UNIVERSIDAD NACIONAL DE INGENIERÍA

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

INTERCONEXIÓN DE SERVICIOS TCP/IP CON ATM.

INFORME DE SUFICIENCIA

PARA OPTAR EL TÍTULO PROFESIONAL DE:

INGENIERO ELECTRÓNICO

PRESENTADO POR:

NAPOLEÓN STEIN CERNA ROSAS.

PROMOCIÓN
1987- I

LIMA – PERÚ
2006

INTERCONEXIÓN DE SERVICIOS TCP/IP CON ATM.

Dedicado a mi Madre

A quien todo le debo.

SUMARIO

En un mundo Globalizado como el actual e indispensable el compartir información y uno de los medios mas eficientes en este aspecto son las computadoras, Internet es la red publica mas usada a nivel mundial, como es conocido por todos nosotros la arquitectura usada es TCP/IP .

Oe otro lado la tecnología ATM a demostrado con creces la eficiencia de su funcionamietpo a nivl WAN

En este Informe se trata de dar a conocer las diferentes características de la tecnología ATM. Protocolos de comunicación, tipos de celdas, tipos de enlaces y relacionarla con la Arquitectura TCP/IP aprovechando lo mejor de ambas estructuras y asi obtener mejores resultados para la transmisión de voz, datos y video con QoS bajo demanda

En ste informe se trata de resaltar la importancia de la tecnolgía ATM en Redes WAN, MAN y LAN, con calilidad de servioo , la optiización en costo bajo demanda y realacionar esta con la arquitectura TCP/IP la cuál es un estandar en la comunicación via Internet, relacionando los protocoos en la estructura jerarquica de los diferentes capas para poder obtener una adaptación coherente y aprovechar los beneficios de ambas estructuras , tanto en calidad de servicio, control de trafico, costo bajo demanda y nlac de servicios en sistemas conmutados.

INDICE

CAPITULO IV
IP SOBRE ATM

CAPITULO V
TECNOLOGÍA DE CONVERGENCIA

CAPITULO VI
CONEXIÓN AVANZADA DE TCP/IP SOBRE ATM

CAPITULO VII
MODELO DE CONEXIÓN

PRÓLOGO

El objetivo de este informe es el de analizar los diferentes servicios, protocolos, utilidades y herramientas que TCP/IP brinda para la interconexión con redes ATM Debido a la gran demanda de Redes con Tecnología ATM..

Lograr así una mayor eficiencia en este tipo de comunicaciones, mayor rendimiento, diferentes técnicas a nivel de hardware y software, entendiendo mejor la comunicación y conociéndola para poder hacer cambios puntuales de acuerdo a una realidad especifica.

Los administradores de red encontrarán información importante, estructurada y sencilla de utilizar, sobre todo, aquello que necesiten configurar y sobre la forma de configurarlo.

Los ingenieros de soporte obtendrán información práctica y detallada sobre las herramientas disponibles para la resolución de problemas, así como técnicas sobre la mejor forma de utilizarlas.

Para los consultores y diseñadores se han incluido análisis detallados, y los programadores obtendrán información sobre las distintas interfaces de desarrollo

CAPITULO I
TECNOLOGÍA ATM

1.1 Introducción

Es una tecnología que divide la información en paquetes, cada paquete tiene un tamaño constante de 53 bytes de los cuales 5 conforman la cabecera y el resto es la información, se basa en la conmutación por paquetes teniendo este tipo de transmisión una ventaja sobre la conmutación por circuitos, pues permite difusión, flexibilidad y facilidad en la transferencia de información.

Las redes ATM son orientadas a la conexión eso quiere decir para lograr el enlace se debe en principio enviar señales de control que permitan entablar la conexión, una vez entablada la conexión se define la trayectoria y los paquetes viajan en forma ordenada uno tras otro es decir el paquete 1 llega primero luego el paquete 2 y así sucesivamente

Las velocidades desarrolladas en esta tecnología van desde los 155Mbps , 622 Mbps hasta los Gbps

La tecnología ATM se puede usar en LAN, MAN y WAN, es orientada a la conexión, full duplex, punto a punto y de celda conmutada

1.2 Concepto y Operación

Se dice orientada a la conexión por que para lograr la comunicación debe haber primero señalización, full duplex por que la comunicación es en ambos sentidos a la vez y de celda conmutada por ser similar al frame relay, el ATM usa celdas de longitud fija 53 bytes 48 para datos y 5 para sobrecarga, también el ATM es conocido como celda conmutada por emplear multiplevaje en el dominio del tiempo TDM y conmutación de paquetes

Es una tecnología basada en celdas inicialmente se empezó a usar en redes de banda ancha (B-ISDN) el modelo de referencia está dado por 3 capas. La capa física la capa ATM y la capa de adaptación ATM.

El ATM tiene sus propios protocolos de adaptación y administración, facilidades de calidad de servicio, control de flujo y rendimiento, es un sistema que soporta otro sistema encima de él, Es independiente y transparente a otros sistemas

La tecnología ATM fue diseñada desde el principio para transmitir voz, datos y video a la vez. Por lo tanto puede usarse para cualquier tipo de ambiente de red actualmente y en el futuro.

Características:

Garantiza la entrega de la información sensible al tiempo por ejemplo la exactitud en la entrega de información implica que se debe reservar de manera apropiada el espacio apropiado (ancho de banda) sin degradar el rendimiento de la red, para lograr ello todas las etapas intermedias (dispositivos de conmutación ATM) deben lograr el espacio necesario para poder trasladar la información en cuestión en el intervalo de tiempo requerido (comunicación isócrona) esto requiere de varias confirmaciones de conexión en el camino

Características:

De las Celdas ATM

Transparencia para las tecnologías que están encima de é{, manejando sus propios controles de administración.(es como una red dentro de otra red), protocolos de comunicación como el TCP/IP y SNA no saben que están sobre ATM.

De la Rapidez

Rendimiento

Calidad en el servicio.

De operación Operación

Las celdas son transmitidas de fuente al destino mediante una subred conmutada ATM los nodos finales se comunican con un dispositivo ATM por medio de una interfaz usuario a red (UNI), y los conmutadores ATM se comunican entre si con una interfaz de red a red (NNI), UNI y NNI son solo 2 de una variedad de componentes estándar ATM

En ATM hay una jerarquía de interfaces estandarizadas llamadas el **Anchorage Accord** (acuerdo de anclaje)

Los estándares ATM están definidos en 2 grupos Básicos y de Aplicaciones y Servicios .los básicos definen funciones primarias de comunicación y las últimas Inter. operan con componentes ATM de red.

En una red ATM el tipo de hardware es transparente a la red esto significa que datos (computador a computador) video (cámaras) y voz (PBX) pueden compartir simultáneamente la misma infraestructura de la red, y esto se logra usando los estándares funcionales para proporcionar interfaces exactas para diferentes tipos de datos.

El ATM usa direcciones para cada punto de conexión para redes privadas el ATM usa los puntos de red de acceso al servicio **(NSAP**) de OSI , en las redes publicas ATM emplea el **formato E.164** especificado por la ITU-T, pero se pretende estandarizar todos los servicio públicos y privados en base a NSAP

Para conectar un dispositivo a una red ATM debe instalarse una tarjeta de red con tecnología ATM, luego este es conectado a un conmutador local proporcionado por la empresa que da el servicio, la conexión física puede ser con cable de cobre UTP. F.O. o inalámbrico la especificación funcional para que se enlacen es la especificación de interfaz de usuario a red UNI.

La función de ésta especificación es definir exactamente como se comunica la estación en un extremo con el siguiente nivel de interfaz de red, especifica el establecimiento de conexión entre la NIC y el conmutador, la calidad de los requisitos de servicio para el tráfico entre la NIC y el conmutador y todas las consideraciones orientadas a la conexión que ATM requiere para garantizar el correcto enlace entre la estación y el conmutador, además especifica atributos de hardware tales como aspectos referente a la interfaz de cobre, de fibra y funcionalidades de los componentes inalámbricos, a lo que se conoce como **el señalamiento UNI**

La comunicación conmutador a conmutador en una red privada se llama interfaz privada de red a red (PNNI) la PNNI proporciona una funcionalidad especifica que conmuta la necesidad entre cada uno pero que las estaciones finales no atienden en absoluto.

Los conmutadores ATM son los responsables de transmitir la información dentro de ATM

Hay diferentes funcionalidades en ATM que permite la interconexión con otros tipos de redes tales como ISDN , (B-ICI), LAN (LUNI) etc.

Un conmutador ATM implementa un método de enrutamiento llamado **tejido conmutador** se presentan una amplia variedad de implementaciones pero los más comunes son **Batcher-banyan y delta** .los conmutadores de la red intercambian información para saber en el contexto de la red ¿dónde están ubicados los SAP fuente y destino? empleando técnicas de alta velocidad , cuando los conmutadores por primera vez son energizados se configuran así mismos les permite saber automáticamente ¿que otros conmutadores están conectados? También envían direcciones NSAP temporales a puntos de conexión que puedan necesitarse para ínter operar y administrar el ambiente completo de la red desde una perspectiva de administración y configuración de tráfico. Ver figura 1.1.

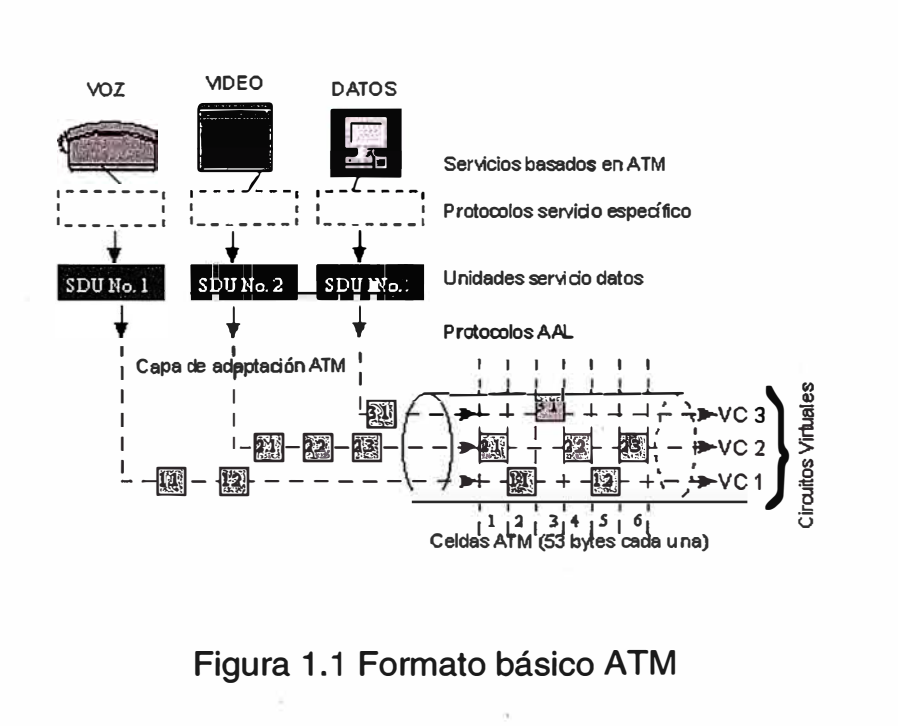


Figura 1.1 Formato básico ATM

ATM es la unica red construida desde el principio para soportar datos voz y video al mismo tiempo por tanto una red ATM puede usarse para casi cualquier tipo de ambiente de red actualmente y en el futuro.

CAPITULO II
FORMAS DE CONEXIÓN

2.1 Conexiones Virtuales y Direccionamiento.

Una de las principales características de la tecnología ATM es que es orientada a la conexión esto hace que las transferencias se realicen a través de los circuitos virtuales (VPI/VCI) VPI (identificador de ruta virtual) VCI (identificador de canal virtual) establecidos extremo a extremo y los cuáles se mantienen abiertos durante toda la comunicación, estos circuitos virtuales son creados en la etapa de establecimiento de la conexión que es cuando el usuario de la red puede especificar los parámetros del tráfico que va a generar o los recursos de la red que van a requerir asimismo el usuario negocia la calidad del servicio que espera recibir de modo que la propia red dispone de mecanismos de gestión de recursos como **la función CAC** (control de admisión de la comunicación) usada para la negociación de conexión evita la entrada de usuarios en donde la red no dispone, evita la sobrecarga de recursos en ese momento y la **función UPC** (control de parámetros de uso) vela por el buen funcionamiento del sistema en la comunicación verificando el cumplimiento del contrato establecido ATM no solo se encarga del establecimiento de las conexiones sino también del estado general de la misma evitando el sobre dimensionamiento del sistema.

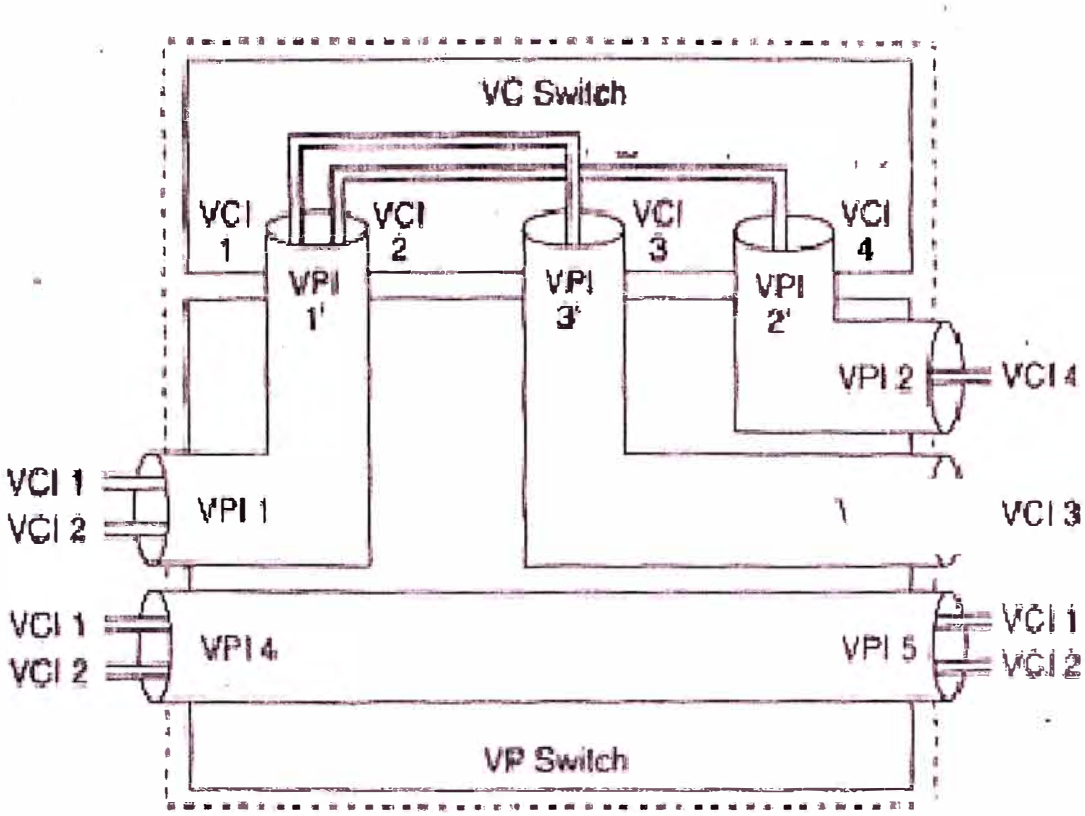


Figura 2.1 Direccionamiento en ATM

2.2 Direccionamiento de celdas

Cada célula lleva en su cabecera un identificador VPI de 8 bits y un identificador VCI de 16 bits, el campo PT de 3 bits es el que identifica el tipo de carga útil que tiene, el bit CLP lo emplea el emisor para especificar la prioridad deseada cuando aparecen situaciones de congestión. el campo HEC empleado para la detección de errores detecta todos los errores de los 32 bits restantes de la cabecera

Una conexión ATM, consiste de "celdas" de información contenidos en un circuito virtual (VC). Estas celdas provienen de diferentes fuentes representadas como generadores de bits a tasas de transferencia constantes como la voz y a tasas variables tipo ráfagas (bursty traffic) como los datos. Cada celda compuesta por 53 bytes, de los cuales 48 (opcionalmente 44) son para información y los restantes para uso de campos de control (cabecera) con información de "quién soy" y "donde voy"; es identificada por un "virtual circuit identifier" VCI y un "virtual

path identifier" VPI dentro de esos campos de control, que incluyen tanto el enrutamiento de celdas como el tipo de conexión. La organización de la cabecera (header) variará levemente

Dependiendo de sí la información relacionada es para interfaces de red a red o de usuario a red. Las celdas son enrutadas individualmente a través de los conmutadores basados en estos identificadores, los cuales tienen significado local ya que pueden ser cambiados de interface a interface.

La técnica ATM multiplexa muchas celdas de circuitos virtuales en una ruta (path) virtual colocándolas en particiones (slots), similar a la técnica TDM. Sin embargo, ATM llena cada slot con celdas de un circuito virtual a la primera oportunidad, similar a la operación de una red conmutada de paquetes..

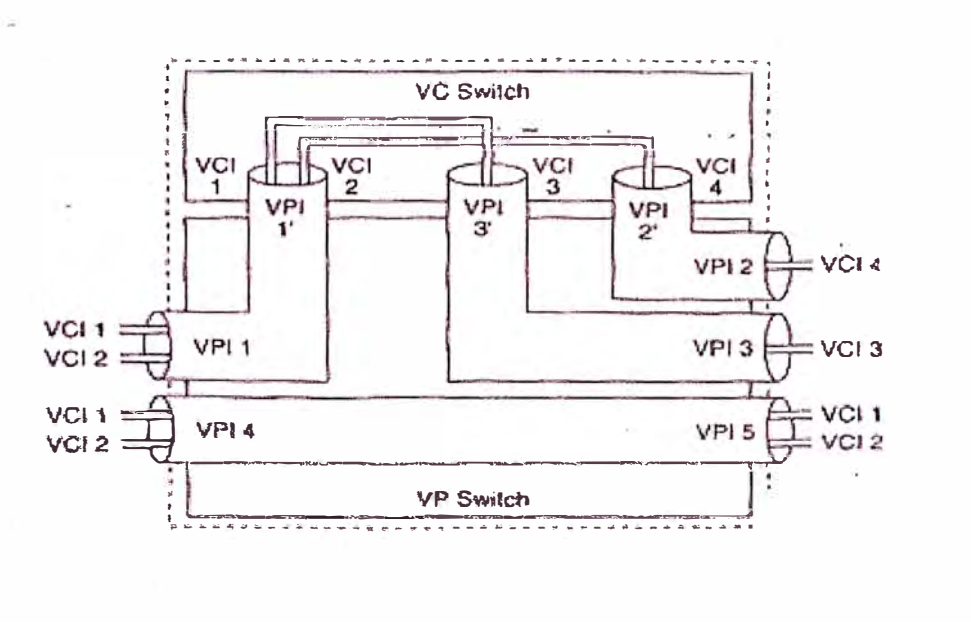


Figura 2.2 Enrutamiento en ATM

Los slots de celda no usados son llenados con celdas "idle", identificadas por un patrón específico en la cabecera de la celda. Este sistema no es igual al llamado "bit stuffing" en la multiplexación Asíncrona, ya que se aplica a celdas enteras.

Diferentes categorías de tráfico son convertidas en celdas ATM vía la capa de adaptación de ATM (AAL - ATM Adaptation Layer), de acuerdo con el protocolo usado.

2.3 Trama ATM

Existen 2 tipos de tramas, en el caso de ATM se les llama celdas, la celda UNI (interfase entre Nodo y usuario) y la NNI (interfase entre Nodo y Nodo).

ATM ha sido definido para soportar de forma flexible, la conmutación y transmisión de tráfico multimedia comprendiendo datos, voz, imágenes y vídeo. En este sentido, ATM soporta servicios en modo circuito, similar a la conmutación de circuitos, y servicios en modo paquete, para datos

Sin embargo, a diferencia de la conmutación de circuitos, ATM no reserva "slots" para la conexión. En su lugar, una conexión obtiene "slots" o celdas, solo cuando está transmitiendo información. Cuando una conexión está en silencio no utiliza "slots" o celdas, estando estas disponibles para otras conexiones. Con esta idea en mente, se decidió que la unidad de conmutación y transmisión fuese de tamaño fijo y longitud pequeña. Esta unidad es conocida como Celda, y tiene una longitud de 53 bytes divididos en 5 de cabecera y 48 de información o carga útil. Esta celda es quien viene a sustituir al "Time Slot" o contenedor del STM

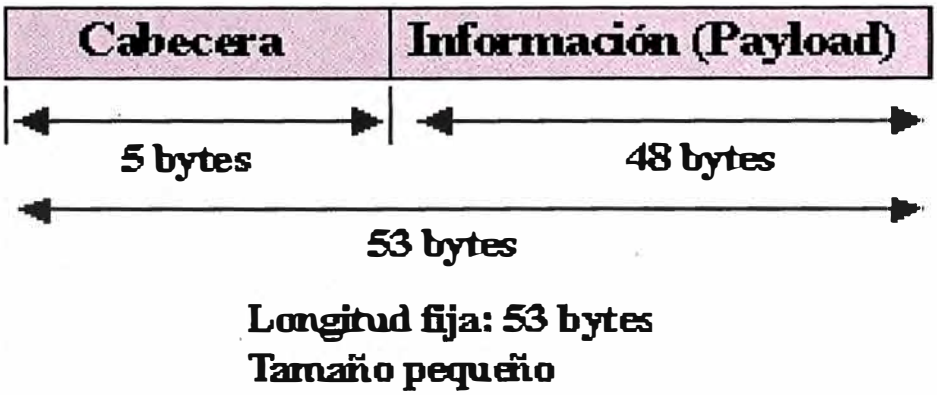


Figura 2.3 Celda ATM

En una red ATM, donde las celdas no están reservadas sino asignadas bajo demanda, el conmutador receptor no puede determinar por adelantado a que canal corresponde cada celda. La Celda ATM a diferencia del Time Slot en STM, debe transportar la identificación de la conexión a la que pertenece, de esta forma no existirán Celdas vacías ya que serán utilizadas por conexiones pendientes. Esta es una diferencia fundamental del ATM frente al STM. La cabecera presente en cada celda, consume aproximadamente un 9.5% del ancho de banda, siendo

este el precio que hay que pagar por la capacidad para disponer de ancho de banda bajo demanda, en lugar de tenerlo permanentemente reservado y eventualmente desperdiciado.

