Wide Web*,(también conocida como *Web*), el servicio de multimedia de *Internet*. En los ultimos años los recursos y servicios basados en *Internet* han crecido exponencialmente. En base a proyecciones actuales, este rápido crecimiento puede continuar dentro de la próxima decada tanto como los negocios y consumidores tomen la decisión de moverse dentro de *Internet*.

## *Antecedentes*

*Internet* nació en 1973, surgió por la necesidad de interconectar la red *ARPAnet (Advanced Research Projects Agency)* del Departamento de Defensa Estadounidense con varias redes enlazadas por medio de satélite y radio. *ARPAnet* era una red experimental que apoyaba la investigación militar, en particular la investigación sobre cómo construir redes que pudieran soportar fallas parciales (como las producidas por los bombardeos) y aún así seguir funcionando.

Al mismo tiempo que *Internet* se consolidaba, las redes locales *Ethernet* eran desarrolladas. La tecnología de redes locales maduró hasta 1993, cuando aparecieron las primeras estaciones de trabajo para escritorio y las redes locales se multiplicaron. La mayor parte de las estaciones de trabajo tenía el sistema Unix de Berkeley instalado, que incluía el *software* de red *IP* (*Internet* Protocol).

Esto creó una nueva demanda : en lugar de conectar una computadora de tiempo compartido en un centro de cómputo, las organizaciones requerían conectar toda su red local a *ARPAnet*, lo cual permitiría que todas las computadoras que estuviesen en la red usaran los servicios de *ARPAnet*. Al mismo tiempo, muchas compañías y organizaciones empezaron a construir redes privadas usando los mismos protocolos de comunicación de *ARPAnet*, es decir, *IP* y sus protocolos asociados.

De estas nuevas redes una de las más importantes fue la *NSFNET*, auspiciada por la Fundación Nacional de la Ciencia (*NSF : National Science Foundation*), una agencia del gobierno de E.U. . Al final de los 80´s *NSF* creó cinco centros de supercómputo en universidades importantes. Hasta ese entonces, las computadoras más rápidas del mundo sólo estaban a disposición de los fabricantes de armamento y de algunos investigadores de compañías muy grandes. Con la creación de centros de supercómputo, *NSF* ponía estas fuentes a disposición de cualquier investigador escolar. Sólo se crearon cinco centros porque su costo fue elevado y fue necesario compartirlos. Esto provocó un problema de comunicación : se necesitaba interconectar a los centros y permitir a los usuarios tener acceso a ellos. Al principio, la *NSF* trató de utilizar la red *ARPAnet* para la comunicación de los centros, pero esta estrategia falló debido a problemas burocráticos.

En respuesta a esto, *NSF* decidió construir su propia red basada en la tecnología *IP* de *ARPAnet*. Esta red conectaba a los centros mediante enlaces telefónicos de 56000 *bits* por segundo. Sin embargo, era obvio que si se trataba de conectar cada universidad a los centros de cómputo, el proyecto se podría venir abajo. El costo de la línea telefónica depende de la distancia. Una línea por universidad con un centro de cómputo como eje, al igual que los rayos de una bicicleta, requeriría de muchos kilometros de líneas telefónicas. Por esta razón, se decidió crear redes regionales. En cada región del país las escuelas podían conectarse a su vecino más cercano.

El hecho de compartir supercomputadoras permitió a los centros de cómputo compartir recursos no relacionados con los centros. Repentinamente, las escuelas que participaban en la red contaron con un amplio universo de información y colaboradores al alcance de sus manos. El tráfico en la red se incrementó con el tiempo hasta que las computadoras que las controlaban y las líneas de teléfono conectadas a ellas se saturaron. En 1987 la red fue mejorada con líneas telefónicas de mayor velocidad (por un factor de 20) y con computadoras más poderosas.

### *La Red*

Las diferentes partes de *Internet* están conectadas por un conjunto de computadoras llamadas enrutadores, que interconectan las redes. Estas redes pueden ser *Ethernets*, *Token Rings* o en ocasiones líneas telefónicas, como se muestra en la figura 1.1.

Estas redes son el medio a través del cual la información viaja de un lugar a otro. Cuando se transmiten mensajes a través de *Internet*, estos son separado en pequeñas piezas llamadas paquetes, los cuales viajan a través de diferentes enrutadores entre la computadora origen y la computadora destino.

Los enrutadores deciden cómo dirigir la información("paquetes"). No todo enruteador cuenta con una conexión a cada uno de los otros enrutadores de la red. Esto significa que cada enruteador sólo necesita conecer las conexiones con las que cuenta y elegir el enlace más apropiado para enviar la información a su destino.

El protocolo de comunicaciones usado para enrutar los paquetes a lo largo de la *Internet* es *TCP/IP* (*Transmision Control Protocol/Internet Protocol*), el cual es un conjunto de protocolos de red que pemite el envio y recepción de datos entre máquinas de ambientes heterogeneos sobre diferentes redes de comunicaciones. El uso de este protocolo estandar hace posible que computadoras usando diferentes sistemas operativos (DOS, Windows NT, OS/2, Unix, etc.) se comuniquen unas con otras. Este protocolo será revisado en mayor detalle en la sección 2.4 .

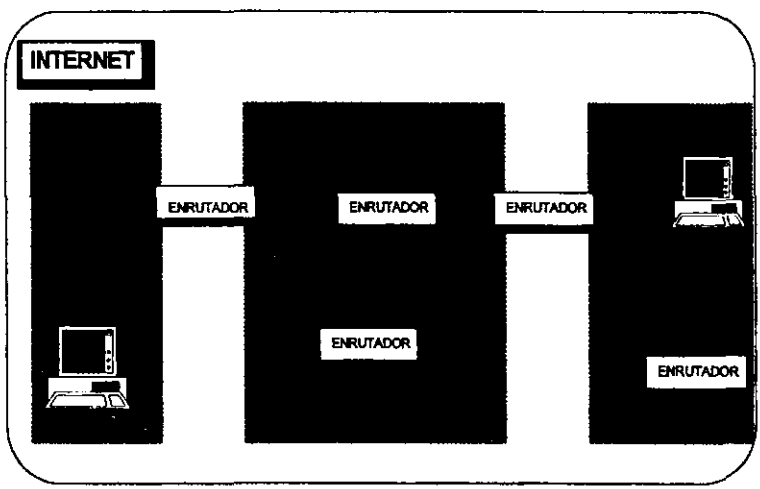



***Fig. 1.1 Hardware de Internet***

## *Acceso a* Internet

Existen dos maneras básicas de conectarse físicamente a *Internet*. La primera, y más ampliamente utilizada, es a través de un modem que conecta una computadora a una línea telefónica analógica normal, que a su vez se conecta a un modem conectado a una computadora central o *host*. Esto es denominado una cuenta de "llamada telefónica", ya que se sigue el mismo proceso que para hacer una llamada telefónica.

Hay varias variaciones para la cuenta de "llamada", que proporcionan diversas facilidades dependiendo del protocolo utilizado. Todas estas conexiones requieren el protocolo *Internet*, por lo que son llamadas cuentas *IP*. Los tres tipos de cuentas *IP* son : *PPP*, protocolo punto a punto ; *SLIP*, protocolo de *Internet* en línea serial y *CSLIP*, versión comprimida de *SLIP*. *PPP* es el tipo de conexión que más se está empezando a utilizar porque es más rápido y más seguro, pero también es más complejo, por lo que todavía hay muchos equipos que sólo tienen soporte interconstruido para *SLIP*.

La otra forma de conexión es a través de *ISDN* (Red Digital de Servicios Integrados), que es un servicio de telecomunicaciones que conecta redes a través de líneas digitales usando un adaptador de terminal. *ISDN* proporciona una conexión mucho más rápida.

Teóricamente, ambos esquemas pueden conectar a *Internet* cuentas de un solo usuario o de múltiples usuarios. Las cuentas de "llamada" son las más recomendadas para conexiones de un solo usuario, pero *ISDN* representa un esquema más económico para conectar varios usuarios que se encuentran en el mismo sitio.

Con el surgimiento de protocolos de *Internet*, modems y navegadores del *Web*, virtualmente cualquier persona puede tener acceso a *Internet*. Pero antes requieren contar con los servicios de lo que pudiera considerarse una compañía utilitaria o *IAP* (*Internet Access Provider*).

## *Identificación de domicilios* Internet

Para que sea posible identificar a cada una de las computadoras dentro de *Internet*, a cada una de ellas se le asigna un domicilio *Internet* único. El domicilio *Internet* es un conjunto de 4 números separarados por un punto, como en este ejemplo : 198.46.8.34.

Los domicilios *Internet* contienen 4 *bytes* de información de ruteo, que consisten de un identificador de red y uno de *host*. El primero es usado para rutear información a través de la red ; el segundo identifica la computadora en particular en la red destino. Un ejemplo sería el siguiente, en donde un punto separa a cada elemento : 192.127.10.1. En este caso, el domicilio de red consiste de 192.127.10 y el domicilio del *host* es 1. La dirección de red es asignada por una autoridad central y es única a través de todo *Internet*.

Al principio, la gente aceptaba que las combinaciones de números como domicilios estaban bien para que las máquinas se comunicaran entre sí, pero las personas prefieren utilizar nombres. Por esto, a las computadoras de *Internet* se les asignaron nombres para la conveniencia de los usuarios. Todas las aplicaciones de *Internet* permiten el uso de nombres en lugar de una combinación de números para definir los domicilios de las computadoras.

Pero el uso de nombres también tiene problemas implícitos, como el hecho de que debe asegurarse que 2 computadoras no tengan el mismo nombre. Para resolver esta situación se buscó tener la forma de convertir los nombres a combinaciones numéricas únicas.

Cuando surgió *Internet* y ésta era una red pequeña, el manejo de los nombres era sencillo. El *NIC*, Centro de Información de la Red (*Network Information Center*), estableció llevar un registro de los nombres. Una persona enviaba una forma, electrónicamente por supuesto, y *NIC* incorporaba la información en una lista de nombres y domicilios. Este archivo, llamado *hosts*, era distribuido en forma periódica a todos los nodos de la red. Los nombres eran simples palabras, no se podían repetir. Si se usaba un nombre, la computadora buscaba en la lista y lo convertía a un domicilio numérico.

Desafortunadamente, cuando *Internet* creció y se multiplicó, lo mismo sucedió con la tabla de nombres. Se requería mucho tiempo para que un nombre quedara registrado y se volvía más difícil encontrar nombres que no hubiesen sido usados. También, se requería de mucho tiempo para poder distribuir el archivo *hosts* a cada máquina de la red. Por lo que fue necesario contar con un sistema distribuido en línea, que convirtiera los nombres en direcciones *Internet* únicas, para satisfacer la rapidez con que cambiaba la información del archivo de nombres. A este sistema se le denominó Sistema de Nomenclatura de Dominios o *DNS* (*Domain Name System*).

*DNS* es una base de datos distribuida. Esto permite el control local de segmentos de la base de datos global, aun así los datos de cada segmento están disponibles para toda la red a través de un esquema Cliente/Servidor[1]. La fortaleza y el adecuado rendimiento de este esquema son logrados a través de la replicación y el manejo de memoria cache (caching).

Los programas llamados *name servers* (servidores de nombres) comprenden la parte servidor del mecanismo cliente/servidor de *DNS*. Los servidores de nombres contienen información acerca de varios segmentos de la base de datos y los hace disponibles a los clientes también llamados *resolvers*.

Los clientes (*resolvers*) son frecuentemente sólo rutinas de librerías que crean las preguntas y las envían a través de la red a un servidor de nombres.

La estructura de la base de datos *DNS* (ver fig. 1.2) es como un árbol invertido, donde la raíz está en la parte superior del árbol. Cada nodo en el árbol representa una partición de la base de datos global o *dominio*. Cada *dominio* puede ser dividido aún en particiones llamadas *subdominios*. Los subdominios son mostrados como hijos de sus nodos padre. Cada dominio tiene un nombre o etiqueta, el cual es relativo a su dominio padre, también tiene un *nombre de dominio* el cual identifica su posición dentro de la base de datos.

En *DNS*, el nombre completo del *dominio* es la secuencia de etiquetas desde el *dominio* hacia la raíz, separando las etiquetas con ".", por ejemplo, en el nombre (*elopez.dgsca.unam.edu.mx*), *elopez* es el nombre del equipo anfitrión, una computadora con un domicilio *IP*. El nombre para esa computadora se asigna y pertenece al subdominio *dgsca*, que es el departamento donde ésta se localiza. El subdominio *dgsca* es parte del dominio *unam.mx*. El subdominio *unam.mx*, a su vez, es parte del dominio de organizaciones educativas (*edu*) de México, *que* es la raíz del domicilio.

[1] El esquema Cliente/Servidor es discutido en el capítulo 2.

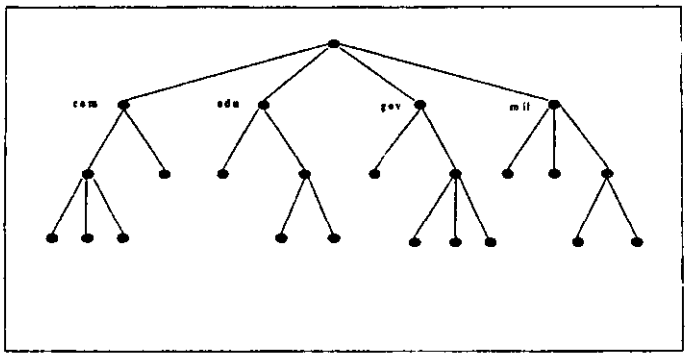



***Fig. 1.2 Estructura de Base de Datos DNS***

### *Administración de Dominios*

Dentro de *DNS*, cada *dominio* puede ser administrado por una organización diferente. Cada organización puede entonces romper su *dominio* dentro de un número de *subdominios* y repartir responsabilidades para estos *subdominios* a otras organizaciones. Por ejemplo, el NIC se responsabiliza del *dominio* **com** (comercial), pero asigna a la empresa Zebra National Bank la autoridad sobre el *subdominio* **zenabank.com**, ver fig. 1.3

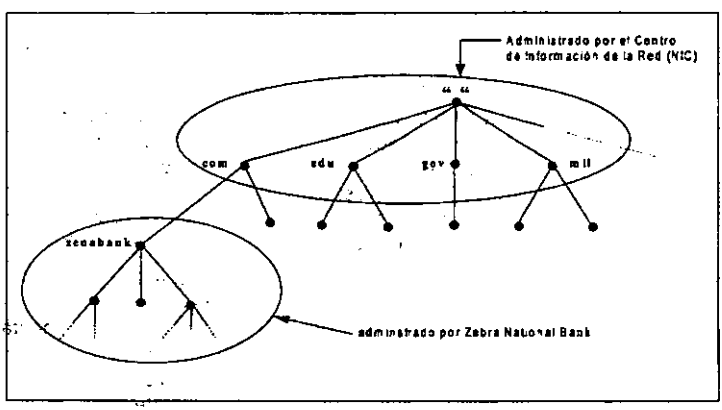



***Fig. 1.3 Delegación de administración de Dominios***

Cada grupo puede crear o cambiar todo lo que esté dentro de él. Si *zenabank* decide crear otro grupo que se llame *corporate*, lo puede hacer sin solicitar ningún permiso. Sólo tiene que agregar el nuevo nombre a su parte de la base de datos mundial y tarde o temprano todo aquel que lo necesite descubrirá el nuevo nombre (*corporate.zenabank.com*). Si cada grupo a partir de *com* respeta las reglas y se asegura de que los nombres que asigne sean únicos, ningun nombre en *Internet* se repetirá. Es posible tener 2 máquinas llamadas *atm*, pero solamente si están en dominios distintos (por ejemplo, *atm.branch.zenabank.com* y *atm.corporate.zenabank.com*).

Cuando *Internet* se convirtió en una red internacional, se requería que los países tomaran la responsabilidad de sus propios nombres . Por esto, existe un conjunto de dominios de dos letras que corresponden a los dominios de jerarquía superior en un país (ej, mx para México, fr para Francia, etc.).

Los dominios de la jerarquía superior fueron creados por concenso cuando se diseñó el Sistema de Nomenclatura de Dominios. Originalmente, existían seis dominios de jerarquía superior (ver tabla 1.1 ).

***Tabla 1.1: Dominios originales de jerarquía superior***

| DOMINIO | UTILIZACION |
|---|---|
| **com** | organizaciones comerciales (negocios) |
| **edu** | organizaciones educativas |
| **gov** | organizaciones gubernamentales sin incluir a la milicia |
| **org** | otras organizaciones |
| **net** | recursos de la red |

Además de los domicilios *Internet* de cada computadora, Cada servidor *Web* o recurso en la *Internet* tiene su propio identificador de localización o *URL (Uniform Resource Locator)*, una dirección *Internet*. Esto ayuda a los usuarios a identificar la fuente de cualquier información dentro de *Internet*. Cuando se conoce el *URL*, se puede tener la clave para accesar en forma directa esos recursos sin tener que navegar a través de directorios o usar busquedas por palabras claves.

La primera parte del *URL*, indica el método de acceso o protocolo usado por el servidor. Por ejemplo, todos los sitios *Web* pueden tener un *URL* que comienza con "http", ya que éste es el protocolo para accesar el *Web*.

La segunda parte del *URL* es el nombre del dominio de la computadora. Por ejemplo, "*http ://www.zenabank.com*" es el localizador para la página Web de los servicios e información en línea del Zebra National Bank . La "www" indica que es una página Web de inicio, "zenabank" es el identificador de la empresa, "com" indica que pertenece al dominio comercial de la *Internet*.

### *Búsquedas en el Sistema de Nomenclatura de Dominios*

Todas las computadoras en *Internet* pueden hacer uso del Sistema de Nomenclatura de Dominios para convertir el nombre de la computadora con la cual requieren comunicarse, en su domicilio numérico. Para poder hacer esto, empiezan a hacer peticiones de ayuda a los servidores *DNS*, empezando por el extremo derecho del domicilo y recorriendolo hacia la izquierda. Primero, se pregunta por el domicilio al servidor *DNS* local. Hasta este punto, hay tres posibilidades:

1. El servidor local conoce el domicilio debidó a que éste se encuentra en la parte local de la base de datos contenida en el servidor. Por ejemplo, si se está usando una computadora en el departamento de crédito de un banco, es probable que el servidor local tenga información acerca de las computadoras del departamento.
2. El servidor local conoce el domicilio solicitado porque alguien más lo solicitó recientemente. Cuando se solicita un domicilio, el servidor *DNS* lo guarda por un tiempo, sólo por si alguién necesita el domicilio más tarde; esto hace al sistema mucho más eficiente.
3. El servidor local no conoce el domicilio, pero sabe cómo buscarlo.

¿Cómo hace la busqueda el servidor local del nombre *atm.branch.zenabank.com*?. Como se muestra en la figura 1.4, el *software* del servidor local sabe cómo comunicarse con el servidor *DNS raíz*. Este es el servidor que conoce los domicilios de los servidores *DNS* que tienen a su cargo los dominios de la jerarquía superior. Un servidor *DNS* de nombres pregunta al servidor *DNS* raíz el domicilio de la computadora responsable de la zona *com*. Con esta información, se comunica con ese servidor y le pide el domicilio del servidor *zenabank*. Después, contacta a esta computadora y le solicita el domicilio del servidor *branch*. Finalmente, se comunica también con esa máquina y obtiene el domicilio de *atm*, que es la máquina con la que originalmente se quería comunicar y de esta manera obtiene el domicilio numérico de la computadora destino.

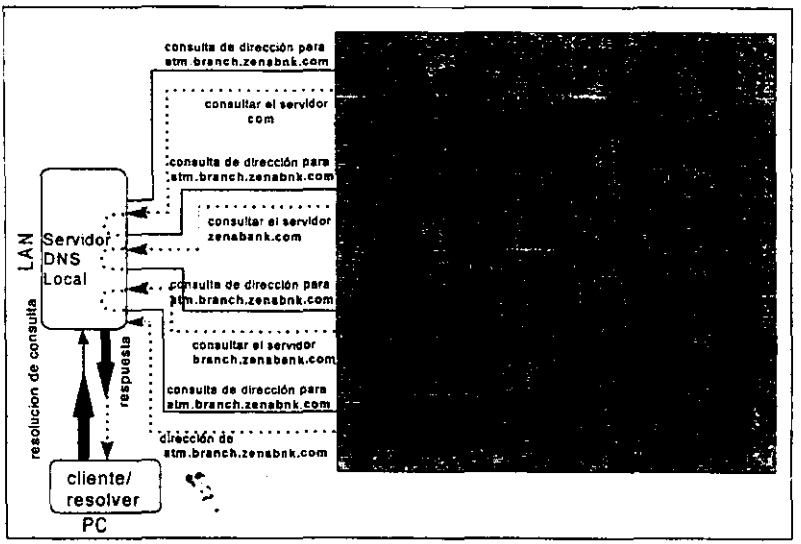



***Fig. 1.4 Resolución de domicilio en Internet***

## *Organización detrás de* Internet

*Internet* no tiene presidente, director ejecutivo o mandatario. Las redes que componen a *Internet* pueden tener presidentes o directores ejecutivos, pero en *Internet*, eso es distinto, no existe la figura de autoridad máxima como un todo. La máxima autoridad sobre la cual descansa *Internet* es la Sociedad *Internet* (ISOC: *Internet Society*). *ISOC* es una organización de membresía voluntaria cuyo propósito es promover el intercambio de información a nivel global mediante el uso de la tecnología de *Internet*. Esta designa a una especie de Consejo de Ancianos cuya responsabilidad consiste en la administración técnica y la dirección de *Internet*.

Quienes integran *ISOC* son un grupo de voluntarios invitados llamado Consejo de Arquitectura de *Internet* (*IAB*: *Internet Architecture Board*). *IAB* se reúne con regularidad para "bendecir" estándares y asignar recursos, como los domicilios. *Internet* funciona porque existen formas estándar para que las computadoras y las aplicaciones de *software* se comuniquen entre sí. Esto permite que las computadoras de diferentes fabricantes puedan comunicarse sin ningún problema.

Los usuarios de *Internet* expresan sus opiniones a través de las reuniones del Grupo de Trabajo de Ingeniería de *Internet* (*IETF*: *Internet Engineering Task Force*). *IETF* es otra organización voluntaria que se reúne con regularidad para discutir problemas operacionales y problemas técnicos a corto plazo. Cuando considera que un problema amerita su atención, *IETF* define un "grupo de trabajo" para realizar una investigación a fondo.

## 1.2 Servicios en Internet

La mayor parte de la gente no se emociona cuando tiene o cuenta con un flujo garantizado de *bits* entre dos máquinas, sin importar lo veloz que éste sea o lo exótico de la tecnología que lo hace posible. Generalmente desean usar ese flujo de *bits* para hacer algo útil, ya sea mover un archivo, obtener información o divertirse. Las aplicaciones son parte del *software* que permite que lo anterior suceda fácilmente. De hecho son otra "capa" del *software*, que actúa por encima de los servicios de *TCP/IP*. Hay tres aplicaciones básicas en *Internet* : Sesión de Trabajo Remota; Transferencia de Archivos y Correo Electrónico.

### *Sesión de Trabajo Remota (Telnet)*

*Telnet* es el protocolo para establecer sesiones de trabajo remotas en *Internet*. Permite estar frente al teclado de una computadora y establecer una sesión en una computadora remota en la red. La sesión puede ser en una máquina en la misma oficina, en la misma universidad o al otro lado del mundo. Permite tener acceso a todos los servicios que esa máquina provee a sus terminales locales.

### *Transferencia de Archivos (FTP)*

*FTP (File Transfer Protocol)* es el protocolo de aplicación que se usa para mover un archivo de una computadora a otra. No importa dónde se localicen estas dos computadoras, cómo estén conectadas o si tienen o no el mismo sistema operativo. Dado que ambas computadoras "hablan" el protocolo *FTP* y tienen acceso a *Internet*, es posible utilizar comandos de ftp para transferir archivos. Algunas de las características de su uso cambian con cada sistema operativo, pero la estructura básica de comandos es la misma en cualquier máquina.

Esto ha provocado la proliferación de una vasta gama de bases de datos y servicios; se puede encontrar desde opiniones legales y recetas, hasta *software* gratuito en una gran cantidad de bases de datos en línea disponibles o repositorios.

*FTP* tiene dos formas comunes de transferir datos : "binaria" y "ASCII". En una transferencia "binaria", se preserva la secuencia del archivo, de tal forma que el original y la copia del archivo sean idénticas *bit* por *bit*, aunque el archivo contenga una secuencia de *bits* que no tenga ningún significado en la máquina receptora.

El concepto de modo "ASCII" realmente no es apropiado: debería llamarse modo de "texto". En el modo "ASCII", las transferencias son tratadas como un conjunto de caracteres, el cliente y el servidor tratan de asegurar que los caracteres que se transfieren tengan el mismo significado tanto en la máquina en que se transfieren como en la máquina receptora.

Las facilidades básicas de ftp hacen muy difícil poner un archivo a disposición de muchos usuarios. Por ejemplo, si se desea distribuir un paquete de *software*, se tendría que poner en algún sistema y después repartir combinaciones de clave de usuario para todo aquel que quisiera el *software*. Esto sería una carga, especialmente para el administrador, pero también para el usuario.

Con *FTP anónimo* es posible evitar esta limitación, lo que permitiría a los usuarios evitar la necesidad de contar con una clave de usuario o una contraseña para poder tener acceso a archivos en una máquina. Obviamente existen algunas restricciones: normalmente los usuarios anónimos sólo pueden copiar los archivos, no pueden instalar archivos nuevos o modificar archivos que ya existen. Y existen límites estrictos sobre qué archivos se pueden copiar.

Cuando se habilita un servidor *FTP anónimo*, se crea una clave de usuario especial llamada *anonymous*. Si se inicializa *FTP* con la clave de usuario *anonymous*, *FTP* aceptará cualquier conjunto de caracteres como contraseña. Después de haber establecido la conexión mediante la clave de usuario *anonymous*, se podrán copiar aquellos archivos que estén permitidos expresamente a los usuarios de los servidores *FTP anónimo* .

### *Correo Electrónico*

El Correo Electrónico (*E-Mail*), envío y recepción de mensajes electrónicos, es actualmente el servicio más popular en *Internet*, es también usado en la mayoría de los servicios en línea, y para mucha gente, es la principal razón para incorporarse a la *Internet*.

El Correo Electrónico permite a los usuarios que se incorporan a *Internet* intercambiar mensajes a través de la red, con usuarios de todo el mundo. Por medio del Correo Electrónico, los usuarios pueden tomar parte en conversaciones electrónicas sobre los temas más diversos, según su interés.

Para enviar un correo electrónico, se debe conocer la dirección de correo del destinatario o receptor. Esta dirección está compuesta de la identificación del destinatario, seguido por el signo "@", seguido por la localización de la computadora del destinatario. Por ejemplo, la dirección de Miguel Pérez en la UNAM sería mperez@unam.edu.mx

El Correo Electrónico difiere de las otras aplicaciones de *Internet* porque no es un servicio de usuario a usuario: no es necesario que las máquinas emisora y receptora del correo electrónico se comuniquen directamente entre sí. Al Correo Electrónico se le conoce como un servicio de "almacenaje y reenvío". El correo pasa de una máquina a otra hasta que llega a su destino final. Esto es análogo a la forma en la que el Servicio Postal entrega el correo.

Justo como en el Servicio Postal, si el emisor y el receptor no están conectados en la misma red, es necesario colocar el mensaje en algún lugar en donde se concentre todo el correo que vaya a una determinada red. Los puntos de conexión entre redes de Correo Electrónico se denominan *gateways*.

Los *gateways* tienen que conocer las aplicaciones de Correo Electrónico de ambos lados para que los mensajes sean reformateados a una forma congruente al pasar de una red a otra. Para enviar correo a través de un *gateway*, casi siempre es necesario proporcionar en el domicilio del mensaje información sobre cómo llegar al *gateway* e información sobre cómo entregar el correo en la otra red.

Antes de poner la carta en el buzón, se coloca en un sobre. Lo mismo pasa en el Correo Electrónico, excepto que al "sobre" se le llama encabezado del mensaje de correo. El encabezado se refiere a los datos que aparecen en la primera parte del mensaje, o sea la dirección *E-Mail* del destinatario (*To*), la dirección *E-Mail* del remitente (*From*) y el tema del mensaje (*Subject*).

Cuando se accesa *Internet* a través de un proveedor local de servicios, se pueden enviar/recibir correos electrónicos de cualquier parte del mundo, sin incurrir en los cargos de una llamada telefónica de larga distancia. Dependiendo del tipo de servicio seleccionado, el Correo Electrónico puede ser una alternativa económica para la comunicación a larga distancia de las personas.

El tiempo que toma entregar un mensaje de Correo Electrónico consta de dos partes: el tiempo que se lleva entregar el mensaje en la computadora destino y el tiempo que toma la lectura del mensaje una vez que se encuentra ahí. El tiempo de la primera parte está en función de cómo está conectada a la red la máquina donde se maneja el correo. El tiempo de la segunda parte está bajo el control del usuario. Si el Correo Electrónico no se revisa con regularidad, la entrega inmediata de los mensajes carece de sentido. Los mensajes están a la espera de ser leídos por el usuario. El Correo Electrónico se hace más útil cuando se reduce el tiempo de entrega de los mensajes entre la máquina y el usuario. Cuando se entrega y se lee el Correo Electrónico rápidamente, puede llegar a ser tan parecido y fluído como una conversación personal.

El Correo Electrónico, a diferencia de la comunicación telefónica, representa un excelente medio para establecer conversaciones entre los miembros de un grupo de usuarios, sin importar el tamaño de éste. Mandar correo electrónico a millones de personas es relativamente sencillo. Esta cualidad del Correo Electrónico lo hace muy útil para diseminar información y pedir opinión a todo un grupo.

Finalmente, la seguridad del Correo Electrónico por lo general es deficiente comparada con los otros medios de comunicación. Si se es cuidadoso con el correo, una carta puede permanecer en un apartado postal de las oficinas de correo, hasta que llegue a manos de su destinatario. Si se abre durante el trayecto, el daño en el sobre hace evidente cualquier intrusión. Para interferir un teléfono se requiere tener acceso a alguno de los dos extremos de la comunicación. Una vez que la comunicación sale de un edificio y entra en la red telefónica, es técnicamente muy difícil interferir una llamada sin la ayuda de la compañía telefónica. Sin embargo, el correo electrónico toma una ruta predecible a través de computadoras cuya seguridad puede ser cuestionable. También existen modalidades de error en las que un mensaje lo entregará al administrador del correo . Normalmente, los administradores no husmean ni divulgan el correo que les llega de esta forma, pero aun así, si la seguridad es crítica, el hecho de que un mensaje llegue a manos de cualquier persona no es apropiado.

### *Network News (Grupos de Interés)*

*Network News* es el equivalente en *Internet* a los grupos de discusión o el tablero de foros de discusión (*BBS*) como los de *Compuserv* o las instituciones que cuentan con un enlace privado vía línea conmutada. Para el usuario, dentro de *Network News* se organizan pláticas bajo un conjunto de amplios apartados que son conocidos como de grupos de interés (*newsgroups*). Un programa despachador de artículos se encarga de presentar tales pláticas de manera ordenada: un menú de pláticas sobre música clásica, seguido por otro sobre una colección de lápices, seguido por otro sobre asuntos de ingeniería química, etc.. Por lo general, dentro de cada grupo de interés se realizan múltiples conversaciones sobre un tema en específico. Todas estas conversaciones se desarrollan simultáneamente. El despachador de artículos ayuda a mantener todo en orden . Mantiene el rastreo sobre los temas que ya se vieron y sólo muestra los temas nuevos que llegaron desde la última sesión. Una vez que el despachador de artículos ha mostrado qué artículos están disponibles sobre determinado tema, se pueden seleccionar y leer los temas que son de interés. Si se olvida dónde se vió algo en particular, se puede buscar el artículo según el autor, el tema o una sinópsis por autor. También se puede configurar el despachador de artículos para ver o descartar ciertos asuntos automáticamente, basándose en el nombre del autor o en el tema del artículo.

*Networks News* (USENET) usa el protocolo *NNTP* (*Network News Transfer Protocol*), el cual es un protocolo estandar en *Internet* para la distribución, consulta, recuperación y publicación de articulos. Para accesar y participar en los grupos de interés se requiere de un programa especial, la mayoría de los navegadores comerciales tienen la capacidad para accesar a los grupos de interés. Para subscribirse y comunicarse a los grupos de interés que se desee, se utiliza un sistema de mensaje similar al del Correo Electrónico.

Se puede participar en forma pasiva, simplemente viendo un prolongado diálogo, o activamente, mandando artículos y convertirse en parte del foro. Como con el Correo Electrónico, *NetNews* es usualmente una comunicación informal entre individuos, sin embargo varios grupos de interés son manejados por un monitor, el cual puede elegir no publicar/anunciar respuestas que son consideradas inapropiadas para el foro. *Usenet* opera a una gran velocidad, con artículos apareciendo rapida y constantemente. Existe un administrador del grupo que determina el tiempo que los articulos deben permanecer antes de ser borrados del sistema.

Los grupos de discusión pueden ser una excelente fuente de información y de asistencia en asuntos técnicos, hobbies, viajes, etc.

### *Archie (Cómo encontrar archivos)*

Históricamente, uno de los grandes problemas de *Internet* ha sido encontrar lo que se sabe que existe. Los servidores *FTP anónimo*, surgieron rápidamente para brindar la oportunidad de buscar archivos en los repositorios de la Red; pero durante mucho tiempo la existencia de esos archivos era conocida sólo por contacto interpersonal de unos usuarios con otros a través de la Red. Parte del aprendizaje para convertirse en un gurú de red consistía en conocer las suficientes personas y acudir a las conferencias necesarias para saber dónde estaba la información. Esto funcionaba bien cuando *Internet* era una red de dimensiones pequeñas utilizada únicamente por profesionales de la computación. Ahora que *Internet* proporciona recursos para el público en general, el mecanismo de "los viejos tiempos" ya no funciona. La mayoría de los usuarios nuevos no tiene un administrador diestro con los contactos adecuados. Y actualmente existen tantos recursos en línea que ni siquiera el mejor de los administradores podría rastrearlos todos.

*Archie* es un sistema que permite explorar índices disponibles en los servidores públicos especiales. Aquí es donde debe empezarse si se están buscando programas, datos o archivos de texto. Como usuario, se le puede solicitar que encuentre nombres de archivos que correspondan a ciertos criterios de búsqueda o que se muestre archivos que contengan ciertas palabras. *Archie* devuelve los nombres de archivo que concuerdan con el criterio de búsqueda y el nombre de los servidores que contienen esos archivos. Una vez que se decida cuál de esos archivos satisface mejor las necesidades, puede ser trasladado a una computadora con *FTP anónimo*.

Como muchos servicios de *Internet*, se puede tener acceso a *Archie* de muchas maneras. La más sencilla de ellas es a través de *TELNET*, usando un *software* cliente de acceso público. El otro extremo consiste en enviar solicitudes de búsqueda a *Archie* a través del Correo Electrónico. El acceso por Correo Electrónico resulta útil cuando se está conectado con alguna otra red que sólo proporcione acceso a *Internet* a través del Correo Electrónico.

El servicio *Archie* fue construido por ciertas personas de la Universidad McGill, quienes se dedicaron a buscar por toda la red, preguntando a quienes estuvieran ejecutando programas servidores *FTP anónimo* para catalogarlos. Se ejecuta un programa una vez al mes, el cual se conecta con los servidores mencionados a través de ftp (ver figura 1.5 ). Cuando se enlaza con los servidores, construye un directorio para listar todos los archivos que se encuentran en cada uno de ellos, usando comandos ftp estandar.

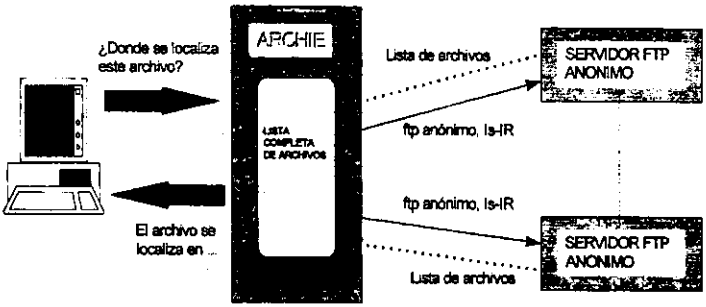



***Fig. 1.5 Funcionamiento de Archie***

Este es el servicio básico como fue creado, sin embargo, se volvió obvio que algunas personas eligieron nombres extraños, nada intuitivos, para sus archivos. Por ello, los creadores de *Archie* pidieron a las personas responsables de cada servidor que les enviaran información sobre los paquetes más importantes que manejaran, y utilizaron esta información para crear un servicio llamado *whatis*. Se trata de un conjunto de palabras clave alternativas y catalogadas para archivos de la Red, que pueden ser utilizadas para localizar *software* o archivos de datos incluso si el nombre del archivo no guarda ninguna semejanza con su contenido. Como este servicio requiere de la intervención humana, tiene altibajos, pero de cualquier manera es muy útil.

Cuando aumentó el uso de *Archie*, el servicio se transformó para satisfacer la creciente demanda. En la actualidad, existen muchos servidores *Archie* repartidos por *Internet*. Cada servidor construye un índice de archivos *FTP* cerca de él y después los diversos servidores comparten información. Esto permite que las actualizaciones sean más oportunas, sin sobrecargar la Red. En la mayoría de los casos, sin embargo, el usuario no tiene que preocuparse por cómo funciona este sistema. La mayor parte de las tareas que el usuario realiza no se ven; todo lo que necesita hacer es conectarse a cualquier servidor *Archie* (todos contienen los mismos datos) y buscar la información que necesita.

## *Gopher*

*Gopher* es otra herramienta ampliamente usada en *Internet*, es un programa que permite buscar recursos utilizando menús. Cuando se encuentra algo que resulta de interés, se puede leer o tener acceso a ello a través de *Gopher* sin necesidad de preocuparse por los nombres de dominio, los domicilios *IP* o por cambiar de programas.

Por ejemplo, si se quiere tener acceso al catálogo en línea de una universidad, en lugar de buscar el domicilio y ejecutar el comando telnet, simplemente se encuentra una opción en un menú *Gopher* y se selecciona. Entonces *Gopher* va por la información.

La gran ventaja que ofrece *Gopher* no es tanto que ahorre la búsqueda de domicilios o nombres de recursos o que no se tenga necesidad de utilizar varios comandos para obtener lo que se desea. La ventaja real consiste en que permite curiosear a través de los recursos de *Internet*, sin importar su tipo, tal como si se hojeara el catálogo de una biblioteca que contiene libros e imágenes, todo agrupado en un solo volumen.

Para imaginarse el uso de *Gopher*, hay que pensar en una biblioteca Se tiene que pensar en *Gopher* de *Internet* como un conjunto de bibliotecas sin catálogos ni bibliotecarios. Para encontrar algo se tendría que vagar sin rumbo fijo hasta encontrar algo interesante. Este tipo de bibliotecas no es muy útil, a menos de que se sepa con detalle lo que se quiere encontrar y su localización exacta. Un servidor *Gopher* sería como contratar a un bibliotecario, encargado de crear los catálogos temáticos del material existente, si no se encuentra lo que se busca en una biblioteca, se puede desplazar electrónicamente a la siguiente y buscar ahí.

*Gopher* no permite tener acceso a cualquier cosa que no esté disponible por otros medios. No existe un formato especial de "recursos Gopher" para que se pueda tener acceso, por lo menos no lo hay en el mismo sentido que en los servidores *FTP* o los directorios de *White pages*. Pero una vez que se encuentre algo que se quiera "revisar", *Gopher* también ayudará. *Gopher* sabe cuál aplicación (*telnet, ftp*, etc) se debe utilizar para tener un elemento particular en el cual se esté interesado, para así ejecutar el trabajo.

Para tener acceso al sistema *Gopher* se requiere un programa cliente gopher. El cual debe estar instalado en una computadora que se encuentre en *Internet*. Existe *software* gratuito de clientes gopher para casi cualquier tipo de computadora que existe. Se puede conseguir cualquier *software* que se necesite en el servidor *FTP* anónimo boombox.micro.umn.edu, dentro del directorio */pub/gopher*.

Cuando se pone en funcionamiento por primera vez un cliente gopher, éste se enlaza con su servidor base y le solicita el menú principal. El servidor envía dicho menú y cierta información oculta al cliente. La información oculta indica a este último lo que representa cada elemento del menú (por ejemplo, un archivo de texto, un directorio, etc), también le indica el domicilio *Internet* del servidor para ese elemento y una ruta de acceso para un archivo. El domicilio puede ser el servidor base mismo, si es ahí donde reside el recurso; pero también puede ser cualquier otro servidor ubicado en un lugar diferente.

### *El World Wide Web*

El *World Wide Web*, también conocido como *WWW*, es un sistema de información abierto diseñado específicamente pensando en su facilidad de uso y de intercambio de documentos. Es la parte multimedia de *Internet*; es el servicio de información de *Internet* más reciente y de mayor crecimiento. El contenido de *WWW* se despliega como una página y , a diferencia de otros sitios de *Internet*, el texto es formateado con varias fuentes, estilos, colores y tamaños. Las páginas también pueden contener dibujos, sonidos y videos.

Lal *Web* es conceptualmente simple (ver fig. 1.6). Es es una manera profundamente efectiva de publicar electrónicamente información. Para el usuario, la aplicación navegadora de la *Web* obtiene la información hipertexto y la despliega en una página a la vez. Estas páginas, construidas en *HTML*, son entregadas en una forma más elegante que el simple texto. Las páginas pueden contener inmersas gráficas y formateos primitivos seleccionados. También contienen ligas, usualmente expresadas como palabras subrayadas o con íconos, la selección de una liga invoca a una página *HTML* en cualquier lado de *Internet*, permitiendo al usuario saltar de una fuente de información a otra con sólo apretar el botón del *mouse*. Varias ligas apuntan no a páginas *HTML* sino a archivos accesados vía *FTP*. El seleccionar una de estas ligas puede automáticamente copiar un archivo a la máquina del usuario y, dependiendo del navegador que tenga instalado, puede invocar automáticamente otra aplicación que no sea el navegador.

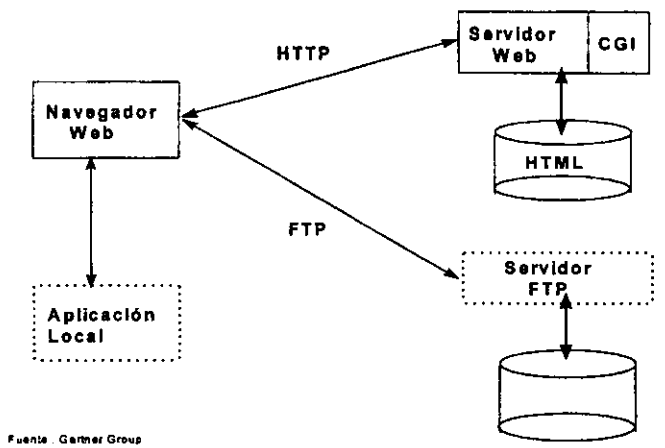



***Fig. 1.6 Vista simplificada del Web***

Detrás de los servidores *Web*, existe una interface llamada *CGI* a través de la cual los programadores pueden construir *scripts* para invocar otros procesos dentro del mismo servidor, pasar la información a éstos y recibir resultados para regresarlos. La comunicación entre la parte cliente del navegador *Web* y el servidor *Web* se realiza con el protocolo *HTTP*.

En este diseño básico ampliamente implementado hoy en día ( y accesado por 24 millones de personas en NorteAmerica), el *Web* consiste de una colección semiestática de páginas con ligas *hard-code* (codificadas) hacia otras páginas o archivos.

En 1989, Tim Bernes-Lee del Laboratorio Europeo para Partículas Físicas (*CERN*)[2], propuso el Web como una manera para que los científicos alrededor del mundo colaboraran usando un sistema de información global basado en hipertexto.

En 1990, el primer navegador de tipo sólo-texto fue implementado y los científicos del *CERN* pudieron accesar archivos de tipo hipertexto y otra información en el *CERN*. Sin embargo, la estructura de los documentos hipertexto y la manera en que éstos pudieran ser transferidos a sitios remotos todavía tenía que ser definida. En base a las propuestas de Tim Bernes-Lee, la estructura de los documentos hipertexto fue definida en un nuevo lenguaje llamado *HyperText Markup Language (HTML)*. *HTML* fue basado en un subconjunto del lenguaje *Standard Generalized Markup Language (SGML)* que estaba siendo ampliamente usado en ese momento. Para transferir documentos *HTML* a sitios remotos, se ideó un nuevo protocolo, el protocolo de transferencia *hipertexto HTTP (HyperText Transfer Protocol)*.

*HTTP* ofrece un medio para moverse de un documento a otro y posicionarse dentro de documentos. El poder del hipertexto está en su simplicidad y transparencia. Los usuarios pueden navegar a través de una red global de recursos con el toque de un boton. Los documentos hipertexto son ligados por medio de palabras claves o de áreas específicas dentro del documento. Estas áreas pueden ser íconos gráficos o partes de mapas indexados. Cuando una nueva palabra o idea es introducida, el hipertexto hace posible brincar a otro documento que contiene la información completa del nuevo tópico. Los lectores ven las ligas como palabras claves resaltadas o como imágenes desplegadas gráficamente.

En 1993 en el Centro Nacional para Aplicaciones de Supercómputo (*NCSA*, por sus siglas en inglés), fue desarrollado un navegador que permitió a los usuarios explotar las capacidades gráficas del *Web*. *NCSA* llamó a este navegador Mosaic. Por un tiempo Mosaic y *Web* parecieron ser sinónimos.

Usando las facilidades de hipertexto del *Web*, se tiene la libertad de proporcionar información a los lectores en poderosos modos innovativos. Las empresas que sostuvieron el crecimiento del *Web* comenzaron creando pequeñas publicaciones que hicieron poco uso de las capacidades gráficas y de multimedia del *Web*. Esto cambió dramáticalmente en pocos años y, hoy en día, las publicaciones del *Web* utilizan muchas de sus características gráficas, interactivas y de multimedia.

[2] (William Staneck,HTML, JAVA, CGI, VRML, SGML Web Publishing, Edit. Sams Net,1996)

Mientras algunos publicadores del *Web* están buscando en los orígenes de la publicación en *Web*, otros están dando saltos gigantescos hacia adelante. Estos saltos son en parte debido a innovadores tales como *Netscape Communications Corporation, Microsoft Corporation, y Sun Microsystems Incorporated.*

En 1994, *Netscape Corporation* liberó el primer navegador para soportar sólo extensiones a *HTML*. El *Netscape Navigator* se convirtió rapidamente en el navegador más popular de la Red.

Actualmente, el navegador que está compitiendo con *Netscape Navigator* es *Internet Explorer de Microsoft.* Las características adicionales que *Internet Explorer* introdujo permitieron agregar sonido y video a las publicaciones del *Web.*

*Sun Microsystems* ha liberó el navegador *HotJava,* el cual está escrito completamente en el lenguaje de programación *Java* desarrollado por *Sun*. Este lenguaje es similar a C y C++, pero es único en el sentido de ser independiente de la plataforma. Usando *Java*, se pueden agregar programas llamados "*applets*" a las publicaciones del *Web*. Los "*applets*" son aplicaciones que los lectores pueden ver previamente y ejecutarlas automáticamente.

### *Hipertexto*

Hipertexto es un método para presentar información donde las palabras seleccionadas en el texto pueden ser "expandidas" en cualquier momento para proporcionar otra información sobre la palabra. Esto significa que estas palabras forman "vínculos" con otros documentos, que pueden ser texto, archivos, imágenes o cualquier otra cosa. Por ejemplo, suponiendo que en una biblioteca se encuentra un catálogo en línea de fichas en hipertexto, si se extrae la ficha de un libro en particular, posiblemente aparecerá de la siguiente manera:

| | |
|---|---|
| TITULO: | El Error de la Luna |
| AUTOR: | *Héctor Aguilar Camin* |
| EDITORIAL: | Alfaguara |
| FECHA: | 1995 |
| TEMA: | *Novela Contemporánea* |

Si las palabras en itálicas son vínculos, se puede expandir el nombre del autor y obtener una semblanza biográfica. Si se expande "Novela Contemporánea" se obtendrá un resúmen de los principales autores y títulos de la Novela Contemporánea.

Como se trata de otro documento en hipertexto, también en éste existen vínculos, permitiendo avanzar más todavía expandiendo cada una de las palabras que aparezcan en itálicas. Este proceso se puede repetir tantas veces como se desee, ahondando cada vez más en un tema específico.

La cantidad de hipertexto que se encuentra en la Red, se ha incrementado considerablemente en los últimos años. Muchas exhibiciones de museo, revistas y otras presentaciones en hipertexto están disponibles. El problema radica en la escasez de herramientas para construir la estructura de vínculos. La mayoría de los documentos en hipertexto actualmente disponibles se construyó manualmente. Los editores de h pertexto son muy recientes; conforme pase el tiempo, aparecerán más documentos en hipertexto y mejores herramientas para crearlos. Los archivos de hipertexto son almacenados en el formato *HTML*.