La adopción de una cabecera de 5 bytes ha sido posible, porque no se realiza recuperación de errores en los nodos intermedios, tampoco se emplean direcciones válidas a nivel de toda la red, tales como la dirección MAC en Ethernet o IP en redes tipo TCP/IP

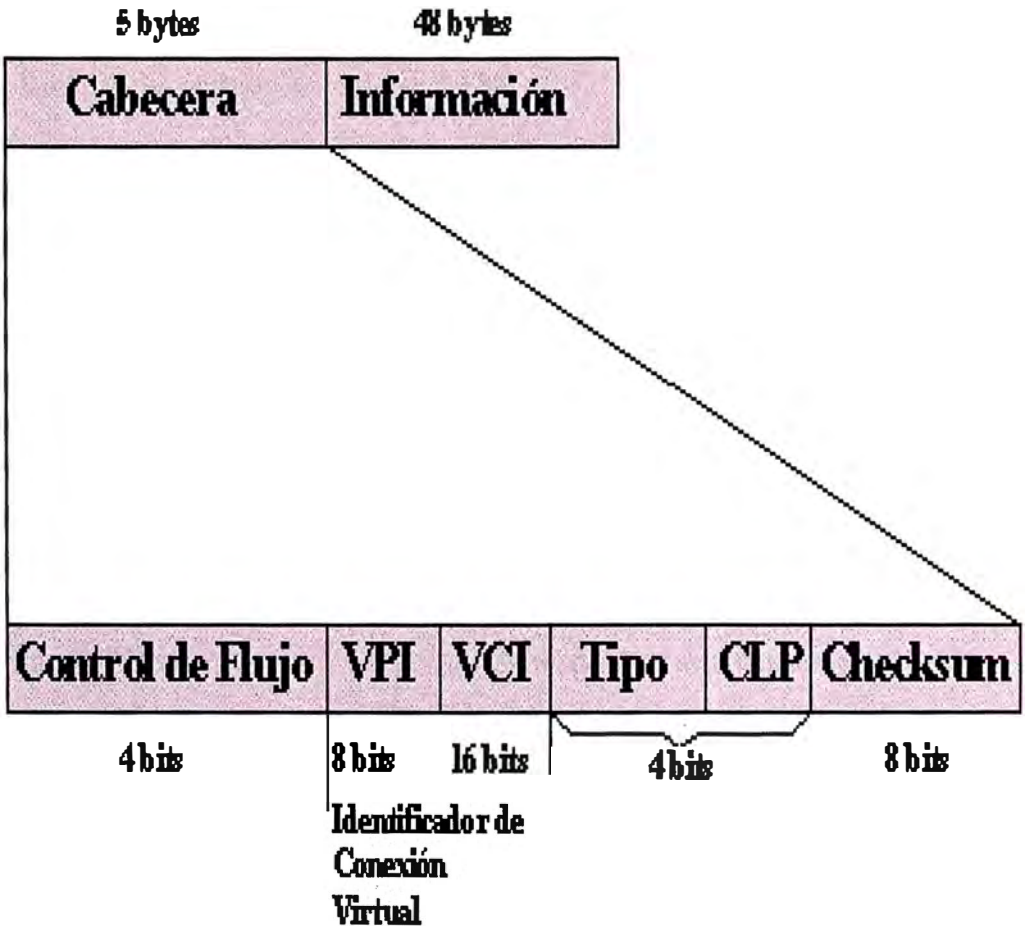


Figura 2.4 Cabecera ATM

Al igual que en las redes de conmutación de paquetes (X.25 y Frame Relay), la tecnología ATM está Orientada a Conexión. Esto significa que antes de que el usuario pueda enviar celdas a la red, es necesario realizar una llamada y que esta sea aceptada para establecer una Conexión Virtual a través de la red. Durante la fase de llamada un Identificador de Conexión Virtual (VCI) es asignado a la llamada en cada nodo de intercambio a lo largo de la ruta.

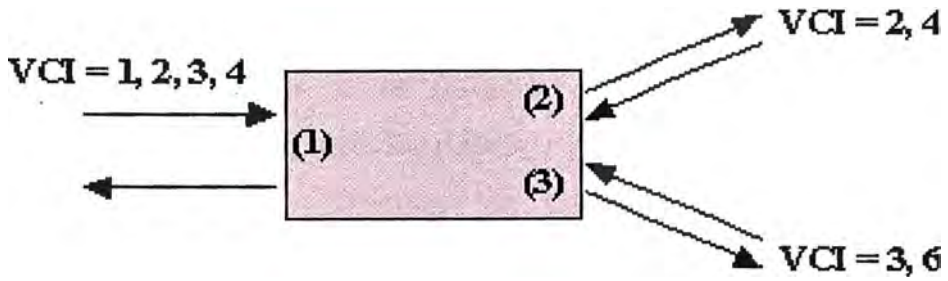


Figura 2.5 Conmutador ATM

2.4 Resolución de direcciones en ATM

La resolución de direcciones en una subred lógica IP de ATM la hace el ATMARP (ATM Address Resolution Protocol) basado en el RFC 826 y en el InATMARP(Inverse ATM Address Resolution Protocol) basado en el RFC 1293. ATMARP es el mismo protocolo que ARP pero con las extensiones necesarias para que ARP funcione en el entorno de servidor unicast de ATM. InATMARP es el mismo protocolo que el InARP original, pero aplicado a redes ATM. El uso de estos protocolos difiere en si se utilizan o no PVCs o SVCs.

Tanto ATMARP como InATMARP están definidos en el RFC 1577, que es una propuesta de estándar con estado electivo

El protocolo ARP se usa para calcular la dirección hardware de un host a partir de su dirección IP. El protocolo InATMARP se usa para calcular la dirección IP de un host a partir de su dirección hardware. En un entorno conmutado, primero se establece una VC("Virtual Connection") o una PVC("Permanent Virtual Connection") o una SVC("Switched Virtual Connection") para comunicar con otra estación. Por lo tanto, se sabe la dirección hardware del otro host, pero no la dirección IP. InATMARP proporciona resolución dinámica de direcciones. Utiliza el mismo formato de trama que el ARP estándar , pero define dos nuevos códigos de operación: InARP request=8

nARP reply=9

El InATMARP básico opera esencialmente del mismo modo que ARP, con la excepción de que no hace las peticiones con broadcasts. Esto se debe a que la dirección hardware ya se conoce. Una estación solicitante simplemente formatea

una petición insertando sus direcciones hardware de IP(fuente) y la dirección hardware del destino. Luego rellena con ceros el campo de dirección IP del destino y envía el mensaje a la estación de destino. Para cada petición InATMARP, la estación receptora formatea una respuesta utilizando la dirección fuente de la petición como dirección de destino para la respuesta. Ambos extremos actualizan sus tablas ARP. El valor *tipo de hardware* para ATM es el 19 decimal y el campo *EtherType* se pone a 0x806, que indica ARP según el RFC 1700

2.5 Resolución de direcciónes en el entorno PVC

En un entorno PVC cada estación utiliza el protocolo InATMARP para determinar las direcciones IP de todas las demás estaciones conectadas. La resolución se hace para aquellos PVCs configurados para la encapsulación LLC/SNAP. Es responsabilidad de cada estación IP que soporte PVCs la revalidación de las entradas de la tabla ARP a medida que pasa el tiempo.

2.6 Resolución de direcciones en el entorno SVC

SVCs requiere soporte para ATMARP en el entorno no-broadcast de ATM. Para hacer frente a esta necesidad, se debe localizar un único servidor ATMARP dentro de la LIS("Logical IP Subnetwork";

Este servidor tiene la responsabilidad de resolver las peticiones ATMARP de todos los miembros IP del LIS. Para una explicación de los términos ATM,

El servidor en sí mismo no establece conexiones de modo activo. El inicio del proceso de registro ATMARP depende el cliente del LIS. Un cliente individual se conecta al servidor ATMARP con una conexión punto a punto VC. El servidor, al completarse la conexión ATM de un nuevo VC con encapsulación LLC/SNAP, transmitirá un petición InATMARP para determinar la dirección IP del cliente. La respuesta InATMARP del cliente contiene la información necesaria para que el servidor construya su caché ATMARP. Esta tabla consiste en:

dirección IP, dirección ATM, Sello de tiempo("Timestamp"), VC asociado

Esta información se usa para generar respuestas a las peticiones ATMARP recibidas.

Nota: El servidor ATMARP requiere que cada cliente sea configurado administrativamente con la dirección ATM del servidor ATMARP.

Algoritmo de inserción/actualización de la tabla ARP:

- Si el servidor ATMARP recibe una nueva dirección IP en una respuesta InATMARP la dirección IP se añade a la tabla ATMARP.
- Si la dirección IP de la respuesta InATMARP duplica una dirección IP de una entrada de la tabla y la dirección InATMARP de ATM no coincide la dirección ATM de esa entrada en la tabla y existe un VC abierto asociado a esa entrada, la información InATMARP se desecha y no se hacen cambios en la tabla.
- Cuando el servidor recibe una petición ATMARP sobre un VC, en el que la dirección IP y ATM de la fuente coinciden con la asociación que ya existe en la tabla, y la dirección ATM coincide con la que está asociada al VC, el servidor actualiza el timeout de la entrada en su tabla para la fuente. Por ejemplo, si el cliente está enviando solicitudes ATMARP al servidor sobre el mismo VC usado para registrarse, el servidor se da cuenta de que ese cliente todavía existe y actualiza su timeout en la tabla.

Cuando el servidor recibe un ARP_REQUEST sobre un VC, examina la fuente de la información. Si no hay ninguna dirección IP asociada a ese VC y si la dirección IP de la fuente no está asociada a ninguna otra conexión, entonces el servidor añade esa estación a su tabla. Este no es el procedimiento normal ya que, como se indica arriba, es responsabilidad del cliente registrarse en el servidor ATMARP

Degeneración de la tabla ATMARP

Las entradas de la tabla ATMARP son válidas: En clientes por un máximo de 15 minutos En servidor por una mínimo de 20 minutos

Antes de invalidar una entrada de su tabla, el servidor ATPARP genera un InARP_REQUEST para cualquier VC abierto asociado con esa entrada y decide lo que ha de hacer de acuerdo con las siguientes reglas:

Si se recibe una respuesta InARP_REPLY, la entrada en la tabla se actualiza en vez de borrarse.

Si no hay ningún VC asociado a esa entrada, la entrada se borra.

Por tanto, si el cliente no mantiene un VC abierto al servidor, debe refrescar su información ATMARP en el servidor al menos cada 20 minutos. Esto se hace abriendo un VC al servidor intercambiando los paquetes InATMARP iniciales.

El cliente maneja las actualizaciones de la tabla con el siguiente criterio:

Cuando una entrada de la tabla degenera, el cliente la invalida, Si no hay un VC asociado a la entrada invalidada, se borra.

En el caso de una entrada invalidada con un VC abierto, el cliente ATMARP revalida la entrada para ese VC antes de enviar cualquier información que no tenga nada que ver con la resolución de direcciones. Hay dos posibilidades:

En el caso de un PVC, el cliente valida la entrada al transmitir un InARP_REQUEST y actualizar la entrada al recibir un InARP_REPLY.

En el caso de un SVC, el cliente valida la entrada al transmitir un ARP_REQUEST al servidor ATMARP y actualiza la entrada al recibir un ARP_REPLY.

Si un VC asociado con una entrada invalidada de la tabla ATMARP se cierra, la entrada se elimina.

Como se menciona anteriormente , cualquier cliente IP de ATM que use SVCs debe conocer la dirección de su servidor ATM para el LIS concreto. Esta dirección se le debe indicar a cada cliente durante la configuración.

CAPITULO III
PROTOCOLOS ATM

3.1 .-Introducción

Los protocolos ATM conforman tres niveles o capas básicas. La primera capa llamada capa física (Physical Layer), define los interfases físicos con los medios de transmisión y el protocolo de trama para la red ATM es responsable de la correcta transmisión y recepción de los bits en el medio físico apropiado. A diferencia de muchas tecnologías como Ethernet, que especifica ciertos medios de transmisión, (10 base T, 10 base 5, etc.) **ATM es independiente del transporte físico**. Las celdas ATM pueden ser transportadas en redes SONET (Synchronous Optical Network), SDH (Synchronous Digital Hierarchy), T3/E3, TI/EI o aún en modems de 9600 bps. Hay dos subcapas en la capa física que separan el medio físico de transmisión y la extracción de los datos:

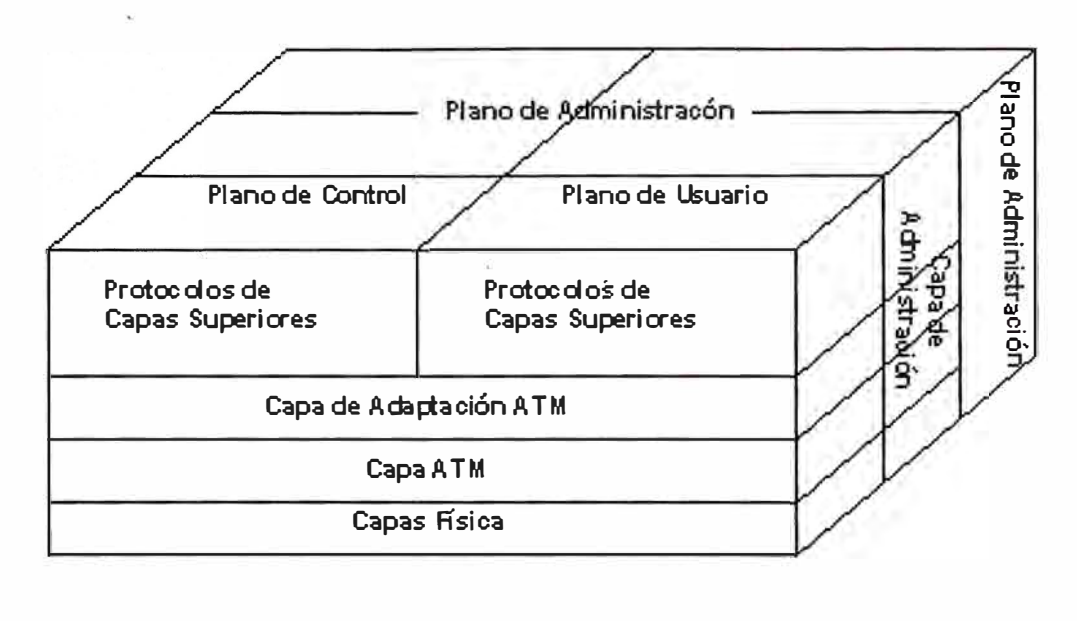


Figura 3.1 Protocolo de modelo de referencia para ATM Banda Ancha

La sub capa PMD (Physical Medium Depedent) tiene que ver con los detalles que se especifican para velocidades de transmisión, tipos de conectores físicos, extracción de reloj, etc.,

.**La subcapa TC** (Transmission Convergence) tiene que ver con la extracción de información contenida desde la misma capa física. Esto incluye la generación y el chequeo de la cabecera Error Corrección (HEC), extrayendo celdas desde el flujo de bits de entrada y el procesamiento de celdas "idles" y el reconocimiento del límite de la celda. Otra función importante es intercambiar información de operación y mantenimiento (OAM) con el plano de administración. La **segunda capa es la capa ATM. Ello define la estructura de la celda** y cómo las celdas fluyen sobre las conexiones lógicas en una red ATM**, esta capa es independiente del servicio.** El formato de una celda ATM es muy simple. Consiste de 5 bytes de cabecera y 48 bytes para información.

Las celdas son transmitidas serialmente y se propagan en estricta secuencia numérica a través de la red. El tamaño de la celda ha sido escogido como un

compromiso entre una larga celda, que es muy eficiente para transmitir largas tramas de datos y longitudes de celdas cortas que minimizan el retardo de procesamiento de extremo a extremo, que son buenas para voz, vídeo y protocolos sensibles al retardo. A pesar de que no se diseñó específicamente para eso, la longitud de la celda ATM acomoda convenientemente dos Fast Packets IPX de 24 bytes cada uno.

Los comités de estándares han definido dos tipos de cabeceras ATM: los User-to-Network Interface (UNI) y la Network to Network Interface (UNI). La UNI es un modo nativo de interfaz ATM que define la interfaz entre el equipo del cliente (Customer Premises Equipment), tal como hubs o routers ATM y la red de área ancha ATM (ATM WAN). La NNI define la interfase entre los nodos de la redes (los switches o conmutadores) o entre redes. La NNI puede usarse como una interfase entre una red ATM de un usuario privado y la red ATM de un proveedor público (carrier). Específicamente, la función principal de ambos tipos de cabeceras de UNI y la NNI, es identificar las "Virtual paths identifiers" (VPIS) y los "virtual circuits" o virtual channels"(VCIS) como identificadores para el ruteo y la conmutación de las celdas ATM

La tercera **capa es la ATM Adaptation Layer (AAL)**. La AAL juega un rol clave en el manejo de múltiples tipos de tráfico para usar la red ATM, y **es dependiente del servicio**. Especificamente, su trabajo es adaptar los servicios dados por la capa ATM a aquellos servicios que son requeridos por las capas más altas, tales como emulación de circuitos, (circuit emulation), vídeo, audio, frame relay, etc. La AAL recibe los datos de varias fuentes o aplicaciones y las convierte en los segmentos de 48 bytes. **Cinco tipos de servicios AAL están definidos** actualmente:

La capa de Adaptación de ATM yace entre la capa ATM y las capas más altas que usan el servicio ATM. Su propósito principal es resolver cualquier disparidad entre un servicio requerido por el usuario y los servicios disponibles de la capa ATM . La capa de adaptación introduce la información en paquetes ATM y controla los errores de la transmisión. La información transportada por la capa de adaptación se divide en cuatro clases según las propiedades siguientes:

1. Que la información que esta siendo transportada dependa o no del tiempo.
2. Tasa de bit constante/variable.
3. Modo de conexión.

Estas propiedades definen ocho clases posibles, cuatro se definen como Clases de servicios B-ISDN. La capa de adaptación de ATM define 4 servicios para equiparar las 4 clases definidas por B-ISDN:.

- AAL-1
- AAL-2
- AAL-3
- AAL-4

La capa de adaptación se divide en dos subcapas:

1)**Sub capa de convergencia** (convergence sublayer (CS)) :

En esta capa se calculan los valores que debe llevar la cabecera y la carga útil del mensaje. La información en la cabecera y la carga útil depende de la clase de información que va a ser transportada.

2) **Sub capa de Segmentación y reensamblaje** (segmentation and reassembly (SAR))Esta capa recibe los datos de la capa de convergencia y los divide en trozos formando los paquetes de ATM. Agrega la cabecera que llevara la información necesaria para el reensamblaje en el destino.

La figura 3.2 aporta una mejor comprensión de ello la subcapa CS es dependiente del servicio y se encarga de recibir y paquetizar los datos proveniente de varias aplicaciones en tramas o paquete de datos longitud variable.

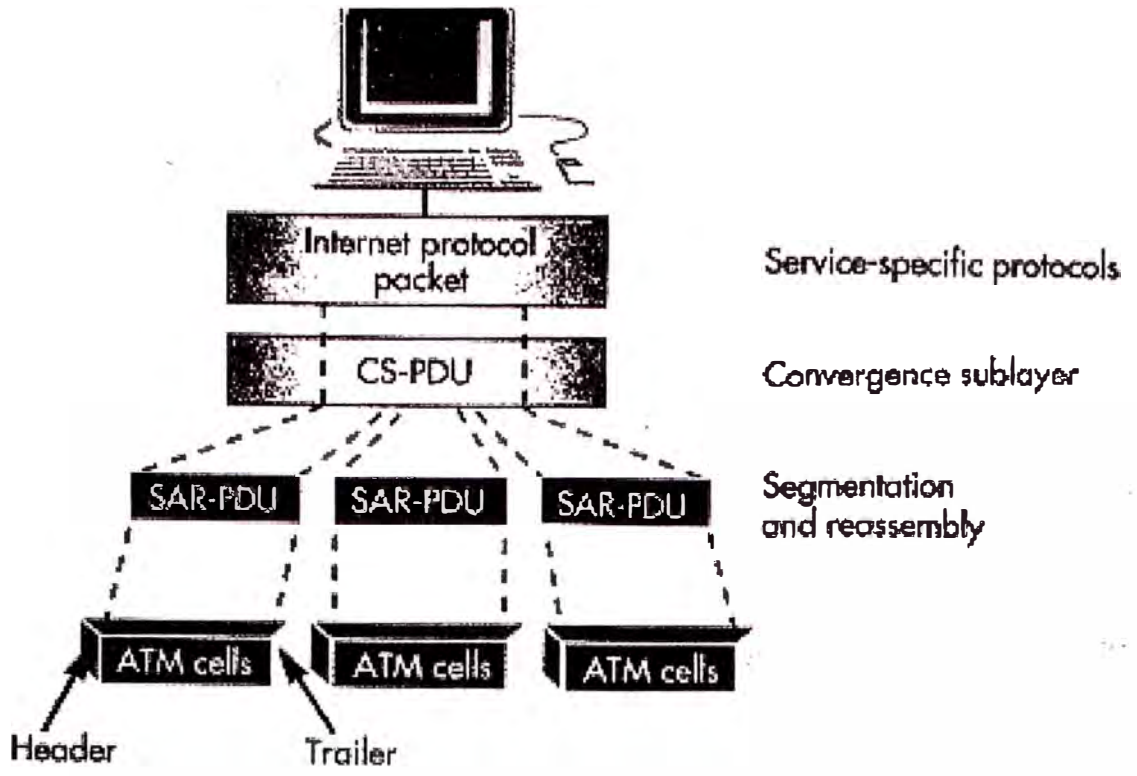


Figura 3.2 Distribución de tramas ATM

Estos paquetes son conocidos como (CS - PDU) CONVERGENCE SUBLAYER PROTOCOL DATA UNITS.

Luego, la sub capa recibe los SAR CS - PDU, los reparte en porciones del tamaño de la celda ATM para su transmisión. También realiza la función inversa (reensamblado) para las unidades de información de orden superior. Cada porción es ubicada en su propia unidad de protocolo de segmentación y reensamble conocida como (SAR PDU) SEGMENTATION AND REASSEMBLER PROTOCOL DATA UNIT, de 48 bytes.

Finalmente cada SAR - PDU se ubica en el caudal de celdas ATM con su cabecera y ensamblado respectivo.

AAL1:

AAL-1 se usa para transferir tasas de bits constantes que dependen del tiempo. Debe enviar por lo tanto información que regule el tiempo con los

datos. AAL-1 provee recuperación de errores e indica la información con errores que no podrá ser recuperada.

Sub capa de convergencia:

Las funciones provistas a esta sub capa difieren dependiendo del servicio que se proveyó. Provee la corrección de errores.

Sub capa de segmentación y reensamblaje:

En esta sub capa los datos son segmentados y se les añade una cabecera. La cabecera contiene 3 campos.

- Número de secuencia usado para detectar una inserción o perdida de un paquete.
- Número de protección usado para corregir errores que ocurren en el numero de secuencia.
- Indicador de capa de convergencia usado para indicar la presencia de la función de la capa de convergencia.

ALL 2:

AAL-2 se usa para transferir datos con tasa de bits variable que dependen del tiempo. Envía la información del tiempo conjuntamente con los datos para que esta puede recuperarse en el destino. AAL-2 provee recuperación de errores e indica la información que no puede recuperarse.

Sub capa de convergencia:

Esta sub capa provee para la corrección de errores y transporta la información del tiempo desde el origen al destino.

Sub capa de segmentación y recuperación:

El mensaje es segmentado y se le añade una cabecera a cada paquete. La cabecera contiene dos campos.

- Numero de secuencia que se usa para detectar paquetes introducidos o perdidos.
- El tipo de información es:
  - BOM, inicio de mensaje
  - COM, continuación de mensaje
  - EOM, fin de mensaje o indica que el paquete contiene información de tiempo u otra.

La carga útil también contiene dos campos :

- indicador de longitud que indica el numero de bytes validos en un paquete parcialmente lleno.
- CRC que es para hacer el control de errores.

AAL 3:

AAL-3 se diseña para transferir los datos con tasa de bits variable que son independientes del tiempo. AAL-3 puede ser dividido en dos modos de operación:

1. Fiable: En caso de perdida o mala recepción de datos estos vuelven a ser enviados. El control de flujo es soportado.
2. No fiable: La recuperación del error es dejado para capas mas altas y el control de flujo es opcional.

Sub capa de convergencia:

La sub capa de convergencia en AAL 3 es parecida al ALL 2. Esta subdividida en dos secciones:

1. Parte común de la sub capa de convergencia. Esto es provisto también por el AAL-2 CS.
2. Añade una cabecera y una carga útil a la parte común

La cabecera contiene 3 campos:

- Indicador de la parte común que dice que la carga útil forma parte de la parte común.
- Etiqueta de comienzo que indica el comienzo de la parte común de la capa de convergencia.
- Tamaño del buffer que dice al receptor el espacio necesario para acomodar el mensaje.

La carga útil también contiene 3 campos:

- Alineación es un byte de relleno usado para hacer que la cabecera y la carga útil tengan la misma longitud.
- Fin de etiqueta que indica el fin de la parte común de la CS (capa de convergencia).
- El campo de longitud tiene la longitud de la parte común de la CS.

Parte especifica del servicio. Las funciones provistas en ésta (que capa depende de los servicios pedidos). Generalmente se incluyen funciones para la recuperación y detección de errores y puede incluir también funciones especiales.

Sub capa de segmentación y reensamblaje

En esta sub capa los datos son partidos en paquetes de ATM. Una cabecera y la carga útil que contiene la información necesaria para la recuperación de errores y reensamblaje se añaden al paquete. La cabecera contiene 3 campos:

1) Tipo de segmento que indica que parte de un mensaje contiene carga útil. Tiene uno de los siguientes valores:

- BOM: Comenzando de mensaje
- COM: Continuación de mensaje
- EOM: Fin de mensaje
- SSM: Mensaje único en el segmento

2) Numero de secuencia usado para detectar una inserción o una perdida de un paquete.

3) identificador de multiplexaje. Este campo se usa para distinguir datos de diferentes comunicaciones que ha sido multiplexadas en una única conexión de ATM.

La carga útil contiene dos campos:

1) Indicado de longitud que muestra el número de bytes útiles en un paquete parcialmente lleno.

2) CRC es para el control de errores.

ALL 4:

AAL-4 se diseña para transportar datos con tasa de bits variable independientes del tiempo. Es similar al AAL3 y también puede operar en transmisión fiable y no fiable. AAL-4 provee la capacidad de transferir datos fuera de una conexión explícita.

AAL 2, AAL 3/4 y AAL 5 manejan varios tipos de servicios de datos sobre la base de tasas de bits variables tales como Switched Multimegabit Data Service (SMDS), Frame Relay o tráfico de redes de área local (Lan). AAL 2 y AAL 3 soportan paquetes orientados a la conexión.

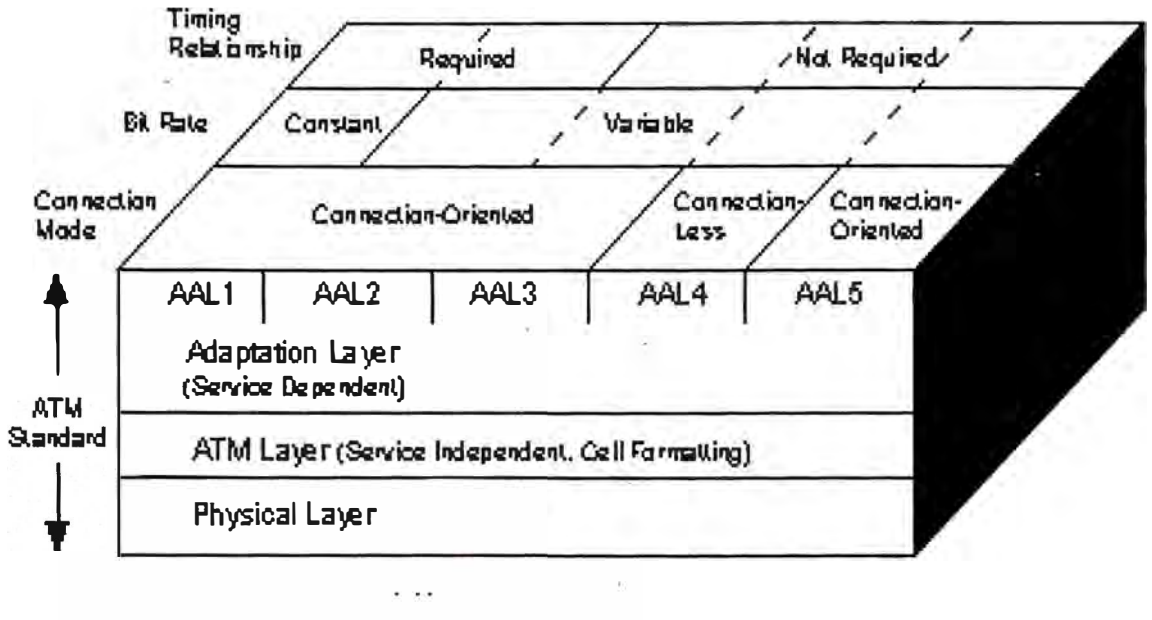


Figura 3.3 Distribución de capas ATM.

(El término orientado a la conexión describe la transferencia de datos después del establecimiento de un circuito virtual).

3.2 Protocolos Nativos ATM.

Son los protocolos específicamente diseñados para usar la tecnología ATM para soporte de Q'S (calidad de servicio) señalización direccionamiento etc. En aplicaciones nativas ATM tales como voz y datos, pizarras compartidas, video conferencias. Los protocolos nativos ATM definen servicios *ATM* específicos disponibles para el software y hardware residentes en dispositivos de usuario *UNI ATM.*

Servicios entre los que se pueden destacar los siguientes:

- Transferencias de datos (fiables o no) usando la capa *ATM* y varias capas de adaptación (*AALs*).
- Disponibilidad de circuitos virtuales conmutados (*SVCs*) y circuitos virtuales permanentes (*PVCs*).

- Consideraciones relativas a la gestión de tráfico (clases de servicio, garantías de *QoS*, etc.).
- Posibilidad de distribución de conexiones y de participación local en la administración de la red **(protocolos *ILMI* y *OAM*).**

Estos servicios nativos también ofrecen soporte a un amplio y heterogéneo rango de flujos con diversas propiedades y requerimientos recomendados.

Los protocolos de transferencia nativos *ATM* gestionan la señalización *UNI* para establecer los *SVCs*, configurar *PVCs* y mapear los perfiles de *QoS* en la correspondiente clase de servicio. Los protocolos nativos también realizan funciones clásicas como las de transporte, mecanismos de control de errores, transferencia de datos, y controles de flujo y de congestión. Las redes *ATM* actuales que usan ***TCP*** **como capa de transporte e *IP-over-ATM* como capa de red**

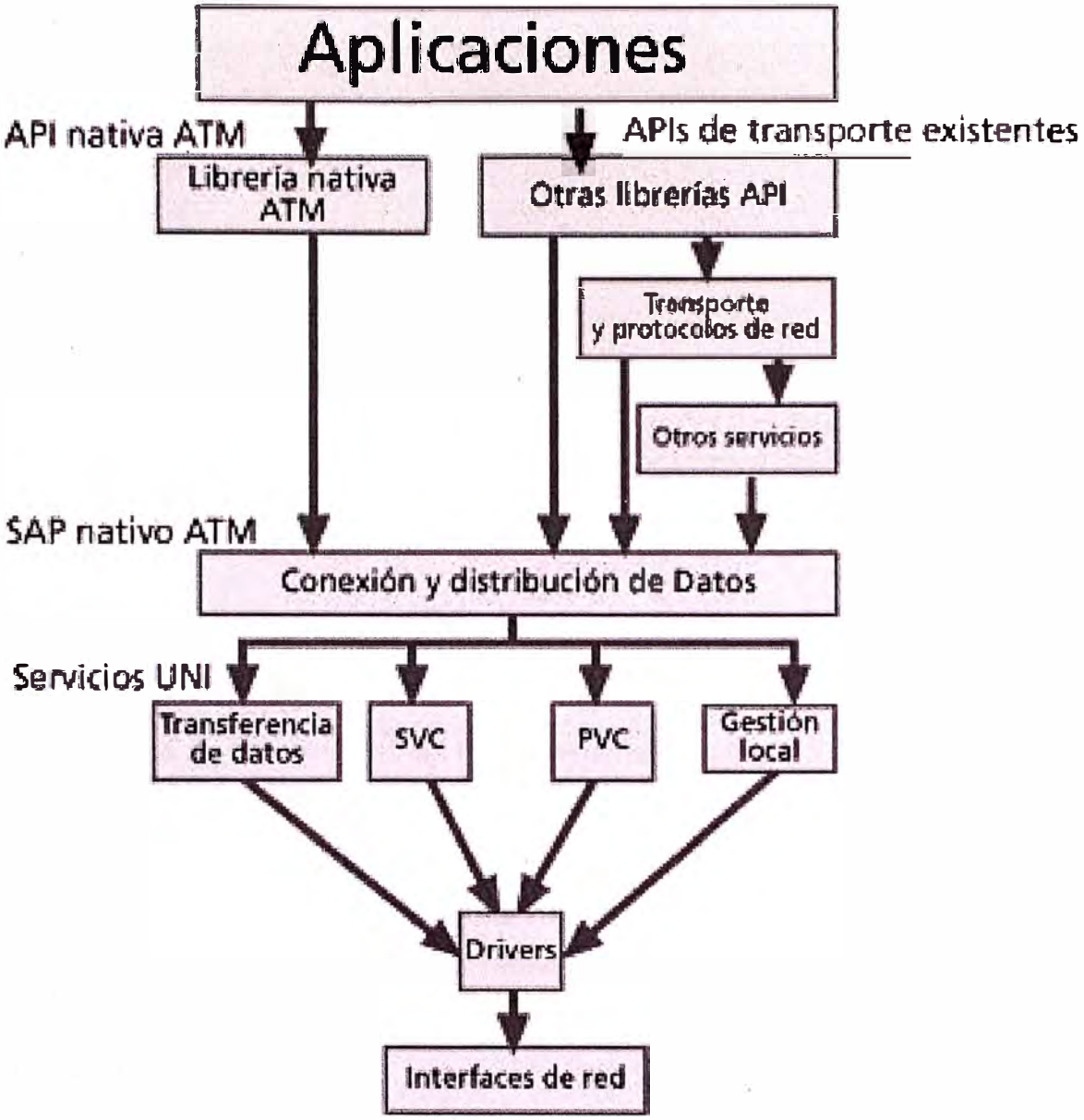


Figura 3.4 Distribución de la información en ATM

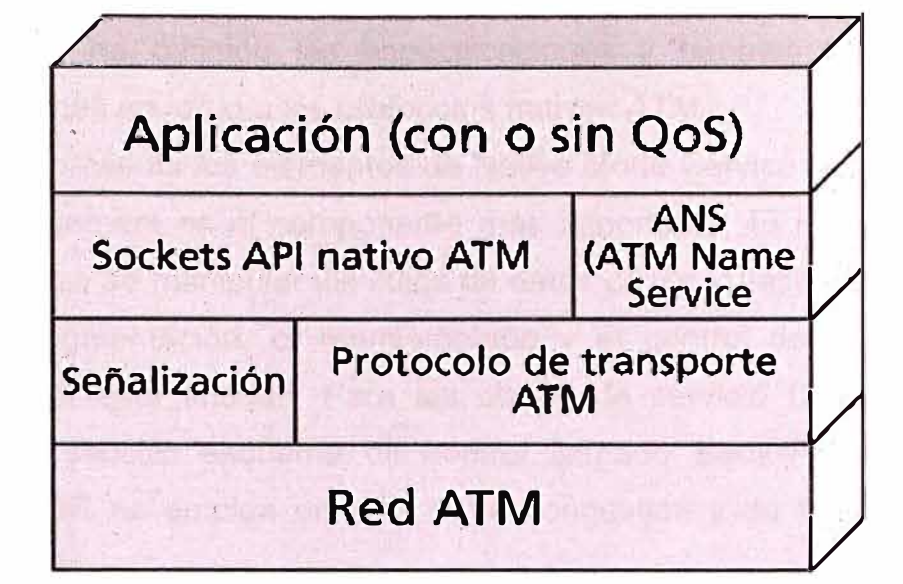


Figura 3.5 Distribución Jerarquica en ATM

Actualmente se forman pilas de Protocolos nativos por las siguientes razones:

- Existen muchas aplicaciones pensadas para explotar avanzados servicios usando tecnología *ATM* y también existen antiguas y no nativas aplicaciones. Este escenario implica cambiar las aplicaciones o proponer nuevas pilas de protocolos nativos *ATM*.

El encapsulamiento consecutivo de paquetes genera problemas de *overhead* y funciones redundantes.

La limitación de recursos en los sistemas finales es otra importante motivación para usar pilas de protocolos nativos y ligeros.

La *QoS* ofrecida por el modo nativo es aprovechada por los usuarios para demandar recursos a los proveedores de servicios en redes privadas. Los proveedores de servicios públicos disfrutan también de estas ventajas.

ATM, *RDSI* y la telefonía ofrecen un esquema de direccionamiento universal basado en ***NSAP/E.164*** el cuál es capaz de enrutar tráfico de forma nativa. Por tanto, aunque *ATM* dispone de protocolos nativos con direccionamiento intrínseco estructurado y jerárquico, éste no es aprovechado por las aplicaciones que están basadas en *IP*. El esquema de direccionamiento *ATM* es una de las principales dificultades en los protocolos propuestos como nativos.

Compatibilidad con aplicaciones tradicionales sin necesidad de recopilaciones.

ATM *Forum* ha definido las especificaciones y también existen importantes investigaciones en torno a los protocolos nativos *ATM*.

La **Figura** presenta los elementos de Native Mode Service Architecture donde el Flow Management es el componente más importante. El Flow Management se responsabiliza de manipular los flujos de datos desde y hasta la red vía la interfaz AAL. La segmentación, el reensamblado y el control de errores es también realizada por esta entidad. Para las clases de servicio CBR, VBR y ABR se emplea un sencillo esquema de control llamado **Back-Pressure Flow**. Para servicios UBR se emplea un control de congestión y de flujo extremo-extremo más complejo.

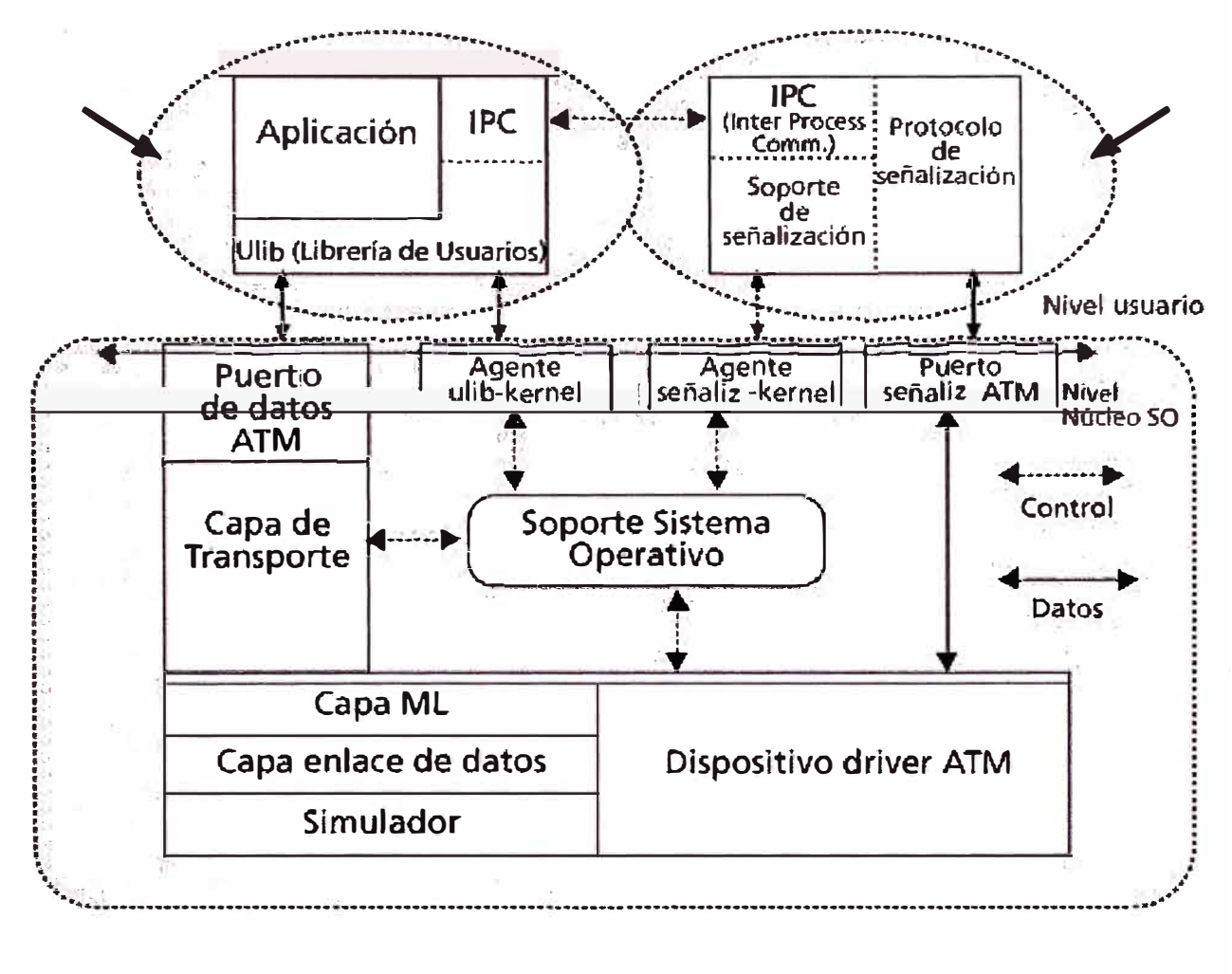


Figura 3.6 Entidad de transporte

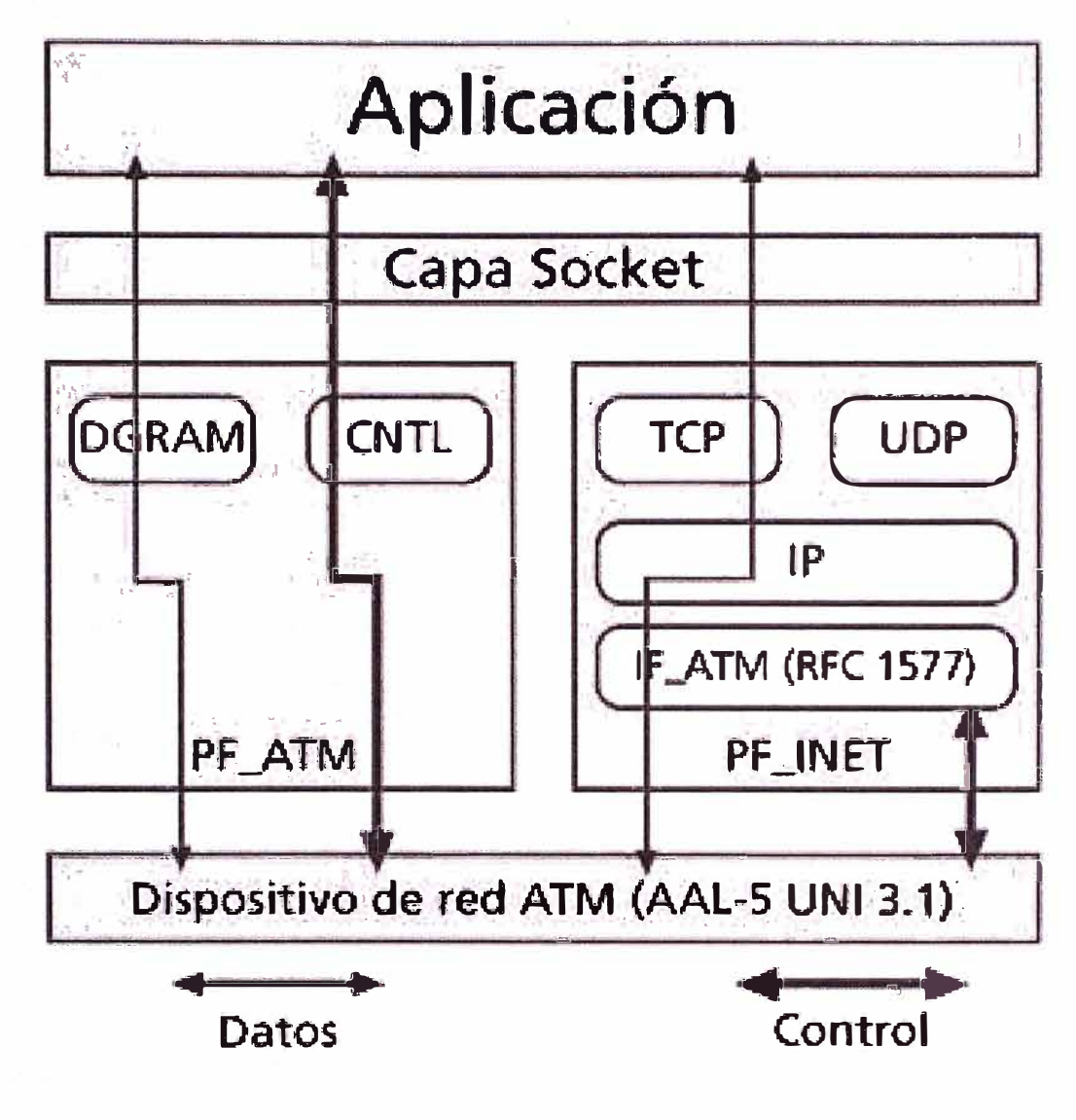


Figura 3.7 Distribución de información ATM y TCP/IP.

La idea es mezclar el soporte nativo ATM en la estructura existente del protocolo que muestra dos caminos separados en el protocolo: la familia nativa ATM y la familia del protocolo IP. Las aplicaciones que tienen acceso transparente a la red ATM usan la familia del protocolo **PF_INET**. El mapeo de IP en ATM es gestionado por la interfaz de red ATM **(IF_ATM)** usando el **protocolo IP-over-ATM**.

La interfaz Native ATM es constituida por la familia de **protocolos PF_ATM** que es directamente soportada encima del dispositivo de red ATM sobrepasando la capa interfaz de red. El **módulo CNTL** abre una conexión de señalización con el dispositivo ATM y establece una gestión de las llamadas de mensajes de configuración.

PF_ATM separa flujos de datos y de control para aliviar el límite de comportamiento en las comunicaciones. Esto permite a los mecanismos de control de tráfico ser rápidos y sencillos, mientras los mecanismos de control pueden ser tan complicados como sea necesario. Esta separación permite también que los dispositivos puedan estar en los puntos finales de una conexión.
La interfaz **PF_ATM** da a las aplicaciones acceso directo a la capa de enlace ATM y extiende las garantías de QoS a los puntos extremos de la comunicación.

3.3 Protocolos de Aplicación ATM.

ATM es una tecnología que trabaja a nivel de las capas 1,2,y 3 del modelo OSI, en este caso los protocolos empleados en este nivel se refieren a los protocolos utilizados por la arquitectura TCP/IP por ejemplo la posibilidad de poder comunicarse con servidores de correo, se debe crear un valor añadido y una interfaz amigable que pueda comunicarse con los servicios suministrados por el sistema operativo
Los niveles superiores son los más cercanos al usuario y tratan con datos más abstractos, dejando a los niveles más bajos la labor de traducir los datos de forma que sean físicamente manipulables.
.HTTP, SMTP, SNMP, FTP, Telnet, SSH y SCP, NFS, RTSP, Feed, Webcal
(Aplicación, Presentación y Sesión) son considerados simplemente como el nivel de aplicación en el conjunto TCP/IP
HTTP, FTP, DNS
(protocolos de enrutamiento como BGP y RIP, que por varias razones funcionen sobre TCP y UDP respectivamente, son considerados parte del nivel de red)
El nivel de aplicación es el nivel que los programas más comunes utilizan para comunicarse a través de una red con otros programas. Los procesos que acontecen en este nivel son aplicaciones específicas que pasan los datos al nivel de aplicación en el formato que internamente use el programa y es codificado de acuerdo con un protocolo estándar.
Algunos programas específicos se considera que se ejecutan en este nivel. Proporcionan servicios que directamente trabajan con las aplicaciones de usuario. Estos programas y sus correspondientes protocolos incluyen a HTTP (*World Wide*

Web), FTP (Transferencia de archivos), SMTP (correo electrónico), SSH (login remoto seguro), DNS (Resolución de nombres de dominio) y a muchos otros.
Una vez que los datos de la aplicación han sido codificados en un protocolo estándar del nivel de aplicación son pasados 'hacia abajo *al siguiente nivel de la pila de protocolos TCP/IP.*
En el nivel de transporte, las aplicaciones normalmente hacen uso de TCP y UDP, y son habitualmente asociados a un número de puerto bien conocido (*well-known port*). Los puertos fueron asignados originalmente por la IANA.
El nivel de Aplicación es a un proceso de usuario que coopera con otro proceso en el mismo o en otro host. Ejemplos son TELNET (un protocolo para la conexión remota de terminales), FTP ("File Transfer Protocol") y SMTP ("Simple Mail Transfer Protocol"
Protocolos en el nivel de aplicación:
BitTorrent, FTP, HTTP, HTTPS, IMAP, ICQ, IRC, NFS, NNTP, POP3, RTP, SIP, SMB/CIFS, SMTP, SNMP, SSH, SSL, Telnet, UUCP,

3.4 Protocolos de Transporte ATM.

Los protocolos de transporte de elevada velocidad son fuente de activas investigaciones y están en constante evolución desde hace más de dos décadas, *TCP* e *ISO TP4* son los dos más populares protocolos de transporte
Se presenta varios protocolos de transporte y arquitecturas para redes de alta velocidad
Se ofrece una excelente y didáctica arquitectura de pila de protocolos para aplicaciones multimedia en modo nativo *ATM.*
TCP

"transporte" se refiere al uso de técnicas de conmutación y multiplexación, en la capa de enlace (Capa 2 del modelo OSI) para el traspaso del tráfico del usuario final de la fuente al destino, dentro de una red. El ATM Forum, grupo de fabricantes y usuarios dedicado al análisis y avances de ATM, ha aprobado cuatro velocidades UNI (User Network Interfases) para ATM: DS3 (44.736 Mbit/s), SONET STS3c (155.52 Mbit/s) y 100 Mbit/s para UNI privados y 155 Mbit/s para UNI privadas. UNI privadas se refieren a la interconexión de usuarios ATM con un switch ATM privado que es manejado como parte de la misma red corporativa.