### *SGML*

El lenguaje de marcado generalizado estandar (*SGML* por sus siglas en inglés) fue diseñado para resolver la compatibilidad para el intercambio de documentos estrictamente formateados entre diferentes plataformas de computadoras. Hasta antes del *SGML*, eran muy limitadas las opciones para el intercambio electrónico de documentos en un formato consistentemente utilizado. Se podría reducir el formateo de un documento a su forma más básica usando el formato estandar de *ASCII* , o bien se podría intentar convertir un formato propietario a otro formato propietario si se contara con un convertidor. Generalmente el documento convertido no se parecía al que el creador había generado originalmente.

Para resolver este problema, se desarrolló un formato para documentos independiente de la plataforma. Esto significa que las computadoras pueden intercambiar documentos *SGML* independientemente del tipo de computadora que reciba el documento.

SGML es muy poderoso, sin embargo tiene unas desventajas. Si se crea un *DTD* (*Document Type Definitions*) para los documentos publicados en el *Web*, el *DTD* debe ser transferido con el documento. Esto representa mayor tiempo en la transferencia de los archivos.

A pesar de ser tan poderoso *SGML* no es tan ampliamente utilizado en la Red debido a la complejidad que implica para desarrollar documentos. Una de las metas de los autores de *HTML* fue reducir drásticamente la complejidad de las soluciones actuales para publicar documentos, por lo que sigue siendo el esquema más utilizado.

### *CGI*

*CGI* (*Common Gateway Interface* ) es un estandar que permite que el servidor *Web*, por medio de *scripts*, pueda interactuar con otras aplicaciones.

A través de *scripts* de *CGI*, se pueden construir publicaciones para Web que resulten ser muy poderosas y personalizadas, con las que realmente se pueda interactuar. Los *scripts* de *CGI* son programas externos que actuan como un *gateway* entre el servidor Web y otras aplicaciones. Los *scripts* de *CGI* se pueden utilizar para procesar datos proporcionados por los lectores de la publicación, estableciendo así un canal bidireccional con ellos. El tipo de entrada puede ser datos capturados a través de una forma electrónica ; palabras clave para hacer consultas a una base de datos o simplemente datos que describan al navegador y la conexión.

Los scripts de *CGI* pueden usar los datos de entrada para agregar elementos a un índice, hacer búsquedas en una base de datos, crear documentos personalizados, entre muchas otras aplicaciones. Una de las características más relevantes de los *scripts* CGI es que ocultan su complejidad al usuario. El concepto de *CGI* ha venido a establecer una alternativa para la integración de información estática en los *Webs*, con información proporcionada por los sistemas legados.

*CGI* permite que cualquier aplicación, sin importar la plataforma, pase información a un *script* CGI, al especificar una forma común de accesar los *scripts*. Por lo que CGI hace que se reduzca el complejo proceso de interactuar con procesos externos, a unos cuantos procedimientos

Al definir la liga entre el *script*, el servidor y otras aplicaciones, CGI hace posible que los programas externos acepten entradas generalizadas y que pasen información a otras aplicaciones.

### *Lenguaje Java*

La primer iniciativa que surgió al mercado pensando en romper la barrera del contenido estático en el *Web* fue *Java*. Liberado al mercado en mayo de 1995 por Sun Microsystems, *Java* es un lenguaje orientado a objetos, que permite que las aplicaciones escritas con este lenguaje se puedan ejecutar en muchas plataformas distintas. Estas aplicaciones escritas en *Java* (las llamadas *Java "applets"*) se compilan bajo un formato intermedio independiente de la arquitectura de cómputo (código a nivel de *bytes*), que posteriormente es leído y ejecutado por intérpretes en una máquina virtual ejecutándose en Windows 95, NT, Macintosh y varias plataformas Unix. A fin de reducir aún más la dependencia con respecto al *hardware* y facilitar el desarrollo para varias plataformas, los desarrolladores de *Java* especificaron los tipos básicos de datos y las funciones de los operadores aritméticos de tal forma que pudieran operar en múltiples plataformas.

También le agregaron algunas características con el fin de hacer este lenguaje particularmente útil para construir sistemas en *Internet*, entre las que se incluyen el soporte integrado para la Red (*HTTP,FTP*, etc) y también medidas interconstruídas en materia de seguridad, de tal forma que la popularidad que ha tenido *Java* se debe no tanto al lenguaje en sí, sino más bien a lo que se puede hacer con él : las "*applets*" en *Java* pueden insertarse dentro del código *HTML* de una página en el *Web* ; por otro lado, dichas "*applets*" se pueden descargar en forma dinámica del servidor del *Web* empleando navegadores que soporten este lenguaje, para posteriormente ser ejecutadas en el "cliente" y dado que *Java* posee la capacidad de procesamiento concurrente de subprocesos (*multithreading*), es posible iniciar la ejecución de varias de ellas simultáneamente.

Es un hecho que entre más diferenciadas y atractivas se elaboren las páginas en el *Web*, más resultados positivos obtendrán las compañías en la promoción y venta de sus productos y servicios ; pues bien, fué precisamente la capacidad que tiene *Java* para hacer que las páginas del *Web* cobren más vida lo que atrajo en gran parte la atención hacia este lenguaje desde su aparición.

Muchas organizaciones están empezando a explorar el uso del *Web* como una forma económica y estandarizada para explotar internamente ciertos tipos de aplicaciones; actualmente se emplean una variedad de interfaces propias para los datos y aplicaciones, mientras que la tecnología de los programas navegadores ofrece un mecanismo estandar para la consulta de los datos y el acceso a otros servicios.

Así varios proveedores están creando productos con varios frentes que permitan a los navegadores interacturar con los sistemas ya existentes y compartir datos con ellos.

Por otra parte, si bien el principal uso de *Java* ha sido para convertir las páginas estáticas del *Web* en otras más dinámicas e interactivas, las "*applets*" pueden llevar a cabo prácticamente cualquier función que se puede programar en C++. Es obvio, pues, el potencial de tales "*applets*", al ser capaces de actuar como interfaces no sólo de sistemas administradores de bases de datos (*DBMSs*), sino también con los servidores de aplicaciones y con el "*middleware*" (como es el caso de los "*object request brokers*", los mecanismos para las invocaciones a procedimientos remotos o *RPCs* y el "*middleware*" orientado al manejo de mensajes, por ejemplo).

Si *Java* tiene el éxito de posicionarse,, como el ambiente operativo estandar para *Internet* y se solucionan otros aspectos técnicos (como por ejemplo la seguridad y la amplitud de banda) y de negocio (como el otorgamiento de licencias), entonces *Java* (y el concepto que implica) podría revolucionar el mundo del *software* y *hardware*.

Desde el punto de vista del *software* es posible que el actual esquema de tener que instalar copias de las aplicaciones en cada PC - con todas las implicaciones que con lleva actualizar sus versiones - sería reemplazado por un nuevo paradigma en el que las "*applets*" se bajarían desde la red en el momento de requerirlas.

Los grandes proveedores de la talla de Oracle, Sun, IBM y otros más, dentro de muy poco tiempo estarán ofreciendo sistemas de *hardware* mucho más baratos que las PCs (las llamadas "terminales para *Internet*"), diseñados específicamente para ejecutar los programas navegadores. Por medio de estos dispositivos sería posible bajar automáticamente de la red un sistema operativo orientado a *Java*, cada vez que se encendiera dicha terminal. Estos nuevos equipos, ni son las tradicionales terminales "tontas" del pasado (sino que tienen cierta inteligencia), ni tampoco se trata de asistentes personales de datos . Si se pudieran extrapolar con validez las tendencias que hasta el momento se perciben, se podría concluir que los sistemas operativos dependientes de cada plataforma y las costosas PCs que tenemos en la actualidad podrían ser reemplazados por un nuevo paradigma basado en otros dispositivos de red más baratos, por interfaces tipo navegadores y por una serie de "applets" que podríamos bajar de la *Internet*, bajo un esquema "justo a tiempo".

## 1.3 Evolución y Tendencias en Internet

Los puntos cubiertos hasta aquí presentan el panorama de como surgió *Internet* , los servicios que proporciona y como está constituido actualmente; en seguida se tocarán algunos aspectos de lo que se espera en el futuro próximo .

Como se mencionó en párrafos anteriores, *Internet* surgió como un medio para intercambiar información. Por mucho tiempo ésta fue la orientación y dado que el sector principal al que estaba dirigido era el universitario y el de investigación, no había mucho interés en facilitar el acceso a la información. En los últimos años la evolución de los servicios en la red ha estado enfocada básicamente a simplificar al usuario su navegación dentro de la Red. El potencial que representa la información contenida en la Red ha despertado el interés de los sectores más diversos, por lo que resulta entendible el énfasis que se ha puesto en el desarrollo de herramientas de navegación que faciliten el acceso a la información a usuarios sin conocimientos informáticos. Es así como ha empezado la guerra entre las grandes firmas, principalmente entre *Netscape Communications y Microsoft*, por acaparar el mercado de navegadores (*browsers*). Muchas empresas y aún personas físicas han encontrado en *Internet* un gran escaparate para anunciar sus productos o servicios, lo que está marcando drásticamente un giro de una red orientada fundamentalmente al intercambio de información, a una red con una orientación adicional : difusión y comercialización.

Si bien *Internet* tiene 25 años de vida, ha sido en el ultimo par de años que ha visto su mayor desarrollo , con un crecimiento explosivo de usuarios y una evolución vertiginosa en los productos que se están desarrollando para ofrecer servicios en la Red. La publicación de información estática (contenido) en el *World Wide Web*, si bien ha abierto la puerta de la modernidad al ofrecer un atractivo medio mercadotécnico, no es más que un paso dentro de la expectativa de las grandes empresas que están poniendo sus ojos en *Internet* : el comercio electrónico.

Para hacer realidad el comercio electrónico a través de la Red, se requiere contar con herramientas que permitan complementar los contenidos estáticos del *World Wide Web* con aplicaciones que permitan interacción con el usuario. Evidentemente, estas facilidades abrirán la puerta a una gran gama de aplicaciones, no solo las referentes al comercio. Es en este sentido que están trabajando la mayoría, por no decir todas, las compañías del ramo informático, al grado de que se está gestando un cambio de paradigma en la forma en que se desarrollan los sistemas informáticos, ya que todos los elementos que están surgiendo para el desarrollo de aplicaciones para ofrecer servicios en *Internet* pueden también utilizarse en el desarrollo de aplicaciones de uso interno en una corporación, esquema al que se le denomina ***Intranet***

Gartner Group[3] predice que para 1998 se incrementará en un 50% el número de empresas que tendrán su sitio de *WWW* y para el año 2000 el 25% de las transacciones de negocios en general se realizarán sobre el *Web*. El comercio electrónico que hoy se encuentra en su infancia, madurará hasta convertirse en un estandar para transacciones de negocios. Esta transición depende y es enfatizada por los cambios en la infraestructura que hace que los programas de navegación sean la interface gráfica prevaleciente y que *TCP/IP* y *HTTP* sean los protocolos de datos comunes y que herramientas de desarrollo como *Java* sean los predominantes. Todo esto implicará para el personal de Sistemas nuevas aproximaciones a esquemas de administración de seguridad y riesgo, así como de administración de desempeño, tráfico y operaciones.

Como parte de la evolución de *Internet* varias extensiones están surgiendo (ver fig. 1.7). En el servidor *Web*, existen ahora varias extensiones hacia *CGI* rivales cuyo objetivo es encapsular el *CGI* en extensiones *CGI* propietarias que puedan proporcionar mejores accesos desde el cliente *Web* hacia los procesos *back-end*. Estas extensiones pueden incluir monitores de transacciones, sistemas manejadores de bases de datos relacionales, máquinas de flujo de trabajo *(workflow)*, sistemas manejadores de documentos, sistemas de encolamiento de mensajes entre procesos y otros procesos *Middleware o back-end*.

En el lado cliente, *Netscape, Sun Microsystems y Silicon Graphics* han hecho equipo para promoveer ampliamente *Java* y *JavaScript* como un estandar "cross-platform" para descargar *applets* que puedan ser movidos de un servidor *Internet* al dispositivo del usuario para cuando este lo requiera.

La mayoría de los vendedores de productos de *groupware*, *workgroup* y cliente/servidor están planeando ofrecer una versión del cliente *Web* que pueda permitir a los usuarios basados en el *Web* interactuar con una versión futura de su servidor, sobre *HTTP*. También están intentando seguir ofreciendo sus propios clientes *no-Web* como una alternativa de mayor valor a los clientes *Web*.

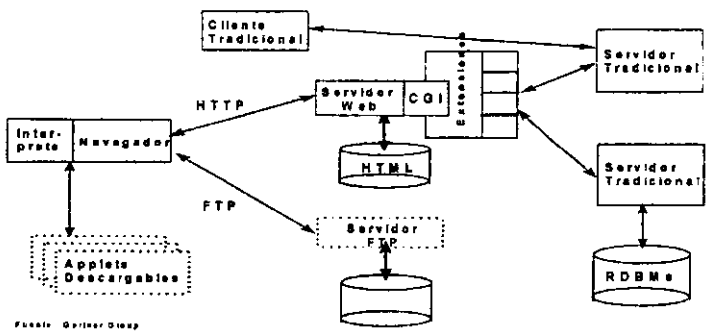



***Fig. 1.7 Escenario Intermedio de la Evolución del Web***

[3] Reconocido grupo consultor, enfocado a la evaluación de tecnología informática

### *Herramientas "Data Web"*

Han surgido en el mercado nuevas herramientas que están ayudando a ligar las aplicaciones existentes dentro de una empresa con el *Web*, llamadas "herramientas para *Data Web*", su propósito es permitir que se utilice al *Web* para distribuir datos provenientes de varios tipos de fuentes a lo largo y ancho de la organización. El objetivo es explotar al máximo las capacidades de acceso a múltiples plataformas que ofrecen los navegadores en el *Web*, tanto para la consulta como para la manipulación de los datos.

El proceso de integración de un *Data Web* consta de tres etapas : en la primera de ellas, la instalación del *Web* se utiliza principalmente para efectos mercadotécnicos ; en la siguiente es cuando las compañías empiezan a considerar la posiblilidad de relacionar las aplicaciones y/o fuentes de datos existentes en los *Webs*, ya sean internas o externas ; finalmente en la tercera, ya se estará en la posiblidad de construir aplicaciones especialmente diseñadas para el ambiente del *Web*.

Así las cosas, los proveedores de *software* están desarrollando herramientas para ligar datos residentes en bases de datos, aplicaciones en Cobol y otros tipos de almacenes "hechos en casa" con los navegadores del *Web*, vía *Intranets*[4] y la *Internet* misma.

### *VRML*

La realidad virtual no es una idea nueva. Se ha estado fantaseando con mundos de realidad virtual por decadas. El lenguaje de modelado de realidad virtual (*VRML*) ha sido el resultado del fuerte interés en la realidad virtual y las tecnologías de modelado en 3 dimensiones ; sin embargo, el incorporar la realidad virtual al *Web* son ya palabras mayores. Mientras gran parte de los usuarios del *Web* estaban bajando Mosaic como producto de navegación en dos dimensiones, un grupo de usuarios visionarios ya estaban contemplando cómo se vería modificado el *Web* con la realidad virtual. Estos usuarios se reunieron en Marzo de 1994. Uno de los objectivos de la reunión fue disparar un esfuerzo para crear un lenguaje común para realidad virtual en el *Web*. Lo que la posibilidad de presentar imagenes complejas en tercera dimensión usando simples instrucciones de marcado. Al igual que *HTML*, *VRML* está diseñado para ser independiente de la plataforma, extendible, y para usar anchos de banda limitados.

Aunque *VRML* fue desarrollado para ser independiente de *HTML*, depende de la estructura de Hipertexto para transferir archivos a través de la red, por lo que todos los conceptos en torno a *HTTP* y *URLs* también aplican a *VRML*. Los documentos de tipo *VRML* se accesan con un navegador para *VRML* o con un navegador para *HTML* con un módulo incluido para *VRML* .

[4] Redes intraempresariales que utilizan la misma tecnología de Internet para compartir información, difundir comunicados e implementar aplicaciones de uso interno en la empresa.

### *Comercialización*

Muchas compañías grandes han participado en *Internet* por años. La mayor parte únicamente con sus departamentos de investigación e ingeniería. Las mismas compañías emplean otra red (normalmente una red privada) para satisfacer sus necesidades de comunicación interna.

Las empresas están descubriendo ahora que mantener varias redes es muy costoso. Por lo que algunas han empezado a ver a *Internet* como la "solución completa" a sus necesidades de conectividad. Ahora, las empresas pueden usar *Internet* como una herramienta para resolver la problemática de sus negocios.

Mucho más allá del beneficio que por años se ha encontrado en *Internet* como un medio para intercambiar información, aparece ahora como principal atractivo de grandes empresas, el encontrar en *Internet* un gran escaparate para que sus productos y servicios lleguen a un mercado potencial cada vez más extenso. Es así como podemos esperar estar viendo en los próximos años un crecimiento vertiginoso en el comercio electrónico en *Internet*, tendencia a la que, desde luego, no están ajenas las instituciones financieras, que ya hoy empiezan a incursionar en la publicación de sus productos en el *Web*, caminando rápidamente hacia el concepto de Banca Virtual. El First Security Bank es el pionero en este concepto y presenta hoy día ya la posibilidad de abrir cuentas, consultar saldos y hacer transferencias de fondos, todo a través de la "magia" de *Internet*, y , lo que resulta más asombroso, con sólo 5 empleados.

### *Tendencias Dominantes*

Gartner Group dice que *no espera que la Internet se convierta ni en una utopía ni en un desastre. Más bien, espera que Internet y sus tecnologías puedan ser usadas cada vez más tanto para comunicaciones intra-empresa como inter-empresas por los siguientes años, y que la empresa esté intentando proporcionar información común y un ambiente de comunicación para todos sus empleados, así como a clientes importantes, proovedores y socios de negocios.* Además menciona que las tendencias dominantes serán :

- La *Internet* estará atravesando la adolescencia en 1997 : experimentando un tremendo brote de crecimiento, eliminando algo de las despreocupaciones de la juventud y adquiriendo algo de la madurez, disciplina y responsabilidades de la edad madura.

- El atractivo de una conectividad universal y del acceso a la información y gente es abrumadoramente irresistible, y seguirá impulsando a *Internet* en una rápida tendencia ascendente (a pesar de sus fallas), acompañada de períodos de mejoras.

- El crecimiento (en número de usuarios y aplicaciones avanzadas) está llevando a *Internet* más allá de las especificaciones para las cuales fué diseñada. Su modelo de negocio se está volviendo cada vez más frágil.

- Llegará un momento en que la infraestructura de comunicaciones de *Internet* reventará y será reconstruida en una base de continuidad, pero esto no inhibirá significativamente su crecimiento o uso cotidiano.

- El *Web* se está expandiendo rápidamente hacia el mundo corporativo a través de una membrana semipermeable que se volverá más porosa con el tiempo.

- El *Web* evolucionará a partir de un medio de publicación pasivo hacia un ambiente de cómputo interactivo que incluya Cliente/Servidor llegando hasta procesamiento de transacciones, comercio electrónico y funcionamiento dinámico de Red.

- Para el año 2000, *Internet* será para el mundo de la comunicación de datos, lo que hoy en día es el "marcado de tonos".

- *Internet* creará la oportunidad para que algunos vendedores puedan crecer a la altura de corporaciones como Microsoft.

⇒ Esta oportunidad y las escasas barreras para acceder a ellas ocasionarán la proliferación de vendedores de *Internet*.

⇒ Innumnerables vendedores con creatividad ilimitada generarán una explosión de productos caracterizados por ciclos de desarrollo y de vida muy cortos.

⇒ Este juego de compartir y dominar conducirá a adquisiciones y consolidaciones de empresas, así como al surgimiento de un grupo de grandes protagonistas que se estarán repartiendo el mercado.

En opinión de Gartner Group, en 1998 los más grandes obstáculos para el desarrollo comercial de *Internet* pueden estar resueltos. En la tabla 1.2 se muestra la evolución comercial de *Internet* hacia el año 2000.

| | |
|---|---|
| 1996 | • El 15% de las grandes corporaciones tienen sitios Web sencillos.<br>• Los procesadores de páginas Web hacen tan sencilla la publicación de Webs como la publicación de documentos integrados.<br>• La publicación de elementos de audio se vuelve una práctica cotidiana en el Web. |
| 1997 | • El correo electrónico basado en Internet se convierte en la medula de las Intranets.<br>• Líneas conmutadas confiables de 28.8 Kps se encuentran disponibles en E.U., Europa Occidental, Japón y Australia. |
| 1998 | • La próxima generación constará de una plataforma implementada con una columna vertebral (backbone) de ruteo.<br>• Ruteadores personales accesibles a los mismos precios que los modems.<br>• Herramientas de desarrollo para Internet suficientemente maduras para Intranet.<br>• 50% de las grandes corporaciones tienen Web corporativo.<br>• Desempeño y seguridad de Internet viable para la mayoría de las aplicaciones. |
| 1999 | • 30% de los sitios Web tiene contenido ejecutable.<br>• Garantías de calidad del servicio de Red.<br>• En E.U. 50% de los accesos por vía conmutada es través de ISDN ; 75% en Europa<br>• Occidental.<br>• La mayoría de los usuarios de PC´s son usuarios de Internet.<br>• El comercio en Internet es tan aceptado como el comercio a través de números 800. |
| 2000 | • El comercio electrónico es mandatorio para la mayoría de los negocios.<br>• La videoconferencia a nivel escritorio de alta calidad es una práctica común.<br>• 25% de las aplicaciones de procesamiento de transacciones se encuentran en el Web.<br>• Las herramientas de desarrollo para Web están suficientemente maduras. |

***Tabla 1.2 Evolución comercial de Internet***

### *Crecimiento de* Internet

Gartner Group considera que *Internet* está creciendo y evolucionando tan rápidamente y su explosión hacia su uso generalizado es tan reciente que no existen datos creíbles en relación al número de usuarios. No existen datos confiables e históricos en terminos significativos y previsiones, acerca de algo que está cambiando tan rápido como *Internet*; sólo pueden ser usados como aproximaciones a modo de guía. La fig. 1.8 muestra la previsión que hace Gartner Group sobre el crecimiento de *Internet* para los proximos años.

Por otra parte un estudio de Forrester Research muestra que el 75% de las compañías que aparecen en la lista "Fortune 1000" estarán operando transacciones en línea vía *Internet* para 1997, lo que representaría el doble respecto a las que lo hacían en 1996. Según esta firma de investigación, cada vez son más las compañías que implementarán sistemas que les permitan a los clientes efectuar sus compras con tarjetas de crédito, pagar servicios, administrar interactivamente sus inventarios a través de la Red, sobre todo a medida que se vayan solucionando los aspectos concernientes a seguridad, estándares y la estabilidad de los proveedores.

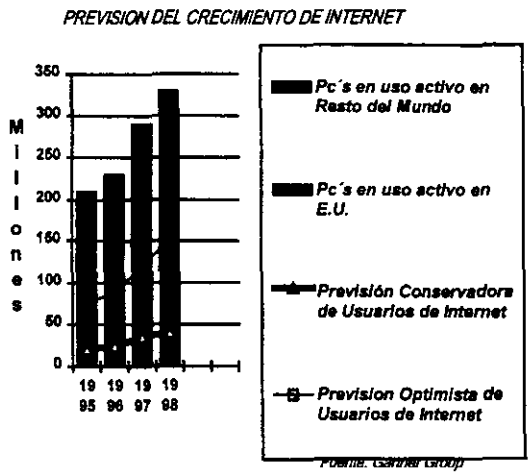



*Fig. 1.8 Crecimiento de Internet*

La fig. 1.9 muestra a los países que, con un alto nivel de adopción de la tecnología y con menores restricciones gubernamentales en cuanto al contenido de la publicación, pueden experimentar el mayor crecimiento de *Internet*.

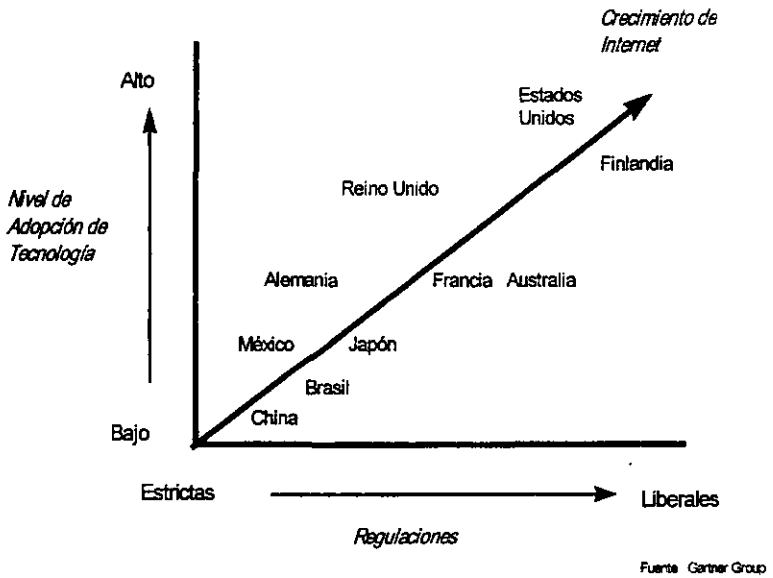



***Fig. 1.9 Países con el mayor crecimiento de Internet***

# CAPITULO II

# Cliente Servidor, toda una revolución

Algunos de los conceptos que han generado gran interés en el sector de Tecnología de la Información en los 90´s son *downsizing/rigthsizing*, orientación a objectos, recursos distribuidos, etc.. Tal vez la tecnología más importante entre todas estas sea Cliente/Servidor, que ha cobrado la más alta prioridad entre las planeaciones de los usuarios para migrar a sistemas abiertos distribuidos.

El fundamento de los cambios originados por los conceptos anteriores, principalmente aquellos asociados con la tecnología Cliente/Servidor, es bien conocido. La industria de cómputo está moviéndose paulatinamente de un ambiente centrado en *mainframes* a uno dominado por redes distribuidas compuestas por PC´s, estaciones de trabajo y servidores de bases de datos.

Esto no implica, sin embargo, que los *mainframes* vayan a desaparecer por completo del escenario. Por el contrario, seguirán jugando un importante rol en el futuro computacional, servirán como motores de bases de datos, servidores de red, plataformas de desarrollo, etc.; así mismo, seguirán proporcionando los servicios de procesamiento tradicionales.

Hay algunas tendencias muy poderosas en el escenario tecnológico que se encuentran, sin embargo, disminuyendo la relevancia de los *mainframes*. Las soluciones Cliente/Servidor se encuentran a la cabeza de estas tendencias y han sido posibles gracias a los siguentes eventos que se han dado en forma paralela :

- Avances acelerados en la tecnología de microprocesadores.

  Procesadores multiusuarios muy poderosos basados en un conjunto reducido de instrucciones de cómputo (*RISC*), la arquitectura de procesadores escalables (*SPARC*), y el diseño del microprocesador 80586-Pentium, ofrecen un desempeño superior a un precio fácilmente competitivo con sus antecesores monolíticos en *mainframes*. La flexibilidad está basada en servidores escalables y dispositivos cliente que se puedan adaptar a un amplio rango de aplicaciones de usuario.

- Estándares de *Software* y *Hardware*.

  El factor menos controlado de los 80´s y principios de los 90´s fue la estandarización de la tecnología, por lo menos en términos legales. Eso no quiere decir que los estándares sean innecesarios; son necesarios, de hecho resultan cruciales para la tecnología Cliente/Servidor. Lo que ha caído en disputas ha sido el aspecto legislativo ya que los comités han trabajado muy lentamente. De aquí que los estándares formales no estén siendo seguidos y que los estándares de facto estén dominando el mercado. En otras palabras, si algo funciona, será ampliamente utilizado. Ejemplo de tales productos son los procesadores *Intel*, *Microsoft Windows* y los protocolos *TCP/IP*.

- Avance Tecnológico de la Comunicaciones.

  El avance de las telecomunicaciones (fibra óptica, comunicaciones satelitales más accesibles, mayores anchos de banda, etc.) han permitido la implementación de esquemas Cliente/Servidor, que requieren usar grandes anchos de banda para el manejo de imagenes, voz y el envio de grandes cantidades de datos desde los mainframes que permita la distribución de las aplicaciones y el procesamiento local de los datos. Este avance de las comunicaciones también ha logrado que las transmisiones de datos se realicen cada día en menor tiempo y con mayor confiabilidad.

- Modelo Cliente/Servidor

  Este modelo explota todos los avances tecnológicas mencionados en los puntos anteriores. Ofrece una arquitectura descentralizada que le ayuda a los usuarios a obtener acceso a información que está distribuida en un nivel u otro entre plataformas de múltiples vendedores. Debido a su gran flexibilidad, los sistemas Cliente/Servidor pueden soportar una gran variedad de aplicaciones distribuidas.

- Nuevas herramientas de desarrollo

  Con la llegada de las Interfaces Gráficas de Usuario (*GUI*) ya sea en Macintosh, Windows u OS/2 Presentation Manager etc. ha surgido una nueva generación de herramientas para el modelo Cliente/Servidor. A menudo, estas herramientas están ligadas a productos *4GL*. La adición de las capacidades de las *GUI*, junto con el soporte a técnicas distribuidas, ha incorporado a los antiguos *4GLs* al dominio de Cliente/Servidor. Algunas características típicas asociadas a estos nuevos paquetes incluyen un ambiente orientado a ventanas para facilitar la interacción con el usuario ; la presencia de técnicas de orientación a objetos, reducción de formalismo en metodologías y la aparición de soluciones *CASE*.

La tecnología Cliente/Servidor es la implementación de todo el procesamiento, *software*, y avances arquitectónicos planteados previamente. Consiste de elementos claves como las redes locales, que pueden ser integradas ; interfaces gráficas de usuarios para simplificar el acceso de los mismos ; módulos de bases de datos distribuidas para la recuperación eficiente de la información, herramientas *SQL* para acceso a esas bases de datos y una gran variedad de herramientas de *software* que ayudan al usuario a construir el modelo Cliente/Servidor.

El modelo Cliente/Servidor no pertenece a ningún fabricante en particular. Es una arquitectura cuya premisa básica incluye estaciones de trabajo o computadoras personales (típicamente clientes) conectadas a repositorios de datos y programas (servidores) vía una red (generalmente una red de área local). No es, sin embargo, sólo otra aplicación de red local, representa un completo rediseño de las interacciones con usuarios, información y tecnología, pudiendo llegar a cambiar el modelo operacional de toda una corporación.

Hay varios modelos para implementar la arquitectura Cliente/Servidor (ver figura 2.1). Los niveles de distribución de servicios de presentación, lógica de la aplicación, localización de bases de datos y *software* de soporte difieren entre los distintos diseños :

- Presentación Distribuida. El programa cliente corre en la plataforma servidora y maneja una terminal como interface para el usuario.
- Presentación Remota. El programa cliente corre en la plataforma servidora y maneja una interface de usuario gráfica en una estación de trabajo (*wokstation*).
- Lógica Distribuida. El programa cliente corre en una estación de trabajo, llama a un servicio que corre en la plataforma servidora y ejecuta una transacción definida por el usuario.
- Acceso Remoto de Datos. El programa cliente corre en una estación de trabajo y llama sentencias de *SQL* que son ejecutadas por un sistema administrador de Bases de Datos (*DBMS*) en un servidor de Bases de Datos.
- Acceso Distribuido de Datos. Los administradores de Bases de Datos en una red forman un administrador lógico único de Bases de Datos ; cada uno analiza las peticiones de *SQL* y envía los elementos apropiados a los motores *DBMS* remotos como sea necesario. Los administradores de Bases de Datos proporcionan acceso transparente a los datos dondequiera que éstos se encuentren.

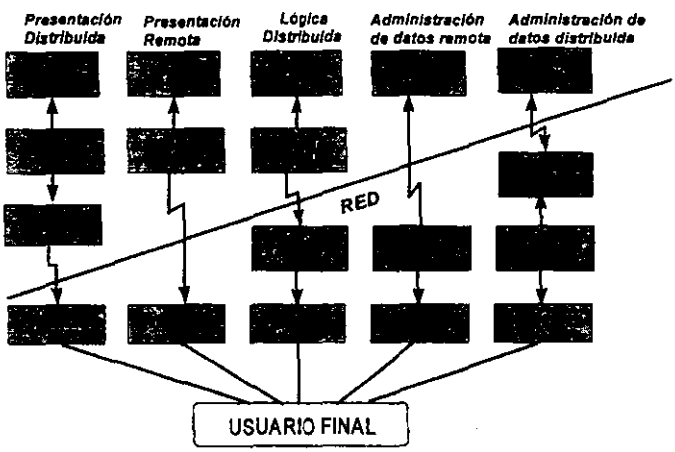



***Fig. 2.1 Modelos de Arquitectura Cliente/Servidor***

La aproximación más común y tal vez la más sencilla que se ha seguido más frecuentemente es la mostrada en la figura 2.2 . En dicha figura se ilustra una arquitectura relativamente robusta, en su forma más básica, el cliente proporciona la interface gráfica y los servicios de lógica aplicativa, mientras que el servidor proporciona los servicios de bases de datos. No hay más distribución de procesamiento o de datos que eso.

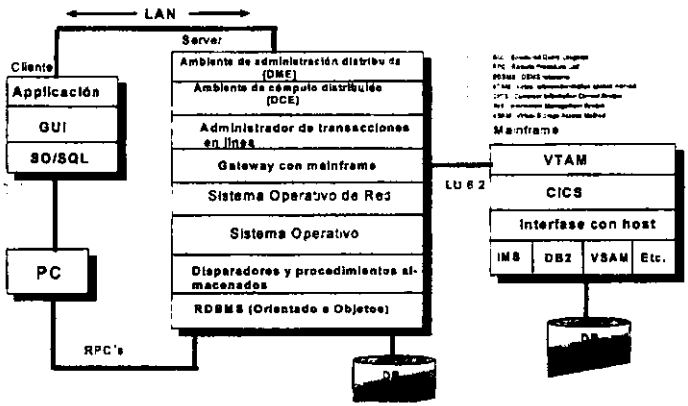



***Fig. 2.2 Componentes en un modelo Cliente/Servidor Distribuido***

Hay muchos elementos importantes que componen un diseño Cliente/Servidor pero tres de las tecnologías mandatorias son los servicios de red de área local, de *SQL* y de *GUI* (ver figura 2.3). Cualquier implementación básica de Cliente/Servidor contará con estos elementos.

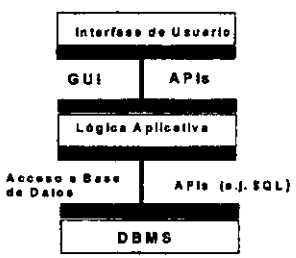



***Fig. 2.3 Principales componentes en un modelo Cliente/Servidor***

Uno de los componentes vitales que dan poder a la tecnología Cliente/Servidor es el medio de comunicación: la Red de Area Local (*LAN*, por sus siglas en inglés). Aunque hay otros elementos potencialmente involucrados como puentes y ruteadores usados para extender e interconectar múltiples redes locales, es la tecnología de redes locales la que distribuye los datos entre clientes y servidores.

Las redes de área amplia (*WAN*, por sus siglas en inglés) también pueden jugar un papel importante cuando los servidores se encuentran dispersos sobre una amplia área geográfica, pero las redes locales son típicamente el punto focal en las comunicaciones Cliente/Servidor.

Otro elemento clave para la tecnología Cliente/Servidor es la interface con el usuario. Es un componente inherente al manejo inteligente de la red. Básicamente los *GUI's* son imágenes que sirven como *front-end* a los sistemas operativos. Su funcionalidad incluye iconos que representan aplicaciones y recursos, menús y cajas de diálogo.

## 2.1 Definición y modalidades de la Arquitectura Cliente/Servidor

La computación Cliente/Servidor es una excitante tecnología con beneficios reales para los usuarios de computadoras. Esta tecnología promete un nuevo mecanismo de computación para los años 90´s. Puede redefinir la manera como los programas de computadoras son escritos, ejecutados y mantenidos. Mientras la tecnología Cliente/Servidor es por sí misma muy básica, el poder y la flexibilidad que se pueden derivar de su uso son impresionantes.

Tradicionalmente, los programas de computadoras se ejecutaban en una máquina sencilla. Los sistemas han sido aplicaciones de *mainframe*, donde todo el cómputo es hecho en las máquinas centrales (*mainframes*) o aplicaciones aisladas, donde todo el procesamiento toma lugar en una computadora personal o en una estación de trabajo (*Workstation).* Las aplicaciones Cliente/Servidor son diferentes porque se ejecutan en ambas tecnologías y combinan lo mejor de ellas en un ambiente cohesivo.

Por definición técnica, el cómputo Cliente/Servidor es la distribución de una aplicación dentro de dos componentes lógicamente separados, cada uno realizando funciones muy específicas y diferentes. Generalmente, un cliente genera un requerimiento para que el servidor realice cierto trabajo a favor de él. La tarea del servidor es procesar el requerimiento del cliente y regresarle los resultados. Este proceso, la mayoría de las veces, ocurre entre dos computadoras físicamente separadas sobre alguna infraestructura física de red local. Típicamente, las computadoras servidores son mucho más grandes y rápidas y están mejor equipadas para manejar el trabajo de otros sistemas.

La tecnología Cliente/Servidor basicamente está compuesta de un cliente o *front-end*, que son programas que coordinan la lógica de la aplicación con un servidor o *back-end* usando la red local como mecanismo de comunicación. Esta distribución de aplicaciones permite que dos computadoras procesen el trabajo en lugar de una. De hecho, con la evolución hacia el cómputo distribuido varios servidores pueden estar involucrados en el procesamiento de la lógica aplicativa en el *back-end.* La combinación de estos sistemas ofrece una gran oportunidad para tener aplicaciones de alto desempeño con un alto nivel de escalabilidad.

El esquema Cliente/Servidor ofrece la flexibilidad de realizar el proceso de cómputo en donde éste sea más eficiente. Resulta muy benéfico el permitir que los componentes lógicos de la aplicación residan ya sea en el cliente o ya sea en el servidor. A través de migrar porciones de la aplicación Cliente/Servidor entre el cliente y el servidor ; se puede determinar en base a los resultados, dónde es conveniente tener cada componente. De hecho algunos productos para desarrollo de sistemas Cliente/Servidor permiten que la localización de la ejecución se determine en el momento de ejecución.

Debido a que el procesamiento puede estar localizado en cualquier parte de la red, el esquema de Cliente/Servidor ofrece escalabilidad. Para alcanzar el balance apropiado en este modelo, un componente de una aplicación debe ser ejecutado en un servidor sólo cuando resulte más eficiente procesar el trabajo centralmente. Por ejemplo, resulta una buena idea el hacer residir porciones de una aplicación en un servidor en donde residen datos centrales. De esta manera, los requerimientos de red pueden ser minimizados, debido a que la lógica aplicativa que interactua con los datos centrales estaría localizada en la misma máquina en donde se encontrarán los datos, evitando que éstos se transmitan por la red.

Es importante resaltar, sin embargo, que conforme la cantidad del trabajo para el servidor se incremente, el sistema se convierte más en un esquema tipo *mainframe*. Dado que la tecnología Cliente/Servidor es implementada en máquinas pequeñas, una aplicación puede degradar su desempeño si la computadora usada como servidor se sobrecarga.

### *Evolución de la tecnología Cliente/Servidor*

El cómputo en el esquema Cliente/Servidor difiere mucho del esquema de cómputo centralizado en equipos centrales o *mainframes*. Lógicamente son esquemas muy diferentes, pero tienen algunas características similares. Ambos esquemas procesan centralmente la lógica aplicativa en favor de sus usuarios remotos. Adicionalmente, dichos usuarios remotos se encuentran generalmente conectados a través de algún medio de conexión física a la red. Desde un punto de vista arquitectónico, la diferencia más importante entre ambos modelos estriba en que en el cómputo Cliente/Servidor el cliente es inteligente en tanto que en el esquema de *mainframes* las terminales son generalmente tontas. Las terminales del *mainframe* básicamente funcionan como un dispositivo de entrada y salida para la computadora (que procesa centralmente toda la lógica aplicativa), mientras que los clientes en el modelo Cliente/Servidor tienen una unidad de procesamiento con la capacidad de procesar información real (ver figura 2.4).

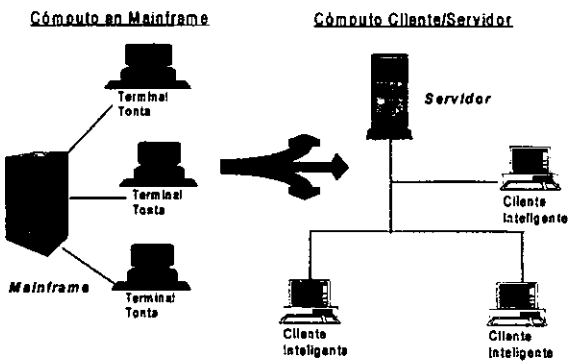



***Fig. 2.4 Esquema Mainframe vs Esquema C/S***

Históricamente, ha habido una tendencia hacia el cómputo Cliente/Servidor. Primero, los Sistemas Operativos se descompusieron en componentes cuya interacción fue administrada como procesos cliente y servidor corriendo en la misma máquina. Más tarde, aparecieron los sistemas de archivo de la red y se empezaron a presentar requerimientos de archivos o datos desde una máquina remota. En tales ambientes, los requerimientos de datos eran ruteados sobre la red y atendidos por un servidor de archivos de la red. Algunos sistemas operativos de red, como Novell Netware, han usado la tecnología Cliente/Servidor por más de una década.

El esquema Cliente/Servidor ha sido transformado, de componentes a nivel del sistema para ejecutar tareas tales como compartir archivos, a aplicaciones mismas con proceso Cliente/Servidor. De hecho, la aplicación Cliente/Servidor más común hoy en día es la Base de Datos Relacional. La mayoría de los proveedores de Bases de Datos tienen una implementación Cliente/Servidor con sus productos. Con tales productos, las consultas generalmente son realizadas con servidores de Bases de Datos. Los requerimientos son formulados en el cliente y enviados al servidor para su ejecución. El servidor procesa las consultas con la base de datos almacenada localmente y regresa el resultado sobre la red al cliente.

La tendencia actual en el esquema Cliente/Servidor consiste en distribuir la lógica de la aplicación. Ya ha sido probado que el modelo Cliente/Servidor ofrece grandes ganancias en el desempeño en los servicios de archivos y bases de datos. Hoy en día, la mayoría de los sistemas operativos tienen la capacidad de ejecutar sistemas de cómputo Cliente/Servidor. Las características medulares de estos sistemas deben incluir sistemas robustos y soporte a nivel aplicativo así como componentes de red. Algunos sistemas operativos se desempeñan mejor al jugar el papel de cliente, mientras que otros lo hacen mejor como servidores. Hay otros más que hacen muy bien ambos papeles. Algunas compañías como Novell primero proporcionaron facilidades de servidor de red y más tarde agregaron servicios aplicativos. Tanto Unix como Windows NT proporcionan ambos servicios concurrentemente y nacieron preparados para Cliente/Servidor.

### *Cómputo distribuido y heterogéneo*

Uno de los factores a los que se enfrenta el personal de sistemas hoy en día es la diversidad de ambientes de cómputo existentes. Plataformas tecnológicas ampliamente esparcidas como las computadores personales, minicomputadoras y *mainframes*, así como sistemas operativos aún más diversos, representan un problema para el personal de sistemas, así como una oportunidad para los desarrolladores de aplicaciones. Las organizaciones más grandes, como las contenidas en el reporte de las mil compañías más grandes de Fortune, y aún compañías mucho más pequeñas, tienen diferentes sistemas de cómputo. Para los sistemas de administración de información (*MIS*), el punto crítico es cómo hacer que todos los sistemas se comuniquen y operen entre sí. Para los desarrolladores de aplicaciones, el reto es proporcionar su *software* en tantas plataformas como sea posible y permitir que sus aplicaciones se comuniquen entre sí. A estos ambientes diversos de cómputo se les denomina cómputo heterogéneo.

Por otra parte, el cómputo distribuido es una arquitectura tecnológica con muchos beneficios prometedores y grandes ventajas . De la misma forma que el cómputo Cliente/Servidor, el cómputo distribuido implica la distribución del trabajo entre más de una máquina. El cómputo distribuido es, sin embargo, más amplio en el sentido de que muchas máquinas pueden estar procesando trabajo para las máquinas cliente. Se puede pensar en que en el cómputo distribuido existe un cliente y muchos servidores.

Este tipo de cómputo beneficia igualmente a usuarios y a corporaciones. Dado que el cómputo distribuido es una forma de cómputo Cliente/Servidor, los usuarios se benefician más de las mismas cosas. El incremento del rendimiento general de la aplicación y la habilidad para realizar multi-tareas son algunos de los beneficios para el usuario. Las corporaciones se benefician de la integración de componentes discretos de red como un todo para incrementar la eficiencia y reducir costos.

El cómputo distribuido pretende que los sistemas y las redes operen eficientemente como un todo. Conforme las nuevas aplicaciones sean desarrolladas e implantadas, muchas de ellas estarán cooperando para alcanzar los resultados deseados. Esta habilidad puede maximizar el uso del equipo existente y optimizar totalmente los ambientes heterogéneos.

Sin embargo, el cómputo distribuido es una arquitectura compleja. Involucra el rediseño de las aplicaciones, reimplantación de los sistemas, e incrementa las dificultades para manejar la red como un todo. La complejidad de su implantación deriva también en otros aspectos tecnológicos como Sistemas Operativos Distribuidos, actualmente se tienen varios pero a nivel académico y experimental, a la distribución de las Bases de Datos y a las comunicaciones aún son insuficientes. Aunque los beneficios son evidentes, los aspectos anteriores pondrían a prueba a un área de sistemas ya existente. Para que la tecnología se convierta en efectiva y revolucionaria, los desarrolladores de aplicaciones distribuidas deben hacer todo lo posible para minimizar la complejidad de la implantación y del mantenimiento.

### *Beneficios del Cómputo Cliente/Servidor*

El cómputo Cliente/Servidor beneficia a todos los participantes en un sistema de información corporativo. Los usuarios se benefician con la flexibilidad, rendimiento y la utilización completa de todos los recursos. Las corporaciones se benefician porque preservan sus inversiones, ya que los recursos son compartidos eficientemente ( lo cual trae reducción de costos), y porque la tecnología permite la interoperabilidad de datos. Los gerentes de sistemas de información se ven beneficiados con la posibilidad de desarrollar aplicaciones más rápidamente; con una significativa reducción de tráfico sobre la red y la interoperabilidad de los sistemas.

Además, apoya a los usuarios debido a que la tecnología les permite realizar tareas de uso intensivo de datos en varias máquinas. Esto significa que el cliente puede responder mejor a requerimientos locales que tengan los usuarios finales. Como la ejecución se realiza en otras máquinas, el cliente es relevado de las tareas de procesamiento intensivo y puede realizar otras tareas. Las aplicaciones se ejecutan, de esta manera, más rápidamente, desde la perspectiva del cliente.

El cómputo Cliente/Servidor ofrece también muchos beneficios para las corporaciones. Las computadoras más costosas (servidores y periféricos) y con mayor poder de procesamiento, pueden estar compartidas por muchos clientes, lo cual maximiza el uso de los sistemas más costosos y la utilización integral de los recursos disponibles. Adicionalmente, dado que el servidor está realizando trabajo para muchos clientes, el poder de cómputo existente en los clientes es extendido conforme los requerimientos de procesamiento de la estación del usuario son reducidos. Esto significa que las corporaciones pueden preservar sus inversiones en la tecnología de cliente existente e incorporar computadoras más caras pero de desempeño más alto para procesar la carga adicional de trabajo. Un beneficio adicional es el incremento de poder de procesamiento que se tiene que incorporar a las estaciones de usuario. Los ambientes tradicionales de *mainframes* ejecutaban todas las aplicaciones en una computadora central, sin embargo, con la llegada de los clientes inteligentes, las aplicaciones Cliente/Servidor que se migraron a partir de los equipos centrales, pueden disfrutar del incremento de poder de procesamiento de los equipos distribuidos. La escalabilidad de las redes distribuidas es una gran ventaja potencial de incremento en el desempeño.

Tal vez, el beneficio más relevante de Cliente/Servidor en las corporaciones, es su uso en los esquemas de administración centralizada de datos. De esta manera, los servidores pueden ser usados como un punto central para acceso de datos en Cliente/Servidor, beneficiando de manera importante a los usuarios. Al enfocar el acceso a datos en un solo punto, tanto los desarrolladores de aplicaciones como los usuarios finales se ven beneficiados. Cuando se cuenta con los servicios esenciales de conectividad, los desarrolladores de aplicaciones pueden ofrecer soporte para integración de una serie de máquinas y servicios, permitiendo que diversas plataformas *front-end* compartan los mismos archivos de datos, como se muestra en la figura 2.5.