Aunque la tasa de datos original para ATM fue de 45 Mbit/s especificado para redes de operadores (carriers) con redes T3 existentes, velocidades UNI adicionales se han venido evaluando y están ofreciéndose. También hay un alto interés en interfases, para velocidades EI (2Mbps) y T1 (1,544 Mbps) para accesos ATM de baja velocidad.

DCCP, SCTP, SPX, TCP, UDP,

pueden solucionar problemas como la fiabilidad ("¿alcanzan los datos su destino?") y la seguridad de que los datos llegan en el orden correcto. En el conjunto de protocolos TCP/IP, los protocolos de transporte también determinan a que aplicación van destinados los datos.

TCP realiza continuamente medidas sobre el estado de la red para evitar sobrecargarla con demasiado tráfico. Además, TCP trata de enviar todos los datos correctamente en la secuencia especificada. Esta es una de las principales diferencias con UDP, y puede convertirse en una desventaja en flujos en tiempo real (muy sensibles a la variación del retardo) o aplicaciones de enrutamiento con porcentajes altos de pérdida en el nivel de interred.

Un mecanismo reciente es SCTP, fiable y orientado a conexión. Está relacionado con la orientación a byte, y proporciona multiples sub-flujos multiplexados sobre la misma conexión. También proporciona soporte de *multihoming*, donde una conexión puede ser representada por múltiples direcciones IP (representando múltiples interfaces físicas), así si una falla la conexión no se interrumpe. Fue desarrollado inicialmente para aplicaciones telefónicas (para transportar SS7 sobre IP), pero también fue usado para otras aplicaciones.
UDP (protocolo IP número 17) es un protocolo de datagramas sin conexión. Es un protocolo no fiable (*best effort* al igual que IP) - no porque sea particularmente malo, sino porque no verifica que los paquetes llegen a su destino, y no da garantías de que lleguen en orden. Si una aplicación requiere estas características, debe llevarlas a cabo por sí misma o usar TCP.
UDP es usado normalmente para aplicaciones de streaming (audio, video, etc) donde la llegada a tiempo de los paquetes es más importante que la fiabilidad, o

para aplicaciones simples de tipo petición/respuesta como el servicio DNS, donde la sobrecarga de las cabeceras que aportan la fiabilidad es desproporcionada para el tamaño de los paquetes.

DCCP está actualmente bajo desarrollo por el IETF. Proporciona semántica de control para flujos TCP, mientras de cara al usuario se da un servicio de datagramas UDP.

TCP y UDP son usados para dar servicio a una serie de aplicaciones de alto nivel. Las aplicaciones con una dirección de red dada son distinguibles entre sí por su número de puerto TCP o UDP. Por convención, los puertos bien conocidos (*well-known ports*) son asociados con aplicaciones específicas

RTP es un protocolo de datagramas que ha sido diseñado para datos en tiempo real como el streaming de audio y video que se monta sobre UDP.

Puertos y Zócalos

Cada proceso que se desea comunicar con otro se identifica en la pila de protocolos TCP/IP con uno o más puertos. Un puerto es un número de 16 bits, empleado por un protocolo host – a – host para identificar a que protocolo del nivel superior o programa de aplicación se deben entregar los mensajes recibidos. Como algunas aplicaciones son ya de por sí protocolos estandarizados, como TELNET y FTP, emplean el mismo número de puerto en todas las implementaciones TCP/IP. Estos puertos "asignados" se conocen como *puertos bien conocidos*, y a sus aplicaciones, aplicaciones bien conocidas.

Estos puertos son controlados y asignados por IANA ("Internet Assigned Numbers Authority") y en la mayoría de los sistemas sólo los puede utilizar los procesos del sistema o los programas que ejecutan usuarios privilegiados. Ocupan número de puerto comprendidos en el rango de 0 a 1023. Los puertos con números en el rango de 1024 a 65535 no los controla IANA y en la mayor parte de los sistemas los pueden usar los programas de usuario.

La confusión que se produce cuando dos aplicaciones distintas intentan usar los mismos puertos en un host se evita haciendo que soliciten un puerto disponible a TCP/IP. Como este número se asigna dinámicamente, puede ser diferente en cada ejecución de una misma aplicación.

UDP, TCP y ISO TP-4 están todos basados en el mismo principio de uso de los puertos En la medida de lo posible, se usan los mismos números para los servicios situados sobre UDP, TCP y ISO TP-4.
Un *zócalo* es un tipo especial de *descriptor de fichero* que un proceso usa para solicitar servicios de red al sistema operativo.
Una dirección de zócalos:
{protocolo, dirección local, proceso local }
En la pila TCP/IP, por ejemplo:

{*tcp, 193.44.234.3, 12345*

Una *conversación* es el enlace de comunicaciones entre dos procesos.

Una *asociación* es la quíntupla que especifica completamente los dos procesos comprendidos en una conexión:

{protocolo, dirección local, proceso - local, dirección exterior, proceso exterior}

En la pila TCP/IP, por ejemplo:
{tcp, 193.44.234.3, 1500, 193.44.234.5, 21}
podría ser una asociación válida.

Una medio – asociación es:

{protocolo, dirección - local, proceso local }
o
{protocolo, dirección exterior, proceso exterior}

que especifican cada una de las mitades de la conexión

La *medio – asociación* se denomina también zócalo o *dirección de transporte*. Es decir, un zócalo es un punto terminal para la comunicación que puede ser nombrado y direccionado en una red.

La interfaz del zócalo es una de tantas APIs con los protocolos de comunicación. Se introdujo por primera vez en el UNIX BSDD 4.2. Aunque no ha sido estandarizada, se ha convertido en un estándar.

A continuación se muestran algunas llamadas básicas de la interfaz de zócalos.

- Inicializar un zócalo

FORMAT: int *sockfd* = **socket**(int *family*, int *type*, int *protocol*)
donde:

  - *family* es la *familia de direccionamiento*. Puede tomar valores como AF_UNIX, AF_INET, AF_NS y AF_IUCV. Su fin es especificar el método de direccionamiento que usa el zócalo.
  - *type* es el tipo de interfaz de zócalo a usar. Puede tomar valores como SOCK_STREAM, SOCK_DGRAM, SOCK_RAW, y SOCK_SEQPACKET.
  - *protocol* puede ser UDP, TCP, IP o ICMP.
  - *sockfd* es un entero (similar a un descriptor de fichero) devuelto por la llamada a **socket**.

- Registrar un zócalo en una dirección de puerto

  FORMAT: int **bind**(int *sockfd*, struct sockaddr **localaddr*, int *addrlen*)
  donde:

  - *sockfd* wa wl mismo entero que devuelve la llamada a **socket**.
  - *localaddr* es la dirección local que devuelve la llamada a **bind**.

  Nótese que tras la llamada a **bind**, ya hay valores para los tres primeros parámetros de la asociación:
  {*protocol, local-address, local-process, foreign-address, foreign-process*}

- Indica disponibilidad para recibir conexiones

  FORMAT: int **listen**(int *sockfd*, int *queue-size*)
  donde:

  - *sockfd* es el mismo entero que devuelve la llamada a **socket**.

- *queue-size* indica el número de solicitudes de conexión que se pueden encolar en el sistema mientras el proceso local no ha llamado todavía a **accept**.

- Acepta una conexión

  FORMAT: int **accept**(int *sockfd*, struct sockaddr **foreign-address*, int *addrlen*)
  donde:

  - *sockfd* es el mismo entero que devuelve la llamada a **socket**.
  - *foreign-address* es la dirección del proceso cliente que devuelve la llamada a **accept**.

  Nótese que la llamada a **accept** la efectúa un proceso servidor más que un cliente. Si hay una solicitud de conexión encolada, **accept** toma la primera solicitud de la cola y crea otro zócalo con las mismas propiedades que *sockfd*; en otro caso, **accept** bloquea el llamador hasta que llega una solicitud de conexión.

- Solicita la conexión con el servidor

  FORMAT: int **connect**(int *sockfd*, struct sockaddr **foreign-address*, int *addrlen*)
  donde:

  - *sockfd* es el mismo entero que devuelve la llamada a **socket**.
  - *foreign-address* es la dirección del cliente que devuelve la llamada a **connect**.

Nótese que esta llamada la efectúa un proceso cliente más que uno servidor.
Envía o recibe datos
Las funciones **read**(), **readv**(*sockfd*, char **buffer*, int *addrlen*), **recv**(), **readfrom**(), **send**(*sockfd, msg, len, flags*), **write**() se pueden emplear para recibir y enviar datos en una asociación ya establecida.

Nótese que estas llamadas son similares a las llamadas estándar del sistema de E/S **read** y **write**.

- Cierra un zócalo

  FORMAT: int **close**(int *sockfd*)
  donde:

sockfd es el mismo entero que devuelve la llamada a **socket.**

Ejemplo

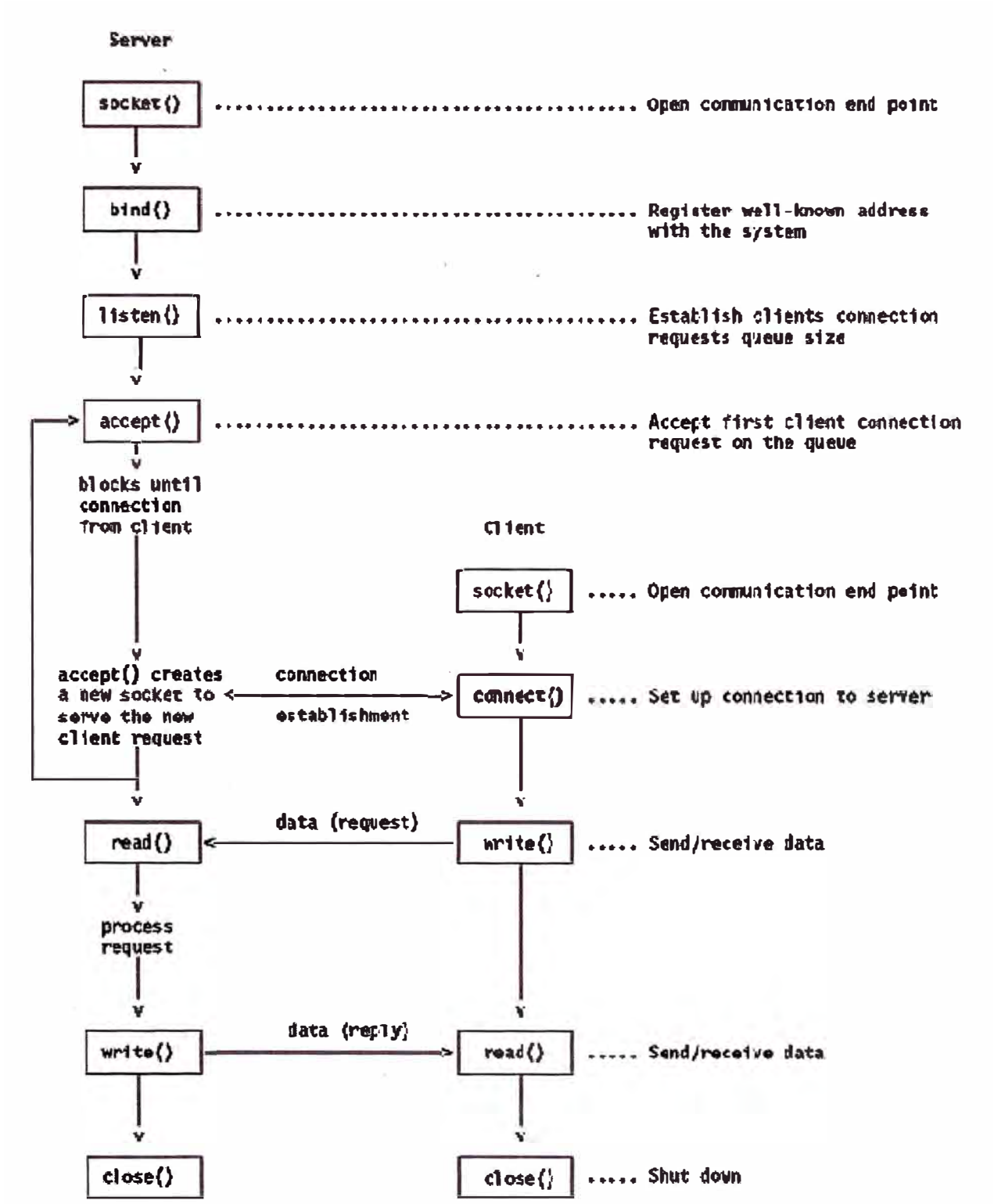


Figura 3.8 Diagrama de bloques de una comunicación.

Llamadas del sistema para zócalos para un protocolo orientado a conexión especificando los elementos de la asociación

	Protocol	Local Address , Local Process	Foreign Address , Foreign Process
connection-oriented server	socket()	bind()	listen() accept()
connection-oriented client	socket()	connect()	
connectionless server	socket()	bind()	recvfrom()
connectionless client	socket()	bind()	sendto()

Figura 3.9 Comunicación con diferentes tipos de conexión.

Llamadas del sistema para zócalos y asociación

La interfaz de zócalos se distingue por los diferentes servicios suministrados. Flujo, datagramas y zócalos a bajo nivel definen cada uno distintos servicios disponibles para las aplicaciones

- **Interfaz de zócalos orientada flujo** (SOCK_STREAM): Define una conexión fiable en un servicio orientado a conexión. Los datos se envían sin errores y sin duplicados y se reciben en el mismo orden en el que se envían. El control de flujo está integrado para evitar el desbordamiento de datos. No se imponen límites sobre los datos intercambiados, que se consideran un flujo de bytes. Un ejemplo de aplicación que usa esta interfaz es el FTP.
- **Interfaz de zócalos orientada a datagramas** (SOCK_DGRAM): Define un servicio no orientado a conexión (sobre UDP, por ejemplo). Los datagramas se envían como paquetes independientes. El servicio no proporciona garantías; los datos se pueden perder o duplicar, y los datagramas pueden llegar fuera de orden. No se realiza ningún tipo de ensamblaje o desensamblaje de los paquetes. Un ejemplo de aplicación que usa esta interfaz es el NFS ("Network File System").
- **Interfaz de zócalos a bajo nivel** (SOCK_RAW): Permite acceso directo a protocolos de bajo nivel tales como IP e ICMP. Esta interfaz suele usarse

para probar implementaciones de nuevos protocolos. Un ejemplo de aplicación que usa esta interfaz es el comando Ping.

UDP es un *protocolo estándar*, y su número STD es 6. El *RFC 768 - "User Datagram Protocol"* describe UDP. Su status es *recomendado*, pero en la práctica cualquier implementación de TCP/IP que no se use exclusivamente para el encaminamiento incluye UDP

Para IP, UDP es básicamente un interfaz de aplicación. No añade fiabilidad, control de flujo o recuperación de errores a IP. Simplemente sirve como "multiplexor/ demultiplexor" para enviar y recibir datagramas, usando *los puertos* para dirigir los datagramas

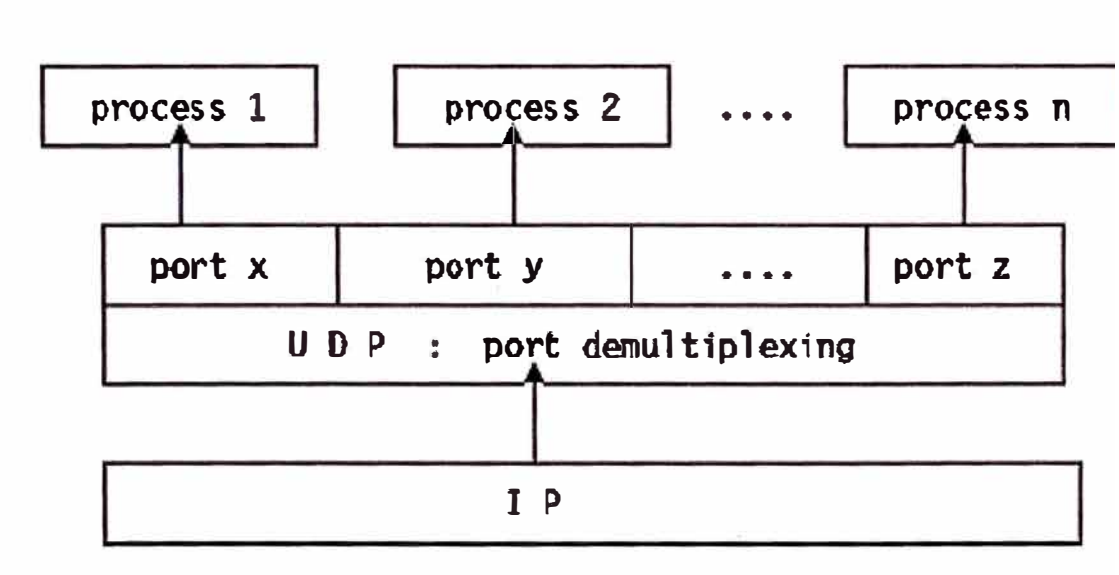


Figura 3.10 Comunicación sin acuse en TCP/IP.

UDP suministra un mecanismo para que una aplicación envíe un datagrama a otra. Se considera que la capa de UDP es extremadamente delgada y en consecuencia tiene poco "overhead", pero requiere que la aplicación se responsabilice de la recuperación de errores y todo lo que ello conlleva.

Las aplicaciones que envían datagramas a un host necesitan identificar un objetivo más específico que la dirección IP, ya que los datagramas suelen dirigirse a procesos concretos y no a todo el sistema. UDP permite hacer esto al hacer uso de los *puertos*.

Un puerto es un número de 16 bits que identifica en un host que proceso está asociado a un datagrama. Hay dos tipos de puerto:

Bien-conocidos ("well-known").

Los puertos bien-conocidos pertenecen a servidores estándar, por ejemplo Telnet usa el puerto 23. Los puertos bien-conocidos se hallan en el rango de 1 a 1023
Estos puertos suelen tener números impares, debido a que los primeros sistemas que usaron el concepto de puerto requerían para las operaciones en duplex una pareja par/impar de puertos. La mayoría de los servidores requieren sólo un único puerto. Una excepción es el servidor BOOTP que usa dos: el 67 y el 68
La razón de ser de los puertos bien-conocidos es permitir a los clientes encontrar a los servidores sin necesidad de información de configuración. Los números de los puertos bien-conocidos se definen en *STD 2 - Números asignados de Internet("Assigned Internet Numbers")*.
Los clientes no necesitan puertos bien-conocidos porque inician la comunicación con los servidores y los datagramas UDP enviados al servidor contienen su número de puerto. El host en funcionamiento proporciona un puerto a cada proceso cliente mientras este lo necesite. Los números de puertos efímeros tienen valores mayores de 1023, por lo general en el rango de 1024 a 5000. Un cliente puede usar cualquier número en ese rango, siempre que la combinación <protocolo de transporte, dirección IP, número de puerto> sea unívoca TCP también usa puertos con los mismos valores. Estos puertos son totalmente independientes de los de UDP. Normalmente, un servidor usará TCP o UDP, aunque hay excepciones. Por ejemplo, el DNS usa tanto el puerto 53 de UDP como el 53 de TCP
Cada datagrama UDP se envía en un sólo datagrama de IP. Aunque el datagrama IP se fragmente durante la transmisión, la implementación de IP que lo reciba lo reensamblará antes de pasárselo a la capa de UDP. Todas las implementaciones de IP deben aceptar datagramas de 576 bytes, lo que significa que si se supone un tamaño máximo de 60 bytes para la cabecera IP, queda un tamaño de 516 bytes para el datagrama UDP, aceptado por todas las implementaciones. Muchas implementaciones aceptan datagramas más grandes, pero no es algo que esté garantizado. El datagrama UDP tiene una cabecera de 16 bytes

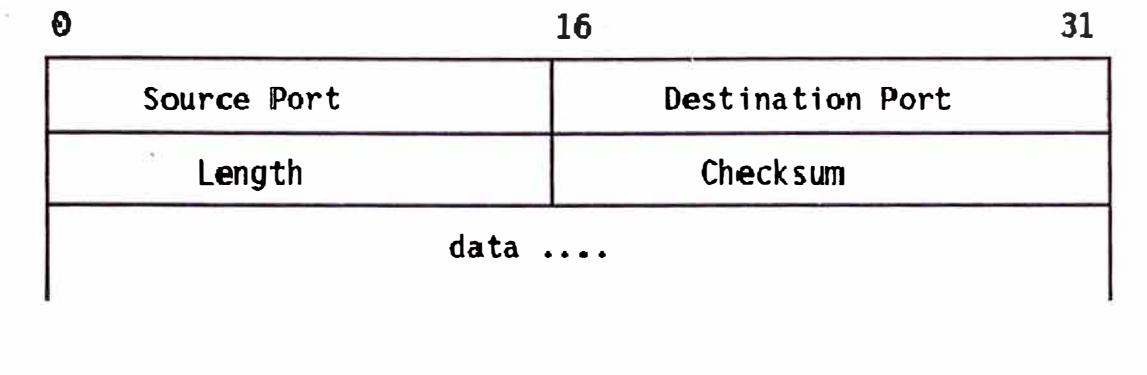


Figura 3.11 Control de comunicación.

Puerto origen:

Indica el puerto del proceso que envía el datagrama. Es el puerto al que se deberían dirigir las respuestas.

Puerto destino:

Especifica el puerto destino en el host de destino.

Longitud

Es la longitud (en bytes) del mismo datagrama de usuario, incluyendo la cabecera.

Checksum

Es un campo opcional consistente en el complemento a uno de 16 bits de la suma en complemento a uno de una pseudo cabecera IP, la cabecera UDP y los datos del datagrama UDP. La pseudo cabecera IP contiene las direcciones IP de origen y destino, el protocolo y la longitud del datagrama UDP:
La API para UDP se describe en el en RFC 768. Proporciona:
La creación de nuevos puertos para la recepción.
Operación de recepción que devuelve los bytes de datos recibidos y una indicación del puerto y la dirección IP de origen.
Operación de envío que tiene como parámetros los datos, los puertos de origen y destino y las direcciones IP.
La forma en que se implementa esto queda a elección del cada distribuidor.
Hay que ser consciente de que IP y UDP no proporcionan una entrega garantizada, control de flujo ni recuperación de errores, así que estos deberán ser implementados por la aplicación.
Aplicaciones estándar que usan UDP son:

- TFTP("Trivial File Transfer Protocol")

- DNS("Domain Name System")
- RPC("Remote Procedure Call"), usado por el NFS("Network File System")
- NCS("Network Computing System")
- SNMP("Simple Network Management Protocol")

TCP es un *protocolo estándar* con el STD 7. Se describe en el *RFC 793 - TCP("Transmission Control Protocol")*. Su status es *recomendado*, pero en la práctica cualquier implementación de TCP/IP que no se use exclusivamente para el encaminamiento incluirá TCP.

TCP proporciona una cantidad considerablemente mayor de servicios a las aplicaciones que UDP, notablemente, la recuperación de errores, control de flujo y fiabilidad. Se trata de un protocolo *orientado a conexión* a diferencia de UDP. La mayoría de los protocolo de aplicación de usuario, como TELNET y FTP, usan TCP.

Dos procesos se comunican a través de *zócalos TCP*. El modelo de zócalo proporciona a un proceso una conexión con un flujo full duplex de bytes con otro proceso. La aplicación no necesita preocuparse de la gestión de este canal; estos servicios son suministrados por TCP.

TCP usa el mismo principio de puerto que UDP para conseguir multiplexación. Al igual que UDP, TCP utiliza puertos efímeros y bien conocidos. Cada extremo de una conexión TCP tiene un *zócalo* que puede identificarse con la tripleta (TCP, dirección IP address, número de puerto). Es lo que se llama una *medio asociación.* Si dos procesos se están comunicando sobre TCP, tendrán una *conexión lógica* identificable unívocamente por medio de los dos zócalos implicados, es decir, con la combinación TCP, dirección IP local, puerto local, dirección IP remota, puerto remoto Los procesos del servidor son capaces de gestionar múltiples conversaciones a través de un único puerto.

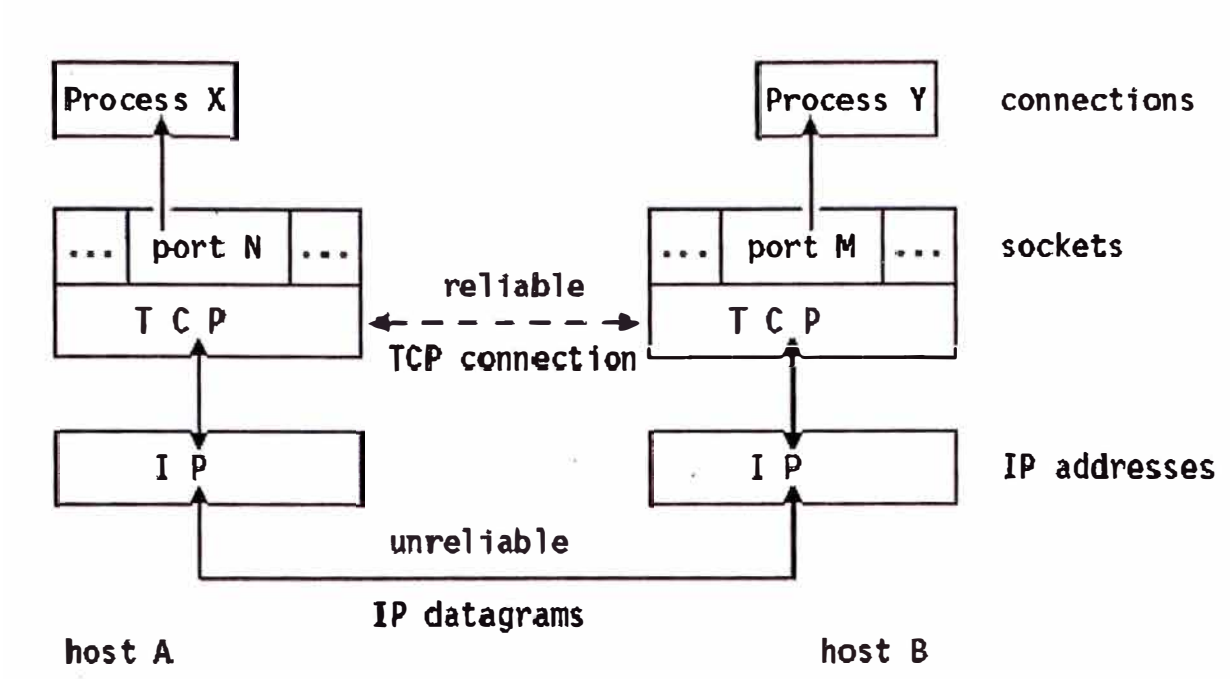


Figura 3.12 Comunicación entre Hosts

Conexión TCP - Los procesos X e Y se comunican sobre una conexión TCP que emplea datagramas IP.

El principal propósito de TCP es *proporcionar* una conexión lógica fiable entre parejas, *procesos. No* asume la fiabilidad de los protocolos de niveles inferiores(como IP) por lo que debe ocuparse de garantizarla.

TCP se puede caracterizar por los siguientes servicios que suministra a las aplicaciones que lo usan:

Transferencia de datos a través de un canal

Desde el punto de vista de la aplicación, TCP transfiere *un flujo continuo de bytes* a través de Internet. La aplicación no ha de preocuparse de trocear los datos en bloques o en datagramas. TCP se encarga de esto al agrupar los bytes en *segmentos TCP*, que se pasan a IP para ser retransmitidos al destino. Además, TCP decide por sí mismo cómo segmentar los datos y puede enviarlos del modo que más le convenga.

A veces, una aplicación necesita estar segura de que todos los datos pasados a TCP han sido transmitidos efectivamente al destino. Por esa razón, se define la función *"push"*. Esta función mandará todos los segmentos que sigan almacenados al host de destino. El *cierre normal de la conexión* también provoca que se llame a esta función, para evitar que la transmisión quede incompleta.

Fiabilidad

TCP asigna un número de secuencia a cada byte transmitido, y espera una reconocimiento afirmativo(ACK) del TCP receptor. Si el ACK no se recibe dentro de un intervalo de timeout, los datos se retransmiten. Como los datos se transmiten en bloques(segmentos de TCP), al host de destino sólo se le envía el número de secuencia del byte de cada segmento.

El TCP receptor utiliza los números de secuencia para organizar los segmentos cuando llegan fuera de orden, así como para eliminar segmentos duplicados.

Control de flujo

El TCP receptor, al enviar un ACK al emisor, indica también el número de bytes que puede recibir aún, sin que se produzca sobrecarga y desbordamiento de sus buffers internos. Este valor se envía en el ACK en la forma del número de secuencia más elevado que se puede recibir sin problemas. Este mecanismo se conoce también como mecanismo de *ventanas*

Multiplexación

Se consigue usando puertos, al igual que en UDP.

Conexiones lógicas

La fiabilidad y el control de flujo descritos requieren que TCP inicialice y mantenga cierta información de estado para cada canal. La combinación de este estado, incluyendo zócalos, números de secuencia y tamaños de ventanas, se denomina conexión lógica. Cada conexión se identifica unívocamente por el par de zócalos del emisor y el receptor.

Full Duplex

TCP garantiza la concurrencia de los flujos de datos en ambos sentidos e la conexión.

Un simple protocolo de transporte podría emplear el siguiente principio: enviar un paquete, y esperar un reconocimiento del receptor antes de enviar el siguiente. Si

el ACK no se recibe dentro de cierto límite de tiempo, se retransmite.

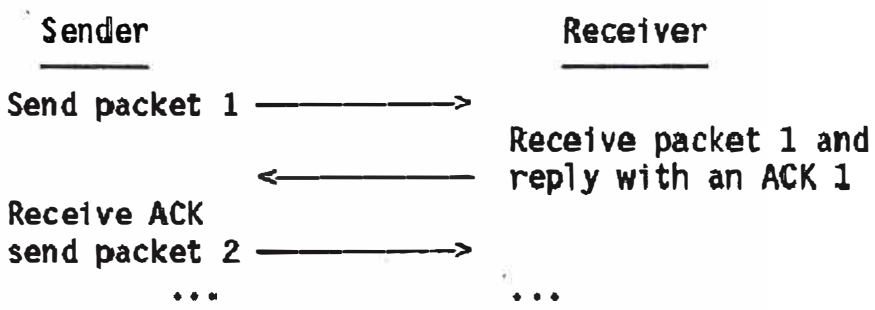


Figura 3.13 Comunicación con reconocimiento.

Aunque este mecanismo asegura fiabilidad, sólo usa una parte del *ancho de banda de la red* que está disponible.

Considerar ahora un protocolo en el que el emisor agrupa los paquetes que va a transmitir como se muestra

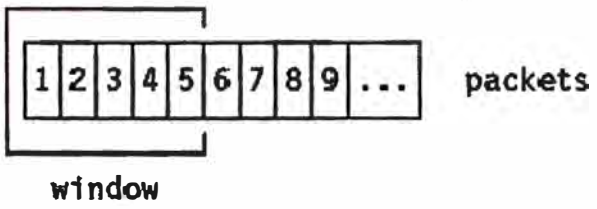


Figura 3.14 Agrupación de paquetes.

Paquetes del mensaje

Y utiliza las siguientes reglas:

- El emisor puede enviar todos los paquetes dentro de la ventana sin recibir un ACK, pero debe disparar un cronómetro para el timeout para cada uno de ellos.
- El receptor debe reconocer cada paquete recibido, indicando el número de secuencia del último paquete bien recibido.
- El emisor desliza la ventana para cada ACK recibido.

En la figura 3.15, el emisor puede transmitir paquetes del 1 al 5 sin esperar respuesta:

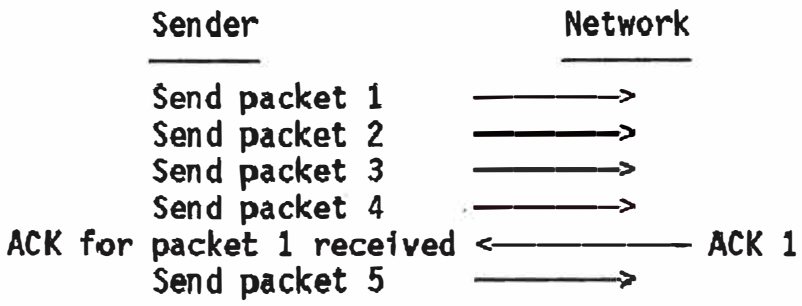


Figura 3.15 Envio de paquetes con acuse por tramos.

En el momento en que el emisor recibe el ACK 1, puede deslizar su ventana para excluir el paquete 1:

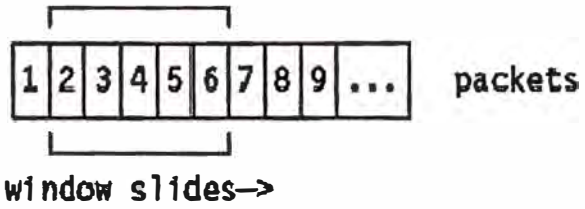


Figura 3.16 Recepción de acuse exclusión de paquete.

En este punto, el emisor puede transmitir también el paquete 6.

algunos casos especiales

- El paquete 2 se pierde: el emisor no recibirá ACK 2, por lo que su ventana permanecerá en posición 1(como se ve en el último dibujo). De hecho, como el receptor no recibió el paquete 2, reconocerá los paquetes 3, 4 y 5 con un ACK 1, que fueron los últimos paquetes recibidos en secuencia. En el extremo del emisor, al final se producirá un timeout para el paquete 2 y se retransmitirá. Notar que la recepción de este paquete en el receptor generará un ACK 5, ya que se habrán recibido con éxito los paquetes del 1

al 5, y la ventana del emisor se deslizará cuatro posiciones al recibir el ACK 5.

- El paquete 2 llegó, pero el reconocimiento se perdió: el emisor no recibe ACK 2, pero recibe ACK 3. ACK 3 es un reconocimiento de *todos* los paquetes hasta el 3(incluyendo el 2) y el emisor ya puede deslizar su ventana hasta el paquete 4.

Este mecanismo de ventanas asegura:

- Transmisión fiable
- Mejor aprovechamiento del ancho de banda(mejora del flujo).
- Control de flujo, ya que el receptor puede retrasar la respuesta a un paquete con un reconocimiento, conociendo los buffers libres de los que dispone y el tamaño de la ventana de comunicación.

El mecanismo descrito anteriormente se utiliza en TCP, pero con unas diferencias:

- Como TCP proporciona una conexión con un flujo de bytes, los números de secuencia se asignan a cada byte del canal. TCP divide el flujo de bytes en segmentos. El principio de la ventana se aplica a nivel de bytes; es decir, los segmentos enviados y los ACKs recibidos llevarán números de secuencia de forma que el tamaño de la ventana se exprese con un número de bytes, en vez del de paquetes.
- El tamaño de la ventana lo determina el receptor, cuando se establece la conexión, y puede *variar* durante la transmisión de datos. Cada ACK incluirá el tamaño de la ventana que acepta el receptor en ese momento.

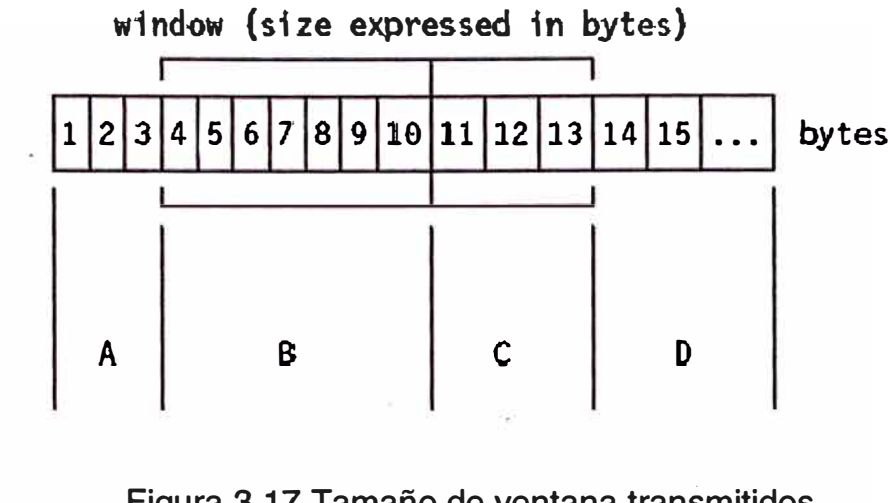


Figura 3.17 Tamaño de ventana transmitidos

Principio de la ventana aplicado a TCP

Donde:

A

Bytes transmitidos que han sido reconocidos.

B

Bytes enviados pero no reconocidos.

C

Bytes que se pueden enviar sin esperar ningún tipo de reconocimiento.

D

Bytes que no se pueden enviar aún.

Recordar que TCP agrupa los bytes en segmentos, y un segmento TCP sólo lleva el número de secuencia del primer byte.

Formato de segmento en TCP

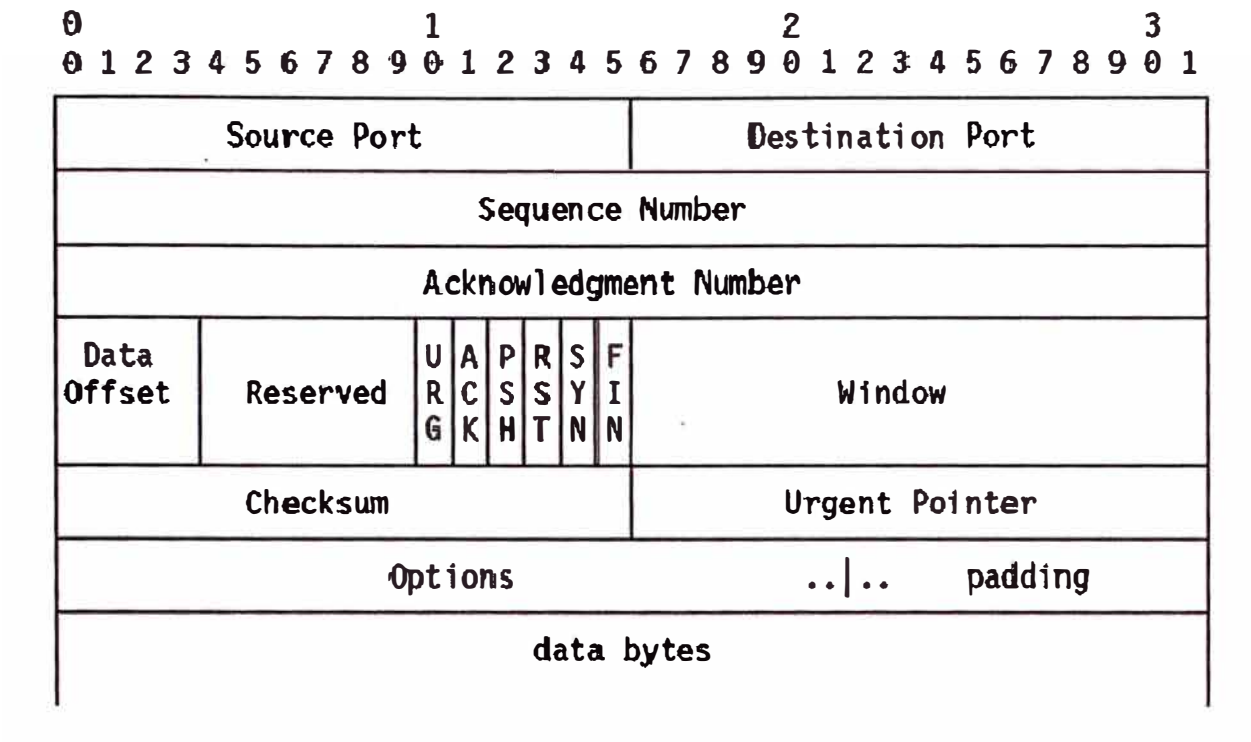


Figura 3.18 Secuencia de comunicación.

Donde:

Source Port

El número de puerto de 16 bits del emisor, que el receptor usa para responder.

Destination Port

El número de puerto de 16 bits del receptor.

Sequence Number

El número de secuencia del primer byte de datos del segmento. Si el byte de control SYN está a 1, el número de secuencia es el inicial(n) y el primer byte de datos será el n+1.

Acknowledgment Number

Si el bit de control ACK está a 1, este campo contiene el valor del siguiente número de secuencia que se espera recibir.

Data Offset

El número de palabras de 32 bits de la cabecera TCP. Indica dónde empiezan los datos.

Reserved

Seis bits reservados para su uso futuro; deben ser cero.

URG

Indica que el campo "urgent pointer" es significativo en el segmento.

ACK

Indica que el campo de reconocimiento es significativo en el segmento.

PSH

Función "Push".

RST

Resetea la conexión.

SYN

Sincroniza los números de secuencia.

FIN

No hay más datos del emisor.

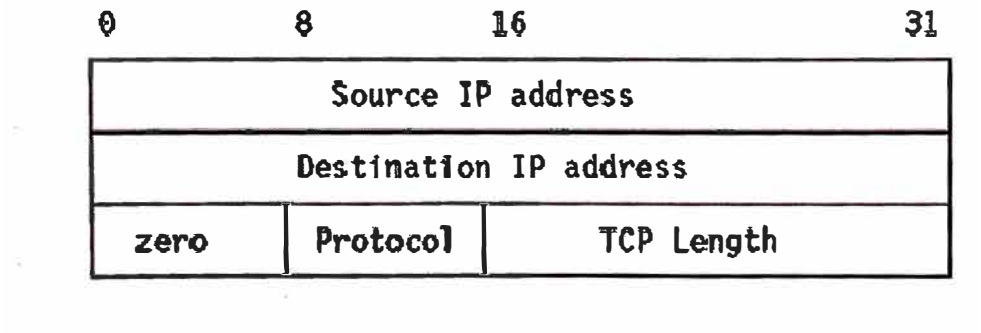
Window

Usado en segmentos ACK. Especifica el número de bytes de datos que comienzan con el byte indicado en el campo número de reconocimiento que el receptor esta dispuesto a aceptar.

Checksum

El complemento a uno de 16 bits de la suma de los complementos a uno de todas las palabras de 16 bits de la pseudo cabecera, la cabecera TCP y los datos TCP. Al computar el checksum, el mismo campo checksum se considera cero.

La pseudocabecera es la misma que utiliza UDP para calcular el checksum. Es una pseudocabecera IP, usada sólo para calcular el checksum, con el formato mostrado



Figuar 3.19 Seudocabecera

Pseudocabecera IP

Urgent Pointer

Apunta al primer octeto de datos que sigue a los datos importantes. Sólo es significativo cuando el bit de control URG está a uno.

Options

Sólo para el caso de opciones de datagramas IP, las opciones pueden ser:

- Un sólo byte conteniendo el número de opción, o

Una opción de longitud variable con el siguiente formato

Figura 3.20 Formato longitud variable.

: Opción del datagrama IP - Opción de longitud variable

Actualmente hay definidas tres opciones:

Tipo	Longitud	Significado
0	-	Fin e la lista de opciones.
1	-	No-Operación.
2	4	Tamaño máximo del segmento

Figura 3.21 Formato longitud fija.

Opción tamaño máximo del segmento

Esta opción sólo se usa durante el establecimiento de la conexión (bit de control SYN puesto a uno) y se envía desde el extremo que ha de recibir datos para indicar la máxima longitud de segmento que es capaz de manejar. Si esta opción no se usa, se admiten segmentos de cualquier tamaño.

Padding

Bytes todos a cero para rellenar la cabecera TCP a una longitud total que sea un múltiplo de 32 bits.

TCP envía los datos en segmentos de longitud variable. Los números de secuencia se basan en una cuenta de los bytes. Los *reconocimientos especifican el número de secuencia del siguiente byte que el receptor espera recibir.*

Ahora suponer que un segmento se pierde o se corrompe. En ese caso, el receptor reconocerá cualquier segmento sucesivo con un reconocimiento referido al primer byte del paquete perdido. Finalmente, se producirá un timeout y el segmento perdido se retransmitirá.

Suponer un tamaño de ventana de 1500 bytes, y segmentos de 500 bytes.

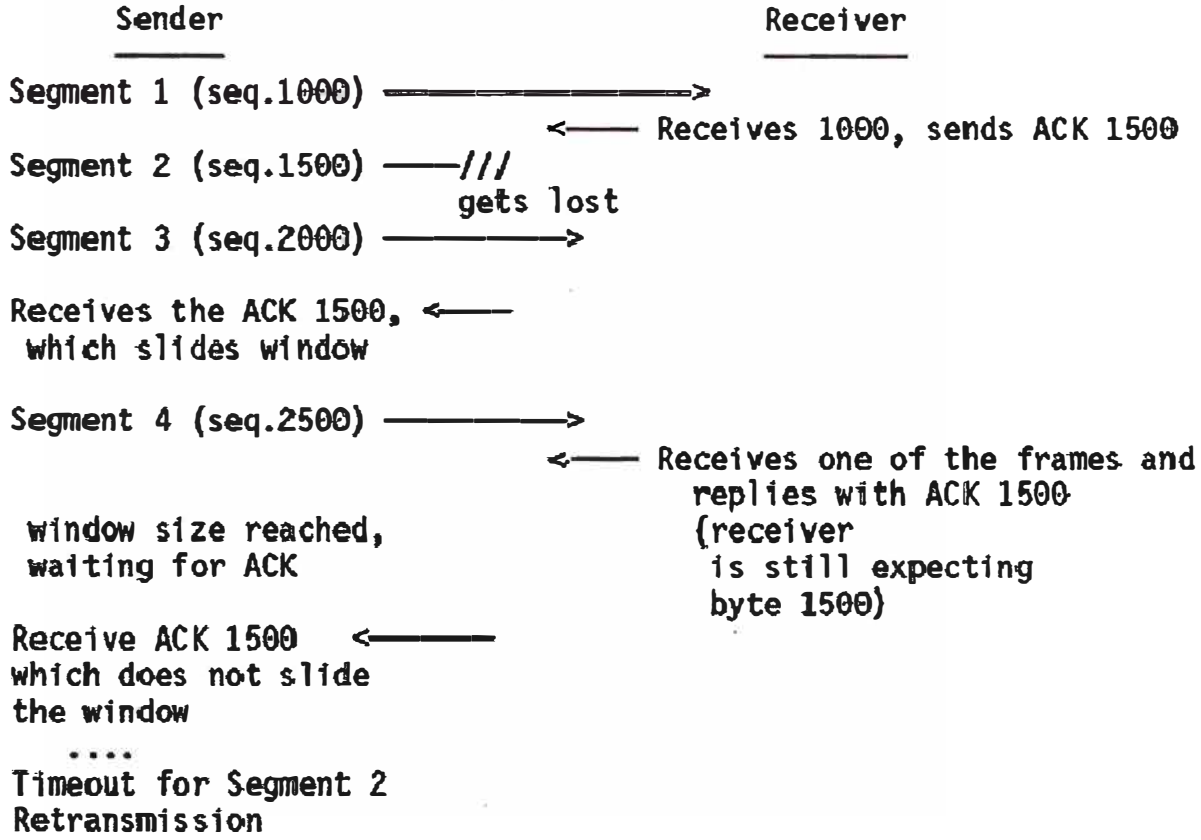


Figura 3.22 Reconocimiento de paquete perdido.