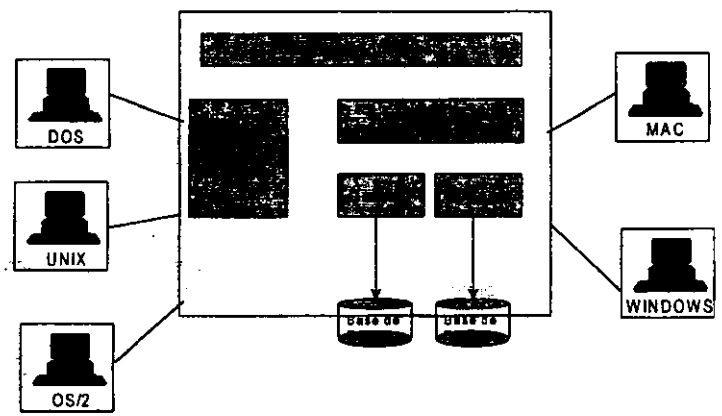



***Fig. 2.5 Esquema centralizado de acceso a datos***

### *Desventajas del Cómputo Cliente/Servidor*

Son muchas y muy atractivas las ventajas que ofrece el cómputo Cliente/Servidor, comparado contra el cómputo tradicional en *mainframes*. Sin embargo esto tiene un costo y es el representado por la dificultad que se tiene para implementar los distintos elementos de administración de sistemas, como son la distribución de software, la administración de configuraciones, la administración de problemas, etc. que en los mainframes existen de manera natural.

El éxito en el desarrollo y en la administración de aplicaciones Cliente/Servidor depende de una estrategia sólida que incluya los siguientes componentes :

- **Distribución de Software y Control de Versiones.** Si una aplicación se tiene que distribuir, una organización debe de entender como es que el código va a llegar físicamente a cada usuario o a cada grupo que lo necesite, es decir, saber si hay alguien que maneje este proceso, si los sistemas están listos para manejar esta carga, si existirá un *software* que permita que las aplicaciones se distribuyan desde un punto central o saber si el nuevo código tendrá conflictos con el código existente y además contar con herramientas y procesos que permitan tener un control adecuado de las versiones del software que se tienen que instalar.
- **Administración del desempeño.** La mayoría de los sistemas complejos inician con un piloto o con un proyecto de pruebas de concepto. Si bien, tales proyectos pueden demostrar las capacidades de los desarrolladores para escribir código, generalmente no tienen que ver con factores de escalabilidad. Las aplicaciones Cliente/Servidor deben de ser diseñadas de tal manera que se comporten bien conforme al código base y en la medida en que el número de usuarios crezca.
- **Administración de problemas.** Se pueden presentar problemas complejos después de que una aplicación haya sido revisada de errores e incluso implantada. Cada vez más organizaciones están creando aplicaciones para trabajar en paralelo con otras aplicaciones. Esto involucra el utilizar un mecanismo de mensajes para mover mensajes entre aplicaciones -lo que no es una tarea sencilla-. Y como siga creciendo la importancia de crear interoperabilidad entre las aplicaciones, este problema se volverá más complejo.
- **Administración de código.** Conforma las aplicaciones empiecen a utilizar librerías de componentes para extender la vida de los sistemas y extender su funcionalidad, seguramente surgirán problemas. Los desarrolladores tendrán que ser capaces de llevar un control de las dependencias de las librerías. También tendrán que entender el impacto en el desempeño por estos cambios, lo cual es especialmente importante si el código esta distribuido a través de diferentes sistemas. Conceptos como la sincronización de código entre múltiples sistemas se convertirá en un reto.

### *Costos del Cómputo Cliente/Servidor*

Con la introducción de la tecnología se ha asumido que las implementaciones de Cliente/Servidor han proporcionado beneficios intangibles, pero no un ahorro en costos reales. Implementadores y firmas de investigación han dispersado la idea de que hoy en día el cómputo Cliente/Servidor es una alternativa más económica que el cómputo en *mainframes* o minicomputadoras. De hecho, las implementaciones Cliente/Servidor están a la par o aún más caras en varios escenarios que los sistemas tradicionales. Mientras el sistema en forma global es comparable, los costos varían ampliamente.

Forrester Research[1] realizó una investigación en donde compara los costos asociados del cómputo Cliente/Servidor con soluciones en minicomputadoras. La fig. 2.6 muestra un desglose de los costos del cómputo Cliente/Servidor por área funcional. Es interesante notar el alto porcentaje de costos atribuidos al soporte del sistema, mantenimiento y entrenamiento, en otras palabras, a costos laborales. Los costos laborales incluyen casi la mitad de los costos totales de un sistema de información Cliente/Servidor. La expectación acerca del tremendo potencial de ahorrar costos fueron desvanecidos cuando los investigadores descubrieron estos grandes costos asociados.

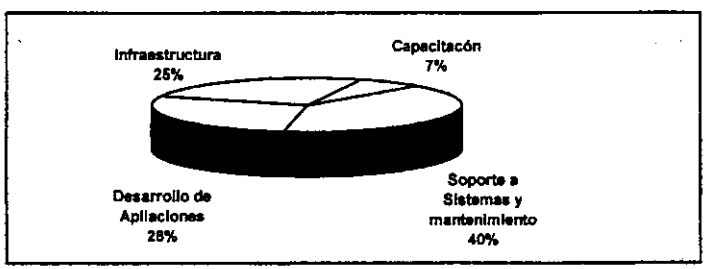



***Fig. 2.6 Desglose de costos por área funcional***

Aún cuando existen tremendos beneficios y ventajas en la tecnología Cliente/Servidor, no es económicamente ventajosa en su primera implementación. Muchos costos laborales ocultos están conduciendo el precio de los sistemas Cliente/Servidor hacia arriba.

[1] Novell's Guide to Cliente-Server Applications and Architecture

Como se ha mencionado, los costos de crear un nuevo sistema Cliente/Servidor partiendo de cero están a la par con los costos de los sistemas tradicionales, como se muestra en la tabla 2.1.

| COMPONENTE DEL COSTO | CLIENTE/SERVIDOR(%) | MINICOMPUTADORA(%) |
|---|---|---|
| **Desarrollo de la aplicación** | 28 | 40 |
| **Infraestructura** | 25 | 27 |
| **Mantenimiento/Soporte a los sistemas** | 40 | 30 |
| **Capacitación** | 7 | 3 |

*Fuente . Forrester Research*

***Tabla 2.1 Desglose de Costos de Cliente/Servidor contra Minicomputadoras***

El diferencial más grande en costos es el mayormente atribuido a los costos laborales tales como capacitación y mantenimiento. Estimaciones recientes de los costos de la tecnología Cliente/Servidor no tomaron en consideración los costos iniciales caros y prolongados asociados a estas labores. Sin embargo, comparados con los sistemas tradicionales, los sistemas Cliente/Servidor son mucho más caros en esta área. Respecto a los costos de *Hardware* (incluyendo redes e infraestructura) éstos son menores para una solución Cliente/Servidor, aunque solo modestamente.

Los costos de infraestructura del cómputo Cliente/Servidor son menores que aquellos de una solución similar basada en minicomputadoras. Sin embargo, la diferencia es muy pequeña en términos del costo diferencial, debido al hecho que las computadoras personales, redes, *gateways, bridges* y ruteadores necesitan ser adquiridos para la instalación inicial. Una solución similar en minicomputadoras no tiene esos costos asociados. Las subsecuentes implementaciones de sistemas Cliente/Servidor pueden no requerir de tales enormes costos iniciales.

Una de las mayores promesas del cómputo Cliente/Servidor son las herramientas, metodologías y tecnologías para su desarrollo. Los costos de desarrollo son 29 % menos caros para implementaciones Cliente/Servidor que para tecnologías de minicomputadoras. La diferencia del costo es debido a muchos factores, incluyendo las herramientas para desarrollo disponibles en el mercado y su inherente facilidad para desarrollar aplicaciones usando estas herramientas.

El desarrollo de aplicaciones Cliente/Servidor con un componente *4GL* reduce ampliamente la cantidad de trabajo necesario para crear aplicaciones Cliente/Servidor. Como un resultado, existe un período de tiempo más corto de desarrollo asociado con estas tecnologías.

Los costos laborales asociados a la tecnología Cliente/Servidor son altos y numerosos. Como con cualquier nueva tecnología, se requiere entrenamiento inicial y continua capacitación del personal de sistemas, programadores y usuarios debe ser esperada. Los costos laborales por si mismos suman casi el 50% de los costos totales de una implementación Cliente/Servidor.

Otro factor de costo con que se enfrentan las compañías es el continuo mantenimiento y soporte de redes Cliente/Servidor. De hecho, los costos de administración son aumentados con el Cliente/Servidor debido a los problemas de administración inherentes asociados con las redes distribuidas. Los tradicionales *mainframes* y minicomputadoras proporcionan una estrategia de administración centralizada con una maquina central cuyas herramientas son probadas y sometidas a pruebas de tiempo. Por otra parte, las herramientas de administración para Cliente/Servidor apenas empiezan a convertirse en lo suficientemente robustas para soportar implantaciones grandes y dispersas. El administrar redes distribuidas es una tarea difícil que se vuelve aún peor con las redes heterogéneas que existen hoy en día. Las soluciones de administración deben de estar bien integradas para que la tecnología Cliente/Servidor prolifere.

Una vez implantados los sistemas distribuidos, las organizaciones de sistemas de información empiezan a descubrir costos ocultos asociados con el soporte a aplicaciones Cliente/Servidor. Estos costos ocultos incluyen reconfiguraciones continuas para apoyar las funciones de reportes de *help-desk*, la reimplantación de aplicaciones (debido a nuevas plataformas distribuidas o nuevos usuarios de la aplicación) y la administración de cambios. Otro costo oculto que resulta ser muy alto, incluye la pérdida potencial de negocios debido al tiempo de caída de aplicaciones claves y gastos innecesarios de capital para nuevo *hardware*.

Conforme los sistemas distribuidos se vuelven más complejos, corren el riesgo de volverse inadministrables. El esfuerzo administrativo continuo juega un rol clave para preservar la integridad de aplicaciones de misión crítica para miles de clientes distribuidos moviendo millones de archivos. Generalmente, hay tres métodos para distribuir y administrar aplicaciones Cliente/Servidor :

1. Configuración Manual
2. Distribución de Archivos
3. Administración de configuraciones continua y automatizada

La manera más común para implantar aplicaciones Cliente/Servidor en sistemas distribuidos es el método de configuración manual. Una vez que una aplicación ha sido completamente desarrollada y probada, los archivos que conforman la aplicación son copiados a *diskettes*, *cd-rom's* o cintas. Se crean muchas copias del medio de distribución para entregar al personal que físicamente hará la instalación y configuración de la aplicación. Este proceso resulta muy doloroso y altamente consumidor de tiempo. Resulta obvio que no es el mejor método para distribuir *software*.

La evolución natural de la implantación y configuración de aplicaciones se presenta en la forma de transferencia de archivos o sistemas de distribución de *software*. Dado que existe tecnología para apoyar las nuevas aplicaciones Cliente/Servidor, usualmente en la forma de redes de área amplia, se puede usar la red para distribuir los miles de archivos aplicativos a clientes distribuidos. La automatización de la configuración de los archivos usando lenguajes basados en scripts, elimina la necesidad de contar con una persona físicamente para iniciar el sistema, eliminando el tiempo y dinero gastado en visitar la locación remota. Esto sirve para limitar los costos y aumentar la velocidad de implementación.

Es aún difícil de confirmar que todos los usuarios obtuvieron todos los archivos y que los *scripts* funcionaron correctamente y que resulte una implementación completa de la aplicación cada vez. Pero, los costos ahorrados justifican el riesgo asociado.

El más grande y más frecuente costo olvidado en la administración de una aplicación Cliente/Servidor distribuida es la continua reconfiguración y la reimplantación de aplicaciones para nuevos usuarios y la implementación en nuevo *hardware* y *software*. Esto es una administración básica de cambio. Los dos métodos discutidos anteriormente son efectivos en poner fuera una aplicación pero no manejan las continuas necesidades de los ambientes Cliente/Servidor distribuidos.

Una vez que los sistemas son implantados, las compañías descubren que los usuarios son los peores enemigos de las aplicaciones. Frecuentemente sin intención, los usuarios borran archivos obligatorios, cambian configuraciones del ambiente operativo, cambian de trabajos y reponsabilidades, etc. Cada cambio resulta en la necesidad de "refrescar" el ambiente de la aplicación distribuida para asegurar que la gente apropiada tenga la aplicación apropiada y que estén configuradas correctamente.

Basado en estimaciones [2], el costo de administrar la aplicación distribuida será arriba de 5 veces más que el costo inicial de implantar la aplicación. Se debe de destacar que esta cantidad significativa no incluye viáticos para oficinas remotas. Adicionalmente, conforme se van agregando aplicaciones el costo de administrar el ambiente distribuido se escala rápidamente.

La técnica más nueva para implantar y administrar aplicaciones distribuidas es la administración automatizada de configuraciones, la que permite administrar centralmente la implantación y configuración continua de aplicaciones. Usando este método las herramientas automatizadas de administración de sistemas usan modelos de aplicaciones y organizaciones para asegurar la excelencia de la configuración continua. Esto reduce significativamente el costo de mantener una aplicación una vez que es implantada.

La clave para administrar exitosamente aplicaciones distribuidas es la administración continua de configuraciones. Es la manera más segura, práctica y efectiva para reducir los costos ocultos de aplicaciones distribuidas. Los métodos manuales requieren de una inversión significativa de tiempo y personal para conservar un ambiente dinámico y creciente. Los métodos de distribución de archivos sólo cubren la implantación inicial de las aplicaciones e ignoran las necesidades más grandes de configuración continua.

2 Data Management Review Nov. 96 "The Hidden Costs of Client/Server Computing"

## 2.2 El Modelo de Tres Niveles

Actualmente la mayoría de la aplicaciones Cliente/Servidor están basadas en bases de datos departamentales. La arquitectura Cliente/Servidor de 2 niveles existente ha sido probada satisfactoriamente para una base de usuarios de 20 o 30 usuarios por sitio, con datos no-críticos y con una lógica de negocio sencilla. Pero existen limitaciones más allá de estos requerimientos. Es posible extender esta arquitectura agregando un nivel medio de servidores intermedios. Esta extensión crea una arquitectura Cliente/Servidor de 3 niveles, en la cual los servidores intermedios soportan la lógica del negocio y otros servicios del negocio.

### *Arquitectura de 2 Niveles*

Esta arquitectura es una buena solución para compañías tratando de desarrollar aplicaciones departamentales que incorporan sólo a un número pequeño de bases de datos relacionales, de usuarios y una lógica de negocio sencilla. Las aplicaciones Cliente/Servidor de 2 niveles están construidas con herramientas de desarrollo *GUI* que producen aplicaciones que combinan el código requerido para el manejo de la interface del usuario, la lógica de la aplicación y la recuperación de datos. Los datos para estas aplicaciones son cada vez más almacenados en Bases de Datos Relacionales, los cuales pueden ser accesar a través de una variedad de mecanismos de *middleware* (el cual se abordará más adelante).

Una arquitectura de 2 niveles puede ser construida de dos maneras : con un cliente ligero o con un cliente pesado. En la arquitectura de cliente ligero, sólo la porción *GUI* de la aplicación reside en la computadora Cliente. En la arquitectura de cliente pesado, la lógica de la aplicación y la porción *GUI* residen en la computadora Cliente.

En una arquitectura de 2 niveles, los procesos de negocios dinámicos no son fácilmente incorporados. La mayoría de las aplicaciones Cliente/Servidor existentes fueron desarrolladas con herramientas *GUI* que conectan la capa de presentación a la lógica del negocio. Las herramientas *GUI* usan programación por eventos, esto es, el código relacionado a una ventana es activado por un evento del usuario, tal como el oprimir un botón. Cuando la lógica del negocio requiere hacer cambios, el código relacionado debe ser cambiado.

Los beneficios proporcionados por una arquitectura de 2 niveles, incluyendo tiempos reducidos de desarrollo, facilidad de instalación y una interface gráfica con usuarios, motivan a los departamentos de sistemas para ampliar su visión y la complejidad de sus aplicaciones. Esta expansión genera una ampliación del número de usuarios, de la complejidad de la lógica del negocio y del número de bases de datos heterogéneas que deberán ser soportadas por estas aplicaciones. Es cuestionable por lo tanto si una arquitectura de 2 niveles es suficiente para soportar aplicaciones Cliente/Servidor corporativas. Varios vendedores sin embargo, han empezado a introducir productos especializados para resolver esta limitante.

### *Arquitectura de 3 niveles*

Una arquitectura de 3 niveles es una extensión de una arquitectura de 2 niveles a través de un nivel intermedio de servidores intermediarios. Los servidores intermediarios son una serie de servicios compartibles multi-tareas que interactúan con clientes, con otros servidores intermediarios y con servidores de bases de datos. Los servidores intermediarios juegan un rol fundamental para proveer un cómputo distribuido confiable, seguro y eficiente. Cada servidor intermediario entrega un servicio de negocio a todos los usuarios conectados en una red de área local o de área amplia.

Los beneficios proporcionados por una arquitectura de 3 niveles permite tener una base de usuarios mayor, alrededor de unas cuantas centenas de usuarios. Esta arquitectura maneja bases de datos heterogéneas que pueden ser relacionales o propietarias de los *mainframes*. La arquitecturas de 3 niveles tiene más flexibilidad y más opciones para escalar y particionar, así como para reducir el tráfico de la red y mejorar el acceso a múltiples servidores.

La meta en esta arquitectura es obtener un componente para cada función de negocio que esté lógicamente separado y funcionalmente independiente. Cada componente de la arquitectura - presentación de usuario, lógica del negocio y datos - puede ser modificado o reemplazado sin que ninguno de los demás componentes sufran cambios.

Los servidores intermediarios juegan un rol clave cuando se construyen aplicaciones Cliente/Servidor de 3 niveles. Proporcionan escalabilidad y reusabilidad al ambiente Cliente/Servidor. Los desarrolladores pueden escribir una biblioteca de funciones, aplicaciones de negocio y lógica de negocio e instalarla en un servidor y ponerla accesible a los clientes. También pueden soportar *gateways* de comunicaciones y de datos.

La arquitectura Cliente/Servidor de tres niveles (ver fig. 2.7) implica el separar las siguientes funcionalidades aplicativas en componentes o niveles intercambiables :

- Presentación (o interface de usuario), nivel que soporta la interacción humano-computadora utilizando dispositivos como teclado, mouse y monitor. También puede contener otras aplicaciones definidas por el usuario que se comuniquen con el siguiente nivel.
- Funcionalidad, conectividad y servidores de bases de datos, que se ejecutan en una o más computadoras para :
  - ⇒ Conectarse a los sistemas existentes.
  - ⇒ Conectarse a nuevas bases de datos.
  - ⇒ Procesar y formular datos.
  - ⇒ Interactuar con las interfaces de usuarios deseadas.
  - ⇒ Mantener seguridad, auditabilidad, control de versiones y otras funciones de sistemas.
- Datos, aplicaciones y sistemas existentes, que hayan sido encapsuladas para aprovechar las ventajas de esta arquitectura con un mínimo de esfuerzo transaccional.

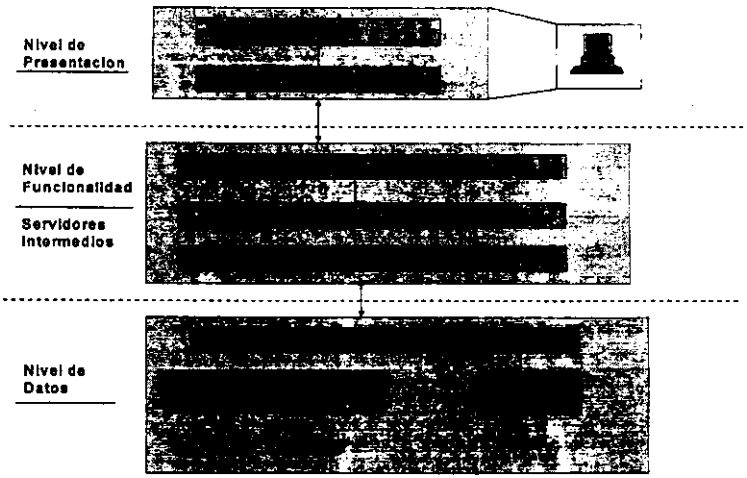



***Fig. 2.7 Arquitectura de 3 Niveles***

Cuando los usuarios interactúan con una nueva aplicación estratégica construida con esta arquitectura, realmente sólo tienen que ver con el nivel de presentación; los servidores interactúan con las bases de datos o sistemas legados existentes para manipular la información que el usuario requiere.

Algunos vendedores promueven arquitecturas de dos niveles mismas que son recomendadas básicamente para modernizar el nivel de presentación de los sistemas legados existentes, incorporando interfaces gráficas de usuario. Sin embargo, esto sólo corresponde al esquema más simple de la arquitectura Cliente/Servidor.

Al separar las aplicaciones en 3 niveles se gana lo siguiente :

- Libertad para seleccionar cualquier manejador de bases de datos.
- Respaldos dinámicos.
- Libertad para seleccionar cualquier interface gráfica de usuario.
- Flexibilidad para agregar nuevas tecnologías como se requieran.
- Acceso en línea a cualquier fuente de datos.
- Menor costo de *hardware*.
- Libertad para seleccionar el tamaño apropiado del *hardware*.
- Soporte para una estrategia de migración de sistemas antiguos.
- Aprovechamiento de inversión actual de *software* y *hardware*.
- Desarrollo paralelo de aplicaciones.
- Diferentes vistas de datos según los usuarios.

***Nivel de presentación***

El nivel de presentación proporciona a los clientes una interface de usuario que soporta la interacción humano-computadora a través de dispositivos como mouse, teclado y monitor .

***Nivel de servidores***

El nivel de servidores es el corazón de la aplicación, es en donde tiene lugar el cómputo crítico. Los servidores pueden realizar varias funciones, dentro de las cuales podemos encontrar las siguientes :

- Autentificar la identidad de los usuarios para proporcionar la seguridad de los sistemas.
- Proporcionar un mecanismo consistente y transparente para el usuario, para nombrar archivos y directorios a través de diferentes plataformas de *hardware* y de diferentes protocolos de sistemas operativos.
- Proporcionar conexiones de comunicación a fuentes externas de datos y traducir dichos datos a los formatos requeridos por el siguiente nivel.
- De esta manera, el agregar una conexión a datos adicional en una aplicación (como incorporar datos de una corporación y facilidades extraidas de un *cd-rom*) es tan simple como conectar un servidor más en este nivel que pueda comunicarse con la nueva fuente de datos.

***Nivel de datos***

El nivel de datos mantiene toda la información que una aplicación finalmente busca alcanzar y usar. Los clientes, con interface de usuario en el nivel de presentación sólo se comunican con los datos a través de un servidor intermedio apropiado. Consecuentemente el tipo de datos que comprenden a este nivel son completamente irrestringidos. Cualquier aplicación puede tener acceso a cualquier tipo de datos en cualquier sistema en cualquier lugar, toda vez que exista el servidor apropiado en su lugar y que se cumpla con las restricciones de seguridad, lo cual representa una gran flexibilidad.

### *La Arquitectura C/S de dos niveles vs la de tres*

Se ha debatido mucho en torno a la arquitectura *Cliente/Servidor* de dos niveles en comparación a la de tres, y sin embargo muchas veces no se entienden claramente las diferencias entre ambas.

Las primeras aplicaciones en *Cliente/Servidor* - que eran sistemas orientados al apoyo a la toma de decisiones para grupos pequeños - fueron construidos de acuerdo al modelo de dos niveles, por una sencilla razón : ofrece una buena transportabilidad entre plataformas y un desarrollo rápido de soluciones.

Pero dado que en este modelo la aplicación se divide en *front-ends* (la parte cliente) y *back-ends* (el servidor) y da soporte a una cantidad limitada de clientes, puede no ser la mejor solución para manejar eficazmente las aplicaciones más complejas, distribuidas a lo largo de toda la empresa. Además, a medida que el negocio cambia, los procesos del mismo también tienen que irse adecuando consecuentemente, lo cual no resulta fácil en el caso de una arquitectura de dos niveles.

Por otra parte, el surgimiento de nuevos productos para apoyar las arquitecturas Cliente/Servidor de varios niveles así como la demanda para nuevas y más flexibles arquitecturas ha ido tenido un lento incremento en medio del creciente conocimiento de las debilidades de las arquitecturas de 2 niveles.

En la arquitectura Cliente/Servidor de varios niveles es el modelo más apropiado para la mayoría de los ambientes computacionales. Diseñar una arquitectura Cliente/Servidor de varios niveles no es menos compleja que desarrollar una arquitectura de 2 niveles. La arquitectura de varios niveles, sin embargo, produce un ambiente Cliente/Servidor con mayor flexibilidad y escalabilidad.

En una arquitectura de 2 niveles, el cliente y el servidor son las únicas capas. El cliente, generalmente una computadora personal con una interface *GUI*, accesa los datos desde el servidor. En este modelo, tanto la capa de presentación como la capa de aplicación son manejadas por el cliente. Una arquitectura de varios niveles tiene una capa de presentación y dos capas de servidores separadas -una capa de lógica del negocio o de aplicación y una capa de datos. El cliente se convierte en la capa de presentación y maneja la interface del usuario. La capa de aplicación funciona entre las otras dos, enviando los requerimientos de datos del cliente a la capa de datos. El cliente está liberado de las tareas de la capa de aplicación, lo cual elimina la necesidad de tecnologías poderosas para clientes.

En estos casos es donde se vuelve indispensable la arquitectura de tres niveles. Conforme las aplicaciones en Cliente/Servidor van dando cada vez más soporte a nivel corporativo, la cantidad de conexiones y plataformas que deben contemplarse van aumentando y tornándose más complejas.

En el caso del modelo de tres niveles, la capa de interface de usuario reside en la parte cliente, la base de datos en el servidor corporativo y la lógica del negocio en una tercera plataforma intermedia.

Los criterios para inclinarse por este modelo pueden ser los siguientes :

- Si las conexiones concurrentes a la base de datos son tantas que el acceso a la misma se este volviendo ineficiente o crítico , lo más seguro es que se requiera una arquitectura de tres niveles.
- Si la aplicación requiere lógica compartida del negocio y se está considerando una fuerte utilización de procedimientos almacenados.
- También hay que considerarlo en el caso de que el uso de memoria caché sea difícil de implementar y administrar a nivel del cliente (en el modelo de dos capas) y más sencillo en los servidores de aplicaciones ( tres capas).
- Si una aplicación necesita validar a los usuarios en forma individual, así como su acceso a la base de datos, el modelo de tres niveles facilita las cosas.
- Si se tienen varias instalaciones físicas conectadas a través de una *WAN*.

Debido a su mayor funcionalidad, la arquitectura de tres niveles también implica varios retos. El principal de ellos es la selección de las herramientas más adecuadas; el segundo consiste en la planeación de la infraestructura requerida, del diseño aplicativo y de los equipos a instalar. Pero a pesar de todo ello, sus beneficios también son grandes: permite el soporte de un gran número de usuarios y de plataformas heterogéneas, un mayor volumen de datos, una mayor cantidad de transacciones y la implementación de aplicaciones más complejas.

El modelo de tres niveles da a los usuarios una mayor flexibilidad y más opciones para el crecimiento y particionamiento de las aplicaciones. Esto, a su vez, le brinda a las compañías la posibilidad de crear y modificar en forma más ágil aplicaciones *Cliente/Servidor* complejas a nivel institucional, con el consiguiente incremento de ventajas en materia de rentabilidad y productividad .

Otras ventajas de la arquitectura de 3 niveles son :

- Los cuellos de botella de las redes son minimizados debido a que la capa de aplicación no transmite datos extras a los clientes, únicamente los que son necesarios para manejar una tarea.
- Cuando se requiere cambiar la lógica del negocio, sólo el servidor tiene que ser actualizado. En una arquitectura de 2 niveles, cada cliente debe ser actualizado con los cambios a la lógica del negocio. Con la arquitectura de 3 niveles, el cliente sólo es modificado cuando las funciones son descontinuadas o los parámetros de la función cambian.
- El cliente es aislado de la base de datos y de las operaciones de la red. El cliente puede accesar fácilmente los datos sin importarle donde se encuentren o cuantos servidores están en el sistema.
- Las organizaciones tienen independencia de la base de datos debido a que la capa de datos esta escrita en *SQL* y es independiente de la plataforma.
- Se crea un medio de comunicación efectivo entre la capa de aplicación y el cliente.

## *Evolución de los niveles lógicos de Software*

En 1991 Gartner Group introdujo el modelo de "5 estilos o modelos" de procesamiento cooperativo (ver fig. 2.1). Este permanece valido y relevante para describir los niveles físicos del *software*, aunque no explique la configuración lógica del *software*. El *software* es dividido en niveles físicos separados para minimizar el *"overhead"* de la red. La lógica del *software* es usualmente colocada cerca de su recurso externo asociado para minimizar el *"overhead"* de la red y para mejorar los tiempos de respuesta, flujo y confiabilidad. En otras palabras, el *DBMS* corre en el sistema al cual los discos son conectados, la presentación lógica usualmente corre en el escritorio y la lógica del negocio es situada cerca del *DBMS* o del usuario, dependiendo de en cual la comunicación se da con mayor frecuencia.

Debido a la familiaridad y sencillez del modelo remoto de administración de datos, las más recientes aplicaciones Cliente/Servidor fueron implementadas en este estilo. Sin embargo, el uso del modelo de lógica distribuida (*message passing*) está creciendo rápidamente porque éste envía menos mensajes y más cortos a través de la red que son requeridos por el administrador remoto de datos para la misma aplicación.

Gartner Group cree que la mitad de todas las nueva aplicaciones de procesamiento cooperativo podrán ser multi-niveles en 1999, un 20% más de lo que existe.

La mayoría de las aplicaciones rutinarias Cliente/Servidor, de pequeño a mediano tamaño, pueden continuar usando los 2 niveles, particularmente el modelo básico de *"data-passing"*, durante los siguientes 5 años. Sin embargo, las aplicaciones que requieren *"message passing"* se pueden mover rápidamente hacia diseños multi-niveles. Las aplicaciones de 2-niveles son fáciles de programar para aplicaciones rutinarias porque ellas solamente usan "procedimientos almacenados" (*stored procedures*) o no usan del todo la lógica del servidor.

Sin embargo, las arquitecturas de 3-niveles ofrecen una amplia opción de modelos de comunicación (ej., encolamientos o publicar-y-suscribir), menor re-uso de aplicaciones, mayor flexibilidad con respecto al modelo de datos, una amplia opción de lenguajes de programación y herramientas de administración ; menor atadura a un vendedor de *DBMS* y mayor escalabilidad. No todos los clientes en un ambiente Cliente/Servidor son computadoras personales en una red local ; los clientes pueden ser dispositivos de puntos de venta, clientes WWW, etc. Estos *front-ends* alternativos no pueden correr aplicaciones cliente de 2-niveles estándar, pero pueden trabajar con aplicaciones de 3-niveles diseñadas adecuadamente.

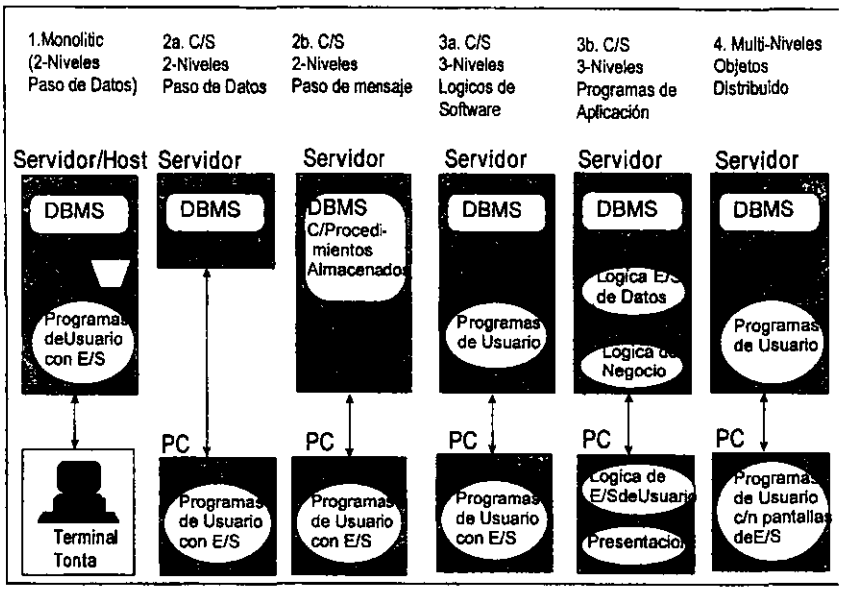



***Fig. 2.8 Evolución de Niveles Lógicos de Software***

Las aplicaciones Cliente/Servidor básicas (columnas 2a y 2b de la fig. 2.8) introdujeron los beneficios de las computadoras personales y de las interfaces gráficas, pero parecían aplicaciones tradicionales, de terminales tontas (columna 1), en su arquitectura lógica de *software*.

Las aplicaciones Cliente/Servidor más sofisticadas (columnas 3a y 3b) son más durables y prolongadas porque usan 3 o más niveles de lógica de *software* (desplegadas sobre dos o mas niveles de *hardware*). Sin embargo, como el *software* se vuelve más complejo, es cada vez más necesario ensamblar aplicaciones usando segmentos de código de múltiples fuentes. De esta manera es cada vez más deseable encontrar mecanismos de vinculación que sean flexibles.

Muchas aplicaciones Cliente/Servidor (columna 4) emplean el cómputo de objetos distribuidos, tecnología de *software* orientada a objetos en el cliente y en el servidor. Las aplicaciones orientadas a objetos están sujetas a las mismas consideraciones de diseño que las aplicaciones Cliente/Servidor tradicionales. Múltiples niveles de *hardware* aún tienen un papel importante; la colocación inteligente de la lógica del *DBMS* en el servidor y la presentación lógica en el cliente todavía es requerida en esta fase; y la idea de niveles de programas de aplicación para aislar la lógica de E/S de los datos de la lógica del negocio todavía es relevante.

## Middleware

El *middleware* es una herramienta crítica para ayudar a los usuarios a construir aplicaciones distribuidas. Existen algunos atributos que pueden ser asociados con el *middleware*. Primero, es un nivel de *software* que interactua entre una aplicación y una red. Segundo, maneja interacciones entre múltiples aplicaciones a través de computadoras multiplataformas.

El *middleware* es una capa de software que soporta múltiples protocolos de comunicaciones, múltiples lenguajes de programación y múltiples plataformas de ejecución. Reside entre las aplicaciones del negocio (capa superior) y la red de plataformas hetereogéneas y protocolos (capa inferior), su importancia radica en que permite separar las aplicaciones del negocio de cualquier dependencia con el sistema operativo, con el *hardware* o con los protocolos de comunicaciones.

Para el desarrollo de aplicaciones, *middleware* da varios beneficios tangibles

- Los desarrolladores corporativos se pueden enfocar en los requerimientos de las aplicaciones del negocio en lugar de tener que desarrollar código para las capas inferiores.
- Las aplicaciones del negocio pueden ser desarrolladas independientemente de los sistemas operativos y protocolos de comunicaciones. Esta independencia trae consigo beneficios tales como la portabilidad de aplicaciones y la protección de las inversiones.
- Los mejores paquetes de aplicaciones pueden ser adquiridos. De esta manera, las decisiones de compra pueden estar basadas en las funcionalidad del *software* en lugar de la capacidad del *software* para correr en un ambiente específico, asumiendo, que la adquisión del paquete ha sido planeada dentro de una arquitectura que contenga una capa de *middleware*.

El *middleware* se pued categorizar dentro de 5 tipos de productos :

1. Database Middleware
2. Message-Oriented Middleware (*MOM*).
3. Portable Transaction Processing (TP) Monitors.
4. Object Request Brokers (*ORB*) y OLE.
5. Remote Procedure Calls (*RPC*).

El *middleware* es un elemento clave para ayudar a integrar ambientes computacionales heterogéneos. Permite a los desarrolladores que construyan aplicaciones sin preocuparse por los ambientes operativos y las redes. Provee un ambiente de sistemas operativos heterogéneos, servicios de mensajeria para aplicaciones basadas en objetos como *CORBA*, soluciones para servicios orientados-a-transacciones que los monitores de transacciones como *Tuxedo*, *Encina*, *Top End* entregan y puede proporcionar un lenguaje *SQL* intermedio común como *ODBC (Open Database Conectivity)*.

También, el *middleware* está dirigido a dos necesidades básicas de interoperabilidad : servicios y datos. Los *middleware* orientados a servicios como *CORBA*, se enfocan a la necesidad de las arquitecturas de sistemas de habilitar la comunicación a través de mensajes entre sistemas que de otra forma no tendrían un lenguaje común. Utilizando *Object Request Brokers (ORBs)* , la cual es la tecnología central de *CORBA*, un sistema puede demandar servicios de objetos de otro sólo conociendo los requerimientos de *los ORBs* , sin importar que no conozca nada acerca de la comunicación a través de mensajes.

Los *middleware* orientados a datos, en contraste, están diseñados para hacer la vida más simple a los administradores de bases de datos que necesitan que uno o varios servidores *DBMS (Database Manager System)* interoperen con una gran variedad de clientes. Usando *ODBC*, la herramienta más común de *middleware* para datos, los arquitectos de sistemas pueden tener desarrolladores trabajando en los clientes y administradores de bases de datos trabajando en los servidores utilzando las interfaces de aplicación de *ODBC* para resolver transparentemente cualquier diferencia entre los formatos de los datos y comandos esperados por cada cliente.

Toda esta funcionalidad intermedia proporcionada por *middleware* permite la integración de arquitecturas de 3 niveles. El esquema de funcionalidad intermedia de 3 niveles básico ("*3 tiers lite*") permite que un programa cliente invoque un programa independiente en un servidor. Cada programa independiente tiene su propia liga al *DBMS*. El *DBMS* coordina la integridad de transacciones y limita las fronteras de integridad transaccional a un único programa independiente. Las herramientas para este tipo de funcionalidad intermedia incluyen *RPCs* y herramientas de mensajeria. La funcionalidad intermedia de los esquemas complejos de 3 niveles ("*3 tiers heavy*") tienen todas las facilidades del esquema básico y en forma adicional expanden las fronteras de integridad transaccional más allá de un programa único, tomando la tarea de coordinar la transacción.

La funcionalidad intermedia en 3 niveles complejos hace esto agregando al ambiente un administrador de transacciones y extendiéndolo a la funcionalidad intermedia para comunicar el contexto de transacciones entre programas independientes. Esta funcionalidad intermedia incluye a los monitores de transacciones como *Tuxedo, Encina*, etc..

Hoy en día, existen tres tipos básicos de *middleware*: *Remote Procedure Calls (RPCs)*, *Message-Queueing Routines* y *Object Request Brokers (ORBs)*.

1) Los *RPCs* (figura 2.9) conforman una de varias intefases de programación incluidas en el protocolo *TCP/IP*. Los productos *TCP/IP* generalmente incluyen librerías de programación para comunicaciones, junto con la interface de *sockets*, que es una interface para comunicaciones punto-a-punto. Los *RPCs* son efectivos en configuraciones Cliente/Servidor o de Proceso Cooperativo. Permiten que una aplicación inicie una función o procedimiento en un sistema remoto.

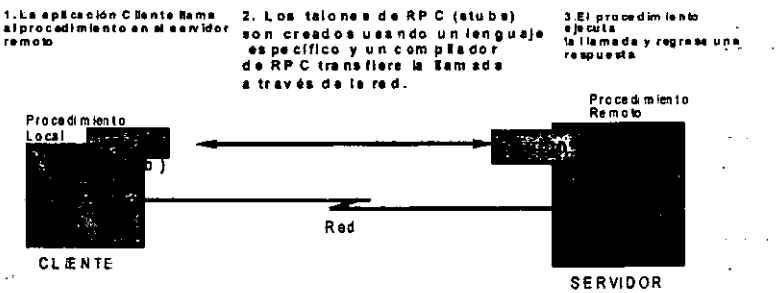



***Fig. 2.9 Llamada a un Procedimiento Remoto (RPC)***

*2) Message-Queueing Routines* (figura 2.10) son recomendables para conectar aplicaciones dispersas que pueden ser operacionales en cualquier punto dentro de la organización. Como su nombre lo implica, los mensajes son almacenados en colas, permitiendo que el procesamiento continúe una vez que la información es colocada en una cola de destino. Este proceso es asíncrono de tipo almacena y envía (*store and forward*), es decir, no se espera por una respuesta.

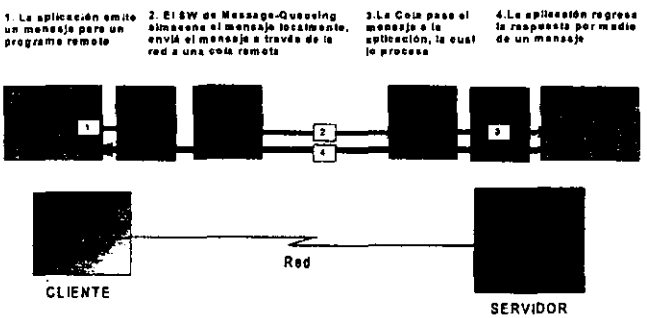



***Fig. 2.10 Message-Queueing Routines***

3) Los *ORBs* son similares al *software* de encolamiento de mensajes, pero son usados en ambientes distribuidos donde se manejan objetos. Los *ORBs* permiten que estos objetos que representan aplicaciones y recursos de cómputo, se comuniquen vía mensajes a través de la red. (figura 2.11).

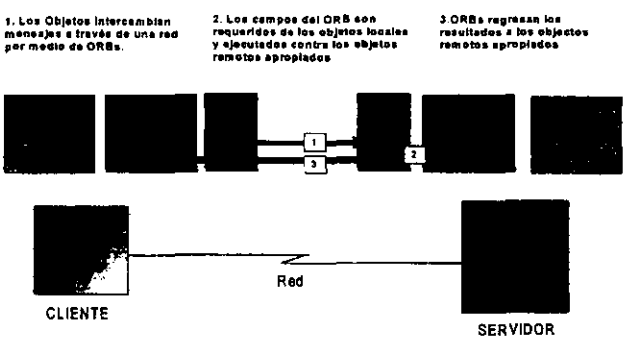



***Fig. 2.11 Object Request Broker***

## 2.3 Evolución de los medios de Telecomunicación

Los principales sistemas de telecomunicaciones empleados para la transmisión a distancia se originaron a finales del siglo XVIII, aunque su aparición definitiva se produjo a finales del siglo XX, como consecuencia del acelerado avance de la electrónica y las ciencias de automatización de sistemas. Entre las que destacan :

• **La telegrafía.** El telégrafo fue el primero y durante muchos años el más importante sistema de telecomunicaciones. Alcanzó su moderna identidad al incorporar los conocimientos sobre transmisión eléctrica de señales que representan letras, números, signos de puntuación y símbolos diversos.

• **La telefonía.** La invención del teléfono desplazó paulatinamente al telégrafo como medio cotidiano de telecomunicación. La telefonía se basa en la conversión del sonido en señal eléctrica, su transmisión inmediata a lo largo de un medio conductor y su transformación de nuevo en sonido en el aparato receptor. Aunque en las últimas décadas del siglo XX extendió su campo de acción hacia la comunicación de imágenes y signos gráficos.

• **La televisión.** Se basa en los mismos principios que la radiotelefonía, con la salvedad de codificar en las frecuencias de las ondas emitidas al espacio elementos de imagen y sonido de forma conjunta. EL medio televisivo adquirió el papel de elemento primordial en la comunicación de informaciones.

Las telecomunicaciones comprenden un conjunto de sistemas, dispositivos y técnicas empleados para la transmisión de información a larga distancia de modo instantáneo. Los principales medios utilizados en estas transmisiones son :

• **La radiocomunicación.** Fue impulsada desde principios del siglo XX y pretende la transmisión del sonido a través de ondas electromagnéticas que acompañan a los campos eléctricos y magnéticos producidos por diversos medios y proyectados hacia el espacio desde una antena emisora sin utilización de cables o hilos conductores.

• **La transmisión por cable.** Se basa en la transferencia de datos a través de un canal de comunicación. Mientras que un canal de transmisión será el camino entre dos nodos de una red. Puede referirse al cable físico, a la señal transmitida por el cable o a un subcanal dentro de una frecuencia portadora.

• **Satélites artificiales.** Los satélites ofrecen prestaciones de comunicación de datos, por lo que reciben señales digitales y analógicas de estaciones terrenas. Todas las señales son transmitidas en una frecuencia portadora. Asimismo, las señales son amplificadas y retransmitidas a la tierra, al cubrir un área geográfica pequeña o bien casi una tercera parte de la superficie terrestre.

Las telecomunicaciones son las nuevas tecnologías de comunicaciones que están dando lugar a la transmisión de grandes cantidades de información que son necesarias para la transmisión de, por ejemplo, audio y video porque son más rápidas y tienen mayor capacidad para la transmisión de mayores anchos de banda en los escritores de los usuarios. Se trata de tecnologías, como *Frame Relay*, *ATM* (*Asynchronus Transfer Mode* ; Modelo de transferencia asíncrona), *ISDN (Integrated Services Digital Network ;* Servicios integrados de red digital).

Este tipo de tecnologías ha hecho posible la interconexión de *Internet* y a su vez la interconexión de millones de usuarios a esta red. El día de hoy, la gran mayoría de los usuarios se interconectan a velocidades de 28.8 Kbps, lo que resulta aceptable para, por ejemplo, transmisiones de video. Sin embargo, en un futuro se verán velocidades cercanas a los 64Kb y 128 Kb a nivel masivo.

La necesidad de comunicar computadoras ha estimulado los avances en redes de comunicación. Por ejemplo, los sistemas para usar más eficientemente los medios de comunicación, han ido de la conmutación de circuitos a la conmutación de paquetes y recientemente a la conmutación de tramas. En el futuro se vislumbra la conmutación de celdas y en conexión de redes de área local, el *SMDS (Switched Multimegabit Data Service* ; Servicio de datos conmutados multimegabit).

Dentro de los sistemas de conmutación de paquetes, X.25 es uno de mayor proliferación y uso, este fue desarrollada para un medio ambiente caracterizado, por los siguientes factores :

- Líneas de comunicación ruidosas.
- Equipos terminales de datos no muy avanzados.

Con esto en mente, en X.25 se realizan las funciones necesarias para garantizar la entrega de un equipo terminal a otro, sin error, y con el control de flujo adecuado.

En los sistemas de comunicación que utilizan fibra óptica- caso de *RDI (Red Digital Integrada)*- y que conectan computadoras con capacidad de proceso apreciable, el medio ambiente es otro. Por ejemplo, el régimen de error en el canal es tan bajo que no es funcional que los equipos de comunicaciones realicen las funciones de control de error y control de flujo. En este ambiente, es aconsejable que esas tareas se dejen a los protocolos, por ejemplo *TCP/IP* que corren en los equipos de cómputo enlazados. Esto trae como consecuencia un aumento en la velocidad de transmisión que se puede manejar, la cual va de 64 y 128 kbps hasta 2.048 Mbps. Este último modo de operación es el que sigue la técnica de conmutación *Frame Relay*. En proceso de investigación o en fase inicial de aplicación, están otras técnicas moderas de conmutación, como *SMDS* o *Cell Relay*.

## *Redes de Comunicaciones*

Las redes de comunicaciones estaban en la era de piedra justo hace una década. La "administración de redes" era un concepto extraño debido a que no existían redes a administrar- justo algunas redes locales experimentales. La mayoría de las conexiones eran entre *mainframes* o minicomputadoras y terminales y las redes de área amplia eran raras. Los sistemas de información hoy en día son diferentes, más del 75% de todas las nuevas computadoras personales de los negocios están conectadas directamente dentro de una red.

La tabla 2.2 detalla el progreso de la tecnología de redes y otros elementos de las nuevas redes corporativas. Muestra el pasado, presente y futuro de las redes de comunicaciones y sus aplicaciones.

| | | |
|---|---|---|
| • Acceso a Hosts vía Terminales<br>• Compartir archivos e Impresoras en Redes Locales. | • Acceso desde Redes Locales hacia mainframes.<br>Cliente/Servidor, Correo Electrónico, Groupware. | Administración integrada de datos, voz y video. |
| • Ethernet<br>• Token Ring<br>• Sincrono | • FDDI<br>• Switched Ethernet<br>• Fast Ethernet<br>• 100-VG Any LAN | • Switched LAN<br>• ATM en redes locales y Backbone |
| • Par Trenzado<br>• Cable Coaxial<br>• Bridges/Puentes | • Sistemas de Cableado Estructurado<br>• Hubs Inteligentes<br>• Ruteadores Multiprotocolo | Fibras ópticas en escritorios<br>• Switches<br>• Sistemas de Cableado Estructurado con categoría 5 |
| • Acceso a Hosts vía Terminales.<br>• Interconexión de Redes. | • Acceso desde Redes Locales hacia mainframes.<br>• Interconexión de Redes<br>Cliente/Servidor, Correo Electrónico, Groupware, VideoConferencia. | Punto-a-Punto<br>• Interconexión de Redes<br>• Multimed a |
| • X.25t<br>• Switched Leased Line T-1<br>• Asincronas vía telefónica | • Frame Relay<br>• Switched Digital (ISDN)<br>SMDS | • BISDN<br>• ATM |
| • Modem<br>• CSU/DSU | • Dispositivos integrados de accese digital<br>• Ruteadores Multiprotocolo | SONET<br>• Switches |

***Tabla 2.2 Pasado, Presente y Futuro de las redes de comunicaciones***

### *Evolución de Redes Locales*

El cableado de par trenzado ligado a través de sistemas inteligentes de cableados utilizado por las redes locales ha eliminado muchos problemas físicos de conectividad, las redes *Ethernet* o *Token Ring* actualmente utilizan predominantemente concentradores *10Base-T* y *Hubs*. Pero esta tecnología está cambiando rápidamente. La congestión de las redes locales está forzando a administradores de redes a impulsar las velocidades de transmisión por encima de los límites estándares de *Ethernet* de 10Mbps. Una manera de alcanzar mayores velocidades es segmentando a usuarios de altas cargas con *Switches Ethernet* o *Token Ring*, dedicando trayectorias de 10 o 16 Mbps a cada usuario o servidor.

Los usuarios están también empezando a incrementar el desempeño sobre redes lentas a través de otras tecnologías, encontrándose entre ellas *100Base-T* o *Fast Ethernet*, que transmite hasta 100Mbps. Varios vendedores venden *Switches Fast Ethernet* que crean enlaces dedicados a 100Mbps entre usuarios y servidores.

Otras tecnologías de redes rápidas a 100Mbps incluyen a la interface *FDDI* y a *100Base-VG AnyLan*. *FDDI* generalmente requiere fibra óptica ; *100Base-VG* corre sobre la mayoría de los cableados de cobre blindados o sin blindar y pronto lo hará sobre fibra óptica.

*ATM* es la alternativa más reciente para crear redes rápidas. En contraste con las antiguas tecnologías de switcheo de paquetes, *ATM* es un nuevo esquema de transmisión de datos designado para mover celdas conteniendo datos, voz y video a altas velocidades. Los pioneros en esta tecnología han instalado redes basadas en *ATMs* que transmiten datos a 155 Mbps.

La práctica de equipar las computadoras personales con procesadores más rápidos e interfaces que soporten los adaptadores de redes rápidas continuará mejorando el desempeño de las redes. Pero eventualmente esta practica llegará a ser obsoleta. La relación costo/beneficio será mayor a adquirir computadoras personales preconfiguradas y los componentes para conectividad integral. El objetivo final será soportar una nueva familia de aplicaciones basadas en multimedia. Se requerirá entonces una red que permita el compartir de una manera segura y eficiente dichas aplicaciones para un crecimiento futuro.

### *Evolución de Redes de Area Amplia (WANs)*

La generalizada y extendida necesidad de comunicar a computadoras entre s a altas velocidades ha propiciado la transición de las redes telefónicas públicas de líneas analógicas a líneas digitales. La rápida comunicación de computadora-a-computadora requiere un nuevo y eficiente nivel de servicios de las redes de área amplia. Los actuales servicios, tales como líneas privadas T1 de 56 kbps y 1.544 Mbps, permiten a los usuarios extender sus redes de datos locales privadas sobre una red de área amplia.

Actualmente, cerca de 1000 compañías usan servicios que requieren grandes anchos de banda. Basados en varias tecnologías, todos estos servicios intentan hacer una sola cosa : incrementar la velocidad con la cual las computadoras y servidores se comunican. Una manera de ver estos servicios es dividirlos por su disponibilidad. Servicios dedicados ofrecen líneas privadas que están disponibles en todo momento. Servicios conmutados (*switched*) ofrecen anchos de banda en demanda. Los servicios de banda ancha incluyen *Frame Relay, ATM e ISDN*, entre otros.

El apoyo para localidades remotas, sucursales de oficinas y videoconferencias también ha incrementado la demanda para mayores y más sofisticados servicios conmutados digitales. Recientes tecnologías de conectividad han satisfecho un poco estas demandas.

El más popular de los nuevos servicios de datos basado en la tecnología "*fast-packet*" es *Frame Relay* , una extensión del estándar de "*packet-switching*" *X.25*, el cual permite aumentar la velocidad de transmisión.

*ISDN (Integrated Services Digital Networks)* permite a las compañías proveedoras de servicios de comunicaciones ("*carriers*") extender los servicios de circuitos-conmutados digitales a una mayor cantidad de empresas. Las redes *ISDN* permiten a las organizaciones ampliar sus redes digitales a todos los sitios remotos. La necesidad de *ISDN* es clara pero su disponibilidad está todavía limitada. La conectividad de corporaciones dispersas basadas en *ISDN* requiere de una disponibilidad extendida. En los próximos años, *ISDN* puede continuamente volverse más rápidamente accesible.

## *FRAME RELAY*

El gran volumen de información intercambiado en un ambiente de negocios basado en aplicaciones Cliente/Servidor requiere que las comunicaciones se realicen a velocidades altas y con un retardo bajo. Las redes locales satisfacen estos requerimientos de comunicación al interior de las empresas proporcionando velocidades que llegan ahora hasta los 100 *Mbps*.

La integración de las diferentes redes locales que existen geográficamente dispersas en las corporaciones puede realizarse utilizando líneas privadas *E1* que proporcionan una velocidad de 2048 *Mbps*. Sin embargo, en muchos casos esta solución no es económicamente factible, sobre todo si se trata de una red con una gran cantidad de redes locales y enlaces de larga distancia que no se ocupan en un porcentaje alto del tiempo, debido a la naturaleza, por ráfagas (intermitente) del tráfico transportado. Surge entonces *Frame Relay* como la alternativa más viable de implementación de redes de transmisión de datos en la presente década.

*Frame Relay* es una forma simplificada de conmutación de paquetes diseñada para trabajar sobre las líneas de transmisión digitales de los 90s, que presentan una baja probabilidad de errores de transmisión. *Frame Relay* aumenta la velocidad de tránsito a través de una red, en comparación a *X.25*, reduciendo el procesamiento efectuado sobre los paquetes en la red. Los nodos de la red (*switches*) actúan sólo como "relevadores" : reciben paquetes y los envían sobre la línea de salida correspondiente, dejando que las estaciones de los usuarios corrijan los errores eventuales que puedan ocurrir en la red.

Las características más importantes de *Frame Relay* son :

- Altas velocidades de transmisión.
- Bajos retardos sobre la red.
- Alta conectividad.
- Uso eficiente del ancho de banda.

Además *Frame Relay* es una tecnología de alta velocidad que ofrece ancho de banda sobre demanda y que permite multiplexar estadísticamente diferentes circuitos virtuales sobre un mismo enlace de acceso a la red. La existencia de caminos redundantes en las redes públicas *Frame Relay* y el uso de protocolos de enrutamiento dinámicos, como *OSPF* (*Open Shortest Path First*), proporcionan una alta disponibilidad de la red. Todas estas características hacen de *Frame Relay* la tecnología más adecuada en términos de velocidad, costos y disponibilidad para las empresas con un gran número de aplicaciones. Uno de los principales usos de las redes públicas *Frame Relay* será sin duda el transporte del tráfico de Internet.

La demanda por servicios públicos de *Frame Relay* es muy grande en todo el mundo, siendo actualmente la tecnología más usada en las redes de área amplia. Como un ejemplo del fuerte crecimiento experimentado por *Frame Relay* desde sus inicios podemos citar que en Japón había un solo proveedor del servicio en 1992 y que para fines de 1995 ya existían 23 [3]. Según una encuesta realizada entre Marzo y Mayo de 1996, el crecimiento de *Frame Relay* fue de aproximadamente 300% entre 1995 y 1996[4]. De acuerdo a esta encuesta :

- Todos los proveedores de servicios *Frame Relay* ofrecen puertos de acceso de 64 *Kbps*, *E1* fraccional y múltiplos de *E 1*, y algunos ofrecen puertos de velocidades menores de 64 *Kbps*.
- Los *Backbone* de las redes que ofrecen el servicio *Frame Relay* emplean en su mayoría tecnología *Frame Relay* internamente, pero algunas ya utilizan *ATM*.
- Las topologías lógicas de redes privadas virtuales más utilizadas son (en orden decreciente) : estrella, malla parcial, malla completa y punto-a-punto.
- Las dos aplicaciones que más utilizan *Frame Relay* en la actualidad son la interconexión de redes locales y el acceso a Internet.

### *ATM*

*Frame Relay* es, actualmente, la tecnología más utilizada para la implementación de *Backbones* de las redes públicas. Aunque no está realmente diseñada para ello, su funcionalidad y amplia difusión la han llevado a esa posición. Aunque *Frame Relay* se ha comportado hasta ahora de modo satisfactorio, ya empieza ha sentirse la necesidad de transmitir conjuntamente datos, voz y video y las redes públicas comienzan a mirar hacia el futuro y hacia nuevas tecnologías. *ATM* a sido escogido por el mundo de las telecomunicaciones como la tecnología universal. *ATM* está diseñado para implementar tanto redes locales como *Backbones* de redes públicas, funciona en un amplio rango de velocidades (1.5 a 622 *Mbps*), puede utilizar variados medios de transmisión (par trenzado, cable coaxial y fibra óptica) y soporta el transporte de datos, voz y video de manera natural sobre una misma red. El desarrollo de *ATM* prevee su interoperabilidad con los protocolos más utilizados actualmente. En particular, se han publicado normas que permiten interconectar los equipos actuales del usuario y las redes *Frame Relay* con tecnología *ATM*.

### *Comunicación Satelital*

[3] Tecnologías de Area Amplia para la Comunicación de Datos Revista Soluciones Avanzadas, Diciembre 1996

[4] 1996 Market Survey by Steven Taylor ; Distributed Networking Associates

http ://www.frforum.com/4000/4013.html

Varios de los primeros satélites de comunicaciones fueron diseñados para operar en un modo pasivo. En lugar de transmitir activamente señales de radio, ellos sirvieron sólo para reflejar señales que fueron dirigidas hacia ellos por estaciones transmisoras terrestres. Las señales fueron reflejadas en todas las direcciones, así pudieron ser recogidas por estaciones receptoras alrededor de todo el mundo. La capacidad de tales sistemas estaba severamente limitada por la carencia de poderosos transmisores y grandes antenas terrestres.

Los satélites de comunicaciones actualmente hacen uso exclusivo de sistemas activos, en los cuales cada satélite lleva su propio equipo para recepción y transmisión. *Score*, fue el primer satélite de comunicación activa, fue lanzado por los Estados Unidos en 1958. Fue equipado con una grabadora que almacenaba los mensajes recibidos mientras pasaba sobre una estación transmitora terrestre. Estos mensajes eran retransmitidos cuando el satélite pasaba sobre una estación receptora. Cientos de satélites de comunicación activa existen hoy en día en órbita. Ellos reciben señales de una estación terrestre, las amplifican, y entonces las retransmiten en una diferente frecuencia hacia otra estación.

El despliegue y operación de satélites de comunicación para uso comercial comenzó con la fundación de la Corporación de Satélites Comerciales (*COMSAT*, por sus siglas en inglés) en 1963. En 1963 fue formada la Organización Internacional de Satélites de Telecomunicaciones (*INTELSAT*, por sus siglas en inglés). El satélite *Intelsat 1* fue lanzado en 1965, proporcionando ya sea 2400 circuitos de voz o un canal de televisión de dos vías entre los E.U. y Europa. Durante los 60s y 70s, la capacidad de mensajes y el poder de transmisión de los satélites *Intelsat 2, 3 y 4* fue incrementándose progresivamente, dirigiendo el poder del satélite sólo hacia la tierra y segmentando el espectro de transmisión dentro de unidades de transpondedores de cierto ancho de banda. El primero de los satélites *Intelsat 4* , lanzado en 1971, proporcionaba 4000 circuitos de voz. Con el *Intelsat 5* de 1980 la introducción de la operación de múltiples emisiones proporcionó incrementos adicionales en capacidad. El poder de un satélite ahora puede ser concentrado en pequeñas regiones de la Tierra, siendo posible aperturas más pequeñas, estaciones terrestres de menor costo. Un satélite *Intelsat 5* puede soportar 12000 circuitos de voz. Los satélites *Intelsat 6* que entraron en operación en 1989, pueden soportar 24000 circuitos y cuentan con la característica de switcheo dinámico de la capacidad telefónica entre 6 emisiones, usando una técnica llamada *SS-TDMA* (*Satellite Switched Time Division Multiple Access*). Para principios de los 90s, *INTELSAT* tenía 15 satélites en órbita proporcionando el sistema de telecomunicaciones más extenso del mundo. Otros sistemas también proporcionan servicios internacionales en competencia con *INTELSAT*.

Los satélites comerciales proveen una amplia gama de servicios de comunicaciones. Los programas de televisión son transmitidos internacionalmente, dando lugar al fenómeno conocido como "aldea global". Los satélites también transmiten programas para sistemas de televisión por cable así como a casas equipadas con antenas parabólicas.

Además, las *VSATs (Very Small Aperture Terminals)* transmiten datos digitales para una multitud de servicios de negocios. Los satélites *Intelsat* ahora transportan cerca de 100,000 circuitos telefónicos, con el creciente uso de transmisiones digitales. Los métodos de codificación digital han dado por resultado una reducción de 10 veces la frecuencia de transmisión para transportar un canal de voz, mejorando de esta manera la capacidad de las facilidades existentes y reduciendo el tamaño de las estaciones terrestres que proporcionan los servicios terrestres.

Los sistemas de comunicación satelital han entrado en un período de transición de una troncal de comunicaciones de alta capacidad punto-a-punto entre grandes y costosas terminales terrestres a comunicaciones multipunto-a-multipunto entre estaciones pequeñas y a bajo costo. El desarrollo de métodos de múltiple acceso han acelerado y facilitado esta transición. Con *TDMA*, cada estación terrestre es asignada a una ranura en el tiempo en el mismo canal para utilizarse en la transmisión de comunicaciones ; las otras estaciones monitorean estas ranuras y seleccionan las comunicaciones dirigidas hacia ellas. Amplificando una frecuencia de portadora sencilla en cada satélite repetidor , *TDMA* asegura el uso más eficiente del poder de los satélites.

Una técnica llamada reuso de frecuencia permite a los satélites comunicarse con un número estaciones terrestres usando la misma frecuencia para transmitir amplios haces apuntando hacia cada una de las estaciones. El ancho de los haces puede ser ajustado para cubrir áreas tan pequeñas o grandes como se quiera. Dos estaciones lo suficientemente apartadas pueden recibir diferentes mensajes transmitidos en la misma frecuencia. Las antenas satelitales han sido diseñadas para transmitir varios haces en diferentes direcciones, usando el mismo reflector.

## 2.4 TCP/IP como elemento clave para Internet

*TCP/IP* es una colección de protocolos de red, los cuales conjuntamente proporcionan comunicación "máquina a máquina" para equipos conectados a redes heterogéneas. *TCP/IP* es referido como una colección de protocolos de *Internet*, debido a que es lo suficientemente poderoso para permitir que máquinas conectadas en diferentes tipos de redes se puedan comunicar entre sí.

*TCP/IP* e *Internet* están intrínsecamente ligadas en origen, uso actual, y desarrollos futuros. Los lazos no son sólo técnicos, sino también políticos y culturales. Estos vínculos pueden fortalecerse, en tanto gobiernos y vendedores de telecomunicaciones la vean dentro del marco de globalización que se está dando en el mundo, con el fin de apoyar al desarrollo educativo, cultural y económico de los países. Dentro de la "era de la información" en que estamos viviendo, el desarrollo de la "supercarretera de la información" que se está dando, gracias a *Internet* y *TCP/IP*, permitirá que la mayoría de la población tenga acceso a un sinfín de fuentes de información.

Por otra parte, *Internet* está ganando reconocimiento en la comunidad financiera y de negocios, debido a que las estrategias de negocios se están moviendo para ajustarse a las interredes de sucursales ampliamente distribuidas y de los escenarios de cómputo Cliente/Servidor, los servicios de *Internet* se están volviendo de gran ayuda para las grandes corporaciones para proporcionar un mayor número de servicios e información a sus clientes así como a sus ejecutivos y mejorar la comunicación interna de la empresa.

En 1969, el Departamento de Defensa de E. U. construyó una red de cuatro nodos de área amplia llamada *ARPANET* como el primer experimento para demostrar la factibilidad de la tecnología de conmutación de paquetes. Este experimento altamente exitoso fue demostrado en 1972, para ese entonces la red incluía muchos sitios de universidades cuyas máquinas tenían implementados una serie de protocolos para comunicar máquinas entre sí.

Ese mismo año se comenzó a trabajar en una segunda generación de protocolos diseñados para hacer uso del conocimiento obtenido de los experimentos originales. En 1982, una nueva familia de protocolos había sido especificada , implementada y sometida a una extensiva experimentación. Los principales miembros de esta familia son el *Transmision Control Protocol (TCP)* y el *Internet* Protocol (*IP*).

El éxito de *ARPANET* y de *TCP/IP* es atribuido a dos factores : la habilidad del protocolo de *TCP/IP* de ligar equipos de cómputo heterogéneos y una política administrativa abierta y flexible que alentaron a los usuarios a criticar y ayudar a resolver el rendimiento de la red. La fortaleza de *TCP/IP* es su arquitectura abierta y su compatibilidad con las capas bajas de casi cualquier infraestructura de red.

### *Arquitectura del Protocolo de TCP/IP*

Como se ha mencionado, el término *TCP/IP* se refiere a una gran familia de protocolos y servicios. Varios de los protocolos son mostrados en la figura 2.12. Además, los servicios del conjunto de protocolos y servicios de *TCP/IP* pueden ser identificados y acomodados de acuerdo al modelo de siete niveles de *Open Systems Interconections*(*OSI*), ver figura 2.13, establecido por la *International Standard Organization* (*ISO*). Este modelo de referencia describe las funciones proporcionadas por cualquier sistema de red en términos de capas - donde cada capa se construye sobre otras inferiores. El modelo de referencia es, de hecho, una especificación de servicios de comunicaciones , donde cada capa proporciona un tipo particular de servicio a la capa inmediata superior y espera un tipo particular de servicio de la capa inferior.

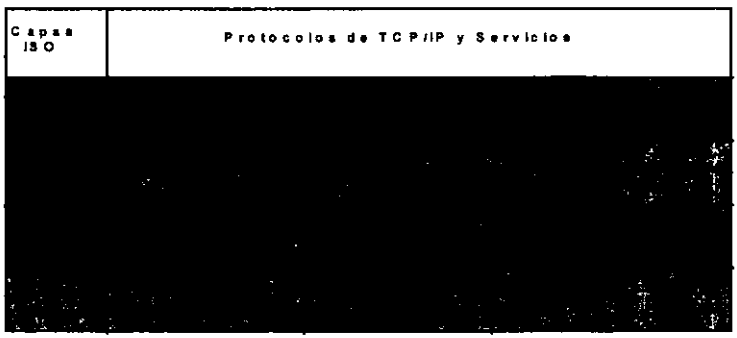



***Fig. 2.12 Protocolos de TCP/IP***

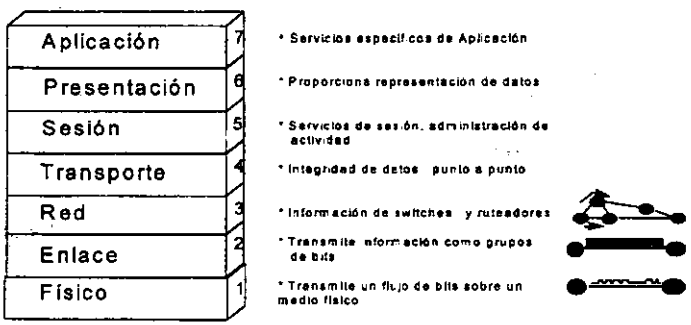



***Fig. 2.13 Modelo OSI***

El protocolo y los servicios de *TCP/IP* se engloban, de acuerdo al modelo *OSI*, en las siguientes cuatro capas (ver. Figura 2.14) :

- La capa de Red [*Network Layer*], proporciona funciones para la entrega de datos a otros dispositivos conectados directamente en una red. Estas incluyen encapsulamiento de "datagramas" de *IP* dentro de tramas[*frames*] transmitidos por la red y mapeo de direcciones *IP* a direcciones físicas usadas por la red. Estos servicios corresponden a funciones de la capa 2 del modelo *OSI*.
- La capa de *Internet* [*Internet Layer*], contiene el Protocolo *Internet* (*IP*), el cual proporciona los servicios básicos de entrega de paquetes para la red *TCP/IP*. Todos los otros protocolos usan *IP* para entrega de datos. Esta capa también contiene al *Internet Control Message Protocol* (*ICMP*) usado para realizar funciones de control, reporte de errores y de información para *TCP/IP*. La capa *Internet* traduce direcciones *IP* a direcciones *Ethernet*. El protocolo que realiza esta función es el *Address Resolution Protocol* (*ARP*). *IP* y los protocolos de mayor nivel pueden ser implementados en varios tipos de redes físicas, como por ejemplo *Ethernet, X.25, Token Ring*.

  El Protocolo *Internet* (*IP*) implementa funciones correspondientes a la capa 3 del modelo *OSI* : la Capa de Red. Este protocolo maneja el ruteo de datos a través de una red, la cual típicamente consiste de muchas y diferentes (heterogéneas) subredes. *IP* no es un protocolo "orientado a conexión"; éste rutea los datos desde una dirección origen a una dirección destino; cada mensaje contiene la información requerida para localizar el nodo destino.
- La capa de Transporte [*Transport Layer*] contiene a los protocolos *Transmision Control Protocol* (*TCP*) y *User Datagram Protocol* (*UDP*).

  ⇒ *TCP* implementa funciones correspondientes a la capa 4 del modelo *OSI* : la Capa de Transporte. Este protocolo es orientado a conexión y proporciona un intercambio confiable de datos entre un sistema receptor y un transmisor, sin importar sobre cuantos nodos intermedios viajan los datos, garantiza que todos los datos enviados pueden ser recibidos por el sistema destino y pueden llegar en el orden en el cual fueron enviados. Es usado por aplicaciones que requieren una transferencia de datos punto a punto confiable. Es un protocolo orientado a flujos de *bytes* ("*stream-oriented*") sin el concepto de delimitadores de paquetes.

  ⇒ *UDP* suministra servicios no-confiables de "datagramas". La integridad de los paquetes enviados es mantenida ; cuando un paquete es recibido, garantiza la exacta igualación de lo que fue enviado. Sin embargo, la entrega de los paquetes no es garantizada, y no hay garantía del orden en el cual los "datagramas" son recibidos. *UDP* proporciona un servicio de entrega de datagramas sin conexión.

- La capa de Aplicación [*Aplication Layer*] contiene a todos los servicios de aplicación que usan a la capa de transporte para la entrega de datos, que corresponden a las funciones de las capas 5,6,7 del Modelo *OSI*. Algunos de los servicios principales son :

  ⇒ ***SMTP*** : *Simple Mail Transfer Protocol* - Proporciona servicios para el envío de correos tipo texto entre computadoras.

  ⇒ ***DNS*** : *Domain Name Server* - Proporciona un método para nombramiento de recursos. La función básica de *DNS* es suministrar información de objetos de la red respondiendo a consultas. La información proporcionada por *DNS* incluye :

  ⇒ ***FTP*** : *File Transfer Protocol* - Está diseñado para facilitar la transferencia de archivos entre computadoras. Los archivos son transferidos sobre una conexión *TCP*, provee al usuario de un método estándar para controlar las funciones de transferencia de datos, establecimiento de la conexión, administración de la conexión, y modos de transmisión de datos.

  ⇒ ***TFTP*** : *Trivial File Transfer Protocol* - Este protocolo permite a un usuario recuperar o almacenar archivos de lectura/escritura de una máquina remota. *TFTP* es un protocolo orientado a "no-conexión" y no utiliza mecanismos para autentificación del usuario.

  ⇒ ***TELNET*** : *Telecommunication Network* - Este protocolo proporciona una comunicación bidireccional orientada a *bytes* que permite una interface entre terminales y procesos. El protocolo *Telnet* es utilizado sobre una conexión *TCP* que proporciona funciones de control que permiten que cualquier extremo de la conexión pueda modificar las características de la misma en forma dinámica.

La mayoría de los servicios de TCP/IP se mencionan a detalle en el Capítulo 1.

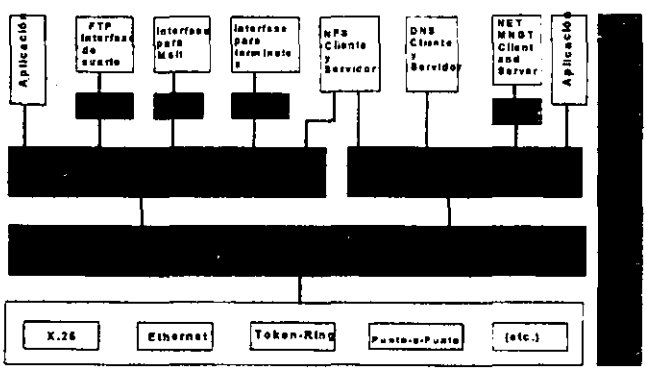



***Fig. 2.14 Arquitectura del Protocolo de TCP/IP***

### *Direccionamiento IP*

Como se ha mencionado uno de los objetivos de *IP* es proporcionar servicios básicos de entrega de paquetes en una gran variedad de redes y ambientes interredes bajo el protocolo de *TCP/IP*, para lo requiere que cada computadora que esté conectada a la red o interred cuente con un domicilio *TCP/IP*. El mecanismo de direccionamiento ha sido diseñado para permitir diferentes clases de configuración de redes, dependiendo del número de computadoras en la red y su distribución.

Un domicilio *TCP/IP* está compuesto por dos niveles de información. En el nivel *TCP* del protocolo *TCP/IP*, el domicilio es llamado *port_id* . Este domicilio identifica al proceso o servicio de aplicación que corre en el *mainframe* o servidor y consiste de una dirección de *Internet* de 32 bits (para identificar el nodo) y un número de 16 bits para identificar el proceso aplicativo en si mismo. *IP* maneja la información de ruteo entre sistemas terminales y sistemas intermedios en la red. En otras palabras, maneja el tráfico entre computadoras y ruteadores. Los domicilios que identifican a los *Hosts* y a los sistemas terminales son manejados por *IP* y son llamados domicilios de *Internet*.

Los domicilios *IP* contienen 4 bytes de información de ruteo, que consisten de un identificador (*ID*) de red y uno de *Host*. El primero es usado para rutear información a través de la red ; el segundo identifica la computadora en particular en la red destino. Un ejemplo sería el siguiente, en donde un punto separa a cada elemento : 192.127.10.1.

En este caso, el domicilio de red *IP* consiste de 192.127.10 y el domicilio *IP* del *Host* es 1. Todos los *Hosts* en la misma red deben tener el mismo número de red pero no pueden tener el mismo número de *Host*.

Los ruteadores también deben tener su domicilio de *Internet*. Todas las redes que interactuan con un ruteador deben tener número de red diferente.

Un domicilio de *Internet* se puede obtener del Centro de Información de la Red (*NIC*, por su siglas en Inglés) que es el repositorio central de información de *Internet* y es responsable de generar domicilios para entidades comerciales y gubernamentales. *NIC* también proporciona domicilios OSI para la milicia estadounidense.

Se han definido cinco clases diferentes de domicilios *TCP/IP* : categorías A,B,C,D y E. La diferencia estriba en el número de bits usados para identificar al *Host* y el número de bits usados para identificar a la red. El formato del domicilio de cada clase está indicado por los primeros tres bits del domicilio.

Las organizaciones con redes privadas *TCP/IP* generalmente se apegan a los formatos de dirección usados en las clases A, B y C. Incluso pueden pedirle a *NIC* un bloque de domicilios de red a fin de garantizar que sus domicilios van a ser realmente únicos. El tipo de dirección que *NIC* asigna depende del tamaño de la red de la organización.

Las direcciones *TCP/IP* son clasificadas en cinco grupos establecidas por *NIC* :

- Clase A, inicia con un número entre 0 y 127. Estas direcciones son adecuadas para grandes redes con muchos host. Los bits 1-7 identifican el segmento de red ; los bits 8-31 identifican a un host en la red.
- Clase B, Inicia con un número entre 128 y 191 ; los bits 2-15 identifican la red ; los bits 16-33 identifican un host en la red. Sólo 16384 direcciones están disponibles.
- Clase C, inicia con un número entre 192 y 223. Existen 2,097,152 posibles direcciones en esta clase, la cual es la más adecuada para ambientes de redes locales.
- Clase D, inicia entre 224 y 239. Estas clases de direcciones son usadas para multicasting *IP*, lo cual facilita la distribución de un mensaje sencillo a un grupo de sistemas dispersos a través de la *Internet.*
- Clase E, inicia con un número entre 240 y 255. Estas direcciones son usadas para propósitos experimentales y para uso futuro.

# CAPITULO III

# Seguridad en Internet

Uno de los obstáculos más grandes para alcanzar el potencial completo de *Internet* ha sido el aspecto de seguridad. El temor a espionaje corporativo, el robo de identidades personales y crédito, la ausencia de privacia en la correspondencia y el acceso potencial a registros confidenciales, ha generado una gran cautela en el uso de transacciones en línea. Algunos de estos temores son justificados, otros, en cambio, se han amplificado desproporcionadamente.

Muchos negocios se han mostrado muy cautelosos para integrarse a la Red, debido a que les implica adecuar sus prácticas y procesos para adaptarse a este nuevo esquema de comunicarse con proveedores, clientes y bancos. Es precisamente la integración del sistema bancario a la Red lo que liberará el verdadero potencial de *Internet* para convertirse en el medio global para el Comercio Electrónico. Sin embargo, existe un factor fundamental que preocupa a la comunidad empresarial : la seguridad. La mayoría de las empresas se encuentran expectantes ante la evolución de los mecanismos de seguridad sobre la Red cuando se trata de hablar de Comercio Electrónico.

Dentro de *Internet* existen varios tipos de redes, las cuales tienen diferentes requerimientos de seguridad. Existen redes para llevar a cabo la administración gubernamental, la investigación militar, la investigación académica y para negocios comerciales. Todas éstas tienen subredes. La seguridad en cada punto de acceso es en varios casos crucial y en otras instancias innecesaria.

El concepto de seguridad se vuelve más complejo conforme los diferentes grupos de usuarios requieren diferentes características del *Internet*. Cada país que usa *Internet* también tiene su propia percepción en relación a tópicos de seguridad. Aunque los productos de seguridad ofrecen mecanismos de encripción, con frecuencia se encuentran países con sus propias especificaciones que obligan a los fabricantes a modificar sus productos. El Instituto Nacional de Estándares y Tecnología *(NIST*, por sus siglas en inglés) está trabajando con la comunidad europea para estandarizar la certificación de productos de seguridad.

En la comunidad empresarial, en donde el *Internet* ha cobrado credibilidad recientemente como un recurso de red confiable, éste sirve a varios propósitos ; en negocios pequeños es usado para fines de correo electrónico ; compañías más grandes lo utilizan para comunicar sucursales dispersas geográficamente y las grandes corporaciones lo escogen como una opción para transmisiones de red a alta velocidad.

## 3.1 Niveles de Seguridad en torno a la Información

De acuerdo a los estándares de seguridad en computación, desarrollados por el Departamento de Defensa de los Estados Unidos, el llamado Libro Anaranjado (*Trusted Computer Evaluation Criteria - Orange Book*), se requieren varios niveles de seguridad para proteger el *hardware*, *software* y la información almacenada en dispositivos electrónicos, contra un posible ataque. Todos estos niveles describen diferentes tipos de seguridad física, autentificación de usuarios, confiablidad del *software* de sistema operativo y sus aplicaciones. Estos estándares también establecen límites en relación a qué tipos de sistemas pueden ser conectados al sistema que se quiere asegurar.

### *Nivel D1*

El Nivel D1 es la forma de seguridad más baja que existe. Este estándar establece que todo el sistema es "inconfiable". En este nivel no existe protección para el *hardware*; el sistema operativo es muy vulnerable y no existe autentificación de los usuarios que deseen ingresar al sistema ni facultades asociadas a éstos en relación a la información que puedan accesar. Este nivel de seguridad se encuentra típicamente en sistemas operativos como *MS-DOS*, *MS-Windows* y el *Sistema 7.x* de *Macintosh*.

Estos sistemas operativos no distinguen entre usuarios y no tienen un método definido para determinar quién está tecleando. Asi mismo, carecen de controles acerca de qué información, de la que albergan en el disco duro, puede ser accesada.

### *Nivel C1*

El Nivel C tiene dos subniveles de seguridad: C1 y C2. El Nivel C1 o sistema de protección de seguridad selectiva (*Discretionary Security Protection System*), describe la seguridad disponible en un típico sistema operativo *Unix*. Existe un grado de protección del *hardware* porque éste no puede ser fácilmente comprometido, aunque ésto sigue siendo posible. Los usuarios deben identificarse ante el sistema operativo a través de un nombre de usuario (*login name*) y una clave confidencial (*password*). Esta combinación se utiliza para determinar los derechos de acceso a programas e información que cada usuario tiene.

Estos derechos de acceso son los permisos sobre directorios y archivos. Este Control de Acceso a Directorios, le permite al propietario de un archivo o un directorio, o al administrador del sistema, prevenir que cierta persona o grupo de personas accesen dicho archivo o directorio. Sin embargo, la cuenta de Administrador del Sistema, no tiene ninguna restricción, por lo que cualquier administrador sin escrúpulos puede fácilmente comprometer la seguridad del sistema sin que nadie se entere.

### *Nivel C2*

El segundo subnivel, C2, fué creado para resolver el conflicto citado en el párrafo anterior. Junto con las características propias de C1, el Nivel C2 incluye características de seguridad adicionales que crean un ambiente de acceso controlado. Este ambiente tiene la capacidad de restringir a usuarios de ejecutar ciertos comandos o accesar ciertos archivos, basándose no sólo en los permisos que tenga el usuario, sino en niveles de autorización. Además, este nivel de seguridad requiere que el sistema sea auditado. Esto implica el escribir un registro de auditoría para cada evento que ocurra en el sistema.

La auditoría es usada para conservar registros de todos los eventos relacionados con seguridad, como aquellas actividades realizadas por el administrador del sistema. La auditoría requiere autentificación adicional, ya que sin ella no hay manera de estar seguros de que la persona que está escribiendo un comando en realidad es quien dice ser. La desventaja de este método es que requiere recursos adicionales de procesamiento y almacenaje de datos.

Con el uso de autorizaciones adicionales, es posible que algunos usuarios realicen actividades de administración del sistema, sin necesidad de contar con la cuenta del Super-usuario (*root* en sistemas *Unix*). Esto permite darle un mejor seguimiento a las actividades relacionadas con la administración del sistema, al conocer exactamente el usuario que las realizó.

### *Nivel B1*

El Nivel B1, o Protección de Seguridad Etiquetada (*Labeled Security Protection*), es el primer nivel que soporta seguridad multinivel, tales como secreto y supersecreto. Este nivel establece que un objeto bajo un nivel de acceso mandatorio no puede sufrir cambios en sus permisos, a través del propietario del archivo.

### *Nivel B2*

El Nivel B2, o Protección Estructurada (*Structured Protection*), requiere que todos los objetos sean etiquetados. Los dispositivos como discos, cintas y terminales pueden tener asignado un nivel de seguridad sencillo o múltiple. Este nivel es el primero que enfoca el problema de un objeto que se encuentre en un nivel alto de seguridad, queriendo comunicarse con otro objeto en un nivel más bajo de seguridad.

### *Nivel B3*

El Nivel B3, o Nivel de Dominios de Seguridad, obliga la existencia de un dominio de seguridad con la instalación de *hardware*. Por ejemplo, se puede utilizar *hardware* de administración de memoria, para evitar que el dominio de seguridad dentro de la memoria, sea accesada sin autorización o modificada por objetos en diferentes dominios de seguridad. Este nivel también requiere que la terminal o estación del usuario esté conectada a un enlace confiable.

### *Nivel A*

El Nivel A, o Nivel de Diseño Verificado, es el nivel más alto de seguridad validado por el Libro Anaranjado. Incluye un proceso muy estricto de diseño, control y verificación. Para que este nivel se pueda alcanzar, todos los elementos de los niveles anteriores deben ser incluidos; el diseño debe ser verificado matemáticamente y debe realizarse un análisis de los canales de cobertura y distribución confiable, entendiéndose por distribución confiable el que tanto el *hardware* como el *software* hayan sido protegidos contra intrusiones durante su embarque.

## 3.2 Elementos de Seguridad

Proporcionar seguridad efectiva en un ambiente computacional de usuario final resulta ser todo un reto. Primeramente debe definirse qué es lo que se entiende por seguridad, y después de esto definir los servicios que son requeridos para cumplir con las expectativas gerenciales en torno a seguridad. La seguridad debe ser analizada en términos de una arquitectura basada en calidad. Para alcanzar la calidad, deben proveerse elementos de continuidad, confidencialidad e integridad. La confidencialidad, en lo que respecta a calidad, puede ser definida como control de acceso. Esto incluye un proceso de autorización, la autentificación de usuarios, capacidad de administración y auditabilidad. Este ultimo elemento, la auditabilidad, se extiende más allá de la definición tradicional del término, para abarcar la habilidad de administración para detectar circunstancias inusuales no autorizadas, trazando un historial de estos eventos. La integridad, otro elemento más de calidad, involucra los componentes usuales de validación y precisión, pero incluye además la contabilización individualizada. Toda implementación de sistemas de Seguridad Informática requiere cumplir con estos elementos de calidad de alguna manera.

Existen tres niveles básicos de actividades "en-línea" que requieren de seguridad en el medio de comunicación : comunicaciones generales, comunicaciones de negocios y transacciones financieras.

La más simple aplicación es comunicación en general, tal como correo electrónico privado o acceso limitado a datos en una página *Web*. Un ejemplo puede ser una situación donde tanto receptor como transmisor (o cliente y servidor) conozcan qué datos están siendo enviados y recibidos, pero ellos preferirían no compartir esta información con otros. En situaciones como esta un sistema de *passwords* es empleado para asegurar que el acceso está siendo autentificado. La autentificación es el proceso por el cual el receptor de un mensaje puede estar seguro que el transmisor es quien él afirma ser.

En comunicaciones de negocio, la autentificación e integridad del mensaje -así como la privacía- pueden ser sumamente críticos. Por ejemplo, un anuncio público de un descuento en las tarifas telefónicas de larga distancia es sólo significante si ésta es valida. La *Internet* ya ha experimentado muchos comunicados de prensa falsos, así como anuncios falsos que no son imaginables.

Las transacciones financieras requieren de medidas de seguridad adicionales. Obviamente, la autentificación es un asunto importante para los usuarios de tarjetas de crédito. Desde el punto de vista de los negocios, es esencial que a los clientes no les rechazen transacciones validas. Los negocios deben estar seguros que los clientes puedan pagar las mercancías antes de enviarlas. Muchas transacciones financieras, tales como las que utilizan tarjetas bancarias, involucran a más de dos partes- el negocio y el tarjetahabiente. Cada parte necesita algunos datos, pero al menos solo una parte no puede tener acceso a todos los datos.

### *Seguridad en redes y en sistemas*

Entrar a *Internet* resulta relativamente simple. El problema con el acceso a *Internet* reside en resguardar las redes departamentales internas así como los sistemas de negocio. Los delincuentes cibernéticos han tenido mucho éxito para tener acceso a datos confidenciales de las compañías que se encuentran públicamente expuestas en *Internet*, al aprovecharse de los huecos de seguridad de éstas. Esto ha ocasionado que muchas compañías se estén dedicando a desarrollar productos que representen soluciones de seguridad.

El concepto general de implementar seguridad en las redes consiste en filtrar todos los accesos a las redes internas a través de la implementación de un *firewall* (decrito más adelante), que permiten el acceso sólo a ciertos servicios predefinidos, como lo puede ser el Correo Electrónico.

Sin embargo, el acceso directo de una persona indeseable a un nodo a través de una conexión de *Internet* no es el único riesgo de seguridad que deba contemplarse, especialmente si hay puntos de acceso remoto a la red. Debido a que un atacante puede tener acceso a través de un modem y atacar a lo que se considera un *firewall* perfectamente seguro, las corporaciones deben empezar por establecer políticas de seguridad para la red completa (incluyendo contraseñas, acceso seguro a *modems*, autentificación y acceso físico a los equipos) y no sólo para un solo punto de acceso.

La mayoría de las personas que se comunican sobre una red asumen que sus datos están seguros ; que todo lo que envían llega intacto y sólo al destino deseado. Esto es usualmente verdadero, pero no siempre.

La falta de seguridad es un problema crítico cuando se está manejando información confidencial y en especial cuando se está realizando algún tipo de comercio. Para crear una red segura se utilizan tres estrategias principales :

1. Encripción - Revolver los datos de modo que no los puedan utilizar si los datos son interceptados.
2. Integridad - Confianza de que los datos recibidos no han sido modificados en el viaje de los mismos.
3. Autentificación - Verificar el origen del mensaje y la identidad del transmisor.

Es vital que las instituciones financieras entiendan estos riesgos y tomen medidas para manejarlos adecuadamente.

### *Encripción*

La encripción es un componente principal en la seguridad de redes. Encriptar es revolver o transformar la información en algo incoherente. Por lo tanto, si alguien es capaz de romper las demás características de seguridad, la información obtenida puede ser indescifrable.

Esta tecnología puede asegurar que un mensaje es recibido tal como fue enviado, y puede confirmar la autenticidad de las partes transmisoras y receptoras. La encripción también puede crear una firma digital, la cual es la equivalencia electrónica de la firma escrita. Las firmas digitales proporcionan un alto grado de autentificación, ofreciendo una prueba al transmisor de que el mensaje fue entregado.

Otro objetivo de la encripción es la privacidad : la confianza de que la información que se envía no pueda ser entendida excepto por el receptor especificado o deseado.

Existen dos tipos básicos de encripción : encripción por llave privada y encripción por llave pública. En el de llave privada, una llave sencilla es usada para encriptar (codificar) así como para decriptar (decodificar). Una de las tecnologías mejor conocidas, de llave privada, es *DES (Data Encryption Standar)*, el estandar del gobierno de Estados Unidos para proteger información sensitiva pero sin clasificación. Este también ha sido ampliamente utilizado por organizaciones del sector privado.

En la encripción con llave pública, ésta se efectua por medio de una llave pública conocida y una llave privada secreta. La autentificación se lleva a cabo añadiendo la llave privada del transmisor al mensaje. Una vez recibido, el receptor puede verificar el origen del mensaje usando la llave pública del transmisor para decriptar el mensaje.

La encripción es aplicada en la seguridad de redes en dos maneras principales :

- Canal Seguro - Un protocolo de red encripta los datos antes de ser enviados y decripta datos al recibirlos. El protocolo puede encriptar todos los datos , pero debido a que la encripción y decripción significan una sobrecarga de procesamiento, y mucho de lo que las personas envian no requiere seguridad, el protocolo de seguridad encripta sólo lo que la aplicación requiere. Los protocolos *PCT* y *SSL* (los cuales más adelante se explican) son canales seguros.
- Seguridad Aplicativa - El vendedor de aplicaciones escribe codigo encriptado dentro de la aplicación, de esta manera el puede enviar mensajes con seguridad sin necesidad de un canal seguro.

Kerberos es una de las tecnologias de encripción de llave privada más conocidas. Consiste en crear un paquete de datos encriptado, llamado "ticket", identificando seguramente al usuario. Para hacer una compra, se genera el "ticket" dentro de una serie de mensajes codificados que se intercambian con el servidor de Kerberos, el cual se encuentra situado entre la computadora del sistema o aplicación y la computadora que quiere accesar el sistema o aplicación. Estos sistemas comparten una llave secreta con el servidor de Kerberos para proteger la información y asegurar que los datos no han sido alterados durante la transmisión.

## *Criptografía*

La criptografía proporciona un conjunto de técnicas para codificar datos y mensajes de tal manera que puedan ser almacenados y transmitidos seguramente. La criptografía es utilizada en varias maneras :

- Para lograr seguridad en las comunicaciones, aún cuando el medio (por ejemplo, la *Internet*) sea poco confiable.
- Para encriptar archivos sensibles de manera que un intruso no pueda entenderlos.
- Para asegurar integridad y privacidad de datos.
- Para verificar el origen de datos y mensajes.

Los métodos de criptografía son una serie de operaciones realizadas sobre los datos. Las dos fundamentales son encripción (con decripción como su inversa) y el firmado (con la verificación de una firma como operación de verificación). El firmado es similar a firmar físicamente un documento y escribir las iniciales en cada sección del documento para mostrar que éste no ha sido cambiado. La verificación es equivalente a comparar la firma en un archivo de firmas, y verificar que ninguna parte del documento ha cambiado.

La criptografía de llave pública confia en funciones de un sentido. Estas son funciones que son fáciles de calcular, pero difíciles de invertir, sin algún tipo de conocimiento anterior. La criptografía de llave pública explota esta asimetría para crear funciones donde :

- Es facil realizar una operación (por ejemplo, encriptar o verificar una firma)
- Es extremadamente difícil invertir la operación (decriptar o crear una firma) sin tener toda la información.

La criptografía implementa funciones de un sentido usando dos llaves diferentes pero relacionadas, llamadas llaves pares. Estas llaves son creadas al mismo tiempo. Las llaves están relacionadas matemáticamente en que la llave privada es requerida para invertir operaciones realizadas con la llave pública, y la llave pública es requerida para invertir operaciones realizadas con la llave privada.

Una función de muchos-a-uno es cuando la llave pública es ampliamente distribuida y la llave privada permanece privada. Esto significa que cualquiera puede usar la llave pública para realiza operaciones de criptografía, pero sólo la persona que tiene la llave privada puede invertirlas. Esta es también una función de uno-a-muchos. Esto significa que una persona retiene la llave privada puede realizar una función que cualquiera que tenga la llave pública puede invertir. Esta dos funciones son usadas para encriptar (muchas personas pueden encriptar, solo una persona puede decriptar), y firmar (una sola persona puede firmar pero muchas personas pueden verificar una firma).

Los algoritmos simétricos son los algoritmos de encripción más comunes. Son llamados simétricos porque la misma llave es usada tanto para encriptar como para decriptar. A diferencia de la llaves usadas con algoritmos de llaves públicas, las llaves simétricas son cambiadas frecuentemente. Por esta razón, estas son llamadas "llaves de sesión"

Comparando a los algoritmos de llaves públicas, los algoritmos simétricos son muy rápidos. Consecuentemente son preferidos cuando se encriptan grandes cantidades de datos.

Muchos de los protocolos modernos de criptografía usan una combinación de criptografía de llave pública y criptografía simétrica para tener los beneficios de ambos. Los algoritmos de llaves públicas son usados para intercambiar una llave simétrica , la cual es entonces usada para rapidamente encriptar o decriptar los datos.

### *Acceso Remoto vía Internet*

Muchas empresas están dando a otras empresas especializadas el manejo y administración de su creciente, compleja y cara infraestructura de acceso remoto. Los dos principales tipos de proveedores que proporcionan estos servicios son : los proveedores de redes de valor agregado(*VAN*, por sus siglas en inglés) y proveedores de servicios de *Internet* (*ISP*, por sus siglas en inglés). Sin embargo, el bajo costo, las opciones tarifarias y los accesos que se encuentran disponibles en todas partes de los *ISP*s han causado que muchas empresas vuelvan a pensar en el uso de su tradicional proveedor tradicional *VAN*.

Nuevos vendedores de productos y el suceso de plataformas de aplicaciones WWW están propiciando el incremento del uso de *Internet*. Sin embargo, la mayoría de los vendedores no advierten a las empresas de los potenciales riesgos de seguridad y problemas de rendimiento que se pueden encontrar. Las empresas no deben ser engañadas creyendo que los factores críticos concernientes al trabajo remoto sobre *Internet* han sido resueltos satisfactoriamente.

*Internet* es una nube pública a la cual cualquiera se puede conectar. Su naturaleza pública y su falta de garantías de rendimiento son las bases de los aparentemente bajos costos de los *ISPs* por conectarse a ellos. Sin embargo, los bajos costos tienden a ocultar los riesgos asociados con el acceso a *Internet* -especialmente cuando el acceso remoto está habilitado. Muchas empresas han estado usando informalmente el acceso remoto y el comercio por al menos los últimos 10 años, y muchas han pagado un alto precio por este acceso en la forma de robos, virus, violaciones a la seguridad, etc.