Proceso de reconocimiento y retransmisión

surge un problema, ya que el emisor sabe que el segmento 2 está perdido o corrompido, pero no sabe nada de los segmentos 3 y 4. El emisor debería

retransmitir al menos el segmento 2, pero también podría retransmitir los segmentos 3 y4. Es posible que:

1. El segmento 3 haya sido recibido, y no se sabe nada del 4: podría haber sido recibido ya, sin que el ACK haya llegado, o se podría haber perdido también.
2. El segmento 3 se ha perdido, y se ha recibido el ACK del segmento 4.

Cada implementación de TCP es libre de reaccionar ante un timeout del modo que deseen los diseñadores. Podría retransmitir sólo el segmento 2, pero en el segundo caso indicado arriba, estaremos esperando hasta que el timeout del segmento 3 expire. En este caso, se pierden todas las ventajas del rendimiento del mecanismo de ventanas. O bien TCP podría reenviar inmediatamente todos los segmentos de la ventana actual.

Sea cual sea la elección, el rendimiento máximo se pierde. Esto se debe a que el ACK no contiene un segundo número de secuencia indicando la trama actual que se ha recibido.

Cada TCP debería implementar un algoritmo para adaptar los tiempos de timeout a usar para el viaje de los segmentos. Para hacerlo, TCP registra el momento de envío de un segmento, y el de recepción del ACK. Se promedia un valor para varios de estos viajes que se empleará como valor de timeout para el siguiente segmento a enviar.

Esto es una característica importante, ya que los retardos pueden ser variables en la red, dependiendo de múltiples factores, tales como la carga de las redes intermedias de baja velocidad o la saturación de las pasarelas.

Antes de que se pueda transferir cualquier dato, se ha de establecer una conexión entre los dos procesos. Uno de los procesos(normalmente el servidor) lanza una llamada *OPEN pasiva*, el otro una llamada *OPEN activa*. El OPEN pasivo permanece dormido hasta que otro proceso intenta comunicarse con él a través de un OPEN activo.

En la red, se intercambian tres segmentos TCP:

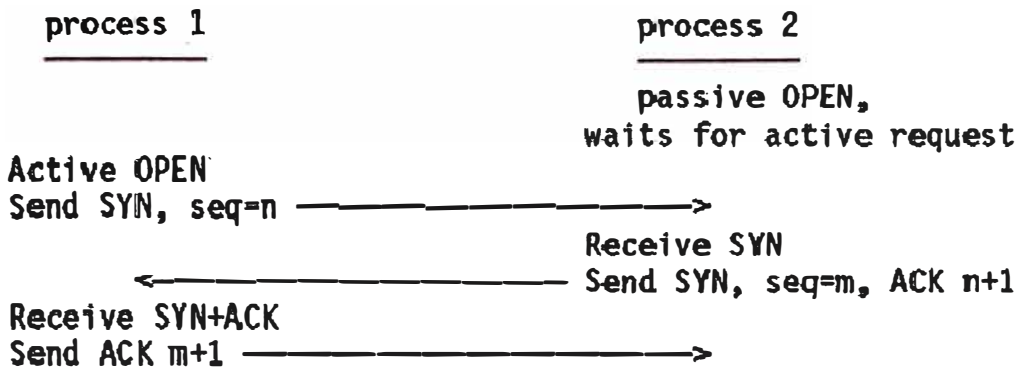


Figura 3.23 Establecimiento de conexión e inicialización.

Establecimiento de la conexión TCP

Este proceso completo se conoce como *three-way handshake*, o acuerdo en tres fases. Notar que los segmentos TCP intercambiados incluyen los números de secuencia iniciales de ambas partes, para ser usados en posteriores transferencias.

El *cierre* de la conexión se hace de forma implícita enviando un segmento TCP con el bit FIN activo. Como la conexión es full duplex, el segmento FIN sólo cierra la conexión en un sentido del canal. El otro proceso enviará los datos restantes, seguidos de un segmento TCP en el que el bit FIN está activo. La conexión se borra (es decir, la información de estado en ambos extremos) una vez que el canal se ha cerrado en ambos sentidos. Los TCP segmentos se transportan sobre datagramas IP con la siguiente configuración de parámetros:

Tipo de servicio = 00000000

es decir: precedencia = rutina

retraso = normal

rendimiento = normal

TTL = 00111100

La API de TCP no está definida del todo. Sólo algunas funciones básicas que deberían ser proporcionadas se describen en el *RFC 793 - TCP("Transmission*

Control Protocol"). Como ocurre con la mayoría de los RFCs de la pila de protocolos TCP/IP, se deja un elevado grado de libertad a los diseñadores, permitiendo en consecuencia implementaciones óptimas (dependientes del sistema operativo), lo que resulta en una mayor eficiencia.

El RFC describe las siguientes llamadas a funciones:

Open

Para establecer una conexión, tiene varios parámetros:

Activo/pasivo

Zócalo remoto

Número de puerto local

Timeout(opcional)

Y muchas otras opciones.

Devuelve un *nombre para la conexión local*, que se usa para referenciarla en todas las otras funciones.

Send

Hace que los datos del buffer del usuario señalado se envíen por la conexión. Opcionalmente puede tener los flags URGENT o PUSH activo.

Receive

Copia los datos TCP que van llegando a un buffer de usuario.

Close

Cierra la conexión; provoca un "push" de todos los restantes datos y segmentos TCP con el flag FIN activo.

Status

Es una llamada dependiente de la implementación que devuelve información como:

- Zócalo local y remoto
- Tamaños de las ventanas de recepción y envío
- Estado de la conexión
- Nombre de la conexión local

Abort

Hace que todas las operaciones de recepción y envío aborten, y se envíe un RESET al TCP remoto.

3.5 Protocolos Multipunto

El crecimiento de las redes *ATM* viene motivado, en parte, por la demanda de servicios multimedia para grupos dispersos de usuarios. El tráfico *multicast* tiene características particulares. La distribución de información *punto-a-multipunto* (*uno-a-muchos*) o *multipunto-a-multipunto (muchos-a-muchos*) es un objetivo básico propuesto por varios protocolos y arquitecturas *ATM* que ofrecen el soporte multimedia y/o *multicast* como audio-conferencia, video-conferencia, trabajos colaborativos o *VoD*.

ATM es aún una tecnología emergente diseñada para ser usada por aplicaciones de datos, audio y video, lo que requiere un buen comportamiento de las transferencias *unicast* y *multicast*. *User Network Interface (UNI 3.0*) para *ATM* define conexiones *punto-a-multipunto*, y las conexiones *multipunto-a-multipunto* sólo pueden ser obtenidas de las dos siguientes formas:

El primer esquema consiste en configurar N conexiones *punto-a-multipunto* para conseguir conectar todos los nodos en una topología completa *todos-con-todos*. Aunque esta topología ofrece conexiones *multipunto-a-multipunto*, hay que destacar que **no escala bien cuando el número de participantes es elevado.**

Una alternativa al anterior esquema es el uso de un servidor que actúa a modo de raíz en el árbol *multipunto*. Este método sólo requiere un nodo raíz para almacenar información, pero la desventaja de este método son las potenciales congestiones en el servidor cuando debe encargarse de envíos y retransmisiones de las conexiones multipunto-a-multipunto.

Para solventar las limitaciones de *UNI 3.0* y *UNI 3.1* que soportan conexiones *uno-a-muchos*, pero no directamente (nativamente) conexiones *muchos-a-muchos*, y ofrecer a *ATM* verdadero servicio *multicast*, *ATM Forum*, *ITU-T* e *IETF* han realizado varias propuestas al actual mecanismo de señalización *ATM* (*UNI 3.1*, *UNI 4.0*),

Los protocolos de transporte *multicast* más importantes en *Internet* son **MTP-2, XTP, RTP, SRM, RAMP, RMTP, MFTP, STORM**, etc.. *IETF* e *IRTF* (como *ITU-T* y

ATM Forum) también impulsan una importante actividad en este campo, los más representativos protocolos *multicast* se clasifican de acuerdo a la taxonomía de **varias características (propagación de datos, mecanismos de fiabilidad, retransmisiones, control de congestión y de flujo, gestión de grupos *multicast***, etc.). En *Internet* los mecanismos efectivos de control de congestión son una de las prioridades en las investigaciones de las transferencias *multicast* fiables. Los **mecanismos de seguridad y las técnicas escalables de recuperación de errores** son algunos de los aspectos actualmente en estudio en el campo de los protocolos de transporte *multicast*.

SMART (shared many-to-many ATM reservations es un protocolo para controlar un árbol *ATM multicast* compartido soportando comunicaciones *muchos-a-muchos (many-to-many)*. Esta propuesta tiene importantes características como que: reside completamente en la capa *ATM* y no requiere ningún servidor; soporta uno o varios *VCCs* (y también *VPCs*) cuyo número es libremente configurado y es independiente del número de puntos finales; usa el concepto de bloques de datos como en la **clase de servicio *ABT*** y también permite *VCCs* de las **clases *CBR*, *VBR* o *UBR*;** el protocolo garantiza que no existen puntos de interrelación en los *VCC* del árbol; son respetadas las garantías del contrato de tráfico asociado con los *VCCs*, etc. *SMART* puede ser entendido como un protocolo completamente distribuido para coordinar la distribución de *VPIs/VCIs*.

Para solventar las conocidas dificultades debidas al soporte y uso de *muchos-a-muchos VCCs, SMART* usa el mecanismo de **Cell Interleaving** (sobre un *VCC muchos-a-muchos*, las células de datos desde diferentes fuentes pueden llegar intercaladas a un destinatario) y también **Demand Sharing** (los recursos asignados a *conexiones muchos-a-muchos* son dinámicamente compartidas entre todas las potenciales fuentes.

Otras investigaciones como son: ofrecer equidad en los accesos a los árboles *multicast*; investigar las **células *RM*** periódicamente para aliviar las congestiones en la red o disminuir el tiempo de acceso del usuario a los árboles de distribución *multicast*; análisis de las células *RM* dentro de cada *VCC* o fuera enviando todas las células *RM* en un *VCC* dedicado.

***MWAX* un algoritmo dinámico** y escalable para *routing multicast* en el marco **PNNI** de redes *ATM*. Se ha identificado el problema para conseguir la escalabilidad con protocolos *multipunto-a-multipunto*

Un esquema jerárquico basado en ***CBT*** para incorporar *routing multipunto-a-multipunto* en *PNNI*. En el algoritmo los nodos *core* actúan como participantes pasivos para eliminar la dependencia en la selección de estos nodos. Con un mecanismo de *backup* se consigue un algoritmo tolerante a fallos en los nodos *core*, lo cual puede ser fácilmente extendido para incorporar *QoS* en el *routing multicast*. El protocolo- **algoritmo MWAS** es recursivo, esto es, el mismo protocolo es ejecutado en cada nivel de la jerarquía.

SEAM (Scalable and Efficient ATM Multicast) propone una arquitectura escalable, eficiente y *multicast multipunto-a-multipunto* para redes *ATM* que usa un sólo *VC* para un grupo *multicast* de múltiples emisores y receptores y todo ello sin realizar cambios en la capa *AAL5* de *ATM*. Esta propuesta permite a los grupos *multicast* aprovechar el soporte de *QoS* y la escalabilidad del ancho de banda. También realiza aportaciones para conseguir soportar *IP multicast* sobre redes *ATM* extensas.

SEAM usa un sólo árbol de distribución compartido para todos los emisores y receptores. Cada grupo *multicast* tiene un *core* asociado, el cual se usa como punto focal para todos los mensajes de señalización del grupo. Este trabajo deja abiertas investigaciones referentes a la gestión de tráfico y a la entrega fiable de tráficos *multicasting*.

MCMP (Multiparty Conference Management Protocol) que, sin estar pensado específicamente para *ATM*, es un protocolo de nivel sesión/transporte distribuido *extremo-extremo* y desarrollado para gestión de grupos de aplicaciones de conferencia. *MCMP* es un conjunto de algoritmos de control distribuido para configuración de conferencias *multipunto* y gestión de miembros de grupos de usuarios.

Conceptualmente, *MCMP* reside en el nivel de sesión en el que se establece la infraestructura para activar la transferencia de información entre los participantes en la conferencia. Pero funcionalmente, el protocolo acompasa los niveles de sesión y de transporte pues utiliza directamente servicios del nivel de red. Son destacables las condiciones de corrección (conectividad, validación, unicidad,

consistencia y terminación) que deben ser satisfechas una vez que la conferencia ha sido configurada por el algoritmo de configuración de **MCMP.**

CAPITULO IV
IP SOBRE ATM

4.1 Capa de interfase de Red sobre ATM

Las definiciones de implementaciones de IP clásico sobre ATM se describen en el **RFC 1577**, que es una propuesta de estándar con status electivo según el **RFC 1720(STD 1)**. Este RFC considera sólo la aplicación de ATM como una sustitución directa de los "cables"("wires"), segmentos LAN locales que conectan estaciones IP como extremos de la conexión("members") y "routers" que operan sobre el paradigma LAN clásico. Las consecuencias derivadas de los puentes a nivel MAC y de la emulación LAN no se toman en cuenta.

Una distribución inicial de ATM proporciona una sustitución de los segmentos LAN por:

Ethernets, redes en anillo o FDDI .

Troncales de área local entre LANs ya existentes(no ATM)

Circuitos dedicados a PVCs por retransmisión de tramas("Frame Relay") entre "router" IP

Cuando se necesita implementar calidad en el servicio se propone IP sobre ATM como una solución posible, el control de la calidad en el servicio se encarga la parte de ATM y de la expansión y conectividad se encarga IP

En general la utilización de redes IP sobre ATM es ciertamente compleja debido a las diferencias de diseño que existen entre ambos, la naturaleza orientada a la conexión de ATM no constituye el entorno ideal para un protocolo no orientado a la conexión como IP.

Habitualmente las redes ATM han sido orientadas a las redes Wan. Como redes de acceso basados en tecnologías XDSL , es estática basado en PVC (circuitos virtuales permanentes) esto simplifica la utilización conjunta de IP y ATM pero presenta inconvenientes no permite la realización de reservas de QoS dinámicas

que deben traducirse en el establecimiento dinámico de circuitos ATM (SVC) ni el aprovechamiento del servicio multipunto ATM para soportar tráfico multicast IP Sin enbargo las soluciones basadas en SVC son complejas debido principalmente al gran número de circuitos ATM a gestionar lo cual redunda en una mayor dificultad en la implementación y problema de escalabilidad

La solución planteada para llevar datagramas IP sobre SVC se especifico siguiendo el modelo IP clásico que se basa en como definir un datagrma IP viaja sobre una subred dicha solución denominada CLIP (clasical IP over ATM) engloba 2 entidades el servidor de ATMARP y el servidor MARS (Multicast Addres Resolution Server), el primero se encarga de las resoluciones IP a ATM esta solución que se resuelve de una manera sencilla en redes locales usando mecanismos de difusión a de realizarse en forma centralizada en ATM

Todos los clientes mantienen una relación sobre el servidor y a él dirigen sus preguntas cuando necesitan obtener la dirección ATM que corresponde a una dirección determinada sobre IP

El segundo servidor MARS se encarga de la gestión de los grupos multicast a diferencia de lo que sucede en la redes locales la correspondencia directa entre las direcciones IP multicast y las direcciones ATM no es posible puesto que en ATM no existen dirección multicast asignable dinámicamente. La solución pasa por gestionar la correspondencia entre direcciones IP multicast y el conjunto de direcciones ATM de los clientes que en cada momento desean recibir el tráfico dirigido a dicha dirección de grupo esto es precisamente la función del servidor MARS

Al momento de enviar tráfico multicast el servidor MARS define 2 escenarios el centralizado esta basado en la existencia de un servidor MultiCast Server MCS y todos los emisores que envían tráfico a un grupo multicast establecen una conexión con una misma maquina que es raíz de un único circuito multipunto compartido con todos los miembros del grupo. El segundo escenario es el distribuido (VC mesh) donde cada emisor realiza una consulta al servidor de MARS para obtener las direcciones ATM de los miembros del grupo.

Abriendo posteriormente un circuito multipunto directamente con ellos

LIS("Logical IP Subnetwork")

El término LIS se introdujo para mapear la estructura lógica de IP a la red ATM. En el contexto LIS, cada entidad administrativa independiente configura sus hosts y "router" dentro de una red IP. Cada LIS opera y se comunica con independencia de otros LIS de la misma red ATM. Los host conectados a una red ATM se comunican directamente con otros hosts dentro del mismo LIS. Esto implica que todos los miembros de un LIS sean capaces de comunicarse con otros hosts del mismo LIS por medio de ATM. La comunicación con hosts externos al propio LIS requiere un "router". El "router" es un extremo ATM conectado a la red ATM que se configura como un miembro de uno o más LISs. Esta configuración puede dar lugar a un número de LISs distintos operando sobre la misma red ATM. Los hosts de diferentes subredes deben usar un "router" aunque se pueda abrir un VC entre ellos a través de ATM.

4.2 Capa de adaptación ATM

Así como SONET tiene características que no pueden ser abarcadas por las definiciones tradicionales de los protocolos de capa física, ATM tampoco puede ser descrita completa mente como un protocolo de capa Data-link. La conversión de todo el tráfico a celdas de tamaño fijo, tal como lo hace ATM, no es requerida por las reglas de la capa 2, aunque una celda ATM es similar a un marco de capa 2 en tanto cuenta con funciones de corrección de errores. También dispone información de dirección que sólo tiene sentido para el enlace de datos local. ATM cuenta con dos mecanismos de QoS (Quality of Service: Calidad de Servicio), Sin embargo, ATM corresponde claramente con la definición de capa 2 cuando entra en interfaz con la capa física. Sus opciones de capa 1 incluyen muchos métodos de transmisión de fibra óptica que conducen celdas ATM, incluyendo a SONET (a casi cualquier tasa de transmisión), el cable trenzado de cobré categoría 5 y las líneas T1. Los conmutadores ATM pueden ser considerados dispositivos de capa Data-link, aunque generalmente necesitan funciones auxiliares para lidiar con IP u otros protocolos de capa de red.

La principal ventaja de ATM es su capacidad para manejar tráfico a tiempo real, tal como voz y video, sin introducir saltos y retrasos, a la vez que permite la

inclusión de una serie de flujos de datos que no tienen consideraciones estrictas de temporización.

Las redes sincrónicas tradicionales pueden soportar tráfico a tiempo real de forma impecable, pues reservan un circuito para uso exclusivo. La desventaja es que cuando no hay tráfico, la capacidad de ese circuito se desperdicia. Las redes de datagramas, como Internet, utilizan su capacidad de transporte muy eficientemente compaginando ráfagas de tráfico, mediante el uso de colas en los nodos intermedios, y obligando a los nodos finales a retransmitir los paquetes perdidos. El precio de esta eficiencia en la utilización del ancho de banda es un rendimiento impredecible, retrasos largos e intermitentes y, algunas veces, niveles inaceptables de saltos para consumidores de datos a tiempo real. ATM crea una red única para voz, video y datos, en la que los flujos de voz y video pueden mantener retrasos y saltos mínimos aceptables. Simultáneamente, los datos que no son tan sensibles al tiempo pueden conformarse con la capacidad restante del canal y, posiblemente, pagar una tarifa menor por sacrificar algo de la alidad de servicio garantizada. En su concepción original, ATM debía extenderse desde el escritorio hasta el núcleo de la nube de datos., La velocidad de las interfaces Ethernet aumentó en un factor de 10 y luego en un factor de 100 sin el correspondiente aumento de costo que las equiparara con las interfaces ATM con las que compiten. Si bien ATM domina el mercado de proveedores de servicio y tiene una fuerte presencia en los segmentos vertebrales corporativos, sólo ha sido adoptado en sistemas de usuario final en aplicaciones muy específicas, donde la máxima velocidad vale el alto precio que debe pagarse. Además, la combinación de ATM con redes Ethernet es compleja. Sin embargo, en la actualidad, la decisión de conectar una empresa con Internet a través de ATM en lugar de framerelay es, principalmente, una decisión financiera, y no técnica. Es muy fácil configurar un puerto ATM en su enrutador de acceso, un conmutador ATM, o un concentrador de acceso con un circuito permanente con el proveedor de servicio Internet sin causar problemas en la red local o dificultades en la instalación.

ATM acomoda diferentes tipos de tráfico mediante el protocolo capa de adaptación ATM ALL, que consiste en 2 sub capas convergencia y segmentación y reensamnlajje SAR en el lado de envío la subcapa de convergencia acepta

mensajes con datos de los protocolos de aplicación de capa superior, interpreta el tipo y formato de los mensajes y los prepara luego para su procesamiento por la subcapa SAR. La sub capa SAR traduce entonces los mensajes a formato ATM empacándolos en la parte de carga útil de 48 bytes de una celda ATM lo inverso se realiza en el receptor.

La interpretación de la subcapa de convergencia de tipo y formato de dfatos se basa en al clase de servicio (QoS) especifico asignado a los datos en la capa de aplicación. El AAL proporciona el soporte para 4 diferentes clases de servicios Estos se encuenbtran resumiddos en la siguiente tabla

Tabla 4.1

Clase de servicio	Descripción
Calse A	Transmisiones Orientadas a la conexión de velocidad bit constante CBR que requieren de una relación estricta de tiempo entre los nodos fuente destino ejemplo circuito de emulación y transmisión de voz.
Clase B	Transmisiones Orientadas a la conexión de velocidad bit variable VBR que requieren de una relación estricta de tiempo entre los nodos fuente destino ejemplo transmisiones de datos sincronizados o video por paquetes, video conferencias.
Clase C	Transmisiones VBR Orientadas a la conexión que no requieren de una relación estricta de tiempo entre los nodos fuente destino ejemplo aplicaciones de transferencia de datos LAN como Frame Relay.
Clase D	Transmisiones VBR no orientadas a la conexión no requieren de una relación

	estricta de tiempo entre los nodos fuente destino ejemplo aplicaciones de transferencia de datos LAN como el SMDS.