En 1996, la mayor parte de los evidentes y previamente conocidos hoyos en la seguridad de *Internet* fueron teóricamente resueltos a través de la introducción de las tecnologías de encripción, métodos de autentificación y *firewalls*. En la práctica, sin embargo, los riegos permanecen altos, debido a la falta de soluciones completas de cualquier vendedor. Cualquier acceso a una red empresarial sobre *Internet* debe ser autentificado y encriptado sin importar que tan benigna sea la información accesada.

Existen 3 tipos de riesgos de seguridad en *Internet* :

1. El potencial para que intrusos puedan conseguir acceso a la empresa usando los caminos de acceso remoto es alto, debido al creciente número de puertos de acceso remoto y cuentas de usuarios remotos.
2. Todos los datos que viajan sobre redes públicas son vulnerables a la seguridad y modificación, y frecuentemente quedan en riesgo aún cuando la empresa haya tomado medidas de seguridad.
3. Las prolongadas restricciones de exportación del gobierno de E.U. sobre las tecnologías de encripción han hecho difícil para las empresas disfrutar confiablemente de la seguridad transaccional de datos.

Las empresas deben seguir los siguientes 4 pasos para todos los accesos corporativos vía *Internet* :

1. Autentificación estricta a través de un *password* de única vez - Cuando millones de potenciales intrusos pueden intentar violar el *firewall* corporativo, los sistemas de *passwords* de única vez son el único método aceptable de identificación al momento de firmarse. Además los servicios de autentificación deben ser instalados para todos los dispositivos disponibles.
2. Encriptar la secuencia de firmado - Las identificaciones de usuarios y los diálogos relacionados deben ser protegidos.
3. Encriptar el flujo de datos - Una encripción completa de los datos del protocolo de *Internet* (*IP*) en el nivel de Red es necesaria para asegurar que los paquetes enviados sobre redes públicas no puedan ser decodificados. Las especificaciones abiertas, tal como *SSL*, no pueden proporcionar completa protección. Una rigurosa seguridad sobre los datos causa una pérdida de rendimiento, aún con estaciones de trabajo poderosas.
4. Evitar la transmisión fuera de límites de información altamente sensitiva a través de *Internet*. Las restricciones de exportación limitan la efectividad de las claves de seguridad usadas en transmisiones internacionales.

Los proveedores tradicionales de redes de valor agregado comprenden que la popularidad del uso de *Internet* para el acceso remoto no puede ser ignorado, pero este debe ser manejado con mayor profesionalismo para usuarios de negocios. Por lo tanto, los proveedores *VAN* están produciendo nuevos servicios para *Intranets* que intentan combinar la estructura de precios y disponibilidad de *Internet* con la calidad profesional y la administración especializada de los servicios tradicionales. Estas redes ofrecen un acceso universal *IP* combinado con la seguridad de acceso para usuarios autorizados y un *firewall* administrado para la *Internet* pública. En las redes privadas de estos proveedores, la adecuada seguridad puede ser habilitada sin los extensos procedimientos de seguridad recomendados para la *Internet* debido a que el tráafico de la red no es transportado en circuitos públicos. Puesto que los circuitos y el equipo de la red están bajo control privado, los proveedores estan dejando de proporcionar los acuerdos de nivel de servicio para garantizar disponibilidad, confiabilidad y tiempo de respuesta.

### *Seguridad Transaccional*

Para todas las instituciones financieras, la seguridad de los sistemas, las redes y la información tiene una importancia de grado superlativo.

En el sistema bancario tradicional, el banco y los comerciantes usan información redundante y re-verificaciones para validar una transacción financiera. La información redundante incluye el checar la firma de un cliente en un cheque contra la firma en una identificación oficial. Una vez que la identidad de un cliente ha sido autentificada, la transacción es autorizada.

En *Internet*, el mismo mecanismo de usar información redundante para autentificar a un cliente y autorizar una transacción financiera debe ocurrir, sólo que de manera electrónica. Para poder realizar una transacción electrónicamente en *Internet* se debe estar seguro de con quién se está interactuando, que el mensaje que se recibió es el que fué enviado, que nadie leyó el mensaje en el camino o que no se pueda negar haber recibido el mensaje. Todos estos aspectos se garantizan a través de esquemas de autentificación, integridad, confidencialidad y de no-repudiabilidad.

Desde esta perspectiva, la información redundante a utilizar para ofrecer servicios bancarios en *Internet*, puede incluir el domicilio *IP* de una máquina, el número telefónico del *modem*, una contraseña, o una llave de encripción. De esta manera, cuando se quiera realizar una transacción, estos datos redundantes se pueden cotejar contra aquellos almacenados en una base de datos en el banco, reduciendo así (más nunca eliminando del todo) la posibilidad de fraudes.

Para ayudar a cubrir estas necesidades de seguridad, se han desarrollado los protocolos : *STT (Secure Transaction Technology)* para transacciones financieras y *PCT (Private Communications Technology)*, un conjunto de mejoras a *SSL (Secure Sockets Protocol)* y *SET (Security Electronic Transactions)*, los cuales a continuación se explican.

### *SSL- Capa de Sockets de Seguridad*

*Netscape Communications* es un proveedor de *software* para navegar en la *Web*, así como de *software* para servidores *Web*, que se encuentra trabajando en un protocolo para realizar aplicaciones que conlleve el pago con tarjetas de crédito en *Internet*. El protocolo es un protocolo de terceros que gobierna la relación entre el cliente, el comerciante y el adquiriente intermediario. Los aspectos de seguridad en una transacción con tarjeta de crédito se clasifican en dos clases : seguridad de conexión (o de canal) y seguridad de pago específico.

La seguridad de canal puede ser provista por una capa de transporte segura, sin embargo se requiere de una capa de nivel superior para el "protocolo de pago electrónico" encargado de autentificación de firmas, no-repudiación y encripción secundaria. En este esquema la encripción secundaria puede ser decriptada parcialmente por el receptor y parcialmente por una entidad tercera. El protocolo de pago asume la existencia de dos canales de seguridad, uno entre el cliente y el comerciante y el segundo entre el comerciante y el intermediario de pago.

La capa de transporte segura proporciona privacidad de datos e integridad para la comunicación entre dos nodos. Esto implica que los dos canales seguros estén establecidos. La autentificación de los nodos apropiados es también una función de esta capa.

*Netscape* está implementando su protocolo de seguridad para la capa de transporte en una capa llamada *SSL (Secure Sockets Layer)* para proporcionar privacía sobre *Internet*. Este protocolo permite que las aplicaciones Cliente/Servidor sean autentificadas a fin de que se puedan comunicar sin temor de ser infiltradas. *SSL* es independiente del protocolo de aplicación, ya que protocolos superiores *como HTTP, FTP y Telnet* pueden ser colocados sobre *SSL* sin ningún problema.

*SSL* tiene tres propiedades básicas :

1. El canal es privado. Se utiliza encripción para todos los mensajes después de que un sencillo protocolo de saludo es realizado para definir una llave secreta. Se utiliza criptografía simétrica para la encripción.
2. El canal es autentificado. El punto terminal de un servidor en una conversación siempre es autentificado y el cliente es autentificado opcionalmente. Se utiliza encripción asimétrica ( criptografía con llave pública).
3. El canal es confiable. El transporte del mensaje incluye la validación de integridad del mensaje.

Por otro lado, el protocolo de seguridad de *Netscape* se ocupa también de aspectos de pago específico, como :

- Confidencialidad del número secreto de la tarjeta de crédito (*PIN*) y otra información del cliente. Esta información se guarda encriptada aún para el comerciante, mientras maneja la transacción.
- Confidencialidad de la información de ordenes específicas de todas aquellas entidades distintas al comerciante. Esto incluye a los bancos emisores y adquirientes.
- Firma digital en cada uno de los distintos mensajes comunicados entre diferentes entidades para garantizar la autoría del mensaje.

### *STT - Tecnología de Seguridad Transaccional*

En septiembre de 1995, *Microsoft Corporation* y *Visa International* anunciaron su especificación para pagos seguros sobre redes públicas y privadas.

La especificación abierta, conocida como *STT* (*Secure Transaction Technology*), está diseñada para proporcionar un método seguro para manejar pagos a través de tarjetas de crédito en redes electrónicas como *Internet*. Construido como una versión electrónica del sistema de pago con tarjeta que se tiene hoy, *STT* extiende la seguridad actual en torno a transacciones así como las ventajas que se tienen en el Comercio Electrónico. Al proporcionar una tecnología que se integra completamente con el sistema actual de tarjeta bancaria, *STT* servirá como un sistema de pago confiable para que los proveedores de *software* lo incluyan en sus productos.

*STT* usa métodos avanzados de criptografía para proporcionar privacidad, integridad, autentificación, y las firmas de las solicitudes y recibos a todos los participantes en una transacción de tarjeta de crédito.

*STT* habilita la seguridad en el comercio en *Internet* y reduce riesgos en todo el sistema de tarjetas de crédito. Es diferente de otras tecnologías en varias maneras :

- No se entromete dentro de las relaciones de negocios entre los bancos y sus clientes o entre negocios y clientes.
- *STT* usa tecnología de firmas digitales -no sólo una por tarjetahabiente, sino por cada tarjeta- extendiendo las relaciones entre un tarjetahabiente y cada banco.
- *STT* es una arquitectura para pagos. A diferencia de otros sistemas, el no determina el modelo de compras. Del mismo modo que nosotros compramos diferentemente en almacenes que lo hacemos en restaurantes, lo mismo puede ser verdadero en el mundo electrónico. Sin embargo, un pago con una tarjeta Visa es la misma a través de los diferentes modelos de compras.

Todos los participantes en la transacción - consumidores, bancos y negocios- se pueden beneficiar :

- Consumidores : Pueden hacer confiadamente compras seguras en *Internet* con tarjetas de crédito.
- Negocios : por tres razones :
    1. Pueden recibir con seguridad pedidos a través de *Internet*.
    2. Pueden recibir solamente información ligada a las características de una transacción *STT* , reduciendo con esto riesgos.
    3. *STT* puede fortalecer la relación del negocio con sus clientes.
- Bancos : Mantienen su relaciones de negocio con sus clientes y con los clientes de los negocios, y reducen el riesgo de manejar transacciones por *Internet*.

### *Protocolo PCT*

El protocolo *PCT* (*Private Communication Technology*), es un protocolo de seguridad que proporciona privacidad a través de la *Internet*. El propósito del protocolo es prevenir la intromisión en las comunicaciones de aplicaciones Cliente/Servido. *PCT* es algo similar a *SSL* ; sin embargo, la versión 1 de *PCT* corrige o mejora varias debilidades de *SSL*, y la versión 2 agrega nuevas características. Las mejoras incluyen una mayor eficiencia , usando un protocolo más eficiente y una fuerte autentificación, y una separación del proceso de autentificación del de encripción.

El protocolo *PCT* está diseñado para proporcionar privacidad entre dos aplicaciones que se están comunicando (un cliente y un servidor), y para autentificar al menos uno de los dos (típicamente el servidor) con el otro. *PCT* es independiente del protocolo de aplicación ya que protocolos superiores como *HTTP, FTP y Telnet* pueden ser colocados sobre *PCT* transparentemente.

En el protocolo *PCT*, todos los datos son transmitidos en forma de registros de longitud variable, cada uno de los cuales tiene un registro "encabezado". Estos registros son usados para transmitir tanto mensajes propios del protocolo *PCT* como mensajes con datos de la aplicación. El intercambio de registros entre un cliente y un servidor son agrupados dentro de "conexiones", las cuales son agrupadas dentro de "sesiones". Cada conexión *PCT* pertenece a una sesión particular.

Cada conexión del protocolo *PCT* comienza con una fase de sincronización, durante la cual una serie de mensajes de sincronización son intercambiados, los cuales negocían una llave simétrica de sesión para la conexión, al mismo tiempo que se solicita la autentificación basada en llaves publicas asimétricas. Una vez que inicia la transmisión de los mensajes del protocolo de aplicación en una conexión, todos los datos (incluyendo mensajes de errores y de administración de llaves) son encriptados usando llaves de encripción derivados de la llave maestra intercambiada durante la fase de sincronización de la sesión de la conexión. Además de encripción y autentificación el protocolo *PCT* incluye la verificación de la integridad de los mensajes utilizando código de autentificación de mensajes basado en la función de *hash*.

***SET (Security Electronic Transactions)***

El nuevo estándar *SET* propuesto por *Visa* y *MasterCard*, detalla cómo las transacciones bancarias en la *Internet* y otras redes abiertas pueden estar aseguradas usando tecnología de encripción. Se espera que a partir de 1998 se puedan ofrecer soluciones en *Internet* basadas en *SET* a consumidores, negocios e instituciones financieras. Los participantes en este esfuerzo con *Visa* y *Master Card* son : *GTE, IBM, Microsoft, Netscape Communications, SAIC, Terisa Systems* y *Verisign.*

*SET* es un protocolo seguro de mensajes para transacciones de tarjeta de crédito. Provee autentificación para tarjetahabientes, negocios y adquirientes. *SET* mantiene la confidencialidad de los datos del pago.

Es un esquema transaccional basado en *RSA*, el cual es el sistema de criptografía de llave pública más conocido. Esto permite a los clientes usar llaves privadas para codificar su identidad y permitir a los bancos usar llaves públicas para codificar las firmas de los clientes, verificando la autenticidad de sus demandantes.

*SET* difiere de los pagos hechos a través de canales seguros como *SSL* en los siguientes puntos :

- *SET* usa un algoritmo de encripción *DES* de 56 bits.
- *SET* requiere de firmas digitales para verificar que el cliente, el negocio y el banco son legítimos.
- *SET* encripta la información directamente a los bancos. Esto evita que los números de tarjetas de crédito finalicen en manos equivocadas
- *SET* requiere de integración dentro del sistema de procesamiento de tarjetas de crédito.

La figura 3.1 muestra la manera en que *SET* trabaja.

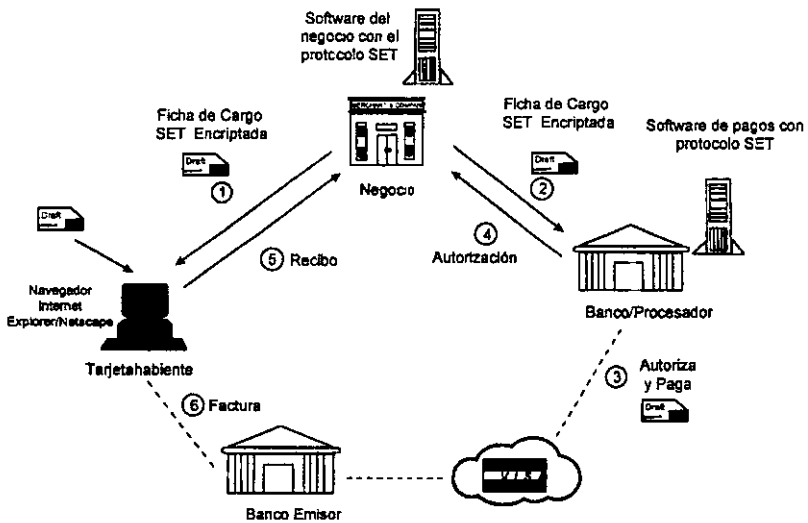



***Fig. 3.1 Security Electronic Transactions (SET)***

Los bancos e instituciones financieras en Europa están trabajando en sus esquemas de banca por *Internet*, por ejemplo, se ha desarrollado *HBCI* (*Home Banking Computer Interface*) que como *SET* utiliza *RSA* para verificación y encripción de mensajes.

En una transacción utilizando *HBCI*, un *password* autoriza al usuario a accesar los sistemas bancarios. Cada usuario tiene firma electrónica única basada en un algortimo *RSA*. A diferencia de un algoritmo simétrico, como *DES*, un algoritmo asimétrico como *RSA* trabaja con dos llaves, una privada y una pública. La privada permanece segura en la PC del usuario. El banco utiliza la llave pública para autentificar al usuario y validar su firma.

La arquitectura de *HBCI* también incluye un mecanismo para checar la integridad de los mensajes colocando un código *hash* sobre el mensaje a transmitir. Un código *hash* es *checksum* criptográfico que es único a cada transacción pero permite la identificación de usuarios y mensajes. El servidor *Web* receptor del banco checa la integridad del mensaje recibido recalculando el código *hash*. Si un mensaje es alterado durante la transmisión, el código *hash* refleja esa alteración y la transacción es invalidada.

### *Seguridad en el Comercio Electrónico*

El Instituto Nacional de Estándares y Tecnología (*NIST*, por sus siglas en inglés) define el Comercio Electrónico como una evolución del Intercambio Electrónico de Datos (*EDI*, por sus siglas en inglés) hacia otros tipos de datos y transacciones. La visión del Instituto acerca de las características de un Comercio Electrónico avanzado comprende :

- Redes de comunicación interconectadas.
- Servicios y herramientas de *software* y *hardware* para computadoras avanzadas.
- Estándares establecidos para transacciones de negocio, intercambio de datos e interoperabilidad.
- Esquemas aceptados de seguridad y privacia.
- Prácticas culturales y administrativas apropiadas.

La infraestructura prevista por el Instituto facilitará que diversas compañías distribuidas mundialmente, puedan intercambiar, y sobre todo, utilizar información, de manera más rápida, flexible y segura, para apoyar sus procesos de negocio. Esto propiciará que los empleados se dediquen más a definir soluciones más creativas en torno a su negocio, que a estar realizando procesos rutinarios.

El *IITF* (Information Infrastructure Task Force, por sus siglas en inglés) establece que, aunque *EDI* permite el intercambio de transacciones rutinarias, relativamente simples, entre distintos sistemas de información automatizados, el requerimiento existente de acuerdos rígidos acerca de la estructura y significado de los datos, dificulta que los estándares establecidos cumplan con el reto de proveer un uso eficiente de los datos a partir de distintas fuentes. Estos estándares también resultan caros, inflexibles y difíciles de mantener, especialmente en un ambiente cambiante. En vez de definir el Comercio Electrónico en función de las tecnologías requeridas, el *IITF* basa su definición en las actividades de negocio que el Comercio Electrónico debe soportar :

- **Transferencia electrónica de fondos** : extensión complementaria al proceso de transferencia de fondos, en donde se busca proporcionar a los compradores la posibilidad de hacer sus pagos a los vendedores de manera rápida y efectiva, con un menor riesgo financiero, menos errores y reduciendo el manejo y almacenamiento de papel.
- **Intercambio de datos regulatorios gubernamentales** : recopilación ( y regreso) de datos formateados de varias entidades para permitir que el gobierno cumpla con sus responsabilidades.
- **Ingeniería colaborativa** : proporcionar una evaluación temprana de diseños ingenieriles para asegurar la posibilidad de manufactura, confiabilidad y mantenibilidad de productos.
- **Integración corporativa** : extender la integración hacia toda la corporación y hacia otras corporaciones con las que se tenga relación.

Existen una variedad de barreras para la amplia aceptación del comercio electrónico en el mundo hoy en día. Muchas de las más grandes ventajas de los servicios bancarios y de compras en *Internet* también tienen peligros potenciales que necesitan ser atendidos.

Primero, el reciente crecimiento del uso de *Internet* ha estado dirigiendo la atención mundial a un problema evidente—privacidad. Hasta ahora, no ha habido una real garantía para asegurar que los mensajes que se envían y reciben no hayan sido interceptados, leídos o alterados por algún intruso desconocido puesto que nadie controla la *Internet*.

Segundo, en el emergente dominio de *Internet*, el potencial para un fraude y engaño es bastante grande. La habilidad para obtener información desde cualquier lugar en el mundo es visto por muchos como un beneficio de la *Internet*. Sin embargo, esto plantea varios inconvenientes prácticos.

Cuando la otra "persona" es meramente un punto en la pantalla de la computadora, ¿ cómo se sabe que ella tiene una cuenta valida ?, ¿ cómo se sabe que se puede confiar en un negocio y tú cuándo lo has visto ? , después de todo, los negocios pueden existir sólo en un disco duro fraudulento/engañoso. Y, ¿ cómo puede un negocio real sentirse cómodo aceptando un número de cuenta de una tarjeta Visa sin alguna forma de identificación ?.

Para que el Comercio Electrónico pueda prosperar, todos los participantes necesitan una manera de verificar las identidades de cada uno de los demás y establecer confianza. Una de las cuales puede ser la utilización de Certificados Digitales.

### *Certificados Digitales*

En muchas maneras, los certificados digitales representan el corazón de las transacciones electrónicas seguras. Proporcionan una manera fácil y conveniente de asegurarse que los participantes en una transacción de Comercio Electrónico puedan tenerse confianza uno del otro. Esta confianza es establecida a través de un tercer participante común, tal como Visa. Por ejemplo, Visa puede proveer certificados digitales a las instituciones financieras emisoras de tarjetas y la institución entonces a su vez puede proveer un certificado digital al tarjetahabiente. Un proceso similar toma lugar para los negocios.

Al momento de la transacción, cada participante en *SET* valida tanto el certificado del negocio como el del tarjetahabiente antes de que cualquier información sea intercambiada. La validación se lleva a cabo checando los certificados digitales los cuales fueron emitidos por un tercer participante autorizado.

Exactamente, los certificados digitales aseguran que dos computadoras que están dialogando una con otra puedan conducir Comercio Electrónico. La base de esta tecnología son los códigos secretos. El proceso es simple, un mensaje puede ser convertido o encriptado en un código utilizando una llave. Para decriptar el mensaje, el receptor simplemente necesita conocer la llave secreta. Existen dos tipos principales de criptografía comúnmente usadas hoy en dia. La más vieja y simple llamada Criptografía de llave sencilla o llave secreta.

## 3.3 Barreras de Acceso (Firewalls)

El objetivo principal de un *firewall* es el proteger a una red de otra. Generalmente la red que se pretende proteger pertenece a una empresa u organización preocupada por su seguridad y la red de la que se quiere proteger es una red externa en la que no se puede confiar. Proteger a una red implica impedir que usuarios no autorizados tengan acceso a datos sensitivos, en tanto que se permita que los usuarios autorizados tengan pleno derecho de accesar los recursos de la red.

Dentro de una empresa u organización sus redes locales pueden estar conectadas entre sí y sólo ocasionalmente establecer una conexión temporal hacia el mundo exterior a través de *modem*, conectándose tal vez a Compuserve o, a través de éste, a *Internet*. Generalmente este tipo de conexiones temporales no ofrecen mayor riesgo de ser atacado por algún usuario externo.

Los riesgos se presentan cuando se cuenta con una conexión permanente, en donde uno o más de los sistemas con los que se cuente, es un nodo permanente de una red más grande. *Internet* es la red más grande que se conoce. Hoy en día incluye redes en más de 100 países con más de un millón de computadoras ( y más de 10 millones de usuarios) alrededor del mundo. Todos estos usuarios podrían, potencialmente, accesar cualquier nodo que se encuentre dentro de la red. *Internet* resulta tan vulnerable como útil es y es quizás capaz de causar más daño a una red local interna, que cualquier otro medio. Recordemos que *Internet* atendió inicialmente, y continua atendiendo, a la comunidad estudiantil para fines de investigación. Pero es esta comunidad precisamente, el semillero de atacantes, ya que tienen las habilidades para accesar redes internas, cuentan con todo el tiempo en sus manos y sobre todo, son curiosos.

Los nodos permanentes en redes como *Internet* representan una vulnerabilidad crítica, y cientos de individuos tienen las habilidades para accesar un sistema en particular a través de dichos nodos. El trabajo de un administrador de un sistema es el filtrar llamadas perfectamente, de tal manera que los usuarios autorizados siempre tengan acceso, pero que quienes no lo estén, nunca tengan oportunidad de entrar al sistema. Esto es precisamente lo que se puede lograr a través de los denominados *firewalls*, *bridges*, *routers* o *gateways*.

En general, un *firewall* es colocado entre la red interna (y confiable) y la red externa (no confiable). El *firewall* actúa como un punto de obturación y rechaza tráfico de red a nivel de la capa de aplicación (*OSI*). También puede trabajar en los niveles de red y de transporte, examinando los encabezados de *IP* y *TCP* de los paquetes que entren o salgan y rechazar aquellos paquetes que no correspondan a las reglas que tenga definidas para filtrado de paquetes.

## Arquitectura de Firewalls

Los *firewalls* se pueden implementar en distintas arquitecturas :

- **Host Bimodal**

Entre las redes *TCP/IP*, el término *host multimodal* (*multi-homed host*) describe a un *host* que tiene múltiples tarjetas de interface de red. Usualmente, cada tarjeta de interface está conectada a una red. Históricamente, este *host* también ha podido rutear tráfico entre los segmentos de la red. El término *gateway* fue usado para describir la función de ruteo que estos *host* realizaban.

Si la función de ruteo está deshabilitada en el *host* multimodal, el *host* puede proporcionar aislamiento en el tráfico de red entre las redes a las que se conecta y aún así cada red podría seguir procesando aplicaciones en el *host*. Más aún, si la aplicación lo permite, las redes pueden compartir datos.

Un *host* bimodal es un ejemplo especial de un *host* multimodal, que sólo tiene dos tarjetas de interface de red y que tiene la función de ruteo deshabilitada. La figura (3.2) muestra un ejemplo de un *host* bimodal con la función de ruteo deshabilitada. El *host* A en la red 1 puede accesar la aplicación A en el *host* bimodal. En forma similar, el *host* B en la red 2 puede accesar la aplicación B en el *host* bimodal. Ambas aplicaciones pueden incluso compartir datos. Ambas aplicaciones pueden compartir información a través de los datos comunes sin tener intercambio de tráfico sobre la red.

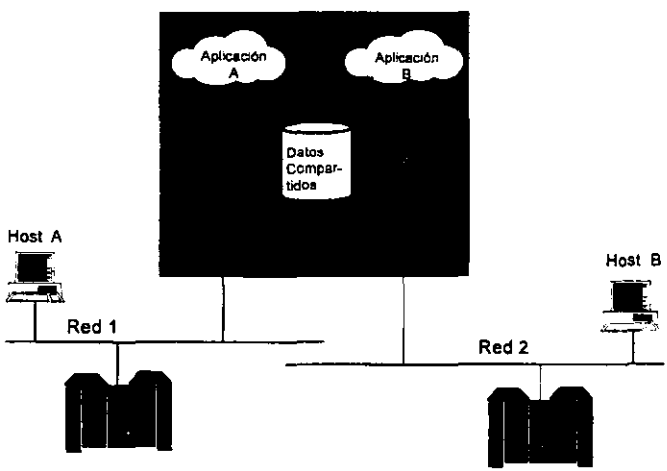



***Fig. 3.2 Host Bimodal***

- **Un Host Bimodal como Firewall**

El *host* bimodal puede ser usado para aislar una red interna de una red externa y no confiable (ver fig 3.3) Debido a que un *host* bimodal no pasa hacia adelante ningún tráfico de *TCP/IP*, bloquea completamente cualquier tráfico de *IP* entre la red interna y la externa.

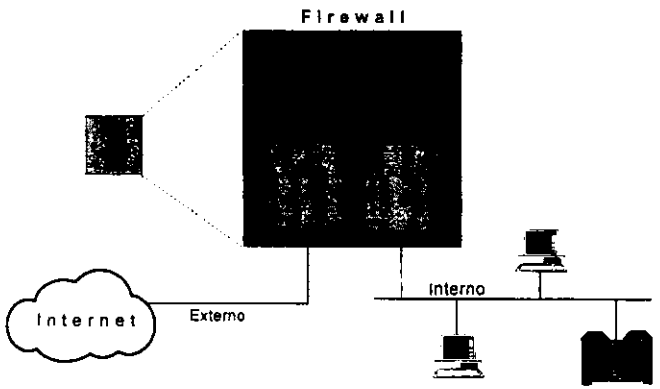



***Fig. 3.3 Host Bimodal como Firewall***

Muchos de los servicios de *Internet* son del tipo *almacena-y-envío*. Si estos servicios corren sobre un *host* bimodal, pueden ser configurados para transmitir servicios aplicativos de una red a otra. Si los datos aplicativos deben atravesar el *firewall*, se requieren "agentes enviadores" (*software* especial para enviar requerimientos aplicativos entre redes conectadas entre sí) ver figura (3.4)

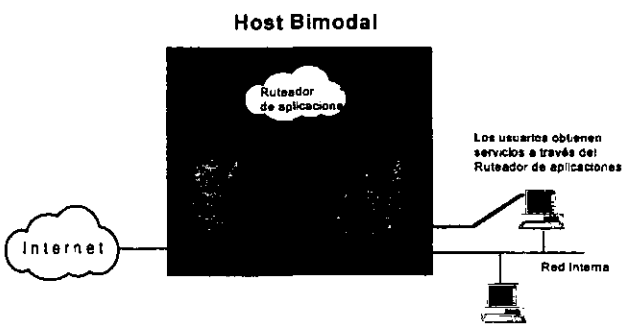



***Fig. 3.4 Host Bimodal con ruteador de aplicaciones***

- **Host Bastión**

Un host bastión es cualquier host firewall que resulte crítico para la seguridad de la red. El host bastión es el host central en la seguridad de la red de una organización. Debido a su papel crítico dentro de la seguridad, debe encontrarse perfectamente fortificado. Debe de poder ser monitoreado, debiendo someterse a auditorias periódicas. Un host bimodal es un ejemplo de un host bastión, por su criticidad dentro de la seguridad de la red.

Debido a que los hosts bastión actúan como punto de interface con una red externa, son con frecuencia objeto de intrusión. La implementación más simple de un host de este tipo consiste en colocarlo como el primer y único punto de entrada para tráfico de redes externas, como se muestra en la fig. (3.5)

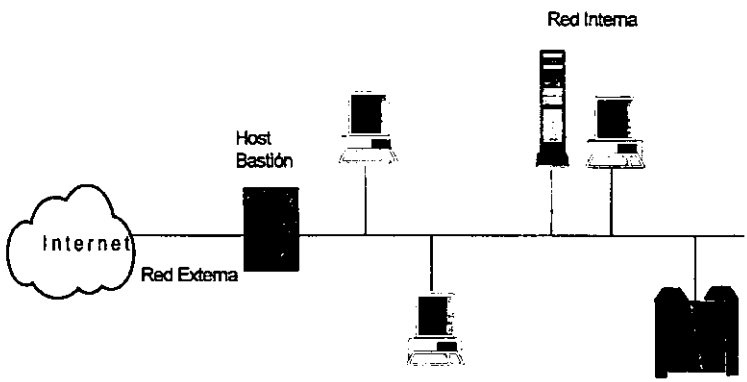



***Fig. 3.5 Host Bastión***

- **Modelo con dos interfaces configuradas**

La figura (3.6) muestra el uso de un host bastión con un "ruteador de chequeo", pero ambas interfaces del host bastión se encuentran configuradas. Una interface se encuentra conectada a la red exterior y la otra interface se encuentra conectada a la red interior. Uno de los puertos del "ruteador de chequeo" se encuentra conectado a la red interior y el otro puerto está conectado a Internet.

El "ruteador de chequeo" debe ser configurado de tal manera que envíe todo el tráfico que reciba de redes externas destinado a la red interna, hacia la interface interna del host bastión. Esto es, antes de dejar pasar el tráfico, el ruteador debe aplicar sus reglas de ruteo.

En la red exterior no existe otro host aparte del "ruteador de chequeo" y una de las interfaces del host bastión. La red exterior forma una Zona Demilitarizada (DMZ). Debido a que la DMZ sólo tiene dos conexiones de red, puede ser reemplazada por una liga punto-a- punto.

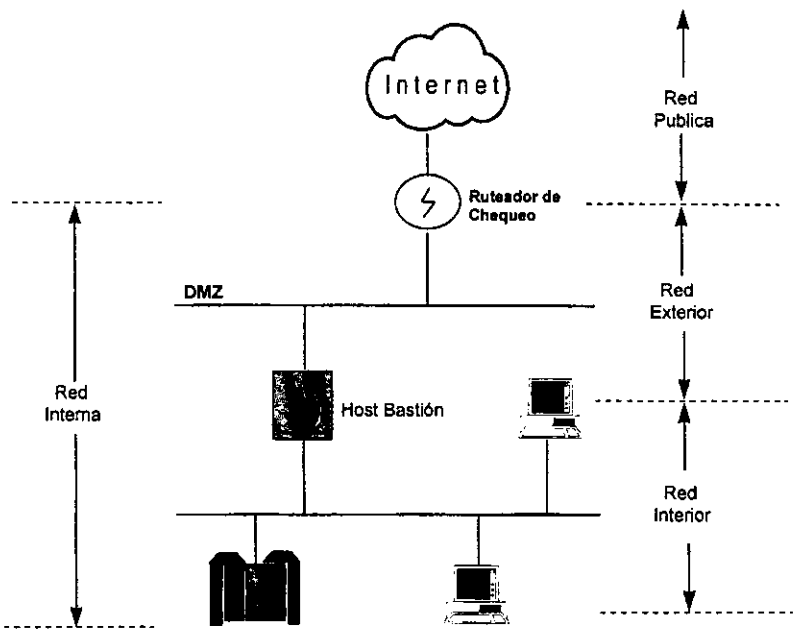



***Fig. 3.6 Host Bastión con dos interfaces y un ruteador de chequeo***

- **Modelo con doble host**

La figura (3.7) muestra el uso de dos hosts bastión con un "ruteador de chequeo". Ambas interfaces de red de los dos hosts se encuentran configuradas. Se conforman tres zonas de red en la red interna: la red exterior, la red privada y la red interior. En la zona privada pueden colocarse algunos hosts que no sean muy críticos para la organización, o bien, puede ser que no se coloque ningún equipo y que esta zona simplemente represente una barrera más para acceder a los equipos que se encuentren en la red interior.

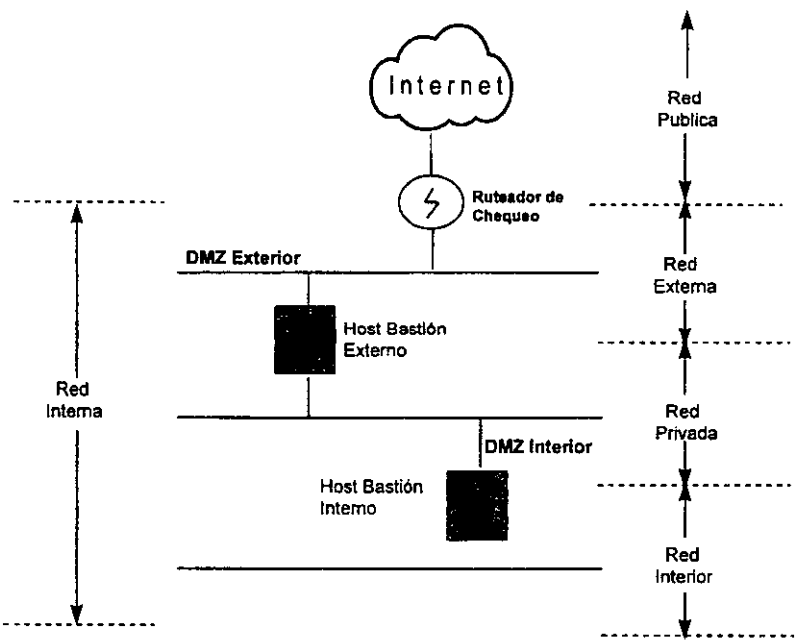



***Fig. 3.7 Modelo con dos host bastión con dos interfaces y un ruteador de chequeo***

### *Bridges, Routers, Gateways y Firewalls*

Los términos puente (*bridge*), ruteador (*router*), *gateway* y *firewall* son en ocasiones usados tan indistintamente que nadie puede especificar claramente la diferencia entre ellos. Algunas veces un ruteador es considerado como un *gateway*; algunas veces los términos *gateway* y *firewall* son usados en forma indistinta; y algunas veces un ruteador es llamado un *firewall*.

Independientemente de la confusión general, los *bridges* trabajan en el nivel 2 del modelo de 7 niveles de *OSI* (ver fig. 2.13), el nivel de ligado de datos (data link) y dentro de este nivel, en el subnivel *MAC* ( media access control ).

Los *routers* funcionan en el nivel 3 del modelo *OSI*, el nivel de red.

Los *gateways* funcionan a partir del nivel 4 del modelo *OSI*, nivel de transporte.

Los *firewalls* no corresponden a un nivel en particular, son más bien una metáfora para cualquier barrera que impide la entrada de intrusos indeseados, por lo que los *firewalls* pueden ser *bridges*, *routers* o *gateways*.

Lo ideal sería tener un dispositivo que protegiera a un sistema en cualquier nivel de *OSI*, que se llamara, no importa qué fuera, un *firewall* y que parara cualquier cosa indeseada perfectamente. La realidad es que no importa el nombre del dispositivo de seguridad, sino qué protección ofrece, qué requerimientos se tienen y cuánto se puede gastar. Los *bridges* son la opción más económica, más conveniente y menos segura. Los *routers* son intermedios, en tanto que los *gateways*, o el par de *gateways*, pueden ser la opción menos económica, menos conveniente pero más segura.

Los *firewalls* contienen dos componentes básicos: compuertas (*gates*) y obturadores (*chokes*). Las compuertas dejan pasar los datos libremente entre las redes. Los obturadores bloquean la entrada de paquetes que no sean destinados a la compuerta, o bloquean los paquetes que salen y que no provienen de la compuerta. Es decir, cualquier paquete que no tenga la dirección de la compuerta ya sea como origen o como destino, es bloqueado. La compuerta es generalmente una computadora *gateway*, en tanto que el obturador normalmente es un ruteador inteligente. El ruteador es localizado entre la compuerta (o *gateway*) y la red externa.

En los últimos años han surgido los *bridges*, *routers* y los dispositivos híbridos llamados *brouters*, como medios para conectar redes locales en las organizaciones, extendiendo las limitantes de distancia impuestas por los cableados de *Token Ring* y *Ethernet*. Al filtrar algunos usuarios, estos dispositivos incrementan el ancho de banda disponible en un segmento de la red.

***Bridges y Routers***

Entre los *bridges* y los *routers* exsisten varias similitudes :

Los dos pueden ser usados para ligar redes locales que físicamente se encuentran separadas. Tanto los *bridges* como los *routers* son programables, pueden ser configurados para filtrar paquetes, así que ambos pueden ser usados para dividir grandes redes en redes más pequeñas.

Debido a que los *bridges* trabajan en un nivel tan bajo, normalmente se pueden mezclar dispositivos de diferentes fabricantes. También se pueden combinar ruteadores de diferentes proveedores siempre y cuando utilicen el protocolo *PPP* (*point-to-point protocol*), un estándar para ruteadores que apareció recientemente.

Pero cuentan con características que los diferencían, como :

- Los *bridges* funcionan en el nivel 2 de *OSI*, en tanto que los *routers* trabajan en el nivel 3 del mismo modelo.
- Los *bridges* leen los 48 bits de la dirección destino de cada paquete individual en las redes conectadas, leyendo entonces una tabla de ruteo ( de alrededor de 8000 registros, cada uno de los cuales le indica al *bridge* dónde se localiza un domicilio y su dispositivo asociado dentro de la red) para tomar una decisión. Los paquetes de entrada que tienen una dirección conocida, de acuerdo a la tabla, son direccionados a dicho domicilio, los demás se dejan pasar de largo.
- Los *routers* son más inteligentes que los *bridges*, permitiendo ligas lógicas entre redes separadas. La inteligencia en los *routers* puede ser usada para re-rutear el tráfico en caso de que algún componente de la red falle. Esta inteligencia también se puede usar para traducir de un protocolo a otro, como de *IPX* a *TCP/IP* o viceversa. Algunas veces esta inteligencia incorpora el estándar de *Internet* recientemente desarrollado, *OSPF* (*open shortest path first*), que le permite al *router* en todo momento encontrar el camino más corto entre dos puntos.
- La mayor inteligencia de los *routers* tiene su costo, por supuesto. El examinar cada paquete antes de enviarlo hace que el procesamiento de paquetes de un *router* sea más lento que el de un *bridge*.
- Los ruteadores normalmente son filtros de un sólo sentido, mientras que los *bridges* no filtran en lo absoluto. Los *bridges* son transparentes para el usuario, ya que no tienen que conocer ni especificar explícitamente la dirección de los mismos, cosa que no ocurre con los *routers*.

Muchos *routers* comerciales cuentan con la capacidad de filtrar los paquetes utilizando como criterio el tipo de protocolo, el domicilio origen y destino para un protocolo en particular, y campos de control particulares de cada protocolo. Dichos *routers* son llamados *"ruteadores de chequeo"* y representan un mecanismo muy poderoso para controlar el tipo de tráfico de red que puede existir en cada segmento de la red.

### *Gateways*

Los *gateways* son, normalmente, computadoras diseñadas para manejar una liga entre una red interna y una o más redes externas. Aunque la mayoría de los *gateways* son diseñados para manejar conexiones tanto internas como externas, existen algunos que sólo manejan tráfico externo. Muchas organizaciones cuentan con una sola máquina para manejar su tráfico tanto interno como externo. Aunque dicho dispositivo puede ser configurado inteligentemente para filtrar el flujo de paquetes, el *gateway* puede verse comprometido en términos de seguridad. Normalmente los *firewalls* se construyen a partir de un par de *gateways*.

### *Ruteadores de Chequeo y Firewalls en relación con el modelo OSI*

La figura 3.8 compara a los *"ruteadores de chequeo"* con los *firewalls* en el marco del modelo *OSI*. Esta figura muestra que los *"ruteadores de chequeo"* corresponden al nivel de red *(IP)* y al nivel de transporte *(TCP)*. Los *firewalls* normalmente son descritos como *gateways*, dispositivos que pueden realizar procesamiento en todos los 7 niveles del modelo. Típicamente, el procesamiento que realizan los *gateway* se encuentra en nivel 7, lo cual es cierto para la mayoría de los *firewalls* .

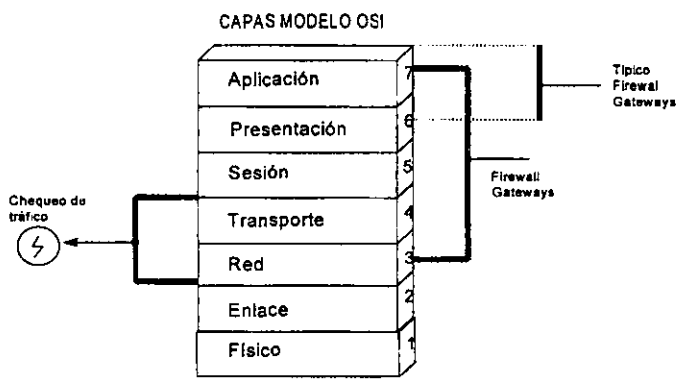



***Fig. 3.8 Ruteadores de Chequeo, Firewalls y el Modelo OSI***

La figura anterior también muestra que, debido a que los *firewalls* cubren los niveles de red y de transporte, también pueden hacer funciones de filtrado. Algunos fabricantes, por razones mercadotécnicas, no hacen distingos entre los *"ruteadores de chequeo"* y los *firewalls*, tanto que venden sus *routers* como si fueran *firewalls*.

### *Zonas de Riesgo*

La figura 3.9 muestra un ejemplo de un servicio de filtrado de paquetes implementado por un "ruteador de chequeo". La figura ejemplifica una red corporativa conectada a *Internet* a través de un *router* que realiza un filtrado de paquetes.

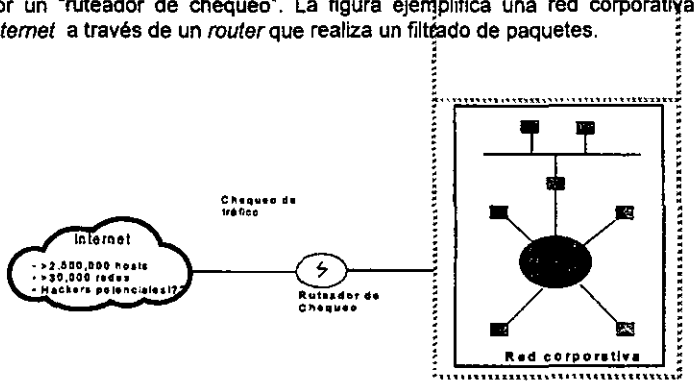



***Fig. 3.9 Ruteadores de Chequeo formando un perímetro de seguridad***

En la figura 3.9, la frontera de la red corporativa es llamada perímetro de seguridad. Debido a que los *hackers* abundan en *Internet*, resulta bastante útil el definir una zona de riesgo. La zona de riesgo incluye todas las redes con protocolo *TCP/IP* que pueden ser accesadas directamente desde *Internet* e incluso se puede extender hacia equipos que no utilicen este protocolo pero que se encuentren conectados a equipos que sí lo utilicen, por ejemplo si se encuentran en el mismo segmento de *Ethernet*. De aquí que lo más deseable es sacar a las redes y equipos centrales de la zona de riesgo, lo cual no se puede hacer sin el apoyo de un dispositivo especializado para este fin. Los "ruteadores de chequeo" pueden realizar esta función,como se muestra en la fig. 3.10.

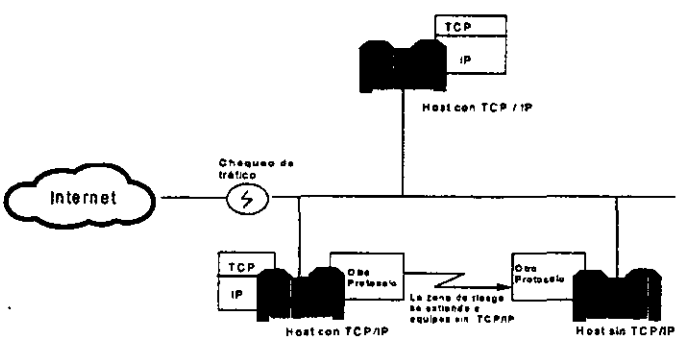



***Fig. 3.10 La zona de riesgo se puede extender a hosts no-Tcpip***

## 3.4 Vulnerabilidades en Internet

1994 y 1995 fueron años que atestiguaron un gran incremento en el interés por el acceso a la "Supercarretera de Información". Estos años trajeron una advertencia pública de que *Internet* podría ser considerada como una valiosa fuente de información. Trajeron una explosiva aparición de libros acerca de Cómo Usar *Internet*. Vieron un incremento en el número de nuevos usuarios de *Internet*, nuevos domicilios asignados y ventas de *hardware* y *software* para permitir el acceso a *Internet*.

Los últimos dos años fueron también años de un despertar general al entendimiento de que hay problemas de seguridad que pueden presentarse al momento de tener una conexión a *Internet*. Mientras que el público en general parece estar consciente de este problema, es muy poca la gente que conoce en detalle en qué consisten los problemas. La mayoría de la gente piensa que los *firewalls* son objetos genéricos, como los extinguidores, que probablemente todos son igualmente efectivos y que seguramente con la instalación de un dispositivo de estos tendrán completamente resueltos los problemas de seguridad acerca de los que tanto han oído.

Nada más lejano de la realidad. Existe una gran variedad de problemas de seguridad. La mayoría de los *firewalls* no eliminan ni una fracción de ellos. Entre ellos, los *firewalls* difieren tan ampliamente, en relación a lo que pueden hacer, que en realidad no merecerían estar agrupados bajo un mismo nombre genérico.

La vulnerabilidad en la seguridad, particularmente en la capa de aplicación (OSI), puede seguir inhibiendo la integración de redes privada confiables con la *Internet* al menos hasta al año 2000.

La más crucial vulnerabilidad de red esta en la capa de aplicación. Los *firewalls* no pueden y no deben proteger contra fallas fundamentales en el diseño de la aplicación (por ejemplo, la habilidad de pequeños correos tipo *MIME* para lanzar documentos de Word conteniendo caballos de Trojan). De aquí al año el 2000, los dispositivos de seguridad de red, como *firewalls* no podrán proporcionar protección contra vulnerabilidades en la capa de aplicación. Muchos de los riesgos en esta área es el resultado de la amplia variedad de dispositivos heterogéneos, sistemas, protocolos y aplicaciones típicas en sitios de grandes empresas. Cada nueva aplicación, ya sea desarrollada internamente o externamente, que es desplegada puede requerir de un ajuste al sistema del *firewall*. Los *firewalls* deben ser modificados para soportar altos niveles de filtración, lo cual es difícil sin la suficiente experiencia de red, con las nueva aplicaciones.

El *firewall* es la primera línea de defensa, el cual controla el acceso al flujo de la información desde y hacia la empresa. Esto es crítico, el que las empresas vean al *firewall* como la única medida de protección necesitada. Las empresas deben diseñar e implementar una política de seguridad y una arquitectura en la cual el *firewall* sea solo una pieza de la estrategia global.

En este capítulo no se pretende cubrir el problema de seguridad en *Internet* en toda su extensión, lo que merecería un libro completo dedicado a este rubro. Lo que se presenta a continuación son los 20 puntos más relevantes de vulnerabilidad a la que se encuentra expuesta una red al conectarse a *Internet*.