Cuando la subcapa SAR recibe datos de la sub capa de convergencia procesa los datos para que sean consistentes (es decir cumplan con las necesidades de transmisión) con el tipo y formato especificado. Dependiendo del tipo de datos el protocolo AAL proporciona 5 tipos diferentes de AAL para acomodar una clase de servicio particular , por ejemplo el AAL1 se usa para datos que requieren transmisiones orientadas a la conexión con velocidad de bit constante un ejemplo es el servicio tradicional de voz.El AAL2 se usa para datos que requieren transmisiones orientadas a la conexión con velocidad de bit variable un ejemplo es el video en paquete que podría usarse en aplicaciones de videoconferencia. AAL3 y AAL4 se usa para datos que requieren transmisiones orientadas a la conexión con velocidad variable ejemplo datos en ráfaga tipiuco de redes LAN como los que se encuentran en redes Frame Relay (orientadas a conexión) y SMDS sin conexión y finalmente AAL5 que inicialmente fue llamado la simple y eficiente capa de adaptación SEAL se usa para transmisiones en que los protocolos de capa superior proporcionan recuperación de datos, AAL% es una mejora de AAL3 como resultado por medio del protocolo AAL y sus capas de adaptación una transmisión ATM es capaz de acomodar diferentes tipos de transmisiones en forma simultánea sobre la misma red. La correspondencia entre tipos AAL y clase de servicio se resume en la tabla.

Tabla 4.2 Clasificación por tipo de AAL

Tipo AAL	Clase de Servicio
AAL1	Clase A
AAL2	Clase B
AAL3	Clase C/D

AAL4	Calse C/D
AAL5	Calse C/D

Observando las tablas se puede ver que parece haber una correspondencia 1 a 1 entre las clases de servicios y los tipos AAL el diseño inicial consideraba que cada tipo AAL tuviera una clase de servicio correspondiente, □robablemente□e desde el AAL1 al AAL4 fueron diseñados □robablemente□e para soportar las clases de servicio de la A a la D sin enbargo cuanldo AAL3 y AAL4 estaban siendo desarrollados se observó que habia un traslape considerable entre las 2 especificaciones, en consecuencia se conbinaron en un solo tipo al que se llama ahora AAL tipo ¾ , el AAL5 fue desarrollado para tener un AAL tipo ¾ menos complejo.

Muchos de los servicios soportados en las clases C y D no garantizaban el nivel de sofisticación incorporado por el tipo ¾ como resultado surge AAL5

□robablemen AAL5 es el mas usado la razón uno de los □robableme ATM , ATM □roba definio AAL5 para una transmisión eficiente del TCP/IP como AAL5 usa una muy poca cantidad de la celda para sobrecarga se ha vuelto muy popular entre blos proveedores e inplementgadores de tecnología ATM , si el AAL se usa o no en una □robable depende de si el proveedor del paquete de protocolo usa el □robable ATM en una topología de red. Alghunas de las otras capas AAL podrían ser implementadas con el tiempo, nuevas capas □robablemente serán inventadas y algunas de las antiguas descartadas.

4.3 Interfase Controladora de Red

Un examen más cercano del protocolo ATM y cómo opera ayudará a explicar cómo los circuitos virtuales, las rutas virtuales, los conmutadores y los servicios que ellos acarrean se afectan entre sí.

La figura 4.1 muestra un formato básico y la jerarquía de ATM. Una conexión ATM, consiste de “celdas” de información contenidos en un circuito virtual (VC). Estas celdas provienen de diferentes fuentes representadas como generadores de bits a tasas de transferencia constantes como la voz y a tasas variables tipo

ráfagas (bursty traffic) como los datos. Cada celda compuesta por 53 bytes, de los cuales 48 (opcionalmente 44) son de información y los restantes para uso de campos de control (cabecera) con información de "quién soy" y "donde voy"; es identificada por un "virtual circuit identifier" VCI y un "virtual path identifier" VPI dentro de esos campos de control, que incluyen tanto el enrutamiento de celdas como el tipo de conexión. La organización de la cabecera (header) variará levemente dependiendo de sí la información relacionada es para interfaces de red a red o de usuario a red. Las celdas son enrutadas individualmente a través de los conmutadores basados en estos identificadores, los cuales tienen significado local ya que pueden ser cambiados de interface a interface.

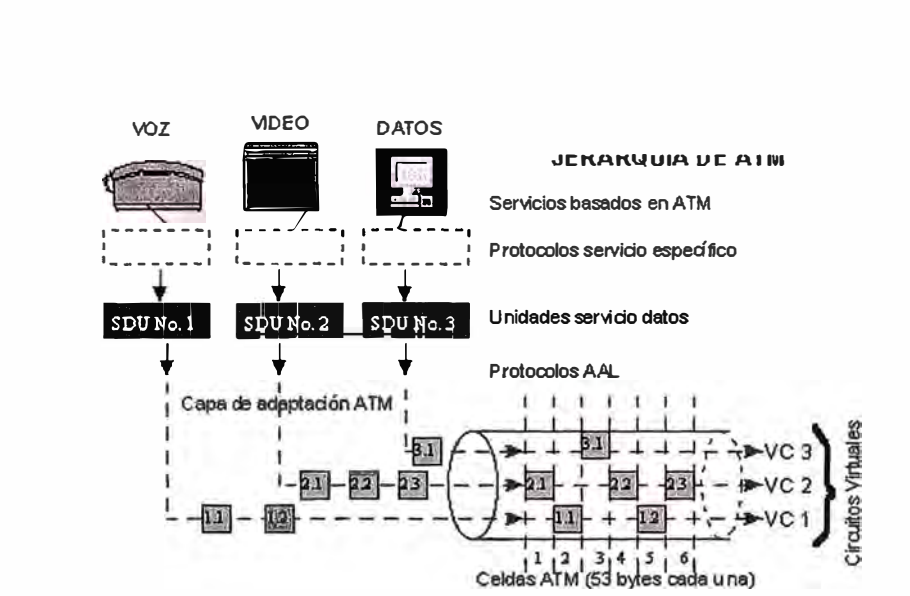


Figura 4.1 Formato básico ATM.

La técnica ATM multiplexa muchas celdas de circuitos virtuales en una ruta (path) virtual colocándolas en particiones (slots), similar a la técnica TDM. Sin embargo, ATM llena cada slot con celdas de un circuito virtual a la primera oportunidad, similar a la operación de una red conmutada de paquetes. La figura siguiente describe los procesos de conmutación implícitos los VC switches y los VP switches.

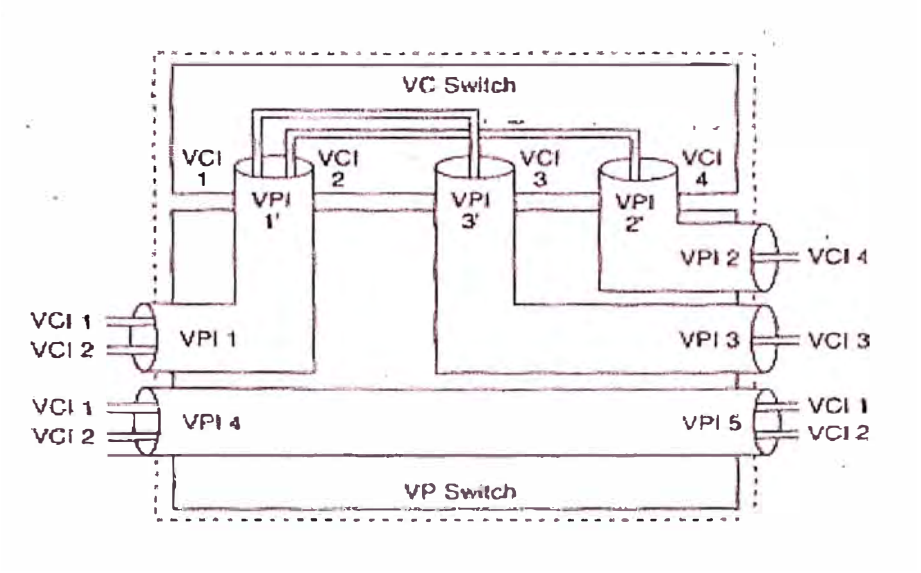


Figura 4.2 Direccionamiento ATM.

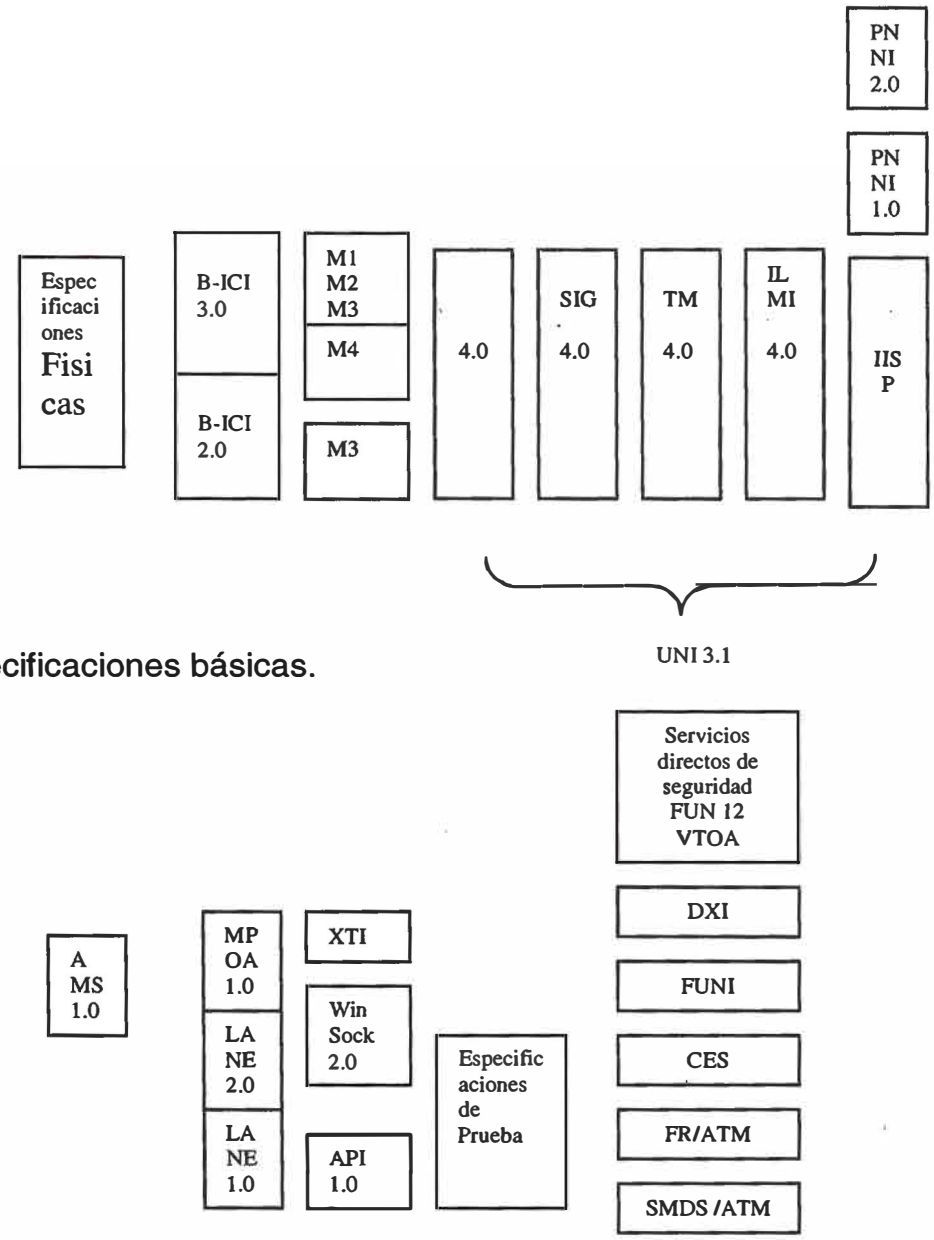
Los slots de celda no usados son llenados con celdas "idle", identificadas por un patrón específico en la cabecera de la celda. Este sistema no es igual al llamado "bit stuffing"en la multiplexación Asíncrona, ya que aplica a celdas enteras.

Diferentes categorías de tráfico son convertidas en celdas ATM vía la capa de adaptación de ATM (AAL – ATM Adaptation Layer), de acuerdo con el protocolo usado.

La tecnología ATM ha sido definida tanto por el ANSI como por el CCITT a través de sus respectivos comités ANSI T1, UIT SG XVIII, como la tecnología de transporte para la B-ISDN (Broad Band Integrated Services Digital Network), la RDSI de banda ancha. En este contexto "transporte" se refiere al uso de técnicas de conmutación y multiplexación en la capa de enlace (Capa 2 del modelo OSI) para la transferencia de tráfico del usuario final de la fuente al destino, dentro de una red. El ATM Forum, grupo de fabricantes y usuarios dedicado al análisis y avances de ATM, ha aprobado cuatro velocidades UNI (User Network Interfases) para ATM: DS3 (44.736 Mbit/s), SONET STS3c (155.52 Mbit/s) y 100 Mbit/s para UNI privados y 155 Mbit/s para UNI privadas. UNI privadas se refieren a la interconexión de usuarios ATM con un switch ATM privado que es manejado como parte de la misma red corporativa. Aunque la tasa de datos original para ATM fue de 45 Mbit/s especificado para redes de operadores (carriers) con redes T3 existentes, velocidades UNI adicionales se han venido evaluando y están

ofreciéndose. También hay un alto interés en interfases, para velocidades El (2Mbps) y T1 (1,544 Mbps) para accesos ATM de baja velocidad.

En ATM hay una jerarquía de interfaces estandarizada llamadp el anchorage Accord (acuerdo de anclaje), la jerarquía básica como es definida por el acuerdo se muestra en la figura, estandares separados definen la manera como estan interconectados los componentes de la red y como interactuan en la figura de izquierda a derecha los diferentes componentes del ambiente ATM estan definidos por las □specificaciones funcionales de la lista, los diversos números indican versiones sin enbargo ese número de versiones variaran con el tiempo conforme los estandares progresen.



Especificaciones básicas.

Figura 4.3 Estandares ATM.

Espacificaciones de aplicación y servicios.

AMS	Servicios de multimedia audiovisuales.
B-ICI	Interfaz portadora ISDN de banda ancha.
CES	Servicio de emulación de circuito.
DXI	Interfaz de intercambio de datos.
FR/ATM	Frame relay sobre ATM.

FUNI	interfaz de bloque usuario a red.
IISP	protocolo provisional de señalamiento entre commutadores.
ILMI	interfaz de administración integrada local.
LANE	emulación de LAN.
M	multiplexión.
MPOA	multiprotocolo sobre ATM.
PNNI	interfaz privada de red a red.
SIG	especificaciones de señalamiento para UNI
SN}MDS/ATM	Servicio de datos conmutado multimegabit sobre ATM.
TM	Admistración de tráfico.
UNI	Interfaz de usuario a red.
VTOA	Voz y telefonía sobre ATM.

Los estandares estan divididos en 2 grupos:

Los básicos y las aplicaciones y servicios.los primeros definen conjuntos ATM en funciones básicas la segunda define como otras redes y aplicaciones interoperan con componentes ATM.

En una red ATM el tipo de computadora es irrelevante para las capacidades de transporte de la red esto quiere decir que los PBX y las redes de datos,las facilidades de video a exhibición pueden todas compartir simultáneamente la misma infraestructura , esto se hace usando los estandares funcionales para proporcionar interfaces exactas para tipos diferentes de datos a la infraestructura de la red. De esta manera la red soporta todos los tipos diferentes de interconexión de servicios requeridos asi como todos los tipos diferentes de requisitos de transmisión.

ATM usa direcciones para cada punto de conexión se usan 2 tipos de direcciones para redes privadas NSAP (puntos de red de acceso del servicio) de OSI los NSAP son direcciones de 20 bytes e incluyen un prefijo de 13 bytes que puede ser usado para identificar una localización especifica, incluida una región.

Las redes públicas usan E.164 el formato estandar global de la ISDN para numerar direcciones especificado por la ITU-T se pretende estandarizar todos los dispositivos públicos o privados sobre direcciones con base NSAP.

Para conestar un dispositivo a una red ATM se debe tener un NIC con base ATM, estas tarjetas de red son dispositivos de capa 2 y contiene circuitería ATM son

entonces conectados a un conmutador local que es usualm ente proporcionado por gente corporativa de tecnología de redes.

La conexión física entre la estación de trabajo y el conmutador local (llamado tambien privado) puede ser cobre (tipicamente UTP cat 5 corriendo a 25 Mbps o 155 Mbps y Terabits) o inalámbrico (10 Mbps o mayores) la especificación de un NIC fiable a un conmutador es lka especificación funcional de interfaz usuario a red UNI, el propósito de la especificación UNI es definir exactamente como se comunica una estación en un extremo con el siguiente nivel de interfaz de la red que es un conmutador privado en un sitio. Esta especificación es bastante grande y compleja , entre otras cosas especifica el establecimiento de conexión entre la NIC y el conmutador, la calidad de los requisitos de servicios para el tráfico entre ellos y los otros facilidades orientadas a la conexión que ATM requiere para garantizar que el NIC y el conmutador se entienden, además la UNI también especifica secciones de los atributos del hardware físico tales como aspectos de la interfaz de cobre, de fibra y funcionalidades de los componentes inalámbricas esas especificaciones colectivas y funciones son llamadas por lo general señalamiento UNI. Assi la UNI pretende específicamente la comunicación de punto final al ambiente del conmutador ATM.

Una comunicación de conmutador a conmutador es diferente en una red privada (premisas) se llama interfaz privada de red a red (PNNI), la PNNI proporciona funcionalidad especifica que conmuta la necesidad entre cada uno pero que las estaciones no atienden en absoluto, existen además otras especificaciones 2 en especial son la interfaz interoperadora de banda ancha ISDN (B-ICI) y la interfaz LAN usuario a red LUNI.

La B-ICI se usa para conectar 2 redes publicas. ATM entre diferentes proveedores de servicio de red, la LUNI especifica como un usuario emulado como una LAN se conecta a un ambiente ATM. Por ejemplo si tenemos una ethernet 802.3 existente u otra LAN entonces hay una especificación llamada LANE (emulación de red de área local) que ayuda a la conexión de una red ATM de igual manera si tenemos una facilidad WAN de Frame Relay que necesita interoperar con ATM , entonces se usa la interfaz de bloques de usuario a red (FUNI) hay muchas otras, así donde hay un punto de conexión de red sobre ATM que se conecta a un

punto de conexión disímil, se define una especificación funcional (y por supuesto una abreviación relacionada) asociada con el método de conexión.

Los conmutradores ATM son los resposables de transmitir datos dentro de una red ATM, un conmutador ATM implementa un metodo de enrutamiento llamado tejido conmutador. Los algoritmos del tegido del conmutador se presentan en una amplia gama de variedad de inplementaciones, pero los tipos mas comunes son los llamados ambiente de conmutación Batcher-Bayan y delta. Los algoritmos en el conmutador intercambian información con otros conmutadores interconectados prara saber donde estan en el conjunto de la red las localidades SAP de fuente y destino. Usando esos métodos de alta velocidad una red muy grande de componentes puede ser mapeada rápidamente y descubrier cuando el tráfico llega a la red física y necesita ser enviado a la localidad apropiada . Cuando son primero energizados los conmutadores se configuran asi mismos pasando a un modo de auto configuración que les permite saber automáticamente que otros conmutadores están conectados.

Ellos tambien proporcionan direcciones NSAP temporales a puntos de conexión ATM que pueden necesitarse para interoperar y ayudar a administrar el ambiente completo de la red desde una perspectiva de configuración y administracióbn del tráfico.

CAPITULO V
TECNOLOGÍA DE CONVERGENCIA

5.1 Introducción

La tendencia en redes es una fuerte convergencia a las aplicaciones de voz, datos y video para satisfacer los 3 tipos, algunas funciones básicas deben ser creadas para permitir que los 3 tipos de tecnologías de redes coexistan. Los datos de las operaciones con redes permiten caracterísrticas de tráfico que pueden ser en su naturaleza de tipo ráfaga y los datos pueden tener paquetes de longitus variable, el tráfico de datos pueden támbien tolerar una cierta cantidad de demora de transmisión especialmente en tráfico que no está en tiempo real, esto implica que las velocidades de llegada de datos pueden ser de naturaleza variable lo que sugiere que las transmisiones de tasa de bit variable VBR son aceptables para aplicaciones de datos.

Por otro lado el tráfico de voz es mas sensible al tiempo de llegad del tráfico es una buena idea que los paquetes que llevan parte de la conversación lleguen lo suficientemente rápido para enviar intervalos de tiempo en blanco en la conversación esto sugiere usar una transmisión con tasa de bits constante para la voz, existe un problema con el lenguaje de full duplex donde 2 o mas personas hablan exactamente al mismo tiempoy pueden oirse simultáneamente, las redes no necesariamente concuerdan muy bien con este concepto, sin eenbargo la vboz no ocupa gran ancho de banda por lo que redes rápidas de datos como VBR pueden soportar comunicaciones de voz siempre que la red sea rápida y no sufra congestión o demoras por pérdidas

El video es una de las señales más sensibles al tiempo tales señales esperan que el número de bloques transmitidos lleguen en orden y en un intervalo de tiempo muy específico (usualmente medidio en milisegundos) las tecnologías como las de velocidad de bit constante, donde el numero de bits de llefada en una transmisión son constantes y consistentes, son esenciales para ser posible una

velocidad estándar de llegada de tráfico sin enbargo el CBR no es sufuciente los bloques deben ser transmitidos en el orden aporopiado y deben llegar con la velocidad correcta dentro de un intervalo de tiempo específico para hacer esto la red debe reservar anchos de banda en la trayectoria de la fuente al destino para garantizar que todos los bits lleguen en orden y a tiempo. Este método general de proporcionar al CBR una secuencia garantizada de envío en un intervalo de tiempo específico donde se ha reservado espacio de trayectoria, se llama comunicación isócrona y es comun a todas las redes ATM.

Es claro que ATM es la única red actualmente que proporciona la tecnología de transmisión, las velocidades y la calidad de servicio requeridas para enfrentar las necesidades de los usuarios por lo menos por ahora.

5.2 ATM frente a otras tecnologías

Actualmente, las incertidumbres y paradojas abundan en el campo de la columna vertebral. Tan pronto como IP demostró su capacidad para soportar redes de gran escala, sin conexión y de mejor esfuerzo, la IETF puso en marcha el grupo de trabajo de servicios integrados para brindar un tráfico orientado a conexiones, con admisión controlada y de garantía de calidad. Las telefónicas partidarias de IP se quejan de la ineficiencia a la que sus redes se ven sometidas por los protocolos de las capas más bajas, pero luego deciden adoptar métodos de calidad de servicio basados en IP que no pueden soportar el tráfico de datos a tiempo real a menos que la red reserve enormes recursos y no permita el fenómeno de congestión.

Los proveedores de servicio se jactan de las enormes capacidades de sus cables de fibra óptica, multiplicadas por el número de longitudes de onda de las más recientes especificaciones DWDM (Dense Wavelength Division Multiplexing: Multiplexión por División Densa de Longitudes de Onda) que pueden viajar sobre un mismo cable, pero se quejan por la ineficiencia de unos cuantos puntos porcentuales en su infraestructura. Justo cuando parecía que alguna combinación de conmutación con paquetes y celdas será la base inevitable de las redes multiservicios del futuro, aparece en el panorama una compañía como Monterrey Networks. Su producto, a pesar de llamarse enrutador de longitud de onda, configura caminos de circuitos conmutados entre diversas longitudes de onda OC-

48 y redirecciona de forma flexible las señales entrantes en una longitud de onda, o tono, a una longitud saliente distinta, bajo el control de un protocolo WARP (Wavelength Routing Protocol: Protocolo de Enrutamiento por Longitud de Onda)..