**1. Ataques adivinando Claves Secretas (*Passwords*)**

El uso de *passwords* "adivinables" puede ser la derrota de casi cualquier sistema, y representa el medio más común a través del cual un sistema es accesado. Un *firewall* apropiado debe poder establecer la no-identificación de todos los *passwords* usados por el sistema al que está protegiendo . También debe proveer autentificación adicional, como por ejemplo, autentificación tanto del usuario como de la máquina (a través del domicilio *IP*). También debe limitar el número de intentos de acceso al sistema.

**2. Ataques adivinando Claves Secretas (*passwords* ) por la fuerza bruta**

El uso de la fuerza bruta para adivinar los *passwords* almacenados en el archivo encriptado de *passwords* en *Unix* (*/etc/password*), normalmente tendrá éxito cuando el atacante utilice herramientas como *CRACK* y el archivo tenga el suficiente número de nombres. Este tipo de ataques pueden extraer hasta el 25% de los *passwords* en el archivo, algunos de los cuales también resultan útiles para accesar otros sistemas. Un buen *firewall* protege este archivo contra su transmisión o su alteración.

**3. Espiar sesiones de terminal**

Esta es una técnica en la que el atacante simplemente monitorea a un usuario activo con algún dispositivo de monitoreo de mensajes transmitidos sobre la línea, capturando cada caracter tecleado y buscando una firma a otro sistema. Tales ataques son posibles hasta con las versiones "seguras" de *Unix* tales como *OSF/1*.

**4. Captura del tecleado del password vía un TSR**

Existen varias herramientas utilizadas por los *hackers*, tales como *THIEF* o *GETIT* con las cuales se puede capturar la secuencia de caracteres del *password* en el momento en que se esté tecleando, a través de un módulo residente en memoria (*TSR* ). Con este método de ataque, se requiere que el intruso tenga la posibilidad de tener acceso al dispositivo local de almacenamiento a fin de recuperar el archivo conteniendo los caracteres tecleados.

**5. Ataque con una secuencia de números**

Un ataque de este tipo ocurre cuando un *hacker* predice las direcciones de origen de usuarios válidos, colocando dicho dato en el domicilio *IP* del origen, comprometiendo entonces a cualquier protocolo que use esta dirección para autentificación (como los comandos **r** en *Unix*).

**6. Engañando paquetes de *UDP***

El engañar a los paquetes de *UDP* es fácil para un atacante si la aplicación en cuestión utiliza *UDP* (*User Datagram Protocol*) para transmitir información. El protocolo *UDP* no utiliza protocolo de sincronía (*handshaking*), ni secuencias de números, y envía todos los paquetes para un puerto determinado a un mismo proceso, independientemente de la dirección origen o del número del puerto. Un *firewall* debe de verificar en forma independiente el origen de un paquete *UDP* antes de procesarlo , aunque el origen sea interno a la organización.

**7. Rompiendo conexiones *ICMP***

El *ICMP* (*Internet Control Message Protocol*) es un mecanismo que les notifica a los equipos *host* cuál es la mejor ruta a seguir; termina las conexiones cuando existen problemas en la red y reporta problemas de ruteo. Versiones antiguas ignoran la información de conexiones específicas de un mensaje de *ICMP* y pueden redireccionar todas las conexiones entre un par de *hosts*, reemplazando la conexión original con la nueva. En el mercado negro existen herramientas para *hackers* que utilizan esta técnica.

**8. Redireccionando conexiones de *ICMP***

Los mensajes de *ICMP* pueden ser redireccionados, estableciendo rutas entre un par nuevo de *hosts*. Muchos ruteadores responderían a estas instrucciones, aunque deberían ser configurados para no hacer tal cosa. Un adecuado diseño de *firewall* respondería a estas instrucciones sólo cuando su ruteador confiable le hiciera el requerimiento.

### 9. Ataque por transferencia de zona

El Sistema de Dominio de Nombres (*DNS - Domain Name System*) es una base de datos distribuida que mapea los nombres de los equipos *host* con sus direcciones *IP*. Los servidores de respaldo pueden generar transferencias de zonas al hacer una consulta al servidor de *DNS*, en la que se crea una copia de una porción del espacio de nombres, a fin de que el servidor de respaldos pueda hacer su trabajo. En un ataque de transferencia de zona, los *hackers* pueden hacer requerimientos similares al *DNS*, obteniendo así una lista de *hosts* potencialmente atacables, junto con su domicilio *IP*.

### 10. Ataque de árbol de mapeo inverso

En muchos sistemas, el *DNS* permite que los subárboles sean almacenados en otros servidores. Debido a que *DNS* mantiene pares de árboles --uno con la relación *nombres-dirección* y otro con la relación *dirección-nombres*-- un atacante puede modificar un registro invertido para mostrar el nombre de un *host* confiable asociado al domicilio *IP* del atacante. Así, usando el comando rlogin, puede convencer a la máquina a la que desee entrar, de que es un *host* confiable. Un buen *firewall* debe ser capaz de evitar este tipo de ataque protegiendo el *DNS* ó realizando más autentificaciones, checando el domicilio *IP*.

### 11. Ataques de cache *DNS*

En la mayoría de los *DNS*, salvo las versiones más recientes, es posible precontaminar el *cache* de respuestas del *DNS*, antes de iniciar una llamada. Cuando el equipo destino checa las respuestas válidas en el *cache*, encuentra que un nombre concuerda y permite entonces el ataque. Un *firewall* debe usar tanto autentificación en base a nombre como en base a domicilio.

### 12. Ataque por resolutor de *DNS*

Este tipo de ataque explota una debilidad del resolutor de *DNS* por la que, para ser más eficiente, el resolutor espera conectarse a destinos en los cuales el mapeo en los nombres del dominio es incompleto. De esta manera, un dominio con un nombre en común con un nombre en el domicilio destino, podría interceptar tráfico correspondiente a otro destino.

### 13. Ataque por sobrecarga de *SMTP*

El protocolo *SMTP* (*Simple Mail Transport Protocol*) proporciona un conjunto simple de reglas para transportar mensajes de 7 bits. Este protocolo puede ser imitado fácilmente, y debido a que no hay ningún tipo de autentificación, cualquier atacante podría introducir mensajes manualmente. Debido a que un atacante puede especificar manualmente cualquier origen para el correo, es posible que éste sobrecargue el sistema con mensajes falsos, ocasionando un ataque por servicio denegado. En este tipo de ataques el correo electrónico pierde su funcionalidad, aunque el servidor no se caiga por el peso del alto número de mensajes falsos que reciba.

### 14. Expansión de Alias

El protocolo *SMTP* permite utilizar *alias* al transmitir correo. Pero existen comandos como vrfy que pueden traducir los alias a los nombres de entrada (*login names*). El comando expn puede expandir los *alias* de las listas de correo.

### 15. Ataque por *sendmail*

*sendmail* es la manera más común de implementación de *SMTP* y junto con miles de líneas de código, también tiene muchos errores. Este es un programa que no debe correrse bajo root, por considerarse extremadamente peligroso. *sendmail* no requiere ejecutarse como root al menos que se desee hacer una entrega especial en equipos *gateway*. Existen alternativas a este comando, incluyendo *front-ends* más seguros.

### 16. Ataque en encabezado *MIME*

Puede ser que un sistema de correo que esté recibiendo correo codificado con *MIME* (*Multipurpose Internet Mail Extensions*) pretenda transportar instrucciones en el encabezado del mensaje *MIME*. Tales instrucciones, si no son cuidadosamente evaluadas antes de su ejecución, pueden sobreescribir el archivo rhosts.

### 17. Ejecutables anexos al Correo Electrónico

Si un atacante puede entrar a un sistema de Correo Electrónico para enviar un mensaje, éste no tendrá problemas para enviar por correo un programa anexo. El programa puede ser diseñado para hacer cualquier cosa que el atacante desee y será un caballo de Troya exitoso si aparenta hacer algo útil para el dueño del recipiente de correo.

### 18. Ataques vía *Telnet* corrupto

*Telnet* le proporciona a los usuarios acceso a terminales. En un sistema inseguro, el programa telnet puede ser usado por un atacante para capturar el nombre de usuario, clave confidencial o incluso una sesión completa. Otra alternativa es que el atacante reemplace el programa telnet para dejar la sesión abierta después de que el usuario piensa que ya se desfirmó.

### 19. Ataque con finger

El protocolo finger le proporciona a los atacantes información muy valiosa sobre los usuarios, incluyendo nombre, domicilio de correo electrónico, fecha de ultimo acceso, de dónde se conectó el usuario la ultima vez, etc., por lo que es considerado como un servicio peligroso.

### 20. Ataques por *NIS*

El *NIS* (*Network Information Services*), anteriormente conocido como *YP* (*Yellow Pages*) es un servicio que distribuye muchas bases de datos importantes desde un servidor central hacia sus clientes. Dichas bases de datos incluyen el archivo de *passwords*, la tabla de domicilios de *host* y llaves públicas y privadas utilizadas para *RPC* seguro. Este tipo de ataque consiste en indicar a *NIS* que transfiera uno o más de estos archivos al atacante.

### *El Caso del Security First Network Bank (SFNB)*

*SFNB* es la primera organización bancaria aprobada por los reguladores gubernamentales para ejecutar sus servicios financieros en *Internet*. El "banco virtual" permite a clientes con cuentas usar sus navegadores para transferir fondos, programar pagos, escribir cheques electrónicos, conciliar sus estados de cuentas – y en corto plazo – proporcionar todos los servicios de un banco las 24 horas de los 365 días del año. La arquitectura de seguridad del *SFNB* combina lo ultimo en tecnología de comercio electrónico, tecnologías de sistemas operativos confiables y tecnologías de *firewalls* para proteger al banco de las amenazas que un servicio en línea puede encontrar.

*SFNB* está hecho de dos distintas partes. La primera es el *Servidor de Información*, el área que clientes potenciales usan para aprender acerca del banco y sus servicios. Una vez que el cliente se decide a abrir una cuenta, el o ella emplean una forma de registro de seguridad para enviar un mensaje encriptado al *Servidor Bancario*, el cual contiene las actuales aplicaciones bancarias.

Usando la información proporcionada en la forma de registro, el banco verifica la información de la cuenta y crea una nueva cuenta para el cliente. Un paquete de creación de cuenta, conteniendo el nombre del usuario y el *password* necesarios para accesar la cuenta, es enviado al cliente a través del Servicio Postal. Este paquete puede eventualmente contener un dispositivo personal de autentificación tal como una tarjeta inteligente, o un diskette conteniendo una llave privada encriptada y certificada.

Los clientes se comunican con el banco usando su navegador. Cada transacción que el cliente envía al banco es encriptada para proteger la información que viaja sobre la red. El servidor del banco recibe la transacción, decripta el mensaje y ejecuta el servicio solicitado.

El *Servidor Bancario* corre en un *Sistema Operativo Confiable* , llamado *CMW+*. Security First es la primera aplicación comercial de la tecnología de sistema operativo confiable. Desarrollado como una plataforma de seguridad multi-nivel para instalaciones gubernamentales por *SecureWare, Inc.*, *CMW+* proporciona una jerarquía de autorizaciones y privilegios que protegen las funciones del sistema de interferencias externas.

Un mecanismo de *separación de información* dentro del sistema operativo crea una pared entre el ambiente de red y las aplicaciones internas del banco. La red recibe las solicitudes de los usuarios y valida su identificación. Una aplicación válida la seguridad y rutea la solicitud al ambiente interno del banco, donde ésta es procesada y regresada la respuesta. Debido a esta separación, nada del ambiente externo puede tocar las funciones bancarias. Más allá, ningún proceso externo puede alterar las operaciones internas del banco.

Security First utiliza varias capas de tecnología para asegurar la confidencialidad de las transacciones a través de *Internet*. La seguridad comienza con el navegador. El protocolo *SSL* es usado para proporcionar privacidad en el flujo de datos entre el navegador y el servidor del banco.

Cuando una cuenta de un cliente es creada, el banco asigna un *password* el cual es enviado al cliente con una carta de verificación de cuenta. Adicional a la protección del *password*, Security First también provee de un servidor de autentificación usando lo ultimo en criptografía de llave pública.

Se utilizan pares de llaves públicas/privadas son usadas específicamente para autentificación. La llave pública puede se distribuida usando un certificado que verifica la identidad del dueño. La llave privada se mantiene en secreto. Un mensaje encriptado con una llave pública puede ser leído sólo después de haberlo decriptado con la llave privada.

Para iniciar una transacción, el cliente usa su navegador para enviar un mensaje seguro vía *SSL* al banco. El banco responde enviando un certificado el cual contiene la llave pública del banco. El navegador autentifica el certificado, entonces genera una llave de sesión la cual es usada para encriptar los datos que están viajando entre el navegador del cliente y el servidor del banco.

La llave de sesión es encriptada usando la llave pública del banco, y regresada al banco. El banco decripta este mensaje usando esta llave privada, y entonces usa la llave de sesión para el resto de la comunicación. Por medio del intercambio de mensajes usando el par de llaves públicas/privadas, el cliente puede estar seguro que se está comunicando realmente con el banco, y no con un tercero que está tratando de interceptar la transacción.

Security First, además está protegido de intrusos con un sistema de ruteadores de filtración y *firewalls*. Este sistema forma una barrera entre el mundo externo de *Internet* y la red interna del banco. EL ruteador verifica el origen y el destino de cada paquete de red, y determina cuando debe dejarlo pasar. El acceso es denegado si el paquete no está dirigido a un servicio específico disponible.

El *firewall* es usado para proteger la red de servicios a clientes del banco de la *Internet*. Todo el tráfico *IP* de entrada es direccionado al *firewall*, el cual está diseñado para permitir sólo correos electrónicos dentro del ambiente de servicios al cliente.

El tráfico a través del *firewall* está sujeto a un proceso especial llamado *proxy*, el cual opera en forma similar a un ruteador de filtración, verificando el origen y destino de cada paquete de información. El proceso *proxy* entonces cambia la dirección *IP* del paquete para entregarlo al sitio apropiado dentro de la red de servicios al cliente. De esta manera, todas las direcciones internas están protegidas de acceso externos, y la estructura de la red interna del banco es invisible a los observadores externos.

# CAPITULO IV
# Medios de Entrega de Servicios Bancarios

La Banca ya no es lo que alguna vez fué y no hay nada más crítico para su futuro que el de las sucursales mismas y la manera en que éstas encajan dentro de un esquema integral de medios de entrega. Las sucursales físicas solían emplearse como una de las principales medidas de los bancos en general para tener contacto con sus clientes. En la actualidad, si bien siguen siendo un ingrediente muy importante, son solamente uno entre muchos otros medios a considerar. Lo que hay que hacer ahora es comprender la conducta evolutiva de los clientes : cómo están cambiando y de qué manera estos cambios están afectando a la Banca.

Si se analiza el comportamiento de la clientela en el transcurso del tiempo, el mensaje es bastante claro : las investigaciones muestran que menos de la tercera parte de los clientes visitan las sucursales una o más veces a la semana ; además con mayor frecuencia muchos de ellos ya no consideran a la sucursal como el punto focal de su relación con el banco, sino que están tendiendo a utilizar toda la gama de medios de entrega que se ponen a su disposición.

Otra de las cosas que se están haciendo evidentes es que, entre más jóvenes sean los clientes, más a gusto se sienten con los medios de entrega diferentes a las sucursales. Estudios recientes han mostrado que la generación de personas que ahora tienen entre 54 y 64 años de edad, emplean la sucursal para realizar casi la mitad de sus operaciones bancarias ; por el contrario, las personas cuya edad anda alrededor de los 35 años las utilizan para menos de la tercera parte.

La tendencia parece clara : a medida que transcurra el tiempo, la sucursal dejará de ser más y más el centro de la relación bancaria para la mayoría de los clientes. Seguirá teniendo su importancia, sí, pero ya no será el medio de entrega por excelencia.

Son varias las fuerzas impulsoras del mercado que aparentemente están guiando este cambio evolutivo en el comportamiento de la clientela ; una de ellas es que la comodidad de la cercanía física de la sucursal está siendo cada vez menos relevante ; por el contrario, el horario de atención -tanto entre semana como en los fines de semana- es más importante para muchas personas, que la cercanía física.

Por otro lado, con la creciente importancia del medio de entrega telefónico y con el surgimiento de la *banca en el hogar* a través de computadoras personales, el concepto general de la ubicación física de un banco se está volviendo prácticamente irrelevante para el manejo de las transacciones rutinarias del día con día.

Muchos clientes no sólo se sienten a gusto con los *cajeros automáticos* y el servicio por teléfono, sino incluso también con las tecnologías más sofisticadas, y todo indica que este segmento de clientes tiende a aumentar cada día. Este segmento de clientes, a los que más les gusta la tecnología, son también, por lo general, los clientes más rentables y también los más exigentes. Ellos serán los mejores prospectos de compra de los productos altamente tecnificados que irán apareciendo en los proximos años, pero también serán los más peleados por otras compañías financieras.

El reto consistirá en satisfacer las expectativas de este segmento, sin olvidar a los que ya hayan dado el salto a la "Supercarretera de Información". Esto significa tener que operar en dos mundos simultáneamente: el de las sucursales tradicionales representativas del pasado y el del futuro altamente tecnificado.

Para afrontar este reto, antes que nada los bancos deben comprender las preferencias de la clientela, y luego actuar en consecuencia. Así, es posible aumentar las probabilidades de éxito si se toman en cuenta algunos factores importantes:

- Las operaciones, las cuentas de cheques y ahorro, las inversiones y los productos de crédito seguirán independizándose. cada vez más, de la ubicación física del banco.
- El teléfono se convertirá en el medio de entrega preferido en lo que toca a la venta de productos y el manejo de servicios bancarios.
- Por lo que los medios de entrega jugarán un papel muy importante en los próximos años en el desarrollo y consolidación de las instituciones bancarias.

## 4.1 Evolución de los Medios de Entrega de Servicios Bancarios

Los *cajeros automáticos* fueron el primer intento que realizaron los bancos para reducir las cargas de trabajo durante las horas pico en las sucursales ; posteriormente hizo su aparición *la banca por teléfono* y, más recientemente, ha surgidó otra corriente, en la medida en que una gran cantidad de instituciones se han ido incorporando al esquema de *la banca en el hogar* (vía computadoras personales).

Sin embargo, actualmente, *la banca en el hogar* tiende a ser más propietaria que basada en *Internet*, aunque esto está empezando a cambiar. Así, la *banca en el hogar* ha salido ya de la etapa de pruebas piloto para adentrarse a una mayor proliferación. El escenario más común en la actualidad es el ofrecimiento por parte del banco de una solución de *software* propietario que se instala en la computadora personal del cliente. Las funciones disponibles son similares a las que se tienen bajo el esquema de la *banca por teléfono* y la mayoría de ellas se pueden realizar sin el cobro de una comisión adicional a la tarifa del servicio en sí, a excepción de ciertas funciones específicas (como el pago a terceros) que sí se cobra por separado.

Los clientes de la *banca en el hogar* pueden también establecer ligas entre la información concerniente a sus cuentas bancarias y los programas de software para finanzas personales como *Quicken* de *Intuit* o *Money* de *Microsoft* ; así, los usuarios pueden descargar información relacionada con sus cuentas (como saldos y movimientos) de las aplicaciones del banco y cargarla al paquete financiero, el cual puede llevar un seguimiento de los portafolios y generar reportes específicos.

### *Cajeros Automáticos*

Los *cajeros automáticos* fueron los primeros medios de entrega diferentes a las sucursales tradicionales que los bancos pusieron a disposición de su clientela con el objetivo de reducir las cargas de trabajo en las horas pico y de ampliar los horarios de servicio (24 horas 7 días a la semana).

Estos empezaron proporcionando una gama muy reducida de servicios, tales como consulta de saldos y retiro de efectivo. Conforme fue evolucionando la capacidad de procesamiento de estos dispositivos y de las telecomunicaciones, la gama de servicios se fue ampliando a pagos de servicios, de tarjetas, consulta de estados de cuenta,etc.

En los ultimos años, con la aparición de las tecnologías de multimedia (voz, video), ésta se ha ido integrando a los *cajeros automáticos*, haciendo más sencillo el uso y acceso a los servicios que se proporcionan por este medio de entrega. Los bancos están aprovechando esta tecnología para que los *cajeros automáticos* realicen funciones de promoción de productos y servicios bancarios, aprovechando los momentos cuando no se está realizando ninguna operación bancaria, a un costo relativamente bajo, lo que hace más rentable para el banco el ofrecer servicios por este medio.

### *Banca por Teléfono*

Las instituciones bancarias, buscando mejorar la atención a sus clientes existentes y atraer nuevos clientes, desarrollaron el servicio de *banca por teléfono*.

Por este medio de entrega los clientes pueden realizar sus operaciones bancarias desde su oficina u hogar y en horarios más amplios que los de las sucursales.

En un principio este servicio se proporcionaba a través de operadores que recibían las llamadas telefónicas que realizaban los clientes y por medio de terminales que se encontraban conectadas al computador central, atiendian la transacciones que los clientes solicitaban.

Con la aparición de los equipos de audio-respuesta y la evolución de la telefonía digital, el servicio de *banca por teléfono* sufrió un cambio en su manera de operar al permitir que los clientes realizaran por si mismos sus operaciones sin la necesidad de un operador. Esto permitó a las instituciones bancarias ampliar su cobertura de clientes con el mismo costo operativo que tenían y reducir el número de operadores con que contaban, ya que con esta tecnología se tiene mayor capacidad de recepción y atención de llamadas telefónicas, lo que hizo más rentable para los bancos el ofrecer este tipo de servicio.

Actualmente la mayoría de las instituciones financieras cuenta con esquemas combinados del servicio de *banca por teléfono* por medio de equipos de audiorespuesta y por medio de operadores, estos últimos utilizados como respaldo en caso de falla de los equipos de audiorespuesta, para dar información a los clientes de como operar el sistema, para aclaraciones o para realizar operaciones más especializadas.

### *Banca Electrónica*

La banca electrónica es el proceso de realizar todo el trabajo bancario a través de una computadora, actualmente a través de enlaces telefónicos a los bancos y en un futuro cercano usando *Internet*. Es el uso de una computadora para recuperar y procesar datos bancarios e iniciar transacciones directamente con un banco vía una red de telecomunicaciones. El acceso a la *banca-en-línea* por medio de la WWW habilita a las entidades financieras la entrega de servicios bancarios completos y de pagos a sus clientes. La banca por medio de la WWW proporciona un medio poderoso y conveniente para fortalecer las relaciones existentes con clientes y adquirir nuevos clientes.

La banca electrónica ofrece mucha ventajas a los clientes y al mundo de los negocios. La mayor ventaja de la banca electrónica es su facilidad de acceso hacia ella. Puede permitir a los bancos proporcionar sus servicios las 24 horas del día, los 7 días de la semana. La banca electrónica también puede permitir a los clientes obtener información de sus cuentas, estados de cuentas, realizar transferencias electrónicas y pagos a cualquier hora del día y en cualquier lugar del mundo. La banca electrónica también ofrece confidencialidad. Con esto se elimina la necesidad de ir a un banco.

Algunos bancos, principalmente los más grandes, empezaron a ofrecer a sus principales clientes y a las empresas más grandes sus servicios de banca electrónica (*banca-en-el-hogar*), mediante terminales instaladas en sus oficinas, la cuales se conectadaban al computador central del banco desde donde los clientes realizaban sus operaciones bancarias.

Utilizando una aplicación especialmente diseñada y desarrollada para brindar estos servicios, donde ya se empezaban a manejar ciertos aspectos de seguridad para el cliente y el banco, como por ejemplo, el uso de números de clientes y claves personales con diferentes niveles de facultades para realizar operaciones.

Estas terminales estaban conectadas por medio de enlaces de radio, líneas telefónicas privadas o microondas.

Estos sevicios de banca electrónica permitió a las tesorerías de las empresas contar con información oportuna de sus cuentas y administrar sus recursos en forma eficiente con la ventaja de contar con horarios más amplios y sin la necesidad de desplazarse a una sucursal.

Al surgir las computadoras personales los bancos aprovecharon esta nueva tecnología para ofrecer un nuevo medio de entrega, *el banco-en-el-hogar/empresa.*

Los bancos desarrollaron el *software* e instalaron la infraestructura telefónica y de telecomunicaciones necesaria para que los clientes pudieran realizar sus operaciones bancarias desde su computadora personal, conectándose al computador central del banco por medio de un *modem* y una línea telefónica.

Aprovechando la capacidad de procesamiento y almacenamiento con que cuentan las computadoras personales, se ampliaron los servicios ofrecidos a los clientes, como información de los movimientos de sus cuentas para realizar conciliaciones sin tener que esperar la llegada de estados de cuenta ; la programación de sus operaciones para realizarse en fechas futuras ; así mismo, se mejoró la interfase del sistema para facilitar su uso.

Este medio de entrega abrió a los bancos la posibilidad de ofrecer sus servicios de banca electrónica a empresas pequeñas y medianas así como a personas físicas, logrando así una mayor penetración en el mercado.

Actualmente, los bancos ya cuentan con versiones de sistema de banca electrónica en el ambiente gráfico Windows, lo que proporciona al cliente mayor facilidad de interacción.

### *Kioscos de Autoservicio*

Otra modalidad de los medios de entrega de servicios de banca electrónica son los kioscos de autoservicio o interactivos colocados dentro de las sucursales o en lugares públicos con mucha afluencia, como tiendas de autoservicio, centros comerciales. Por medio de este dispositivo los clientes pueden realizar sus operaciones bancarias, que no impliquen manejo de efectivo, sin necesidad de formarse en las ventanillas de las sucursales y en forma más rapida. Además por este medio se pueden realizar consultas de información sobre los productos que los bancos ofrecen, consultas de información financiera, realizar simulaciones de créditos, etc. Esto ayuda a descongestionar las ventanillas, disminuir el gasto operativo de las sucursales y hacer más rápida la visita del cliente a la sucursal.

Estos kioscos pueden ser cajeros automáticos o computadoras personales equipadas con equipo de multimedia y conectados al computador central del banco.

Los kioscos interactivos con capacidad de multimedia son una buena alternativa dentro de los medios de entrega ya que pueden efectuar casi todas las actividades que se llevan a cabo en las sucursales, incluyendo el diálogo y asesoría con personas, con la eficiencia de un cajero automático, pero hay quienes opinan que puede no ser una herramienta muy eficaz para la venta cruzada de productos (ya que piensan que las operaciones bancarias deberían manejarse por dispositivos de autoservicio, y que la función de ventas no debería desligarse de la sucursal tradicional) ; pero para que los kioscos tengan éxito es indispensable que incorporen varias funciones y que la clientela los perciba con un cierto valor agregado, éstos tienen que diseñarse de tal forma que encajen perfectamente dentro de la estrategia integral de medios de entrega de la banca al menudeo que mantiene cada institución.

## 4.2 Ventajas competitivas de los Medios de Entrega Electrónicos

La banca es un negocio en donde la gran mayoría de las instituciones ofrecen prácticamente los mismos tipos de productos a través de medios de entrega similares y más o menos al mismo precio. Esto significa que resulta crucial poder entregar un mejor servicio en forma consistente, así como saber establecer una identidad de marca que los consumidores puedan identificar siempre, asociándola a algo positivo. Así las cosas, es indispensable establecer una diferenciación clara de un banco con respecto a su competencia. Pero también es de suma importancia empezar a moverse desde ahora ; no siempre se tiene que ser el primero en lanzar al mercado algo novedoso, ni tampoco ser el primero en hacer cierto tipo de cambios en las sucursales, pero de lo que sí hay que cuidarse es de no ser el último en reaccionar.

Uno de los aspectos clave en el futuro -de hecho lo es en el presente- es la habilidad para atender y satisfacer a la clientela por medio de una amplia gama de medios de entrega, lo cual significa contar con una red de distribución que les brinde una serie de opciones ; pues bien, para ello se requiere tener una sólida penetración en el mercado.

Las sucursales, incluso en el futuro altamente tecnificado, seguirán siendo uno de los activos más importantes. Empero, en los años venideros, tanto los banqueros como su clientela las utilizarán en forma diferente a como lo hacían en el pasado. Para la mayoría de los bancos, las sucursales han estado tradicionalmente orientadas al manejo de las operaciones. El reto consiste en cambiar dicho enfoque, es decir, reducir su papel de gran centro operacional con un alto costo, al de un centro en donde se generen utilidades. Algunas instituciones ya han puesto la muestra : Citibank ha tenido un éxito notable al construir o reconstruir sus sucursales en torno a los medios de autoservicio, reorientando al mismo tiempo su personal hacia actividades relacionadas con las ventas. El Bank of America en Nuevo México está utilizando intensamente el video interactivo en sus sucursales ubicadas en los centros comerciales, en tanto que el Huntington Bancshares en Ohio ha instalado una cantidad de sucursales "sin cajeros", completamente automatizadas, empleando las terminales llamadas "teléfonos inteligentes" en lugares específicos.

Uno de los grandes retos de los banqueros será buscar la manera en que todos los diferentes medios de entrega den buenos resultados en conjunto, incluyendo las sucursales, el teléfono, el *cajero automático* o *Internet*. Pero quizás el reto más grande sea no dejar a un lado a los clientes tradicionales. Esto significa que deben poder manejar los bancos con un pie en las sucursales tradicionales del pasado y con el otro en la intensa tecnología actual y del futuro.

Para que los bancos cuenten con una posición bastante sólida, deberán contar con una estrategia integral de medios de entrega. Se piensa que unos de los puntos que los clientes valoran de los bancos es la facilidad con que pueden accesar sus cuentas, lo cual implica poner a disposición del segmento masivo del mercado una variedad de medios de entrega que complemente a la red de sucursales tradicionales dado que la clientela requiere información constante para poder manejar adecuadamente sus finanzas.

Entre los diversos servicios que los bancos pueden ofrecer por diferentes medios de entrega están:

- Teléfono - Se ofrece información de cuentas, transferencia de fondos, pagos a terceros, la solicitud de talonarios de cheques, aclaraciones sobre préstamos, etc.
- Audio-Respuesta - Se ofrece para hacer consultas y movimientos.
- Cajeros Automáticos (ATMs) - Permiten realizar disposiciones de efectivo, cambio de número confidencial, depósitos, transferencia de fondos, pago de servicios, inversiones en el mercado de dinero, pagos de tarjetas de crédito e impresiones de estados de cuenta.

El objetivo para los bancos es sacar la mayor cantidad de consultas y operaciones fuera de las sucursales, lo cual les redundaría, por un lado, en promover una mayor satisfacción y comodidad para el cliente y, por el otro, en reducir considerablemente los costos.

Parte fundamental de la estrategia de medios de entrega de los bancos, es la planeación integral de los mismos, para lo cual se deben considerar los factores siguientes :

- El aspecto cultural de la clientela y del propio banco.
- El conocimiento del costo operativo por transacción y tipo de medio.
- Tener bien definidos los segmentos del mercado.
- Conocer las necesidades y preferencias de cada segmento.
- Contar con la infraestructura operativa y tecnológica que permita la entrega de los productos y servicios de cada uno de los medios.

## 4.3 Tendencias de los Medios de Entrega de Servicios Bancarios

A medida que nos acerquemos al año 2000 no sólo estaremos cerrando un siglo, sino también dando fin a la era de la "banca tradicional".

Sólo aquellos bancos que sean capaces de adaptarse al nuevo ambiente, que por cierto ya está empezando a surgir, son los que podrán sobrevivir en el próximo siglo. Así como algunos dinosaurios tuvieron que evolucionar hacia otro tipo de animales, y los que no lo lograron desaparecieron, de igual forma los bancos tendrán que evolucionar hacia un nuevo tipo de organizaciones o, de lo contrario, deberán resignarse a la extinción. Y la prueba de fuego de su adaptabilidad será precisamente lo que hagan con su red de sucursales. El futuro estará representado por la banca electrónica y los negocios fuera de sus oficinas, aunado a la atención personalizada a los clientes de alto nivel. Los competidores del futuro, cada vez más, serán compañías tradicionalmente fuera del sector bancario como Microsoft, IBM, AT&T, Fidelity, GE, Time.Warner, EDS, GM y First Data.

Por mucho tiempo, el incentivo más importante para muchos bancos ha sido atraer más gente a sus sucursales, con la idea de que con esto se tenían más oportunidades de vender más productos y servicios. Sin embargo, en la realidad, eso no es cierto la mayoría de las veces.

La oportunidad de venta se da realmente cuando el cliente establece una cuenta, o a través de la mercadotecnia dirigida, o por medio del telemercadeo ; es muy difícil que se pueda vender un certificado de depósito o un fondo de inversión en el momento en que se está cambiando un cheque en la ventanilla del cajero.

Ahora bien, la diferencia entre lo que el cliente desea y lo que le ofrece el banco, genera una buena oportunidad para el ingreso al mercado de nuevos proveedores, más innovadores y creativos. Pero los banqueros no son tan ingenuos ; se están dando cuenta de ésto y están invirtiendo muchos millones de dólares en medios de entrega alternos, al igual que en conocer la relación integral de negocios con su clientela.

No son muy fuertes las barreras que se interponen al ingreso de nuevos proveedores al sector bancario y financiero, en virtud de los cambios estructurales y tecnológicos. No habrá barrera que frene a compañías como Microsoft. Y mientras esto ocurre, los bancos desgraciadamente no cuentan con mucho tiempo para responder. Pudiera ser demasiado tarde cuando se percaten de que sus actuales medios de entrega son obsoletos.

Por otra parte, la banca de menudeo está enfrascada en una profunda reorientación, caracterizada por una serie de cambios importantes en la forma de desarrollar y distribuir los productos al mercado, así como por la creciente necesidad que muestran los bancos por conocer mejor las áreas de negocio que operan así como los costos asociados a ellas, según un estudio conducido por Deloitte & Touche LLP llamado "The Future of Retail Banking - A Global Perspective".

Otro elemento que está impulsando los cambios en el sector financiero y bancario es la tecnología, la cual está actuando como catalizador del surgimiento de nuevas tendencias en el sector bancario mundial ; en concreto en la aceleración de la especialización de la banca de consumo.

En este movimiento que se remonta a varios años atrás, la informática ha estado posibilitando la creación de medios alternos (ATM´s, teléfonos inteligentes, televisión interactiva y computadoras personales) para distribuir productos y servicios bancarios con costos más bajos que los de una sucursal. En otras palabras, el consumidor ha estado pagando por los costos fijos de la infraestrutura de las sucursales. Entonces, las presiones de costo y eficiencia del servicio han provocando la reducción de sucursales hasta en un 50%.

Ante las fuertes presiones derivadas de los altos costos operativos, los bancos desde hace algún tiempo han tratado de reducir los elevados costos que implican las sucursales tradicionales, induciendo a la clientela al uso de canales de entrega electrónicos que son más rentables.

Dentro de los próximos cinco a diez años se estima que desaparecerán algo así como la mitad de las sucursales bancarias y alrededor de 450,000 empleos en el sector bancario de E.U., debido a los avances tecnológicos y a la incesante presión competitiva por parte de los intermediarios no financieros, lo cual obliga a que los bancos se vuelvan más innovadores y a que orienten más sus productos y servicios financieros a la clientela.

Se ha encontrado que la constante desregulación y la proliferación de la automatización a gran escala serán dos de las principales fuerzas impulsoras del cambio en el sector financiero en forma parecida al que han experimentado las industrias de las telecomunicaciones y la aviación comercial. Es más, la tecnología y la aparición del "dinero digital" ya están empezando a erosionar el monopolio bancario como centro para la transmisión del efectivo.

Son varias las fuerzas que han venido afectando en forma simultánea el futuro de la banca al menudeo basada en sucursales, entre ellas : el exceso de la capacidad instalada y los costos elevados ; las nuevas tecnologías ; los cambios en las expectativas y necesidades de la clientela ; y la competencia cada vez más fuerte. A medida que han ido surgiendo nuevas tecnologías en relación a los mecanismos de distribución (como los *cajeros automáticos*, la *banca directa* y las unidades de audiorespuesta), se esperaba que ayudarían a reducir la afluencia de tráfico en las sucursales, al mismo tiempo que permitirían reducir la cantidad y el tamaño de ellas. Pero en los Estados Unidos se ha incrementado la afluencia a las sucursales desde la década de los 70s, cuando precisamente empezó la introducción de los medios electrónicos.

Durante esta década de los 90s, se espera que el porcentaje de las transacciones que se efectúan por los medios de autoservicio (incluyendo *cajeros automáticos*, terminales punto de venta, *banca en el hogar*, *banca directa*, etc.) pase de 22 a 45% ; por el contrario, se piensa que las operaciones en sucursal se reducirán a sólo un 40% del total de las transacciones bancarias.[1]

[1] Revista Retail Banker 10/oct/95

La velocidad de cambio sin precedentes en el sector cambiario está forzando a la banca comercial a replantear sus estrategias en materia de medios de entrega y de enfoque al cliente. El cambio de paradigma implica que los bancos abandonen su filosofía tradicional de "todos los productos para todos los clientes", en aras de emplear la Tecnología de la Información para robustecer sus posiciones actuales en el mercado y capitalizar sus fortalezas en aquellas áreas que dominan y en las que se puede sacar más provecho de sus esfuerzos por elevar la rentabilidad.

Un banco existoso se caracterizará por diversificar sus medios de entrega a efecto de acomodar a un universo de clientes cada vez más sofisticados.

En la actualidad los bancos han empezado a incursionar en la banca electrónica interactiva, es decir, a través de *Internet* y, especialmente, de los "home pages" en la WWW, donde el principal problema ya no es tanto la seguridad, dado que ya se puede garantizar casi cualquier transacción y comunicación entre casi cualquier servidor y para casi cualquier consumidor.

Independientemente que la WWW se convierta en un medio de entrega efectivo o no, los bancos deben comprender que no existe un solo medio que satisfaga a todos y cada uno de los clientes de la banca al menudeo.

Los bancos deben reorientar a su clientela fuera de las ventanillas de sucursal -varios de ellos eliminando, o al menos reduciendo, las comisiones por el uso de los *cajeros automáticos* o de la *banca por teléfono*- aunque varios estudios realizados revelan que las sucursales siguen siendo el medio más popular entre un buen segmento de la clientela ; lo importante en este caso es mantener estos canales convencionales, pero buscando la manera de hacerlos más eficientes y rentables.

## *El futuro de la banca electrónica.*

La banca electrónica ha estado evolucionando en las últimas dos decadas. En los dos últimos años se ha visto el mayor incremento en el número de bancos que están ofreciendo servicios de banca electrónica y en el número de personas que están haciendo uso de estos servicios. La banca electrónica puede parecer un asunto menor comparado a la consolidación de la banca o la competencia con las instituiciones no-bancarias, pero asumiendo la extensa visión del consumidor de servicios financieros, la banca electrónica es una manera crucial de adaptarse y crecer en un ambiente cada vez más competitivo.

Primero, se verá cambiar la perspectiva de *banca-en-casa por banca-directa*, la cual incluye *banca por telefóno, banca por PC, cajeros automáticos* y kioscos, pagos en *Internet* y cualquier otra cosa que no implique una persona en una sucursal. Dentro de 5 a 10 años se puede asumir que esencialmente todos los bancos ofrecerán servicos de *banca-directa* con múltiples maneras de accesar al banco. Casi todos los productos y servicios de los bancos pueden estar disponibles directamente a los consumidores : la *banca-directa* puede ser, la mayor parte del tiempo, el banco para la mayoría de las personas y de los productos. Los bancos necesitan racionar cuidadosamente el uso de la interacción humana, de un alto costo, dirigiendola a actividades de gran valor : ventas de productos complejos, establecer nuevas relaciones, resolver problemas de clientes. En tal mundo, las bases reales para la innovación y diferenciación son los productos y servicios por sí mismos, no los canales de acceso. Los consumidores pueden juzgar al banco en servicio, conveniencia, y en como el banco hace la integración de servicios.

Para el diseño de la integración de productos, para proporcionar mejor servicio a los clientes, y hacerlo a un costo efectivo, los bancos deben enfocarse en los sistemas estratégicos de *back-end* y su conectividad hacia los clientes. La construción de los sistemas bancarios debe enfocarse hacia los clientes en lugar de sistemas separados por productos o medio de acceso, ésta es la mejor estrategia de inversión en sistemas que los bancos puedan hacer. La conectividad de todos los productos a través de múltiples medios de acceso requiere construir una infraestructura enfocada al cliente para la *banca-directa*. Estas son las estrategias reales que deben guiar las inversiones en sistemas de información.

Los bancos deben escoger qué vendedores y sistemas pueden ayudarlos a alcanzar estos objetivos y poder ofrecer *banca vía PC* como uno de los canales de *banca-directa*.

Para *banca-directa vía PC, Internet* puede ser la comunicación más efectiva posible. Se puede pensar en la WWW como un kiosco en casa, excepto que ésta es más rica y más dinámica. *Internet* puede tener la mayor penetración en los hogares que cualquier otro medio de comunicación de computadoras. La seguridad es un problema de corto término. Las apropiadas tecnologías de seguridad ya existen. A largo plazo, *Internet* gana debido a su amplio uso, contínua innovación tecnológica y contínua reducción de costo.

La *banca-directa* podrá obtener otro impulso cuando los consumidores puedan finalmente tomar ventaja del hecho que el dinero ya es casi digital. Todos los cambios de digital a no-digital, efectivo o cheques, y regresarlo otra vez son caros e incovenientes para cualquiera que esté involucrado. La compensación de banco a banco ya es electrónica. Pero, el lado electrónico de la institución está separado del lado del consumidor por cientos de papeles y manejos. Los bancos pueden sostener la posición dominante en depósitos y transacciones de servicios-- y la *banca-directa* es la llave para ligar al consumidor y el sistema de pagos instituicional--. Los servicios de transacciones básicas se pueden convertir en digitales cuando una persona está en casa pagando cuentas o comprando en la Web o pagando tarjetas de crédito. El mundo real y el mundo cibernético pueden convergir y el mundo real puede generar un volumen real en pagos electrónicos.

En este mundo, la *banca-directa* es crucial para los bancos para competir, para ser más eficientes, para ofrecer mejores productos, y para entregar servicios de alta calidad a los consumidores. Los bancos que no se muevan pueden ser dejados con consumidores de bajo valor, con altos costos de operación, productos poco flexibles y menor habilidad para competir otras instituciones no-bancarias.

En el futuro, los consumidores van a querer que sus instituciones financieras les ofrezcan paquetes integrados de servicios financieros (como fondos de inversión, seguros, productos de ahorro, etc.), entregados a través de varios tipos de medios (sucursales, centros telefónicos, *la banca en el hogar vía PC*, etc.).

Según el estudio "Creating the Value Network 1996" de Ernst & Young y la American Bankers Association (ABA), los bancos tienen que establecer proactivamente alianzas estratégicas con otros proveedores de servicios financieros y explotar al máximo sus ventajas competitivas particulares, con el fin de poder ofrecer a la clientela una gama más completa de productos, entregados a través de diversos medios. Estas alianzas le permitirán ofrecer productos diferentes a los tradicionales por medio de lo que el estudio denomina como "redes de valor". (Una red de valor es una forma de ofrecer a los consumidores los servicios financieros merced al establecimiento de alianzas y asociaciones con diversas firmas, cada una de ellas concentrándose en la competencia que domina).

Por otro lado, el estudio menciona que, de los 100 grupos financieros encuestados, el 84% de ellos tienen planeado reducir la cantidad de sucursales no tradicionales, incluyendo minisucursales, kioscos y sucursales dentro de supermercados.

También, los clientes realizarán la mayor parte de sus transacciones vías las sucursales. Mientras que declinará el volumen de las transacciones efectuadas por sucursal en un tercio para 1998, otros medios de *entrega como los cajeros automáticos, los centros telefónicos y la* banca en el hogar, elevarán el volumen operado.

El análisis estima que la cantidad de transacciones bancarias efectuadas por teléfono crecerá en un 50% para 1998, en tanto que las operadas a través de computadoras personales crecerá en un sorprendente 600% (ver. Fig. 4.1).

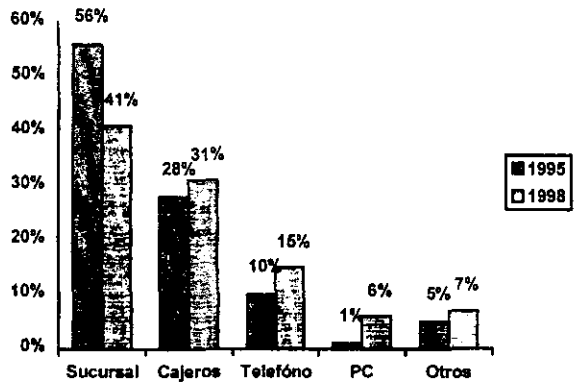



***Fig. 4.1 Transacciones por Medio de Entrega 1995 vs 1998***

Para el mismo año 1998, *los cajeros automáticos* manejarán transacciones más complejas que las disposiciones de efectivo y los depósitos. Más de la mitad (54%) de los entrevistados dijeron que tienen planes de usar los *cajeros automáticos* para la venta cruzada de productos, incluyendo un 21% que expresó planear la venta de productos de mercado de dinero, fondos de inversión y seguros. Finalmente, la mayoría de los bancos (84%) planean instalar más *cajeros automáticos* ubicados en instalaciones fuera de sucursal y que se pueden acceder a través de ellos una gama más amplia de productos.

# CAPITULO V

# INTERNET COMO MEDIO DE ENTREGA DE SERVICIOS BANCARIOS

## 5.1 Internet y los Bancos Hoy

El impacto del mercado de *Internet* parece crecer día a día. Decenas de millones de personas ya están accesando *Internet*, expertos predicen que cada casa con televisión por cable puede estar conectado a *Internet* al final de la década.

El mercado comercial obviamente ha reconocido el potencial de *Internet*, se ha visto cómo el número de negocios que están estableciendo presencia en la WWW ha estado elevándose en los últimos años. Similarmente, cientos de bancos han creado sus sitios de consulta en la *Web*. Todavía, la mayor parte de estos sitios son un poco más que folletos electrónicos, sólo algunos bancos han dado un paso más allá para permitir a sus clientes ver información de sus cuentas.

Uno de los principales riesgos en esta nueva frontera es no estar preparado para una total aceptación de *Internet* como un nuevo medio de entrega. Los bancos que están tomando el enfoque de esperar-y-observar pueden encontrarse ellos mismos esforzándose para tomar un rol activo.

El proporcionar información estática para promover los productos y servicios de las compañías es un buen inicio, pero es sólo el comienzo. La gran promesa de la WWW se apoya en sus capacidades interactivas.

Se puede pensar en los consumidores en *Internet* como televidentes bien educados : en lugar de escoger el contenido con el control remoto utilizan el mouse. Si los clientes de los bancos visitan un sitio y no ven nada nuevo o útil simplemente seleccionarán otro sitio, seguramente otro banco les proporcionará los servicios que buscan.

Los primeros bancos que se incorporen al escenario de *Internet* recibirán beneficios en varios frentes. Estos bancos atraerán a un nuevo mercado potencial que representa un segmento demográfico atractivo : gente educada y profesional. Estos nuevos clientes cuestan menos, debido a que no impactan significativamente en la infraestructura de los bancos, ya que los costos para ingresar a la *banca electrónica vía Internet* son relativamente bajos.

Los bancos líderes de mercado fortalecen su imagen como bancos con enfoque a clientes al desarrollar proactivamente estrategias de servicio-*en-línea* para incrementar los servicios y conveniencias para los clientes. En contraste los bancos que esperan a ver donde se mueve la industria se encontrarán invirtiendo defensivamente y no obtendrán el mismo beneficio económico.

Una falacia, es visualizar a *Internet* como un servicio más de la *banca-en-el-hogar.* Con esto los bancos pierden la oportunidad de fortalecer la relación con sus clientes al no aprovechar las facilidades interactivas de *Internet. Internet* es una alternativa de medio de entrega que requiere esfuerzos en nuevos diseños y nueva mercadotecnia.

Actuando como un impulsador, *Internet* permite que las instituciones más pequeñas y más flexibles se distingan de los grandes competidores por su creatividad. Además provee un medio para proporcionar al cliente más servicios haciendo que los bancos se conviertan en proveedores de valor agregado. Los bancos ya están obteniendo facilidades regulatorias para ofrecer instrumentos financieros adicionales. Los bancos al ser uno de los primeros jugadores en *Internet*, pueden ayudar a definir la industria financiera en línea y posicionarse como entidades que respondan rápidamente a las fuerzas del mercado conforme los servicios financieros electrónicos evolucionen.

Por otra parte, el desarrollo de la *Super Carretera de Información* ha dado origen a un nuevo fenómeno: la convergencia de las industrias de comunicaciones, entretenimiento, información y computación. Esta convergencia, ha estado siendo impulsada por entidades externas al sector de servicios financieros, pero que tienen el potencial de reconfigurar totalmente el marco competitivo dentro del que operan los bancos.

Esta convergencia ha estado transformando el corazón mismo de los negocios de la banca tradicional y está presentando nuevas oportunidades para los competidores *no-bancarios.* Los ejecutivos de la banca están cada vez más preocupados y conscientes de los riesgos y amenazas que se vislumbran en su futuro, así como también de las oportunidades que representa.

Sin embargo, la mayor parte del territorio actual dentro de la *Super Carretera de Información* está aún inexplorado, pero al mismo tiempo en plena evolución. Así pues, no es raro que muchos bancos no sepan qué hacer todavía al respecto. La banca, que apenas si estaba gozando de un pequeño respiro después de una década de problemas financieros, nuevamente se tiene que enfrentar a un período de replanteamiento y cambios en donde ya no se aplican las tradicionales reglas de juego.

En la actualidad, son dos hechos relevantes y relacionados entre sí los que están provocando la serie de amenazas y oportunidades para los intermediarios financieros: primero, los planes de desarrollo y expansión de las redes de banda amplia totalmente interactivas por parte de las compañías telefónicas y televisivas y segundo, que por medio del uso cada vez mayor de computadoras personales, *módems*, líneas telefónicas y redes por cable, los negocios y el público en general están pudiendo acceder con más facilidad a una creciente gama de servicios interactivos.

La mayoría de los participantes interesados en establecer posiciones a largo plazo dentro del evolucionante mundo interactivo están trabajando en algo de lo siguiente: a) desarrollando su propio servicio *en-línea*; b) participando en alguna plataforma existente de servicio en línea a nivel comercial (por ejemplo *America Online*); o c) desarrollando una presencia en *Internet.*

Para acometer este reto, de la transformación de la banca tradicional, el *Bank Administration Institute*, junto con el *Boston Consulting Group*, llevó a cabo un estudio en dos etapas en relación a la *Super Carretera de Información* y sus implicaciones para los banqueros.

Esta investigación arrojo algunos mensajes de alerta para los banqueros en relación al nuevo ambiente:

- La tecnología actualmente disponible ya está en posibilidad de brindar muchas de las características esperadas de las redes de banda amplia a base de fibra óptica, entre ellas la disponibilidad de video, texto y sonido en forma interactiva bidireccional.
- Las organizaciones que están encabezando las tendencias actuales pertenecen a otros sectores fuera del ámbito bancario y financiero, como tal.
- Las aplicaciones de servicios financieros encajan muy bien dentro de los servicios en línea que hoy se tienen.
- El desarrollo de sistemas de pagos seguros para el ambiente *en-línea* constituye un prerrequisito para el desarrollo del comercio electrónico a gran escala.
- La lucha competitiva se centrará en el mantenimiento del acceso por parte del cliente, así como su control.
- La ventaja competitiva de los bancos radica en su experiencia, abolengo y en la confianza de su clientela.

Debido al fenómeno de la convergencia mencionada, los bancos están enfrentando a los competidores *no-bancarios* (como *Microsoft, America Online* y las compañías telefónicas), quienes están replanteando la forma en que sus productos y servicios puedan llegar a las manos de los clientes de la banca, están creando nuevas herramientas para la navegación electrónica y desarrollando nuevos sistemas de pago más seguros. Así pues, es de suma importancia que los bancos también estén trabajando por su cuenta en lo mismo, si realmente quieren entregar sus productos y servicios *en-línea* a la clientela.

El trabajo que se esta realizando se ha enfocado hacia las nuevas oportunidades en materia de "*navegación*" y "*contenidos*", dos de las principales áreas de oportunidad de negocio que están surgiendo a raíz de la citada convergencia.

La "*navegación*" representa los diferentes medios y herramientas por los cuales los clientes suelen encontrar lo que buscan. En el caso de la banca, ésto ha tenido lugar principalmente a través de sus tradicionales medios de distribución, como las sucursales, los cajeros automáticos y el teléfono, apoyados por medios publicitarios. Pero en este mundo convergente gran parte de la información se puede manejar en forma electrónica, con la ventaja de que puede ser interactiva y adecuarse según el mercado al cual va dirigida. En consecuencia, es de suma importancia que las instituciones bancarias se vuelvan muy proactivas y busquen la forma de ayudar a sus clientes para que encuentren el camino que los conduzca a donde quieren ir dentro del ambiente *en-línea*.

El *"contenido"*, por su parte, representa lo que el cliente quiere, en términos de productos y servicios. Los competidores *no-bancarios*, como las sociedades de inversión, por ejemplo, están tratando de ofrecer el contenido bancario tradicional, creando un reto a la habilidad de los bancos para organizar y explotar sus datos transaccionales y de clientes para construir relaciones perdurables con su clientela base. Si no se ponen listos los bancos en esta área, podrían sufrir por la proliferación de productos y una mayor presión en sus márgenes de intermediación.

El *"contenido"* o tema, reformado para el ambiente en línea, es esencial para las plataformas y programas de navegación. Las compañías que cuentan con mecanismos de navegación propios se han dado cuenta que pueden mejorar notablemente la oferta de sus productos y servicios por medio de una integración estrecha con el contenido. Por ejemplo, *Quicken* ha incorporado el procesamiento transaccional (pago electrónico de servicios) a su *software*, con el objeto de incrementar la utilización del mismo y ofrecer un valor agregado a los clientes.

La importancia que representan los servicios financieros para compañías como *America Online* y *Microsoft* se está haciendo más patente con cada nueva alianza que se forma. Estos competidores *en-línea* están invirtiendo agresivamente tanto en navegación como en contenido y no se están confinando a las definiciones de los productos tradicionales, a las capacidades de los medios de entrega, ni a los aspectos económicos.

Para enfrentar estos retos, los bancos necesitan definir cómo van a participar eficazmente, tanto en el contenido como en la navegación, debido a que las compañías *no-bancarias* de servicios financieros se han puesto a la cabeza en la integración del contenido o tema con los mecanismos de navegación. Por lo cual, si los bancos realmente desean mantener su posición competitiva, tienen que buscar la manera de integrar el contenido con la navegación y ofrecerlos en forma tal, que sea intuitiva y relativamente transparente para el cliente, ya que tanto el contenido como el medio de navegación tienen gran influencia sobre la percepción del cliente respecto a la imagen de la institución.

De esta manera, la mayoría de los productos de la banca al menudeo pueden distribuirse ya sea por medio de *Internet* o por medio de compañías proveedoras de servicios en línea como *American Online* y *Prodigy* , acelerando tanto las oportunidades como los riesgos inherentes creados por la convergencia para el sector financiero.

Por otro parte, hay que señalar que, si se explotan convenientemente dichos medios, pueden representar una poderosa ventaja competitiva para los bancos, debido al conocimiento que tienen de sus relaciones con la clientela, a la amplia gama de medios de entrega de que disponen, así como de la fama y renombre que ya tienen en el mercado.

Por el lado de los riesgos, las nuevas iniciativas y alianzas que constantemente están teniendo lugar, por lo general no están relacionadas con los servicios financieros. El hecho de que gran parte de los cambios que están transformando a los medios de entrega provengan de áreas externas dificulta a los bancos, aún más, poder establecer planes eficaces, vigilar las actividades y reaccionar ante ellas.

Algunos bancos ya están ofreciendo los servicios de la *banca-en-el-hogar* usando sus propios programas navegadores, aunque la mayoría de éstos se hallan una generación atrás en términos de la interfase al usuario y de la velocidad de operación de las transacciones. Una de las principales razones por las cuales los bancos están utilizando los paquetes de *software* para finanzas personales como *Quicken* y *Microsoft Money*, es que están tratando de dar un mejor servicio a su clientela, adhiriendo sus contenidos o temas particulares a estas sofisticadas herramientas de navegación. Un posible riesgo para los bancos radica en que los consumidores confundan estas aplicaciones de *software* con la principal interfase bancaria.

Además, no todas las instituciones bancarias cuentan con el tamaño o la experiencia requeridos para desarrollar sus propios sistemas de navegación, por lo que la mayoría de ellas necesitan las herramientas de terceros a fin de crear su propio ambiente en línea. Esto está generando un mayor auge en el empleo de los paquetes de software para finanzas personales, como *Quicken de Intuit Inc.* y *Microsoft Money* de *Microsoft*. Este último introdujó al mercado las especificaciones *OFC (Open Financial Connectivity)*.

*OFC* proporcionó a los usuarios de *Microsoft Money* conectividad hacia bancos ya sea vía *Internet* o vía líneas privadas. *Microsoft* está desarrollando un conjunto de herramientas para la banca por *Internet* , *ActiveX Controls*, diseñados para permitir a los bancos el desarrollo de sitios de servicios bancarios en *Internet*. Los controles pueden trabajar con las especificaciones de *OFC* y *Money*, los cuales también pueden incorporar el protocolo *HTTP*, de esta manera *Money* puede funcionar tanto con líneas privadas como en *Internet*. Las transacciones son aseguradas usando los protocolos *SSL* o *PCT*.

El hecho, es que ahora los bancos pueden alcanzar potencialmente a millones de usuarios de computadoras personales desde sus propios sitios *Web* vía *OFC* y también pueden usar a *Money* como un *front-end* para estos sitios. Esto podrá permitir que los bancos ofrezcan a sus clientes realizar transacciones financieras (incluyendo pagos) en la sucursal virtual del banco desde cualquier computadora personal con un *modem*, usando ya sea algún *software* para finanzas personales o *software* propietario del banco.

Ahora bien, la información del cliente es un elemento muy importante en poder de los bancos, que pueden utilizar para diferenciarse de la competencia. Por ejemplo, si se pudiera combinar la información acerca del perfil del riesgo de las inversiones de un cliente y los saldos de sus cuentas con la planeación financiera disponible en línea, sería una herramienta muy poderosa, tanto para la institución como para el cliente mismo ; es más, esto podría dar lugar a la venta de un nuevo producto o servicio por parte del banco.

Es muy claro el empleo de la nuevas tecnologías dentro del sector financiero en el caso del *Security First Network Bank (SFNB)*, el primer banco de los E.U. que fue creado desde sus inicios como un banco totalmente "cibernético". Este banco, que cuenta con una sola sucursal física y que inició operaciones recientemente, cuenta con más de 1000 clientes en los Estados Unidos. A diferencia de otras instituciones, el *SFNB* surgió del futuro no del pasado. De hecho, su centro de cómputo alberga únicamente servidores HP operando en ambiente *Unix*. Todos estos equipos (servidores de información, de seguridad y de bases de datos) están conectados entre sí vía una red *10BaseT* ejecutando *TCP/IP*, teniendo planes para migrar al segmento de los 100 Mbps.

Actualmente, existen tres modelos diferentes de la implementación de los servicios de la *banca-en-el-hogar* ; el primero es aquél en el que el banco ofrece una solución particular para este servicio y en la que es responsable de la interfase al usuario, de la red y del contenido de la solución. El segundo modelo consiste en un esquema en el cual un intermediario o un proveedor de servicios (como *Intuit Services*, *Prodigy*, etc.) asume la responsabilidad de la interfase al usuario y de la red, en tanto que la institución financiera sigue siendo responsable del contenido. Finalmente, el tercer esquema de la *banca-en-el-hogar* es precisamente en *Internet* ; en este caso, la interfase proviene del visualizador de la *Web (browser)*, la red es la *Internet* misma y el contenido es manejado por un proveedor de contenidos.

Tecnológicamente, *Internet* está teniendo un impacto muy fuerte en la banca al menudeo, ya que logra minimizar la inversión que tiene que realizar un banco en su red de sucursales tradicionales, misma que es sumamente costosa debido a la cantidad de recursos humanos que emplea. Por otra parte, *Internet* permite a la clientela llevar a cabo sus operaciones a cualquier hora desde su computadora personal, en vez de tener que ajustarse a un horario y a la ubicación física de la sucursal.

Por otra parte, se considera que el uso de la *Web* por parte de instituciones financieras puede dividirse en cuatro etapas : en la primera de ellas se utiliza la *Web* únicamente para presentar información (de tipo promocional principalmente) ; en la actualidad varias instituciones emplean la *Web* como un medio publicitario, si bien en la mayoría de los casos es estático. La siguiente etapa, que apenas está empezando a surgir, consiste en el acceso básico al banco por parte del cliente : "se trata de una función meramente del tipo 'solo lectura' en donde los bancos le permiten a sus clientes ver su información, pero sin que puedan efectuar ninguna operación de actualización". La tercera etapa ya es de *interactividad* ; aquí el cliente ya no nada más puede acceder a la información, sino que puede actualizarla o realizar cálculos y simulaciones del tipo "que para si...". Finalmente la cuarta etapa será aquella en la que la institución financiera complementará el uso de la *Web* con otros tipos de medios, como el fax y el correo electrónico.

## 5.2 Tendencias de los Servicios Bancarios en Internet

Un estudio de Booz-Allen & Hamilton sobre la *banca-en-Internet*, muestra por primera vez una visión de la demanda de consumo en el campo de la *banca-en-Internet*, mismo que está creciendo rápidamente. De acuerdo a este estudio, BA&H proyecta que por encima de 16 millones de hogares en E.U. pueden usar los servicios de la *banca-en-Internet* a finales del año 2000. El estudio revela que estos hogares pueden ser algunos de los clientes más rentables del sistema bancario, representando cerca del 30% de las ganancias obtenidas por la banca al menudeo. El estudio concluye que los bancos, con muchos de sus más rentables clientes "*en-línea*", pueden enfrentar a un ambiente en *Internet* mucho más competitivo y riesgoso.

El estudio proyecta que 1,500 bancos pueden tener sitios en *Internet* para 1999 y al menos 500 de estos bancos pueden estar ofreciendo *banca-en-Internet* mayormente desarrollada.

La investigación usa estos estudios así como un análisis de factores principales que están afectando la demanda de los consumidores por los servicios de *banca-en-Internet*, tales como, propiedad de computadoras, uso de *Internet*, y la aceptación de los consumidores de los servicios de la *banca-en-Internet*, para proyectar el número total de hogares estadoudinenses usando servicios de *banca-en-Internet* durante 1997-2000.

Un análisis más amplio revela que mientras los clientes de la *banca-en-Internet* pueden representar solo el 16% de los hogares en E.U. para el año 2000, ellos pueden representar casi el 30% de las ganancias de la banca al menudeo.

## 5.3 Arquitectura para ofrecer Servicios Bancarios en Internet

Ultimamente ha surgido una nueva tendencia en el desarrollo de aplicaciones Cliente/Servidor que ha estado propiciando la evolución del desarrollo de un modelo extremadamente centralizado en el cliente (esto es, el modelo de cliente "pesado") hacia un modelo centralizado en servidores. Esta tendencia ha culminado en aplicaciones en *Internet* e *Intranets* (*I-Nets*). Las aplicaciones *I-Nets* corren en redes públicas y privadas muy flexibles, y son implantadas en servidores *RISC* baratos que pueden ofrecer tanto poder y escalabilidad como los *mainframes*. Las aplicaciones *I-Nets* también pueden ser particionadas de tal manera que la lógica de la aplicación pueda ser ejecutada tanto en el servidor como en el cliente – algo que nunca puede ser hecho usando terminales tontas. Las *I-Nets* son casi una recentralización de la implantación, administración y el mantenimiento. Con la recentralización, puede ser ahora posible implantar a un costo eficaz y confiable aplicaciones de gran escala – sistemas que pueden servir a miles de usuarios sin los problemas que han plagado al cómputo centralizado en el cliente.

La implantación del modelo centralizado en servidores no sucedió de la noche a la mañana. La primera generación de cómputo en computadoras personales comenzó con un cliente muy robusto. Las aplicaciones fueron esencialmente implantadas en computadoras personales. A mediados de los 80's se popularizó la idea de los servidores de bases de datos *SQL*, lo cual movió el manejo de la base de datos de la computadora personal hacia el servidor *back-end*.

Los procedimientos almacenados (*stored-procedures*), los cuales son una forma de particionamiento de la aplicación, fueron el primer paso para mover las aplicaciones hacia los servidores, pero la introducción del modelo particionado de 3-niveles ha sido el impulsor del modelo centralizado en servidores. El particionamiento en 3-niveles transfiere porciones sustanciales de la lógica de la aplicación del cliente hacia los servidores de aplicaciones centralizadas. Esta tendencia hacia la implantación de aplicaciones centralizadas ha sido impulsada por el deseo de incrementar el rendimiento, control y seguridad dentro de un ambiente Cliente/Servidor.

Las aplicaciones *I-Nets* son esencialmente aplicaciones que están completamente implementadas en servidores y accesadas desde navegadores vía los protocolos *HTTP* y *HTML*. Aplicaciones de gran escala que usan esta arquitectura están proliferando en *I-Nets* .

La implantación centralizada significa que en el momento que la aplicación está disponible en el servidor *Web*, está inmediatamente disponible para todos los usuarios autorizados en una *I-Net*. Miles o quizás millones de usuarios pueden accesar este tipo de aplicaciones sin tener que comprar, instalar o inspeccionar nuevo *software*, o renovar las estaciones de trabajo para que se puedan ejecutar nuevas aplicaciones.

No hay necesidad de visitar todas y cada una de las estaciones de trabajo o computadoras personales debido a que no se necesita instalar ni una sola línea de código en el cliente. Este tipo de instantánea y obicua disponibilidad es algo que es simplemente inalcanzable en un ambiente centralizado en el cliente.

Los usuarios no están mayormente a un pequeño conjunto de aplicaciones que puedan ser razonablemente instaladas y mantenidas en cada computadora personal dentro de una organización. Literalmente docenas de aplicaciones en *Intranet* o cientos de aplicaciones en *Internet* están disponibles para cualquier usuario, con poco esfuerzo y a costos mínimos.

Por otra parte, el desarrollo de aplicaciones es mucho más fácil debido a que los desarrolladores no tienen que estar preocupados con diseños complejos, pruebas y consideraciones de implantación asociadas con el particionamiento de la aplicación. El mantenimiento y administración son ampliamente simplificados debido a que todo el código de la aplicación es centralizado. La centralización también significa el fin del extraordinario ritmo de obsolescencia del *hardware* y *software* que la industria del cómputo ha estado experimentando en la última década. Tal rapidez de obsolescencia crea un efecto de domino, el cual es extremadamente caro y desestabilizante debido a que afecta todos los aspectos de un ambiente organizacional. El *hardware* y *software* tienen que ser reemplazado en todas y cada una de las computadoras personales.

La implantación centralizada de aplicaciones no significa necesariamente que todas las aplicaciones tengan que ser ejecutadas en el servidor. Usando tecnologías como *Java* o *Java Script*, es posible bajar "*applets*" del servidor para ser ejecutados en el cliente. Esta variante mantiene la ventaja de la implantación centralizada, pero abre la posibilidad de la ejecución en el cliente.

### 5.3.1 Importancia de los Componentes Reutilizables

Aunque no se trata de un concepto nuevo, la reusabilidad, en términos de mejorar la calidad del *software* y la productividad, ha sido agresivamente perseguida apenas en los últimos años. En torno a este tema se han definido algunos conceptos como *productos de trabajo* (*work products*), que son los productos dentro del proceso de desarrollo de *software* (código, diseño, planes de pruebas, etc) ; *reuso* es el uso de los productos de trabajo sin modificación en el desarrollo del *software* ; *reuso apalancado* (*leverage reuse*), consistente en modificar productos de trabajo para cubrir requerimientos de un sistema específico ; *un productor* es un generador de productos de trabajo reutilizables y un *consumidor* es alguien que los usa para crear nuevos productos de *software*.

Debido a que, en un esquema de reusabilidad, los productos de trabajo han sido creados, probados y documentados previamente, se incrementa la productividad, ya que los consumidores de dichos productos tienen menos trabajo que hacer. Sin embargo, el incrementar la productividad gracias a la reusabilidad, no necesariamente reduce el tiempo total para liberar un producto nuevo. Para reducir este tiempo, el reuso debe ser empleado de manera efectiva en la ruta crítica de un proyecto de desarrollo de *software*. Se ha encontrado que el reuso de productos le permite a las organizaciones aprovechar de manera más eficiente sus recursos humanos porque se basa en su experiencia para generar sinergía : los especialistas en *software* con más experiencia pueden dedicarse a generar productos de trabajo para ser usados por personal con menor experiencia. Sin embargo, la reusabilidad tiene su costo, ya que se tiene que invertir en personal especializado para crear las piezas reutilizables, mantenerlas, administrarlas, etc.

Debido a que los productos de trabajo son usados múltiples ocasiones, la reparación de defectos de cada reuso se acumulan, resultando en una mayor calidad. Lo que es más importante, la reusabilidad incentiva el prevenir o detectar problemas en etapas tempranas del ciclo de desarrollo de un sistema, debido a que los costos de prevención y pruebas pueden ser amortizados entre un gran número de usuarios.

La reusabilidad incrementa la productividad debido a que el ciclo de vida de sistemas requiere menos insumos para generar las mismas salidas. Por ejemplo, el reuso puede reducir costos de elaboración al promover la especialización en áreas como interfases de usuarios. Debido a su experiencia, los especialistas realizan las tareas de manera más eficiente que los no especialistas. Desde otro punto de vista, la productividad se incrementa simplemente porque no se tienen que desarrollar todos los productos de trabajo desde cero. En general la reusabilidad incrementa la productividad al reducir la cantidad de tiempo y el trabajo requerido para desarrollar y mantener un producto de *software*.

## 5.3.2 OLTP (On-line Transaction Processing)

El Procesamiento de Transacciones en Línea (*OLTP*, por sus siglas en inglés) representa el componente vital de muchas organizaciones, el procesamiento de datos en tiempo real en el cual usualmente existe una correlación directa entre el rendimiento de *OLTP* y los ingresos de la corporación --- mientras más asientos y reservaciones pueda procesar una aerolínea, más altos serán sus ingresos.

*OLTP* nos rodea cotidianamente --- en sistemas de reservaciones de aerolíneas, sistemas bancarios, de procesamiento de órdenes y de administración de inventarios. *OLTP* no es otra cosa que la automatización de procesos del negocio altamente repetitivos y de elevado volumen. Independientemente de cómo estén implementados estos sistemas en la actualidad, tienen que tener las siguientes características :

- Disponibilidad ininterrumpida.
- Los tiempos de respuesta deben caer dentro de un cierto rango predecible.
- Deben ofrecer integridad de las transacciones.
- Las transacciones deben poder realizarse con el debido nivel de seguridad requerido.

Típicamente, las aplicaciones más comunes de *OLTP* consisten de sistemas de entradas de órdenes que permiten que cientos de clientes accesen simultáneamente una base de datos proporcionando a los usuarios tiempos de respuesta muy bajos. *OLTP* está formado por una serie de aplicaciones que generalmente se cimentan en bases de datos relacionales y no-relacionales, aplicaciones *front-end* usadas para capturar datos en tiempo real, y herramientas para monitorear el desempeño de las transacciones y de los sistemas que las sustenten. Las aplicaciones *OLTP* generalmente hacen un uso intensivo de las bases de datos, en donde la velocidad y la precisión son esenciales.

*OLTP* raramente implica un sofisticado manejo de datos o análisis, pero requiere amplia velocidad y total precisión. Así, las aplicaciones *OLTP* deben ser capaces de procesar cientos de transacciones por segundo. Debido a estos demandantes requerimientos de rendimiento, combinados con la sensibilidad e importancia de los datos y procesos corporativos, las aplicaciones *OLTP* han tradicionalmente residido en los *mainframes*, ya que sólo los sistemas operativos de los *mainframes* han cubierto los requerimientos de administración del sistema, herramientas de monitoreo y rendimiento necesarios para realizar operaciones *OLTP* exitosamente.

A pesar de la rentabilidad de los sistemas abiertos y los beneficios del cómputo distribuido, muchas compañías creyeron que el *mainframe* era la única plataforma con el suficiente poder, confiabilidad y mecanismos de recuperación para proteger los activos de información corporativa adecuadamente. Aunque, tanto gerentes de negocios como personal de informática hayan visto las ventajas de la flexibilidad de los sistemas abiertos y plataformas Cliente/Servidor para operaciones *OLTP*, las herramientas e infraestructura necesarias para soportar transacciones *OLTP* no son lo suficientemente robustas para satisfacer sus demandantes requerimientos.

Sin embargo, aunque los usuarios se dan cuenta de que las redes locales son plataformas efectivas para muchos tipos de aplicaciones de procesamiento de información, las redes locales no han sido todavía probadas por si mismas en el procesamiento de transacciones en áreas de negocio de misión crítica. En aplicaciones de redes locales, un nivel de 90 % de éxito es suficiente, pero en ambientes *OLTP* se requiere de un 99.9 %. En una aplicación *OLTP* la velocidad, la exactitud y la confiabilidad son esenciales. Por estas razones, el número de compañías que han reemplazado sus aplicaciones *OLTP* en ambientes *mainframe* por aplicaciones *OLTP* distribuidas es aún pequeño ; aunque lentamente, la evolución se está dando. En lugar de implementar un cambio mayor como ocurría con otras aplicaciones, muchas compañías han decidido seleccionar aplicaciones no-críticas para construir prototipos de aplicaciones *OLTP*. De esta manera, si las pruebas son exitosas, las organizaciones gradualmente migrarán hacia el uso de *OLTP* distribuido.

La administración del sistema (*System Management*), la habilidad para asegurar que las computadoras y las redes de comunicaciones funcionen correctamente, es otra área donde las redes locales se encuentran en desventaja respecto a los *mainframes*. La administración de sistema es una frase que engloba el rendimiento de las computadoras, redes, periféricos y *software*. La administración de sistemas también incluye procedimientos de seguridad.

El monitoreo del desempeño es otra área débil. En un *mainframe*, los usuarios pueden recolectar montañas de estadísticas para identificar los problemas. Pero en el ambiente *Unix* sólo una cantidad limitada de información se encuentra disponible.

Actualmente, los *mainframe* aparecen adelante de las redes locales en todas estas áreas. Pero los vendedores han gastado más de 20 años construyendo herramientas para asegurar la confiabilidad de los *mainframe*, por eso no debe ser tan sorprendente que las redes locales se encuentren detrás en muchas áreas de la administración de sistemas. Sin embargo, la brecha puede no ser tan grande. Las herramientas que se requieren para correr aplicaciones *OLTP* en redes locales están madurando, y los usuarios se están sintiendo lentamente más comfortables al construir nuevas aplicaciones en redes locales.

Sin embargo, los departamentos corporativos de información gerencia: se esfuerzan para reducir costos y volverse más competitivos usando más inteligentemente la información corporativa, y viéndose un gradual movimiento hacia esquemas de *OLTP* distribuido. Este movimiento refleja algunas de las actuales tendencias del cómputo tales como : aplicaciones Cliente/Servidor, reducción de plataformas (*downsizing*), incremento en la modularidad y arquitecturas de interfases para sistemas abiertos. La vista de los usuarios se está enfocando gradualmente en los dos principales beneficios de los sistemas abiertos : Interoperabilidad y Portabilidad, vía la arquitectura Cliente/Servidor. Como resultado, tanto usuarios como vendedores, del mismo modo, han comenzado a darse cuenta que la administración de transacciones se puede convertir en una tendencia unificante que puede integrar los diversos componentes del procesamiento transaccional distribuido (tales como clientes remotos, servidores poderosos, etc.) dentro de una imagen sencilla de un sistema.

Aunque ya no son el centro de atracción hoy en día, muchos sistemas *OLTP* en *mainframe* siguen funcionando en la actualidad después de dos o más décadas en producción. Estos sistemas presentan fortalezas muy relevantes y también algunas debilidades, aunque éstas no impiden que operen satisfactoriamente y en forma bastante confiable. Por ejemplo, la disponibilidad de estos sistemas puede ser mayor del 99.9% debido a la madurez del *software* y de las herramientas para su manejo, no obstante, la administración de su crecimiento puede ser un serio problema. Las mejoras funcionales en los sistemas por lo general son lentas de implementarse, la interfase al usuario es bastante limitada y el costo por lo general se considera elevado y relativamente insensibles a las mejoras en precio/rendimiento que presentan las nuevas tecnologías.

Una de las cosas que más han impulsado a la plataforma Cliente/Servidor ha sido la enorme cantidad de paquetes de *software* disponibles en el mercado y que pueden implementarse y distribuirse en forma más rápida y económica que en el caso del *mainframe* ; además, la adopción de estándares de sistemas abiertos significa que están a la mano del usuario una gran variedad de equipos y sistemas administradores de bases de datos relacionales. Por otra parte, la administración del crecimiento del ambiente Cliente/Servidor es más fácil, al igual que las mejoras funcionales, en tanto que la interfase al usuario es excelente. Sin embargo, también tiene sus desventajas : la administración de configuraciones y el control de versiones ; la distribución del *software* ; la administración de los sistemas y equipos ; la administración del rendimiento ; la escalabilidad de los servidores ; el manejo de respaldos y recuperaciones ; así como la seguridad de datos y aplicaciones, esto ha dado lugar a que las aplicaciones *OLTP* que son críticas para las empresas permanezcan en los *mainframes*.

### *OLTP en Internet*

La WWW y los sistemas *OLTP* propietarios de los empresas tienen una relación de amor/odio. La libertad y poderío de la *Web* son irresistibles, pero las aplicaciones *OLTP* corporativas son inamovibles ; ellas están aisladas para mantener la integridad y resguardar las ventajas competitivas. Las empresas resuelven este conflicto dibujando una pared de seguridad entre sus sistemas corporativos y la *Internet*. Como se puede ver en la fig. 5.1, en el lado de *Internet* del *firewall*, el acceso es sin restricciones. En el lado de la empresa del *firewall*, los recursos son altamente controlados. El acceso a través del *firewall* esta limitado y basado en una forma de envío de mensajes o datos. Los sistemas de procesamiento de transacciones detrás del *firewall* están enviando y recibiendo mensajes. Esto describe una relación clásica Cliente/Servidor. En este caso, la *Web* es un cliente multimedia, haciendo su propio procesamiento y enviando requerimientos a los sistemas detrás del *firewall*. El ambiente *OLTP* corporativo es el servidor que proporciona el encapsulamiento de las funciones o datos del negocio a los clientes en el otro lado del *firewall*.

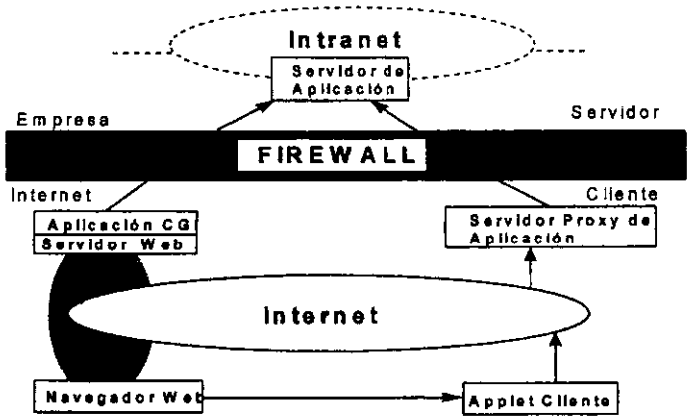



***Fig. 5.1 Acceso a Aplicaciones Corporativas por Internet***

A primera vista, la *Web* posee atributos bastante poderosos para utilizarse como un medio de acceso a las aplicaciones OLTP ; empero, también muestra varias debilidades. La *Web* presenta ciertas ventajas importantes, especialmente su interfase universal al usuario por medio de los navegadores. También, puede utilizarse como un servidor centralizado albergando aplicaciones y datos, en tanto que las estaciones de trabajo remotas únicamente se encargan de la capa de presentación. En este esquema desaparecen los problemas en torno a la distribución del *software*, el control de las versiones y la administración de sistemas y equipos, en tanto que el crecimiento se puede manejar más fácilmente (suponiendo que los servidores *HTTP* sean capaces de escalarse hasta un nivel conveniente para el *OLTP).*

Aunque la *Web* pudiera parecer una buena alternativa en relación al ambiente Cliente/Servidor convencional, todavía tiene mucho camino que recorrer antes de que pueda convertirse en el medio ideal para el procesamiento transaccional a gran escala. Por una parte, no se tiene disponibilidad ininterrumpida ; *Internet* permite el uso de rutas alternas para el tráfico de datos, pero aún no se cuenta con un esquema robusto de recuperación de datos en el momento de falla de un servidor *HTTP* ; los tiempos de respuesta más o menos homogéneos están lejos de ser una realidad ; la integridad de las transacciones requiere aún mejorarse ; y el aspecto de seguridad sigue siendo un problema importante.

La *Web*, dado su estado actual de madurez, puede apoyar en dos áreas : permitir la introducción de nuevos sistemas basados en la facilidad de uso y en un bajo costo de propiedad, y fungir como una extensión a nivel táctico de los sistemas *OLTP* existentes.

Para la implantación *OLTP* en *Internet* Gartner Group ha definido diferentes estilos de aplicaciones de procesamiento de transacciones (ver fig. 5.2) :

- Aplicaciones Privadas : No se tiene acceso a ellas desde *Internet*.
- Aplicaciones de Presentación Distribuida : La capa de presentación es distribuida a través de *Internet*.
- Aplicaciones de Presentación Remota : Los clientes públicos (usualmente basados en la *Web*) accesan los servidores de aplicaciones privadas a través del *firewall*.
- Aplicaciones de Función Distribuida : La lógica de negocio de las aplicaciones esta distribuida a través de *Internet*.
- Aplicaciones de Administración Remota de Datos : Los datos corporativos son accesos a través de *Internet* y del *firewall* ; la interfase del usuario y la lógica de negocio de las aplicaciones son realizadas en el dominio público.
- Aplicaciones de Bases de Datos Distribuida : *Internet* es usada por los *DBMS* de las empresas para integrar segmentos distribuidos de los datos de la empresa.
- Aplicaciones Públicas : Las aplicaciones, sus interfases de usuario y sus datos son accesados directamente desde la *Internet*.

El 80% de estas aplicaciones pueden usar presentación remota, donde la función del cliente es ejecutada en el dominio público (usualmente vía la *Web*) y todos los servidores de aplicaciones funcionalmente están encapsuladas detrás de un *firewall* en la *Intranet* controlada de la empresa.

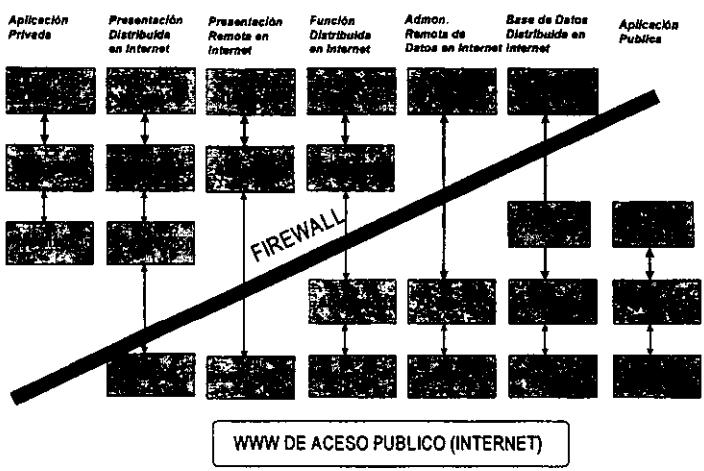



***Fig. 5.2 Estilos de Aplicaciones de Procesamiento de Transacciones relacionadas a Internet***

## *5.3.3 Monitores de Transacciones*

La necesidad de reducción de plataformas (*downsizing*), combinada con los avances tecnológicos del proceso distribuido, ha redundado en el surgimiento de varias aproximaciones a los sistemas de administración de transacciones. Algunas de las opciones más recientes basadas en *Unix* --- Tuxedo de BEA, Encina de Transarc, Top-End de NCR --- están retando agresivamente al campeón de los monitores de transacciones —CICS de IBM. A fin de conservar su posición de liderazgo en esta materia, IBM está reaccionando ofreciendo nuevas propuestas basadas en CICS.

El Monitor de Procesamiento de Transacciones (*TPMonitors*) es un *software* ubicado sobre el manejador de bases de datos para asegurar la precisión de las transacciones. Es utilizado con grandes y complejas aplicaciones y agrega características como recuperación de datos y seguridad, para asegurar la confiabilidad de la infraestructura de cómputo. El Monitor de Transacciones permite deshacer (*rollback*) las transacciones cuando un módulo del sistema falla y puede guardar transacciones en una cola mientras un componente del sistema es actualizado o cambiado.

Los *TPMonitors* realizan las siguientes funciones :

- Proporcionan herramientas de *software* (herramientas de acceso a datos, dibujadores de pantallas, comunicaciones intra-sistemas) que son utilizadas para desarrollar aplicaciones para el cliente.
- Proporcionan un ambiente de ejecución que asegura la integridad, disponibilidad y seguridad de datos, tiempo de respuesta rápido y un alto desempeño de las transacciones al administrar la sincronización y distribución de la carga de transacciones entre los recursos disponibles.
- Proporcionan soporte administrativo que les permite a los usuarios realizar instalaciones personalizadas, así como configuración, monitoreo, y administración de sus sistemas de cómputo.

Conforme las aplicaciones Cliente/Servidor han ido creciendo y soportando más usuarios, ha surgido un mayor auge de los *TPMonitors*. De acuerdo al Grupo Standish, los *TPMonitors* se han convertido en uno de los temas más candentes de la tecnología desde 1996, considerando que un 55% de las aplicaciones de misión crítica han de ser construidas con *TPMonitors*. Los *TPMonitors* juegan un papel trascendental en los sistemas de 3-niveles : como segmento del mercado Cliente/Servidor de crecimiento más rápido, proveen la infraestructura para ejecutar procesos en la capa intermedia.

De esta manera, en un ambiente Cliente/Servidor, un *TPMonitor* se intercala así mismo entre los clientes remotos y los recursos del servidor (ver fig.5.3).

Los *TPMonitors* son un ejemplo de un arquitectura Cliente/Servidor de 3-Niveles, encajan en este modelo debido a que ellos manejan los procesos aplicativos independientemente de la base de datos o del *front-end*. Los *TPMonitors* proporcionan un nivel extra que separa los clientes del manejador de recursos.

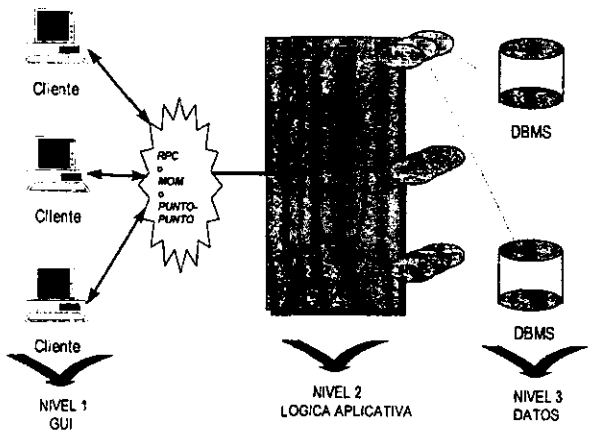



***Fig. 5.3 Arquitectura Cliente/Servidor de 3 Niveles usando un TPMonitor***

Los *TPMonitors* controlan todo el tráfico que liga cientos (o miles) de clientes con programas de aplicación y los recursos en el *back-end*. Los *TPMonitors* aseguran que las transacciones son completadas con precisión, proporcionan balanceo de cargas, y mejoran todo el sistema. Más importante, los *TPMonitors* hacen los procesos independientes de cualquier administrador de recursos. Esto permite trabajar con cualquier recurso en el *back-end*.

Situado entre los clientes y los servidores, el *TPMonitor* puede administrar transacciones, rutearlas a lo largo del sistema, balancear las cargas en la ejecución, reiniciarlas después de fallas. Los *TPMonitors* administran transacciones desde su punto de origen – típicamente en el cliente – a lo largo de uno o más servidores y de regreso al cliente origen. Cuando una transacción termina, el *TPMonitor* asegura que todos los sistemas involucrados en la transacción se quedan en un estado consistente.

Un *TPMonitor* vigila todos los aspectos de una transacción distribuida, independientemente de los sistemas o administradores de recursos utilizados. Muchos *TPMonitors* pueden administrar recursos en un servidor único o en múltiples servidores y pueden cooperar con otros *TPMonitors* en sistemas heterogéneos. Los futuros *TPMonitors* podrán residir en cualquier maquina cliente para hacer que los recursos del cliente –tales como interfase de usuario, almacenes de datos locales o agentes personales– se encuentren al alcance de una transacción distribuida.

Los *TPMonitors* primero aparecieron en los *mainframes* para proporcionar ambientes de ejecución para soportar grandes aplicaciones *OLTP*, tales como sistemas de reservaciones de aerolíneas, de hoteles, sistemas bancarios. Los *TPMonitors* proporcionan lo que sea necesario para conservar las aplicaciones *OLTP* ejecutándose en el estilo de crecimiento usual : altamente reactivo, disponible y bien administrado.

Al moverse *OLTP* a plataformas Cliente/Servidor una nueva variedad de *TPMonitors* está emergiendo para ayudar a convertir el nuevo ambiente en uno hospitalario para aplicaciones de misión crítica.

Se puede pensar en un *TPMonitor* como un sistema operativo de transacciones. los *TPMonitors* hacen 2 cosas extremadamente bien ; administración de procesos : iniciar procesos, canalizarles trabajo, monitorear su ejecución, balancear sus cargas y administración de transacciones ; garantizar las propiedades *ACID* (ver tabla 5.1) de todos los programas que corren bajo su protección.

La tabla 5.1 muestra las propiedades *"ACID"* que fueron diseñadas para asegurar la integridad y confiabilidad de los datos.

| Propiedad | Descripción |
|---|---|
| Atomicity<br>Atomicidad | La secuencia completa de acciones (unidades lógicas de trabajo) deben ser ya sea completadas o abortadas. La transacción no puede ser parcialmente exitosa. |
| Consistency<br>Consistencia | Una transacción toma un sistema de cómputo y sus recursos de un estado consistente a otro. |
| Isolation<br>Aislamiento | El efecto de una transacción no es visible por otras transacciones hasta que la transacción sea ejecutada completamente (committed). |
| Durability<br>Durabilidad | Los cambios hechos por la transacción ejecutada (committed) son permanentes y deben tolerar fallas del sistema. |
| Serialización | Mientras una transacción en proceso dependa de cierta información, esta información es bloqueada para prevenir que cualquier otra transacción la cambie. |

**Tabla 5.1. Propiedades ACID de *OLTP***

Los *TPMonitors* fueron diseñados para ejecutar clases de aplicaciones que puedan servir a cientos y en ocasiones a miles de clientes. La razón es simple : si se le dan a uno de estos cientos de clientes todos los recursos que ellos necesitan en un servidor, aún el *mainframe* servidor más grande puede ser insuficiente. Afortunadamente, no todos los clientes requieren servicios concurrentes. Sin embargo cuando lo hacen, necesitan el servicio en forma inmediata. Los usuarios finales tiene una "tolerancia de espera" de 2 segundos o menos.

Los *TPMonitors* proveen un sistema operativo—encima de los sistemas operativos existentes—que puede conectar, en tiempo real, a miles de clientes con un *pool* de procesos servidores compartidos. Los *TPMonitors* balancean el uso de los recursos entre los clientes en base a la demanda.

En esencia, los *TPMonitor*s eliminan los requisitos de procesos-por-cliente por peticiones encoladas para procesos servidores compartidos. Si el número de peticiones de los clientes excede el número de procesos en una clase de servidor, el *TPMonitor* puede dinámicamente iniciar nuevos procesos—una habilidad llamada balanceo de cargas. Los más sofisticados *TPMonitor*s pueden distribuir la carga de los procesos a través de múltiples *CPUs* en ambientes de memoria-compartida, multiprocesamiento simétrico (*SMP*), o procesamiento paralelo masivo (*MPP*). Parte del balanceo de cargas involucra el manejo de las prioridades de las peticiones recibidas.

Un *TPMonitor* garantiza las propiedades *ACID* mientras mantiene un alto rendimiento transaccional. Para realizar estas actividades, debe de manejar la ejecución, distribución y sincronización de las cargas de trabajo de las transacciones.

Varios de los beneficios obtenidos al usar *TPMonitor*s son :

- Soporte al desarrollo de aplicaciones Cliente/Servidor. Las herramientas de desarrollo están soportando directamente *RPCs* y haciendo a los *TPMonitor*s transparentes para los desarrolladores.
- *Firewalls*. Los *TPMonitor*s implementan *firewalls* entre las aplicaciones y los administradores de aplicaciones y entre las aplicaciones mismas.
- Alta Disponibilidad. Los *TPMonitor*s han sido diseñados para trabajar alrededor de todos los tipos de fallas. Los *TPMonitor*s siempre conocen el estatus de los recursos cliente/servidor que están bajo su control. Forzando las propiedades *ACID*, los *TPMonitor*s permiten detectar una falla exactamente cuando esta sucede.
- Balanceo de Cargas. Los *TPMonitor*s se especializan en administración de procesos y soportan técnicas de balanceo de cargas dinámicas y estáticas.
- Escalabilidad de Funciones. Los *TPMonitor*s alientan a crear procedimientos modulares y reusables que encapsulan los administradores de recursos.
- Sistemas de bajo costo. Los *TPMonitor*s pueden ahorrar dinero, según Standish Group se pueden lograr ahorros mayores a 30% en los costos de los sistemas sobre un modelo centralizado de base de datos. El balanceo de cargas de los *TPMonitor*s proporciona mejor rendimiento usando los mismos recursos del sistema.

### *OLTP en un ambiente distribuido*

El procesamiento distribuido de transacciones es la evolución del tradicional procesamiento centralizado de transaccciones, hacia el mundo del cómputo distribuido y abierto. En un ambiente de procesamiento distribuido de transacciones, cuando las funciones de negocio son distribuidas entre diferentes sistemas, la función del monitor de procesamiento de transacciones debe ser expandida para administrar las transacciones que afecten a múltiples sistemas. Aunque cada nodo distribuido ejecuta su parte de trabajo en su porción de la transacción "local", aún atómica, el procesamiento de transacciones distribuidas requiere de nuevos conceptos de atomicidad, consistencia, aislamiento y durabilidad distribuidos. Estas nuevas propiedades de las transacciones distribuidas son proporcionadas por nuevos manejadores de procesamiento distribuido de transacciones. Cada transacción debe ser diseñada de tal manera que este procesamiento dependa directamente del resultado del procesamiento realizado por otra transacción.

Conceptualmente, un ambiente corporativo de procesamiento distribuido de transacciones debe mantener un *TPM* (*Transaction Processing Manager*) en cada nodo distribuido. Al mismo tiempo, el sistema corporativo de procesamiento distribuido de transacciones debe proporcionar un coordinador global de *TPMs* , lógicamente centralizado, que pueda comunicarse con cada uno de los *TPMs* y controlarlos. La figura 5.4 muestra un ejemplo de ambiente *OLTP* Cliente/Servidor de tres niveles. En esta configuración, las estaciones de trabajo cliente pueden realizar transacciones locales bajo el control de un *TPM* local. Concurrentemente, las estaciones pueden participar en una transacción grupal coordinada por un servidor *TPM*. Por el contrario, el servidor puede participar en una transacción distribuida que abarca múltiple servidores y/o en una transacción global iniciada y coordinada por un *TPM host.*

La clave para este tipo de integración es la existencia de una interfase de programación para aplicaciones transaccionales que sea común, para ser entendida por todos los manejadores de transacciones y manejadores de recursos, particularmente de bases de datos.

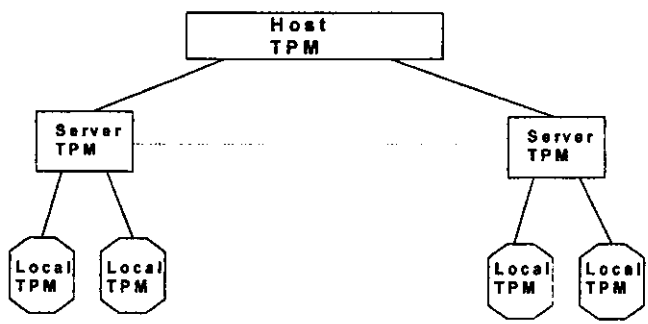



***Fig. 5.4 Ambiente OLTP Cliente/Servidor de 3 niveles***

Hay varias piezas que deben ser tomadas en cuenta por los desarrolladores que busquen implantar aplicaciones Cliente/Servidor. El *software* intermedio (*middleware*) como el de tipo *OLTP*, resulta ser el componente medular ; sin embargo, se requieren productos y servicios complementarios de vendedores de herramientas, compañías de bases de datos, desarrolladores de aplicaciones e integradores de sistemas, si se desea implantar ampliamente soluciones Cliente/Servidor. El siguiente reto para *OLTP* abierto es el proporcionar interoperabilidad con aplicaciones propietarias, tales como CICS, y otras plataformas de sistemas abiertos.

## *5.3.4 Elementos de la Arquitectura Propuesta*

### *Esquema Conceptual de la Arquitectura*

El esquema conceptual de la arquitectura para la entrega de servicios bancarios por *Internet* puede dividirse en 2 entornos (ver fig. 5.5) : el exterior y el bancario.

En el entorno exterior se encuentran los clientes del banco, los cuales utilizan a la red publica de *Internet* como su medio de acceso al banco y a un programa navegador como su interfase a los servicios bancarios, en este entorno también se pueden ubicar otros dispositivos propios de entrega de servicios que se encuentran fuera de la red privada del banco, como cajeros automáticos, kioscos de autoservicio ; los cuales se pueden conectar por otros medios de enlaces (líneas privada, microondas, satelital, etc.).

El entorno bancario esta compuesto por un *firewall* externo, un *firewall* interno, los servidores intermedios y los servidores corporativos.

El *firewall* externo es el punto de contacto del entorno bancario con el entorno exterior, esto permite proteger y controlar el acceso al banco de clientes no-autorizados y de la conexión de dispositivos de entrega no pertenecientes al banco.

El *firewall* interno permite filtrar que usuarios/empleados internos puedan tener acceso a los servicios bancarios ya sea por la *Intranet* o por la *Wan*.

Los servidores intermedios son un conjunto de servidores donde se encuentran las aplicaciones y/o servicios que manejan el acceso externo, también se encuentran las aplicaciones y/o servicios que permiten la interoperabilidad con los servidores corporativos para realizar las transacciones bancarias.

Los servidores corporativos son los servidores en los que se encuentran las aplicaciones y datos corporativos a los cuales no se puede tener un acceso directo, solo se puede tener acceso a ellos a través de los servidores intermedios.

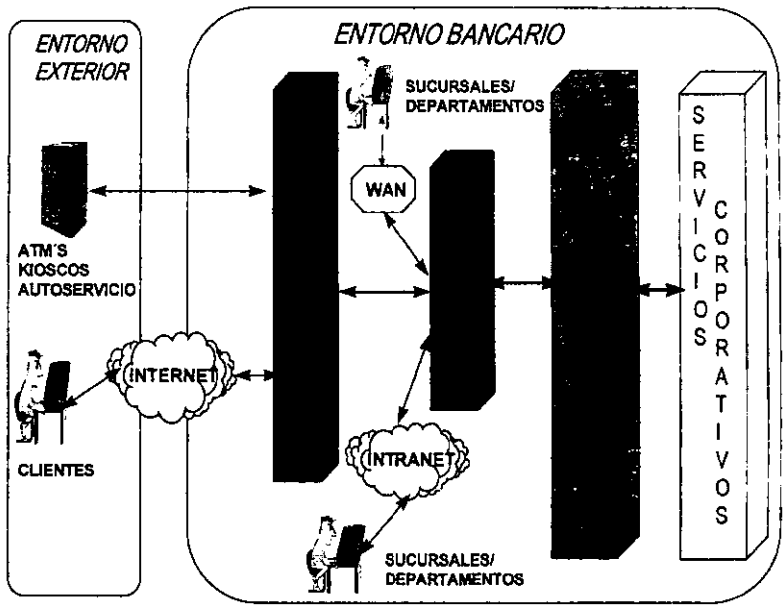



***Fig. 5.5 Esquema Conceptual de la Arquitectura***

El modelo arquitectónico general de la arquitectura para la entrega de servicios bancarios por *Internet* se puede conformar de 5 niveles o capas (ver fig. 5.6) :

1. Cliente
2. Firewall Externo
3. Firewall Interno
4. Servidores Intermedios
5. Servidores Corporativos

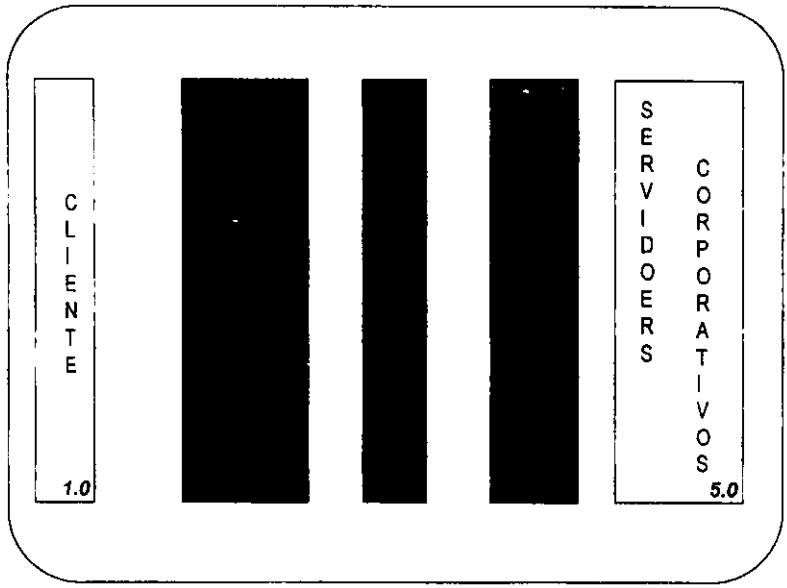



*Fig. 5.6 Modelo Arquitectónico General*

### *Cliente (Nivel 1)*

Este nivel esta conformado por los diferentes dispositivos de entrega de servicios y principalmente por las computadoras personales de los clientes. Los dispositivos utilizan regularmente protocolos de comunicaciones propietarios para comunicarse con los servidores intermedios, en cambio las computadoras personales de los clientes utilizan un programa navegador como su interfase a los servidores intermedios, este es cualquier navegador comercial capaz de desplegar páginas dinámicas a través de la ejecución de aplicaciones interactivas, utilizando "*applets*" de *Java* o controles de *ActiveX*, así como páginas estáticas utilizando el protocolo *HTML*.

Las principales funciones del Cliente son :

- Servicios de Presentación.
- Manejo de protocolos de seguridad como *SSL, PCT.*
- Realizar validaciones semánticas y sintácticas.
- Manejo de los protocolos de comunicaciones.
- Manejo de protocolos de encripción.

### *Firewalls (Niveles 2 y 3)*

En estos niveles se colocan los *firewalls* que restringen los accesos al entorno bancario, se establecen las barreras de seguridad que impiden el acceso a la red interna del banco de usuarios no-autorizados, se filtran los protocolos de comunicaciones no-permitidos dentro de la red interna y se evita que se tenga el acceso a los computadores del banco a través de sesiones de telnet y/o ftp.

Para esta arquitectura se propone utilizar un *firewall* con doble bastión y un ruteador de chequeo (fig. 5.7), lo que permite tener controlado tanto el acceso de los usuarios externos (*Internet*) y de los usuarios internos (*Intranet*).

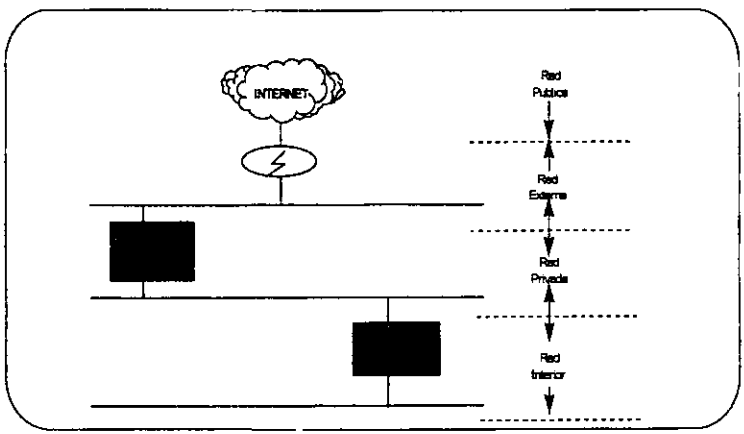



***Fig. 5.7 Firewall con doble Host Bastión y ruteador de chequeo***

Las funciones que realiza un *firewall* o host bastión externo son :

- Recepción y administración de llamadas telefónicas
- Asignación de direcciones *IP*.
- Autentificación de usuarios externos (*Internet*).
- Validación de la zona facultada para usuarios externos.
- Filtrado de una dirección *IP* solicitada como destino contra una dirección única.
- Filtrado de los protocolos *HTML,HTTP,SSL,PCT* y de los servicios de *Tcp/ip*.
- Validación de las reglas de filtrado de paquetes.
- Alerta sobre intentos de accesos no-autorizados.

Las funciones que realiza un *firewall* o host bastión interno son :

- Autentificación de usuarios internos (*Intranet*).
- Validación de la zona facultada para usuarios internos.
- Filtrado de una dirección *IP* solicitada como destino contra una dirección única.
- Filtrado de los protocolos *HTML,HTTP,SSL,PCT* y de los servicios de *Tcp/ip*.
- Validación de las reglas de filtrado de paquetes.
- Alerta sobre intentos de accesos no-autorizados.
- Validación de domicilios *IP* solicitantes contra zonas a las que están facultadas para tener acceso.

### *Servidores Intermedios (Nivel 4.0)*

En este nivel se agrupan el o los servidores intermedios, los cuales soportan la funcionalidad lógica y los servicios. La funcionalidad lógica se refiere a las tareas y reglas que gobiernan la manera en que un negocio opera y los servicios requeridos para implementarlas. Las reglas del negocio se encuentran embebidas en el código de los programas.

Los principales roles que un Servidor Intermedio juega dentro de una arquitectura Cliente/Servidor multi-niveles son :

**1. Flexibilidad en la configuración de aplicaciones.**

En la implementación de una arquitectura de multi-niveles los servidores intermedios apoyan a los componentes de una aplicación distribuida. Debido a que los elementos de una aplicación son módulos independientes ligados por medio de diferentes tipos de interfases, esto da a los desarrolladores la libertad de implantar estos componentes en las plataformas óptimas.

Esta arquitectura permite a los usuarios crear uno o varios niveles intermedios de servidores cuyo propósito es procesar servicios o funciones compartidas por muchas aplicaciones. Los servidores intermedios inyectan escalabilidad y reusabilidad dentro de arquitecturas Cliente/Servidor. En lugar de escribir las mismas funciones o servicios para cada nueva aplicación, los desarrolladores pueden escribir la aplicación una vez y colocar está en un servidor accesible para todas las aplicaciones.

2. **Data Warehouses.**

Los servidores intermedios pueden ser utilizados para organizar y distribu r los datos almacenados en los equipos centrales. Se puede realizar una extracción de los datos pertinentes a un departamento o a un sistema y bajar los datos a un servidor intermedio a donde los usuarios podrán accesarlos para obtener la información que requieran, con el beneficio de que el acceso a estos datos será de una manera más rápida y en forma más eficiente ya que no tendrán que conectarse hasta los servidores corporativos y tampoco tendrán que competir con un gran número de usuarios que también desean accesar las grandes bases de datos corporativas.

3. **Transparencia en la localización de recursos.**

Los servidores intermedios pueden manejar diferentes tipos de *gateways*; de comunicaciones, de bases de datos, etc. Por ejemplo, se puede tener un *gateway* de bases de datos en un servidor intermedio para permitir a los usuarios tener acceso transparente a diferentes tipos de datos (*SQL*, archivos planos, etc.) a través de una red y quitarle a los desarrolladores de aplicaciones la carga de tener que conocer los detalles de cómo se encuentran implementadas las fuentes remotas de datos y de cómo se accesan los datos.

Además de la transparencia en la localización de recursos, los *gateways* en los servidores intermedios también proporcionan transparencia en la migración de recursos. Esto es, los usuarios pueden cambiar la localización o implementación de los recursos de *back-end* sin tener que reescribir el *front-end* de la aplicación. Los servidores intermedios permiten separar a las aplicaciones cliente de los recursos de *back-end*.

4. **Monitores de Transacciones.**

La implantación de los monitores de transacciones en los servidores intermedios permite :

- Aumentar el uso de las bases de datos relacionales con la inclusión de servicios importantes como recuperación (*failover*), balanceo de cargas, deshacer (*rollback*) y recuperación de transacciones. Los monitores de transacciones administran el flujo de las peticiones a los recursos de las aplicaciones, los cuales son típicamente, pero no exclusivamente, motores de bases de datos relacionales.

- Balancear las cargas fluctuantes sobre las múltiples instancias (servidores) del recurso de una aplicación.

- Sensar cuando el servidor de una aplicación ha fallado y automáticamente re-direccionar las peticiones a los restantes servidores disponibles. Esto proporciona alta disponibilidad de los recursos del sistema.

- Implementar un esquema de priorización, en el cual las peticiones de alta prioridad tienen acceso inmediato a los recursos de una aplicación mientras peticiones de baja prioridad son encoladas.

• Asegurar que las actualizaciones a través de una o más bases de datos son manejadas correctamente y que la integridad de los datos no esta en riesgo, debido a que proporcionan capacidades de "*two-phase commit*". Aún cuando, muchos *DBMS* proporcionan servicios de "*commit, rollback, recovery*", los monitores de transacciones corriendo en un gateway intermedio proporcionan un método independiente para implementar estos servicios.

• Concurrencia. En aplicaciones de multi-niveles, la lógica del negocio y los servicios son colocados en servidores intermedios y son accesados por múltiples clientes. Para administrar este ambiente, los servidores aplicativos necesitan poder procesar múltiples peticiones simultáneamente para asegurar adecuados tiempos de respuesta para los usuarios. En esquemas donde las reglas del negocio corren en un servidor separado del motor de la base de datos, la concurrencia debe ser provista por otros medios, típicamente por un monitor de transacciones, el cual puede hilvanar múltiples peticiones en un proceso sencillo.

5. **Servicios Distribuidos**.

Los servidores intermedios proporcionan las bases de la transparencia y extensibilidad para una arquitectura de multi-niveles. Permiten crear un verdadero ambiente distribuido para proporcionar a los clientes acceso a una serie de aplicaciones y servicios corriendo en múltiples servidores intermedios a través de una red de área amplia. En una arquitectura de 3-niveles, el segundo nivel de los servidores intermedios puede funcionar como una capa de servicios distribuidos. Está capa proporciona ruteo dinámico y transparencia de localización para un cómputo distribuido robusto y escalable.

Por otra parte, a los servidores intermedios los podemos clasificar, de acuerdo al tipo de función que realizan, en servidores especializados y servidores funcionales (ver fig. 5.8).

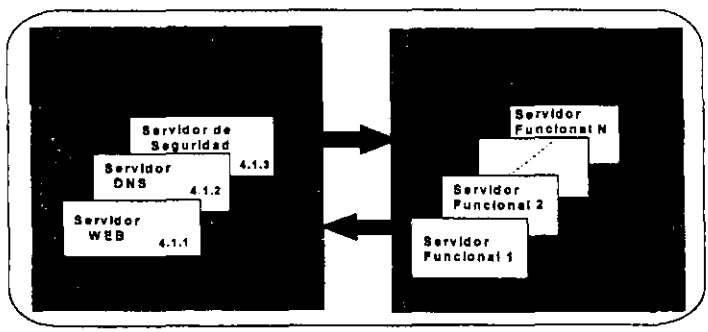



***Fig. 5.8 Clasificación de Servidores Intermedios***

***Servidores Especializados***

Los servidores especializados son aquellos destinados a cubrir una función específica cuyas características demanden una atención especializada en términos de independencia, seguridad, robustez y alta disponibilidad, para cubrir las necesidades planteadas por los ambientes distribuidos. Entre los más importantes están los servidores Web, DNS y de Seguridad :

- El Servidor Web (ver fig. 5.9) se encarga de albergar páginas de contenido y aplicaciones interactivas (*Java applets, ActiveX Controls*), manejar interacción con clientes (navegadores) a través del protocolo *HTTP* y entablar la comunicación con los servidores funcionales. También alberga a los *CGIs* genéricos.

  Un *CGI* genérico es un conjunto de *scripts* o programas externos que actúan como un *gateway* entre el manejador *Web* y los servidores funcionales y/o otras aplicaciones. Sus funciones son :

  ⇒ Conversión de datos.

  ⇒ Procesar datos proporcionados por el cliente.

  ⇒ Enviar los datos a aplicaciones o servidores funcionales.

  ⇒ Facilitar la interfase del cliente.

- Un Servidor *DNS* es el encargado de realizar la conversión de domicilios físicos de *Tcpip* a nombres de Dominios.

- Las funciones de un Servidor de Seguridad son : realizar la autentificación y autorización de acceso de los usuarios tanto internos como externos, manejar los protocolos para la encripción de los datos y para las firmas digitales. También realiza funciones de auditoria.

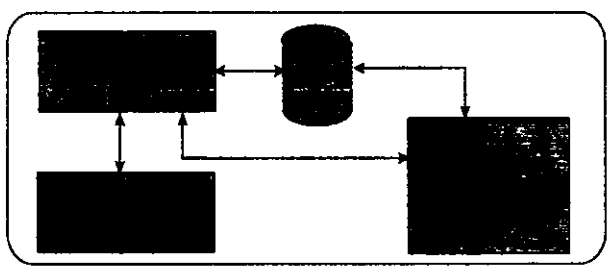


***Fig. 5.9 Servidor Web***

***Servidores Funcionales***

Los servidores funcionales son un conjunto de servidores en los que se encuentran todas las aplicaciones y/o servicios a través de los cuales se realizan las transacciones de datos con los Servidores Corporativos.

Los clientes no tienen acceso directo a estos servidores, sólo por medio de los Servidores Especializados Web, utilizando mensajes a nivel de aplicación.

Los componentes que pueden conformar a un servidor funcional (fig. 5.10) son :

1. Sistema de Acceso.
2. Administrador de Transacciones.
3. Administrador de Salidas.
4. Servicio de Manejo de Datos.
5. Servicios de Interoperabilidad.
6. Servicios de Comunicaciones.
7. Agentes.

**1.** Los elementos que conforman el Sistema de Acceso son :

- Interfase de Comunicación con Dispositivos de Entrega. Está es la encargada de identificar el tipo de dispositivo, manejar los distintos protocolos utilizados por los dispositivos, controlar la interacción con los mismos, tanto para recibir solicitudes de servicio como para dar respuestas a las mismas y comunicar al Servidor Funcional con algunos Servidores Especializados (Web Server).

- Manejador de Dispostivos. Su función, por cada tipo de dispositivo es la normalización de los mensajes (formato común hacia aplicaciones), desnormalización de mensajes (conversión de formato común al formato del dispositivo), validar tipos, justificación y tamaño de datos contenidos en el mensaje de entrada.

- Interfase con el Servidor de Seguridad. Esta interfase valida la completez de los elementos requeridos para autentificar al usuario, interactuar con el o los servidores especializados de seguridad para verificar el acceso y las facultades del usuario y aplicar los criterios para el bloqueo de acceso (intentos, pin, etc).

- Manejador de Sesiones. Se encarga de determinar el tipo de sesión (uni o multi-transaccional), de administrar las sesiones : abrir sesión de trabajo cliente-banco ; cerrar sesión (automáticamente o por petición) ; recuperar sesiones y rutear transacciones al Administrador de Transacciones.

2. Los elementos del Administrador de Transacciones son .

- Analizador de Transacciones. Sus funciones son identificar la transacción, determinar y controlar el flujo de las acciones a seguir y direccionar las transacciones al Administrador de Mensajes.

- Administrador de Mensajes. Este módulo se encarga de armar los mensajes de solicitud hacia los sistemas corporativos, de interpretar los mensajes de respuesta recibidos de los mismos y de mantener las reglas de estructuración de mensajes para cada uno de los sistemas corporativos.

- Gestor de Transacciones. Se encarga de identificar la transacción elemental solicitada, identificar la ubicación de la aplicación que resuelve dicha transacción y ejecutar la transacción elemental enviando los mensajes de solicitud a las aplicaciones involucradas ; genera información para bitácoras.

- Control de Integridad Transaccional. Su función es la de controlar los números de referencia asignados a las transacciones, validar que no se generen transacciones duplicadas, controlar las transacciones que se ejecutaron fuera de tiempo, manejar las reversas de las transacciones, determinar el modo de transmisión de la transacción (síncrona o asíncrona) y generar información para bitácoras.

3. El Administrador de Salidas está compuesto por :

- Formateador de Información de Salida. Se encarga de recibir información del Administrador de Transacciones para complementarla con datos de referencia (fecha, hora, etc.) y formatearla para su registro en bitácora.

- Estadísticas y Auditabilidad. Se encarga de manejar las reglas de conformación de estadísticas, generarlas y formatear la información estadística.

- Generación de Bitácoras. Este modulo se encarga del registro en bitácoras de las operaciones generadas y de la depuración automática de bitácoras.

- Distribuidor de Salidas. Se encarga de formatear y generar las salidas (reportes, estados de cuenta, etc.), mantener los destinos de distribución, realizar la distribución de las salidas y controlar los reintentos de distribuciones fallidas.

4. Servicio de Manejo de Datos.

Su función es la de ofrecer los servicios de administración de datos, de albergar las reglas de negocios, los catálogos institucionales, el catálogo de servicios disponibles y los archivos de bitácoras y estadísticas.

**5.** Servicios de Interoperabilidad.

Su función es la de administrar los diferentes esquemas de conversión de formatos de comunicación entre los distintos ambientes operativos existentes en la organización, además de interactuar con los servicios de comunicación. Parte fundamental de este módulo es la administración y manejo de mensajes entre plataformas.

**6.** Agentes

Los Agentes permiten tener el control sobre el Servidor Funcional y los servicios ofrecidos por el mismo a través de agentes automátizados especializados por función.

Se pueden clasificar de acuerdo a su función en :

- Agente de Monitoreo Operativo. Permite monitorear remotamente el desempeño operativo del servidor.
- Agente de Distribución de Software. Permite la actualización automática de las distintas piezas de *software* que conforman el Servidor Funcional.
- Agente de Inventario. Permite mantener actualizado el inventario de *hardware* de la organización.
- Agente de Monitoreo Aplicativo. Permite monitorear remotamente las señales de alarma de las aplicaciones y procesos que se ejecutan en el Servidor Funcional.

**7.** Servicios de Comunicaciones

Se encarga de manejar los distintos protocolos de comunicaciones, transportar los mensajes entre distintas plataformas, asegurar la integridad en el transporte de los mensajes y controlar los reintentos en el envío de los mensajes. Estos servicios son proporcionados por los diferentes tipos de *middleware*.

***Servidores Corporativos***

Los servidores corporativos son los equipos donde residen las aplicaciones y datos de la empresa y sólo tienen comunicación con los servidores intermedios. Su función es la de procesar las transacciones recibidas desde de los servidores funcionales y mantener las bases de datos corporativas.

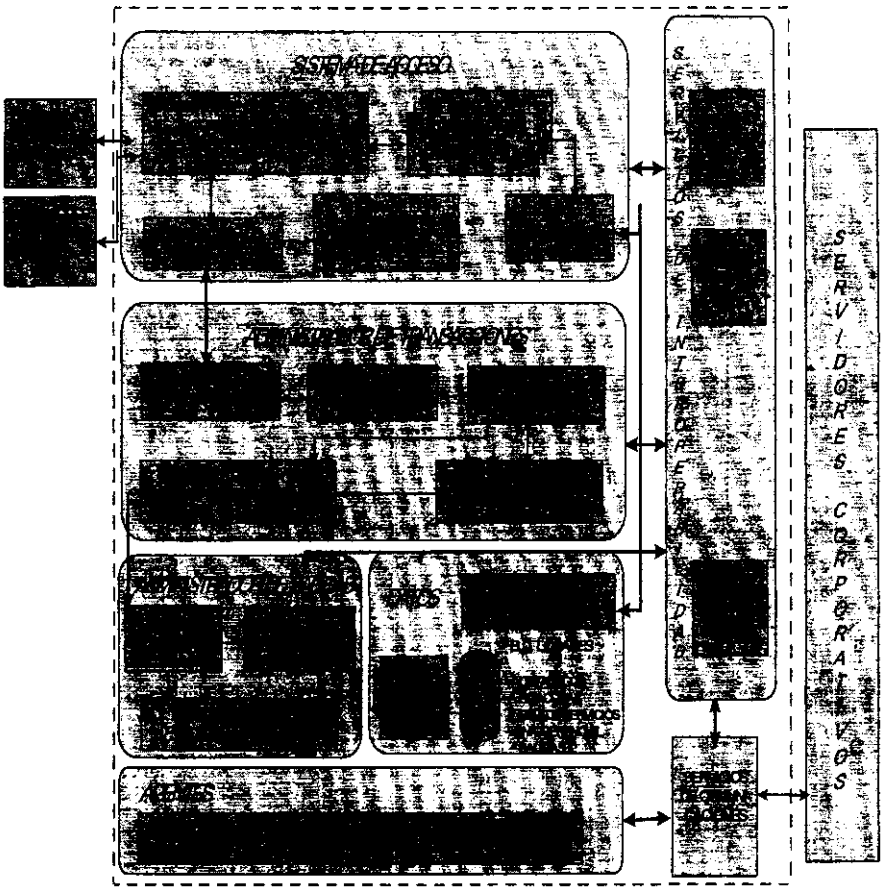


*Fig.5.10 Componentes de un Servidor Funcional*

### *Modelo Físico General*

Para la definición del modelo físico es importante considerar que debido a la evolución de los actuales ambientes computacionales corporativos, los sistemas de *back-end* típicamente constan de una configuración híbrida de diferentes sistemas y equipos. Mientras tanto, el ambiente de *front-end* es una mezcla compleja de diferentes tipos de terminales, computadoras personales, estaciones de trabajo y navegadores de la *Web*, clientes ligeros como computadoras de red y más.

Para unir este conglomerado de elementos, se necesita una tecnología integradora en medio de los sistemas *back-end* y *front-end*, que permita coordinar transacciones y comunicaciones con clientes diversos y muchos diferentes servidores usando un conjunto de reglas para permisos de acceso, políticas de seguridad, perfiles de usuario y procesos de flujo-de-trabajo. Los Monitores de Transacciones (*TPMonitors*) proporcionan esta tecnología integradora (ver fig. 5.11)

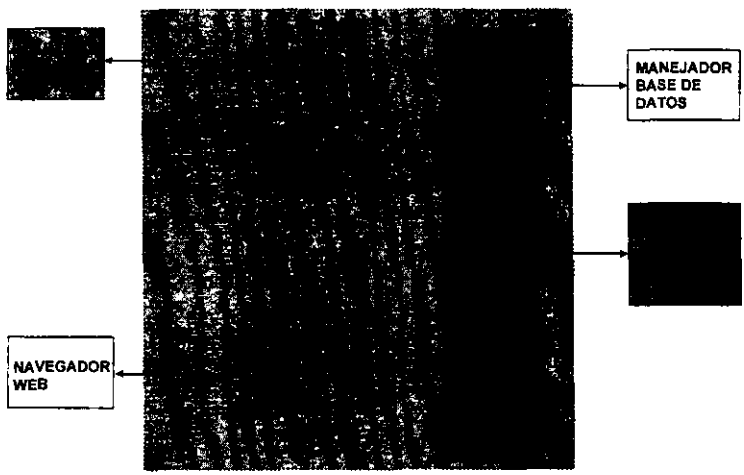



***Fig. 5.11 Arquitectura integradora con Monitores de Transacciones***

Usando las características de los *TPMonitors*, las aplicaciones residentes en las diferentes plataformas pueden ser particionadas para separar aplicaciones complejas, distribuirlas y hacer la administración más fácil moviéndolas a un nivel de componente. Los *TPMonitors* pueden administrar aplicaciones en múltiples *clusters*, sistemas, y máquinas independientemente o con una administración centralizada. Estas características son importantes porque permiten la coordinación de transacciones a través de sistemas heterogéneos, permitiendo abarcar transacciones a través de servidores y sistemas en un ambiente *OLTP* integrado.

Además los *TPMonitors* permiten independencia de sistemas, redes y bases de datos. Una transacción puede incluir cualquier servidor o base de datos soportada por los *TPMonitors*. Esta interoperabilidad permite tomar ventaja de los datos almacenados en los sistemas actuales, mientras da la flexibilidad de escoger diferentes administradores de bases de datos para diferentes partes de la empresa (ver fig. 5.12).

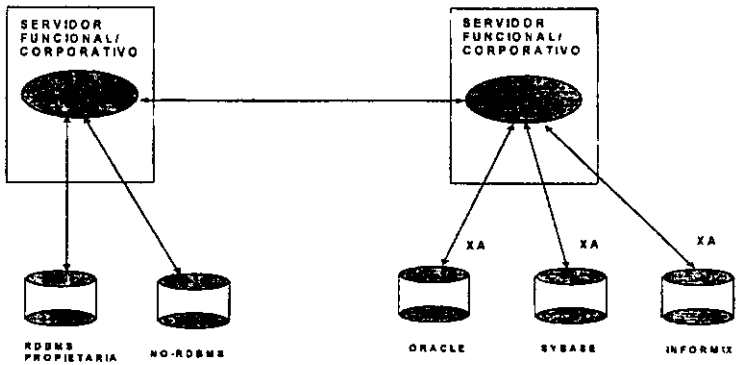



***Fig. 5.12 Acceso a Bases de Datos con Monitores de Transacciones***

Por otra parte, las tecnologías usadas para proporcionar acceso a servicios corporativos de información vía la *Web* han estado desarrollándose conforme pasa el tiempo. Inicialmente, las páginas *Web* estáticas fueron usadas para entregar información de almacenes centrales de páginas hacia los navegadores *Web* a través del protocolo *HTTP*. Posteriormente, las páginas *Web* pudieron ser creadas dinámicamente y permitieron cierto grado de interactividad con el usuario. El mecanismo principal para soportar este estilo de *Web Middleware* es *HTTP*, con una extensión hacia el servidor *Web* llamada *CGI*, la cual facilita conexiones a bases de datos y servidores de aplicaciones.

La utilización de *Java* en la *Web* permite desarrollar componentes de aplicaciones pequeños e independientes de la plataforma, llamados "*applets*" . Estos componentes pueden ser bajados de un servidor *Web* a un cliente con un navegador de acuerdo a la demanda, y al mismo tiempo permanece transparente al usuario final.

Esto permite tener el mismo poder que cualquier aplicación de escritorio convencional, dejando a un lado la pasividad de una página Web estática o las limitadas capacidades interactivas disponibles a través de los *CGIs*.

La integración de las tecnologías *Web* y *middleware* ha permitido el desarrollo de aplicaciones de negocios que necesiten implementarse a través de la *Web*, permitiendo preservar características como escalabilidad e integridad de mensajes.

Las arquitecturas de aplicaciones basadas en la Web pueden hacer uso del *middleware* de 2 maneras : una donde el *middleware* es colocado en el cliente y otra en donde se coloca en el servidor *Web*.

Colocar el *middleware* en el cliente representa una combinación del protocolo de *Web HTTP*, y cualquier tipo de *middleware*. El mecanismo de la *Web* es usado para colocar y bajar un componente de una aplicación cliente desde un servidor hacia la estación de trabajo del usuario. Este componente puede ser un "*applet*" de *Java* o un componente de *ActiveX*.

Además la comunicación entre la aplicación cliente y el servicio *back-end* puede entonces ser conducida usando un producto de *middleware* que reúna los requerimientos del sistema del negocio. Este puede consistir en un acceso directo a una base de datos, o cualquier otro tipo de *middleware*. De hecho, los clientes basados en navegadores pueden ser parte de una arquitectura consistente en cualquier número de capas.

El colocar el *middleware* en el cliente es el escenario más usado en ambientes de *Intranets*, ya que permite que la configuración de las partes cliente y del servidor del *middleware* estén coordinadas.

El colocar el *middleware* en el servidor *Web* representa utilizar el protocolo *HTTP* entre el cliente y el servidor *Web*. Esto significa que la presentación al usuario final es preparada en el servidor *Web* y la *Internet* es solamente el medio de entrega.

Esto requiere que toda la información relevante esté disponible en el servidor *Web* para la construcción de una página. Por lo tanto, cualquier *middleware* puede ser empleado en el servidor *Web* para facilitar el acceso a base de datos corporativas o a servidores aplicativos. Como en el escenario anterior, esté puede ser parte de una arquitectura multi-niveles. La razón obligada para usar *middleware* en el servidor, y no en el cliente, es la significativa reducción de los problemas en la distribución de *software*, esfuerzo de mantenimiento y costo. Esta arquitectura es también adecuada para aplicaciones basadas en *Internet*, por ejemplo, no está restringida a la *Intranet*.

En resumen, la combinación de la *Web* y *middleware* permite a una organización explotar lo mejor de ambos mundos. El paradigma de la *Web* está idealmente situado para la distribución y presentación de información, y proporciona una manera uniforme de navegación para el usuario final. El *middleware*, por otro lado, proporciona un medio flexible para crear esta información de fuentes de datos corporativas, y la integración de la *Web* dentro del contexto de aplicaciones orientadas a transacciones de misión crítica.

La fig. 5.13 muestra una disposición física de los elementos que permiten integrar una solución para la entrega de servicios bancarios por *Internet*. En donde el cliente esta conformado por un navegador de *Internet* .El punto de acceso de los clientes al banco es un(os) firewall(s), estos se comunican con un servidor intermedio especializado donde se localiza el servidor *Web* que maneja la interacción con el cliente a través de "*applets*" de Java o controles *Active X*, también se localiza el servidor que administra y autoriza las sesiones de los usuarios. En este servidor intermedio se ubica la parte cliente del monitor de transacciones, la comunicación entre el servidor *Web* que recibe los datos de la transacción y el monitor de transacciones es a través de un *CGI*.

Por otra parte, se cuenta con un servidor intermedio funcional que lleva el control y la administración de las transacciones. El servidor intermedio especializado y el servidor intermedio funcional se comunican a través del monitor de transacciones una vez que el monitor de transacción tiene bajo su control la transacción la hace llegar a los sistemas aplicativos ubicados en los servidores corporativos. La comunicación la realiza ya sea a través del monitor de transacciones si el servidor corporativo a donde quiere llegar cuenta también con un monitor de transacciones, sino la realiza utilizando algún middleware disponible en el servidor corporativo, en ambos casos el control de la integridad de la transacción lo lleva el monitor de transacciones del servidor funcional.

En los servidores corporativos se encuentran los sistemas actuales y las bases de datos propietarias o relacionales de cada plataforma.

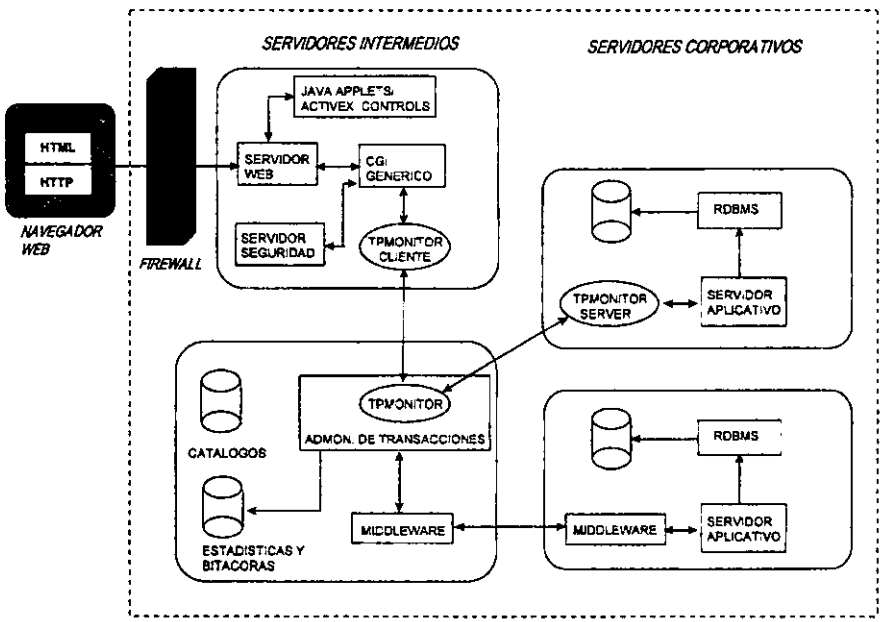



***Fig.5.13 Modelo Físico***

# CONCLUSIONES

A las puertas de un nuevo milenio y ante un proceso de globalización económica, estamos viendo que cada día hay mayor aceptación por parte de los clientes bancarios con respecto a las innovaciones tecnológicas como la banca en *Internet* y el comercio electrónico, siguiendo los patrones vistos con anterioridad con la entrada de la banca por teléfono y los cajeros permanentes. El ritmo vertiginoso de las grandes ciudades, la necesidad de relaciones comerciales internacionales y la necesidad de contar con información más oportuna están haciendo que el cliente bancario sea mucho más sensible a la diversidad de medios de entrega que le ofrezca un banco, buscando la comodidad de operar sus transacciones desde su oficina o domicilio, con horarios cada vez más amplios y con respuestas inmediatas.

La regla del juego hoy en día, y no sólo para el sector financiero, es escuchar al cliente y ofrecerle lo que está buscando. Los bancos que se cieguen por una inercia comercial y que no quieran ver el cambio en el mercado, están destinados a desaparecer. Siempre seguirá existiendo un segmento del mercado con deseos de seguir accediendo a los servicios bancarios por los medios convencionales, sin embargo el segmento de mercado con necesidades de una banca de conveniencia seguirá una tendencia de crecimiento. Será este segmento el que se estarán disputando los bancos en los próximos años.

Las tendencias tecnológicas son realmente impredecibles y difícilmente existe algún banco que tenga todas las respuestas. Sin embargo, las instituciones que basen sus estrategias de banca de menudeo en los cambios de estilo de vida y tecnología serán las que estarán en las posiciones más fuertes en el futuro.

Una de las principales tendencias que están surgiendo es la de la banca por *Internet*. Debido al gran crecimiento de usuarios de *Internet* que se está dando, se estima que cada año se duplique el número de usuarios.

Los bancos deben de aprovechar este crecimiento de usuarios de Internet para ofrecer a sus clientes sus servicios por este nuevo medio de entrega y a la vez atraer a nuevos clientes.

Otra de las tecnologías que está consolidándose, una vez resuelto el problema de la seguridad en las transacciones, es el comercio electrónico por Internet, donde los bancos pueden ser un protagonista en el sistema de pagos con tarjetas de crédito.

Por lo anterior, es importante que los bancos empiecen a crear su infraestructura para ofrecer el servicio de Banca por Internet y responder a los retos de una banca que está ofreciendo sus servicios a través de una gran variedad de medios de entrega (banca por teléfono, cajeros permanentes, kioscos de autoservicio, etc.) y donde la tendencia es ir disminuyendo el número de sucursales tradicionales a la vez que aumentan los medios de entrega electrónicos y el número de servicios que se ofrecen por estos medios.

Además, la banca por medio de computadoras personales está creciendo, debido principalmente a los paquetes para las finanzas personales que los bancos han distribuido entre sus clientes, esto está propiciando que tenga lugar una integración entre el *software* basado en *Internet* y los paquetes para finanzas personales, desde donde se puede tener acceso a los sitios Web de los bancos.

Por otra parte, hoy en día los grandes proveedores de *software* están ofreciendo una gran cantidad de productos para integrar los sistemas actuales a la tecnología de *Internet* , por lo cual el desarrollo de la infraestructura que requieren los bancos para ofrecer sus servicios por Internet puede realizarse en un corto/mediano plazo dependiendo de las características y amplitud de servicios que se quieran ofrecer y de la facilidad de integración que tengan sus sistemas actuales con las nuevas tecnologías.

La arquitectura propuesta en esta tesis es solo una de diferentes formas en que se puede implementar una arquitectura para ofrecer servicios bancarios por *Internet*.

Cada banco podrá implementar su infraestructura de acuerdo a las plataformas con que cuente, a su capacidad tecnológica, y al volumen de transacciones que espera manejar.

Lo que se pretende con este trabajo es mostrar por un lado la importancia y el futuro que *Internet* tendrá en la relaciones comerciales tanto de la empresas como de las personas y por otra lado que toda la tecnología necesaria para que los bancos empiecen a incorporar a Internet como un medio de entrega está disponible.

Debido al gran dinamismo que presenta la tecnología, posiblemente algunos tópicos pudieran parecer obsoletos o atrasados pero lo que se buscar es mostrar los conceptos que permitan diseñar una arquitectura para usar a *Internet* como un medio de entrega de servicios bancarios.

Esta integración de servicios bancarios, de comercio electrónico, de una comunicación globalizada a través de Internet y del dinero electrónico, está dando origen a la economía digital que hace unos años sonaba como una utopía pero que ahora ya se esta volviendo una realidad, a la cual todos debemos de empezar a incorporarnos para ser competitivos en el siguiente milenio.

# GLOSARIO

4GLS. Lenguajes de programación más avanzado que los lenguajes tradicionales tales como C, Pascal o Fortran. Muchas personas consideran a los programas de lenguaje de consulta [query-language] y a los generadores de informes disponibles comercialmente como lenguajes de la cuarta generación.

ActiveX. Piezas de software o componentes desarrollados por Microsoft para ser ejecutados por navegadores (browsers).

Applets. Aplicaciones escritas en Java que se compilan bajo un formato intermedio independiente de la arquitectura de cómputo (código a nivel de *bytes*), que posteriormente es leído y ejecutado por intérpretes en una máquina virtual ejecutándose en diferentes plataformas. Los applets residen en los servidores Web y son bajados a los clientes para que se ejecuten localmente.

Ascii. Es un código que usa 7 bits para representar caracteres gráficos (letras, números y signos de puntuación) y de control, es un estándar que se utiliza con frecuencia para intercambiar información entre la mayoría de las computadoras.

Back-End. Programas que realizan el procesamiento los datos de las aplicaciones en los servidores. Generalmente son la interface con los DBMS.

Bits. El bloque de información mas pequeño que puede manejar una computadora.

Byte. Unidad básica de medida de la memoria de una computadora. Un byte contiene 8 bits. Cada carácter puede representarse por un byte en código ASCII.

Cache. Area especial de la memoria, manejada por un controlador de memoria caché. Permite un mejor rendimiento de la computadora, almacenando el contenido de las ubicaciones de la memoria a las cuales se tiene acceso con mayor frecuencia, así como las direcciones donde se encuentra dicho contenido. Cuando el procesador hace referencia a una dirección de memoria, el caché verifica si tiene almacenada esa dirección; si la tiene, la información se pasa de inmediato al procesador, de modo que no sea necesario recurrir a la memoria de acceso aleatorio [random access memory (RAM)].

Caching. Proceso de referenciación de la memoria cache.

CASE (Computer Aided Software Engineering). Software de desarrollo utilizado para ayudar en todos los aspectos del ciclo de vida del software, incluyendo el diseño, codificación, prueba, documentación y mantenimiento del software. La ingeniería del diseño de software asistido por computadora proporciona un conjunto de herramientas de programación y de desarrollo que ayuda a los programadores a automatizar la producción de software comercial, software técnico y software de ingeniería.

*Checksum*

Clusters. Conjunto de equipos separados físicamente pero que lógicamente unidos como uno solo.

CORBA (Common Object Request Broker Architecture). Un sistema que proporciona interoperabilidad entre objetos en un ambiente heterogéneo distribuido de manera transparente para el programador.

DBMS (Data Base Management System). Software de aplicación que controla los datos en una base de datos, incluyendo la organización global, el almacenamiento, la recuperación, la seguridad e integridad de los datos. Para la salida impresa, un sistema de administración de bases de datos (DBMS) puede dar formato a los informes, y a los datos importados de (o exportados hacia) otras aplicaciones, utilizando formatos de archivo estándar. También proporciona un lenguaje de manipulación de datos para dar apoyo a las consultas de la base de datos.

DCE (Distributed Computing Environment). Estandar establecido por la *Open Software Foundation (OSF)* que permite entre otras cosas, definir cuales usuarios tienen permitido el acceso a un recurso dado.

Downsizing. Rediseño de aplicaciones para negocios (diseñadas originalmente para mainframes) para convertirlas en aplicaciones capaces de ejecutarse en sistemas más pequeños y menos costosos, a menudo en las redes de área local de las computadoras personales. La arquitectura/cliente servidor es el modelo frecuentemente utilizado durante a reducción de tamaño [downsizing].

Para hacer más significativa la concordancia que debe existir entre lo ofrecido y los requerimientos para aplicaciones específicas que tenga una empresa, así como con los recursos operacionales del sistema de hardware y de software que ésta tiene disponibles, un término más apropiado que reducción de tamaño [downsizing] puede ser adecuación computacional [rightsizing].

Ethernet. Protocolo de red y esquema de cableado con una velocidad de transferencia de 10 Megabits por segundo (Mbps), utiliza una topología de bus.

FDDI. Especificación para las redes de fibra óptica. Transmite a una velocidad de hasta 100 Megabits por segundo (Mbps) en una topología de anillo de señal doble contra rotativa. La interface de datos distribuidos por fibra óptica [Fiber Distributed Data Interface (FDDI)] resulta apropiada para los sistemas que requieren la transferencia de grandes cantidades de información.

Front-End. Programas que tiene contacto directo con el usuario y que coordinan la lógica de la aplicación con un servidor.

Gateway. Puente o compuerta. Sistema que permite que una red o un grupo de usuarios tenga acceso a Internet. Las compuertas a menudo se utilizan para convertir el protocolo de una red en el de otra.

Hardware. Comprende todos los dispositivos físicos que conforman una computadora.

Hash. Función o método para transformar uno o más campos (usualmente una llave) dentro de un arreglo diferente (usualmente más compacto).

Host . Es una computadora, generalmente Mainframes, conectada en red que centraliza datos y aplicaciones disponibles para todos los clientes.

Icono. En una interface gráfica de usuario (GUI), pequeña imagen que representa un elemento específico que el usuario puede manipular de algún modo en la pantalla. Un icono se selecciona con el ratón (mouse) u otro dispositivo apuntador.

El icono puede representar a un programa de aplicación, un documento, objetos incrustados y objetos vinculados, una unidad de disco duro, o varios programas recopilados en un mismo icono que representa a un grupo.

Mainframe. Sistema de cómputo multiusuario, grande y rápido, diseñado para manejar grandes cantidades de datos y tareas de computación complejas. Las maxicomputadoras (mainframes) normalmente están instaladas en las grandes corporaciones, las universidades o en proyectos militares y pueden dar apoyo a cientos, o hasta miles, de usuarios.

A medida que el hardware de las computadoras continúa reduciéndose, se hace más difícil catalogar las computadoras basándose sólo en el tamaño, y debe emerger una clasificación basada en la función (que ésta desempeñe).

Mbps (Megabits per second). Medida de la cantidad de información que se mueve por una red o sobre un enlace de comunicaciones en un segundo, medido en múltiplos de 1.048.576 bits.

Modem. MODulator-DEMulator. Dispositivo que convierte señales digitales de una computadora en señales analógicas de un teléfono (señales audibles que pueden mandarse vía telefónica) e invierte el proceso al final de la línea.

MOM (Message-Oriented Middleware (MOM). Middleware que facilita la comunicación asíncrona, punto-a-multipunto—esto es, maneja la comunicación entre los clientes y las bases de datos, con una entrega garantizada del mensaje.

Mouse (Ratón). Pequeño dispositivo de entrada, con uno o más botones incorporados, que se utiliza con las interfaces gráficas de usuario. A medida que el mouse se desplaza, un cursor del mouse duplica todos los movimientos en la pantalla, los cuales son relativos. Una vez que el apuntador del mouse está en la posición correcta en la pantalla, usted puede pulsar uno de los botones del mouse para iniciar una acción o una operación.

Navegadores (Browsers). Un navegador es una aplicación cliente que es usada para ver, buscar y navegar a través de la información en *Internet*. Dentro de los navegadores más populares hoy en día podemos incluir a Netscape Navigator y Microsoft Internet Explorer que ofrecen interface gráfica a la World Wide Web.

Object Oriented (Orientación a Objetos). Término que se puede aplicar a cualquier sistema de cómputo, sistema operativo, lenguaje de programación, programa de aplicación o a la interface gráfica de usuario, todos los cuales dan apoyo al uso de objetos.

ODBC (Open Database Connectivity). Formato estándar para acceso a base de datos.

ORB (Object Request Broker). Herramienta que pasa requerimientos de los clientes a la implementación de los objetos a los cuales son invocados.

Overhead. Es el tiempo que una computadora gasta en realizar cálculos que no contribuyen directamente al progreso de una tarea dentro del sistema. Es la sobrecarga de recursos que un proceso durante su ejecución.

RDB (Base de Datos Relacional). Modelo de base de datos en el cual los datos siempre se muestran desde el punto de vista del usuario, organizados como una tabla de dos dimensiones, presentados en forma de filas y columnas. Las filas en una tabla representan registros, los cuales son colecciones de información acerca de un tema específico. Una base de datos relacional puede enlazar dos o más tablas, produciendo una nueva tabla que contiene los datos que se solicitan de ambas tablas.

RISC (Reduced Instruction Set Computing). Procesador que reconoce un número limitado de instrucciones en lenguaje ensamblador. Los procesadores de conjunto de instrucciones reducidas se pueden diseñar para que funcionen hasta un 70 por ciento más rápido que los procesadores de conjunto de instrucciones complejas (CISC).

Sockets. Concatenación de una dirección de Tcpip y un número de puerto que juntos especifican un proceso particular o un servicio dentro de Internet.

Software. Instrucciones codificadas electrónicamente (programas que dirigen a la computadora para realizar ciertas tareas).

SQL(Structured Query Languaje). Lenguaje estructurado utilizado para crear, consultar, modificar y tener acceso a datos organizados en tablas de bases de datos relacionales.

Unix. Sistema operativo multitareas para una amplia variedad de computadoras desde Mainframes hasta estaciones de trabajo .