Claramente, las telefónicas que brinden calidad de servicio según su mejor esfuerzo pueden eliminar la ineficiencia de sus núcleos enrutados prescindiendo de ATM y APS. Las que intenten ofrecer redes multiservicios, capaces de manejar voz y video así como tráfico de datos no sensibles al tiempo, pueden garantizar una mejor calidad de servicio empleando ATM, al menos a lo largo del núcleo delimitado por los dispositivos de borde. Los servicios de voz y video son necesarios sin importar si los datos vienen en forma de paquetes IP que a su vez se convierten en celdas de ATM, o directamente sobre celdas ATM.
Si el ancho de banda se hace abundante y económico, las soluciones basadas en IP pueden brindar con éxito servicios a tiempo real sobre redes no basadas en ATM, pues cualquier flujo de datos sensibles al tiempo podría recibir sus propias longitudes de onda. Sin embargo, si el ancho de banda baja de precio, habrá pocos incentivos, para invertir esfuerzos en aumentar la eficiencia en los protocolos de las capas más bajas. Si el costo del ancho de banda en redes de área extensa no cambia, entonces una red en la que converjan todos los tipos de datos, incluyendo voz y video, será la mejor forma para reducir al mínimo el costo total de operar una red. En este escenario, IP sobre ATM sobre SONET, con MPLS para allanar los picos en los que la conmutación brinda ventajas sobre el enrutamiento y viceversa, quizás sea la solución más ventajosa.
La ATM fue diseñada inicialmente para WAN para usarse en B-ISDN pero también se puede usar en LAN o MAN, La infiltración de Tamal frente de las LAN fue diseñada para servir como una tecnología troncal a 155 Mbps y entregar 25 Mbps a la estación (llamada ATM de baja velocidad) frente a las velocidades de la Fast Ethernet y el Gigabit Ethernet las velocidades de datos de la ATM son bajas y su despliegue muy costoso. Sin enbargo el Forum ATM esta trabajando en una especificación ATMa 2.5 Gbps para la troncal LAN
La tecnología ATM fue diseñada inicialmente para redes WAN para usarse en la B-ISDN sin enbargo por la gran capacidad de poder transmitir voz. video y datos ahora tambiém puede usarse para redes LAN (o para MAN) la operación der

ATM en una LAN fue diseñada para servir como una tecnología troncal a 155Mbps y entregar 25 Mbps (llamada ATM de baja velocidad) a la estación de trabajo sin enbargo este gran esquema de una ATM en WAN, MAN y LAN unificadas fue considerado por Ethernet , específicamente por Fast Ethernet que entrega 100 Mbps a la estación de trabajo y la Giga bit Ethernet que proporciona una troncal de 1000 Mbps

Compara con la Fast Ethernet o Giga bit Ethernet las velocidades de datos de ATM son bajas y su despliegue muy costoso. Sin nebargo el Forun ATM esta trabajando en una especificación ATM a 2.5 Gbps para la troncal LAN.

Aunque Gigabit Ethernet es probablemente capaz de transmitir datos y voz a niveles aceptables, aunque es una tecnología VBR tiene grandes problemas enm video esto es evidente cuando la red se congestiona o cunado se requiere un intervalo de tiempo especifico de entrega, un buen ejemolo de ello es la televisión de alta definición HDTV que con el tiempo aparecerá en las redes globales como un método de transmisión estandar.

Es de suponerse que nuevos protocolos como el Resourse Reservation protocol RSVP y el Real time transport Protocol RTP remediarán las deficiencias QoS de la Gigabit Ethernet . Ambos protocolos permiten aplicaciones psrs resrvar una cantidad específica de ancho de banda para transmisión de datos. Sin enbargo cuando consideramos que representa en realidad la incorporación de esos protocolos con el Gigabit Ethernet y luego comparamos sus funciones con la ATM surgen varios problemas primero los bloques Ethernet son de losngitud variable esto es de 64 a 1518 bytes, esto solo sugiere que la velocidad de entrega no será consistente. ATM emplea celdas de tamaño fijo lo que garantiza una velocidad de entrega constante, segundo en una transmisión Ethernet los bloques hacen cola en un conmutador sobre la base de que el primero que llega esw el primero que sale además antes que su conmutador transmita n bloques en cola el contenido entero de n-1 bloques en cola debe ser transmitido. Así un conmutador transmite secuencialmente bloques en cola y en el orden en que ellos entraron al buffer eel proble ma de este esquema es que si 2 bloques, ambos reservando ancho de banda. Llegan al mismo tiempo (con unos pocos microsegundos de diferencia) a un puerto de conmutador. El bloque que llega primero es el primero en ser transmitido y el bloque que llega en segundo lugar es colocado en un buffer. Por

otra parte los conmutadores ATM pueden crear y dar servicio simultáneamente a múltiples colas independientes con prioridades diferentes (clase de servicio) y necesidades diferentes de transmisión con base en el tipo de datos (esto es calidad de servicio) además esas transmisiones simultáneas y multiples son efectuadas con uns velocidad constante de entrega

Hacer la red más rápida no resolverá el problema de convergencia incluso el trabajo sobre Terabitethernet sserá una red más rápida que la red ATM comercial, sin enbargo no tendrá los controles apropiados de tráfico y factores de confiabilidad de red que son inherentes en una red ATM, tampoco se emplea factores de confiabilidad

Con referencia a Frame Relay ATM es muy parecido pero la gran diferencia radica en que ATM tiene celdas fijas y Frame Relay celdas variables, ATM ofrece 2 ventajas frente a Frame Relay la velocidad y tipo de tráffico, como todas las celdas ATM tienen el mismo tamaño son mucho mas fáciles de procesar usando ceñldas cortas con demoras de transmisión predecibles, ATM puede conbinar celdas que llevan tráfico sensible a demoras como video y voz interactivo, con celdas de datos. Este concepto llamado entrelazamiento no es posible con el Frame Relay por que bloques de datos mas largos generan demoras mayores y no predecibles al procesar tráfico de voz y video asi el Frame Relay es menos adecuado para video conferencias en tiempo real.

Con respecto a SONET, este no es mas que un mecanismo de transporte, ATM no requiere el uso de ningun protocolo de capa física específica. Como un servicio de portadora de gran ancho de banda SONET puede servir como un medio de transporte par cualquier tecnología de red o servicio como ATM, FDDI, SMDS y el ISDN , sin enbargo se usa con frecuencia trafico ATM para ser llevado por SONET.

Además de servir como una troncal WAN, ATM puedeb también emular redes Ethernet / 802.3 o Token Ring por medio de una interfaz de emulación de red LAN llamada LANE las LAN que incorporan LANE son llamadas redes de área local emuladas, en la jerarquia de ATM la LANE está más arriba que AAL5 en la capa de adaptacón ATM, El protocolo LANE define una interface de servicio para la capa de red que funciona idénticamente a la usada por la ethernet /802.3 y por las

LAN Token Ring. Los datos que cruzan esta interface son enxcapsulados en el formato apropiado de la sub capa Mac.

En un ambiente ELAN los nodos extremos LAN son conectados a un equipo especial de emulación LAN que corre un proceso de emulación LAN del cliente LEC , el LEC funciona como un nodo de terminales proxy ATM, un nodo extremo ATM nativo corre un proceso de emulación LAN del servidor LES que es responsable de resolver direcciones hardware MAC en direcciones ATM.

Por ejemplo se desea transmitir una información donde la fuente emplea tecnología ethernet y el nodo destino ethernet a traves de conmutadores ATM, el nodo fuente transmite el bloque de datos al proceso LEC que se encuentra en un dispositivo de emulación LAN, el LEC emite una difusión ARP solicitando una resolución de dirección MAC a ATM, un proceso LES que reside sobre un dispositivo nativo ATM responde a la difusión ARP y retorna al LEC la dirección ATM del dispositivo remoto de emulación LAN al que el nodo destino está conectado.La fuente LEC establece entonces un circuito virtual al destino LEC, el dispositivo de emulación LAN traslada entonces el bloque Rthernet a una celda ATM a traves de al sub capa SAR.

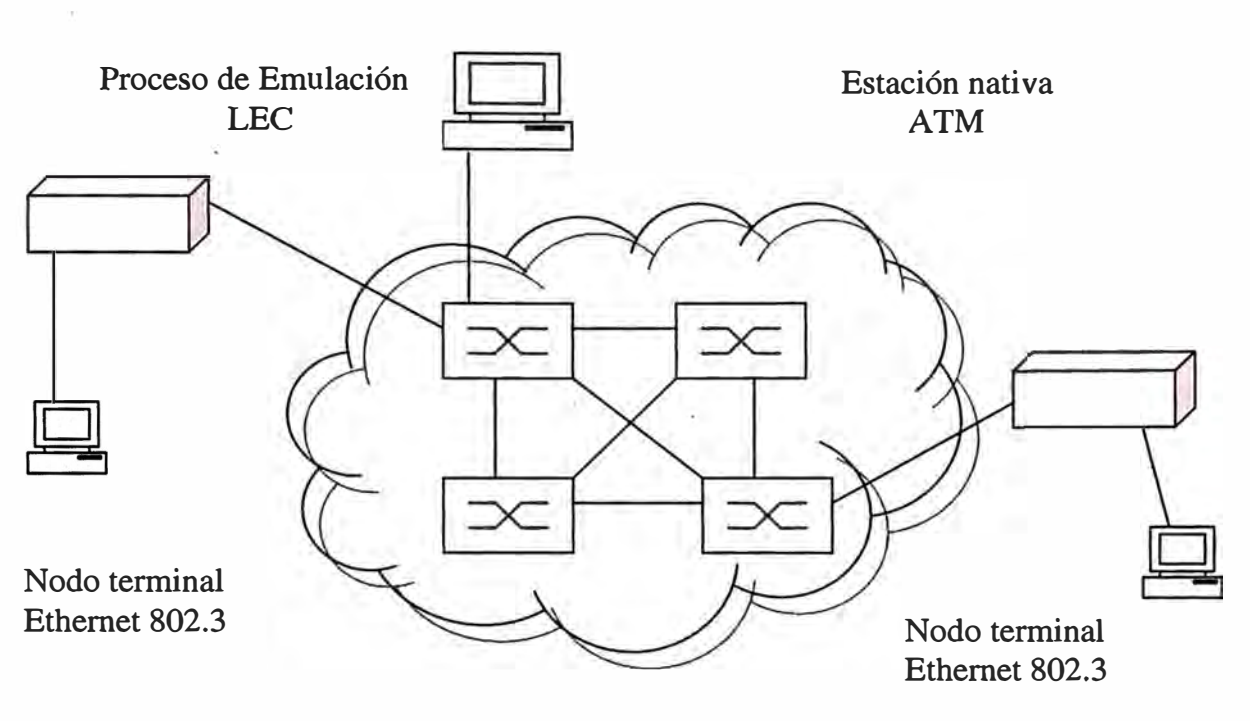


Figura 5.1 Comunicación a traves de conmutadores ATM.

Otro punto importante es el modo de conexión los protocolos LAN son sin conexión, ATM es orientasdo a la conexión entonces para poder emular una LAN debe llevar direcciones comopletas de la fuente y del destino en cada bloque transmitido. Esto presenta un problema dado el encabezado de 5 bytes de ATM, Además una tecnología orientada a la conexión no soporta de manera adecuada una comunicación bidireccional o transmisiones multidifusiones que son inherentes en una LAN para desarrollar estos problemas se ha diseñado una UNI especial para las LAN esta LAN UNI llamado LUNI permite a la ATM emular la naturaleza sin conexión de las LAN a traves de los conmutadores ATM.También se emplea la celda en bloques CIF que define un método para transportar protocolos ATM sobre LANlos CIF son una tecnología LAN que proporciona LAN con características ATM, incluyendo QoS y la integración sin parches de voz, datos y video.

La CIF extiende las conexiones virtuales de ATM a la estación de trabajo a traves de un dispositivo de unión CIF especial que proporciona una interface similar a la interface usuario a red del bloque de la ATM FUNI. Este dispositivo efectua la

mayor parte de la segmentación y reensamble SAR pero el nodo Ethernet debe construir el PDU de la capa ATM.

5.3 RFC de TCP/IP para ATM

La pila de protocolos de Internet sigue evolucionando mediante el mecanismo conocido como *RFC("Request For Comments")*. Los investigadores están diseñando e implementando nuevos protocolos(en su mayoría del nivel de aplicación), que se ponen en conocimiento de la comunidad de Internet en la forma de un RFC. (0)El RFC es descrito por el IAB("Internet Architecture Board"). La mayor fuente de RFCs es el IETF("Internet Engineering Task Force") que es una organización subsidiaria del IAB. Sin embargo, cualquiera puede enviar un informe propuesto como RFC al editor de los RFC. Hay una serie de normas que los autores de RFCs deben seguir para que su RFC sea aceptado. Estas reglas se describen en un RFC(RFC 1543) que además indica como enviar una propuesta de RFC.

Una vez que un RFC ha sido publicado, todas las revisiones y sustituciones se publican como nuevos RFCs. Se dice que un nuevo RFC que revisa o sustituye un RFC ya existente "actualiza" o "desfasa" a ese RFC. Asimismo, el RFC original es "actualizado" o "desfasado" por el nuevo. Por ejemplo, el RFC 1521 que describe el protocolo MIME es una "segunda edición", siendo una revisión del RFC 1341, y el RFC 1590 es una enmienda del 1521. Por tanto el RFC 1521 se etiqueta del modo siguiente: "Deja obsoleto al RFC 1341; Actualizado por el RFC 1590". En consecuencia, nunca hay confusión sobre si dos personas se refieren a dos versiones distintas de un RFC,

Algunos RFCs se califican como *documentos informativos* mientras que otros describen protocolos de Internet. El IAB("Internet Architecture Board") mantiene una lista de todos los RFCs que describen la pila de protocolos. A cada uno de ellos se le asigna un estado y un status.

Todo protocolo Internet puede tener uno de los siguientes estados

Estándar

> El IAB lo ha establecido como protocolo oficial de Internet. Se dividen en dos grupos:

1. El protocolo y superiores, protocolos que se aplican a la totalidad de Internet.
2. Protocolos específicos de redes, generalmente especificaciones del funcionamiento de IP en tipos concretos de redes.

Estándar provisional

El IAB está considerando activamente este protocolo como un posible protocolo estándar. Es deseable disponer de comentarios y pruebas exhaustivas cuantitativa y cualitativamente. Los comentarios y los resultados de las pruebas deberían enviarse al IAB. Existe la posibilidad de que se efectúen cambios en un protocolo estándar antes de que se convierta en estándar.

Propuesto como estándar.

Se trata propuestas de protocolos que el IAB puede considerar para la estandarización en el futuro. Es deseable evaluar la implementación y el testeo sobre un gran número grupos. Es probable que el protocolo se someta a revisión.

Experimental

Un sistema no debería implementar un protocolo experimental a menos que participe en el experimento y haya coordinado el uso que va a hacer del protocolo con el que lo ha desarrollado.

Informativo

Los protocolos desarrollados por otras organizaciones de estándares, o distribuidores, o aquellos que por otras razones son ajenos a los propósitos del IAB, pueden ser publicados a conveniencia de la comunidad de Internet como protocolos informativos. En algunos casos el IAB puede recomendar el uso de estos protocolos en Internet.

Histórico

Son protocolos con pocas posibilidades de convertirse alguna vez en estándar en Internet, bien porque han quedado desfasados por protocolos posteriores o debido a la falta de interés.

Definiciones de los status de los protocolos:

Requerido

Un sistema debe implementar los protocolos requeridos.

Recomendado

Un sistema debería implementar un protocolo recomendado.

Electivo

Un sistema puede o no implementar un protocolo electivo. La idea general es que si vas a implementarlo, debes hacerlo exactamente como se define.

Uso limitado.

Estos protocolos son usados en circunstancias específicas. Esto se puede deber a su estado experimental, naturaleza específica, funcionalidad limitada o estado histórico.

No recomendado.

Estos protocolos no se recomiendan para el uso general. Esto se puede deber a su limitada funcionalidad, naturaleza específica, o a que su estado es experimental o histórico.

Todos los RFCs están disponibles para el público, en forma de documento tanto impreso como electrónico, por medio del Internic("Internet Network Information Center"; internic.net). Antes de 1993, el DNN NIC(*nic.ddn.mil*) realizaba la función del NIC. Consultar el RFC 1400 para tener más información acerca de esta transición.

Los RFCs pueden conseguirse en forma impresa de:

Network Solutions, Inc.
Attn: InterNIC Registration Service
505 Huntmar Park Drive
Herndon, VA 22070
Help Desk Telephone Number:
703-742-4777
FAX Number 703-742-4811

Para conseguir el documento electrónico, los usuarios pueden hacer un FTP anónimo a ds.internic.net (198.49.45.10) y tomar los ficheros del directorio rfc, o un Gopher a internic.net (198.41.0.5).

Para información sobre otros métodos de acceder a RFCs vía E-mail o FTP, envía un E-mail a *"rfc-info@ISI.EDU"* con el mensaje *"help: ways_to_get_rfcs"*. Por ejemplo:

To: rfc-info@ISI.EDU

Subject: getting rfcs

help: ways_to_get_rfcs

Si tienes acceso a Internet, hay muchos sitios que mantienen archivos de RFCs. Uno que podrías probar es el *"MAGIC Document Archive"* en http://www.msci.magic.net/docs/rfc/rfc_by_num.html.

Los RFCs también se pueden obtener a través de la red IBM VNET usando el siguiente comando:

EXEC TOOLS SENDTO ALMVMA ARCNET RFC GET RFCnnnn TXT *

Donde nnnn es el número del RFC.

Para conseguir una lista de todos los RFCs(y saber si están disponibles en formato TXT o postscript), usa el comando:

EXEC TOOLS SENDTO ALMVMA ARCNET RFC GET RFCINDEX TXT *

También están los archivos STDINDEX TXT y FYIINDEX TXT que listan aquellos RFCs que tienen un número ST o FYI.

Tabla 5.1: Estado, status y números STD actuales de protocolos importantes de Internet

Protocol	Name	State	Status	STD
IP	Internet Protocol	Std.	Req.	5
ICMP	Internet Control Message Protocol	Std.	Req.	5
UDP	User Datagram Protocol	Std.	Rec.	6
TCP	Transmission Control Protocol	Std.	Rec.	7
TELNET	TELNET Protocol	Std.	Rec.	8
FTP	File Transfer Protocol	Std.	Rec.	9
SMTP	Simple Mail Transfer Protocol	Std.	Rec.	10
MAIL	Format of Electronic Mail Messages	Std.	Rec.	11
DOMAIN	Domain Name System	Std.	Rec.	13
DNS-MX	Mail Routing and the Domain System	Std.	Rec.	14
MIME	Multipurpose Internet Mail Extensions	Draft	Ele.	
SNMP	Simple Network Management Protocol	Std.	Rec.	15
SMI	Structure of Management Information	Std.	Rec.	16
MIB-I	Management Information Base	Hist.	Not	
MIB-II	Management Information Base-II	Std.	Rec.	17
NETBIOS	NetBIOS Services Protocol	Std.	Ele.	19
TFTP	Trivial File Transfer Protocol	Std.	Ele.	33
RIP	Routing Information Protocol	Std.	Ele.	34
ARP	Address Resolution Protocol	Std.	Ele.	37
RARP	Reverse Address Resolution Protocol	Std.	Ele.	38
GGP	Gateway to Gateway Protocol	Hist.	Not	
BGP3	Border Gateway Protocol 3	Draft	Ele.	
OSPF2	Open Shortest Path First Protocol V2	Draft	Ele.	
IS-IS	OSI IS-IS for TCP/IP Dual Environments	Prop.	Ele.	
BOOTP	Bootstrap Protocol	Draft	Rec.	
GOPHER	The Internet Gopher Protocol	Info.		
SUN-NFS	Network File System Protocol	Info.		
SUN-RPC	Remote Procedure Call Protocol Version 2	Info.		

En el momento de escribir este documento, no hay ningún RFC asociado al protocolo de transferencia de hipertexto("HTTP") usado en implementaciones de la "World Wide Web". Sin embargo, el documento HyperText Transfer Protocol (HTTP) escrito por Tim Berners-Lee se puede obtener en ftp://info.cern.ch/pub/www/doc/http-spec.text.

Adicionalmente, los siguientes RFCs describen el URL("Uniform Resource Locator") y conceptos asociados a él:

- RFC 1630 - identificadores universales de recursos en WWW

- RFC 1737 - Requerimientos funcionales para los URN("Uniform Resource Names")

RFC 1738 - URL("Uniform Resource Locators")

5.4 Datos no ajustados a estándares RFC

Los diferentes organismos encargados del desarrollo y evolución de ATM como ATM Forum dictan una serie de estándares aplicables a esta tecnología tales como se muestra en la tabla 5.2

Tabla 5.2 Estándares de ATM Forum.

Technical Working Group	Approved Specifications	Specification	Approved Date
AIC/ATM-IP Collaboration (formerly LanE)	ATM-MPLS Network Interworking1.0	af-aic-0178.000. (101kb)	Aug, 2001
	LAN Emulation over ATM 1.0	af-lane-0021.000 (494kb)	Jan, 1995
	LAN Emulation Client Management Specification	af-lane-0038.000 (145kb)	Sep, 1995
	LANE 1.0 Addendum	af-lane-0050.000 (24kb)	Dec, 1995
	LANE Servers Management Spec v1.0	af-lane-0057.000 (201kb)	Mar, 1996
	LANE v2.0 LUNI Interface	af-lane-0084.000 (912kb)	July, 1997
	LAN Emulation Client Management Specific Specification Version 2.0	af-lane-0093.000 (218kb)	Oct, 1998
	Multi-Protocol Over ATM Specification v1.0	af-mpoa-0087.000 (321kb)	July, 1997
	Multi-Protocol Over ATM Version 1.0 MIB	af-mpoa-0092.000 (190kb)	July, 1998
	Multi-protocol Over ATM Specification, Version 1.1	af-mpoa-0114.000 (993kb)	May, 1999
	MPOA v1.1 Addendum on VPN Support	af-mpoa-0129.000 (170kb)	Oct, 1999
	MPOA Errata	af-aic-0160.000 (14kb)	November, 2000
	ATM-MPLS Network Interworking, Version 2.0	af-aic-0178.001 (460kb)	August, 2003
	ATM_MPLS Network Interworking (N-to-one mode) Version 1.0	af-aic-0196.000 (75kb)	October, 2003
Architecture	ATM User-Network Interwork Interface (UNI) Specification Version 4.1	af-arch-0193.000 (84kb)	November, 2002
	AAL1 Circuit Emulation over Packet Switched Networks	af-arch-0204.000 (600kb)	January, 2005

B-ICI	B-ICI 1.0	af-bici-0013.000 (426kb)	Sep, 1993
	B-ICI 1.1	af-bici-0013.001 (972kb)	Sep, 1994
	B-ICI 2.0 (delta spec to B-ICI 1.1)	af-bici-0013.002 (553kb)	November, 1995
	B-ICI 2.0 (integrated specification)	af-bici-0013.003 (879kb)	Dec, 1995
	B-ICI 2.0 Addendum or 2.1	af-bici-0068.000 (58kb)	Nov, 1996
Control Signalling	PNNI Addendum on PNNI/B-QSIG Interworking and Generic Functional Protocol for the Support of Supplementary Services	af-cs-0102.000 (376kb)	Oct, 1998
	Addressing Addendum for UNI Signalling 4.0	af-cs-0107.000 (20kb)	Feb, 1999
	PNNI Transported Address Stack, Version 1.0	af-cs-0115.000 (98kb)	May, 1999
	PNNI Version 1.0 Security Signaling Addendum	af-cs-0116.000 (23kb)	May, 1999
	UNI Signaling 4.0 Security Addendum	af-cs-0117.000 (28kb)	May, 1999
	ATM Inter-Network Interface (AINI) Specification	af-cs-0125.000 (331kb)	July, 1999
	ATM Inter-Network Interface (AINI) Specification Version 1.1	af-cs-0125.001 (284kb) **Contents are identical to af-cs-0125.002 with revision marks**	September, 2002
	ATM Inter-Network Interface (AINI) Specification Version 1.1	af-cs-0125.002 (248kb) **Contents are identical to af-cs-0125.001 without revision marks**	September, 2002
	PNNI Addendum for Generic Application Transport Version 1.0	af-cs-0126.000 (86kb)	July, 1999
	PNNI SPVC Addendum Version 1.0	af-cs-0127.000 (93kb)	July, 1999
	PHY/MAC Identifier Addendum to UNI Signalling 4.0	af-cs-0135.000 (124kb)	Nov, 1999
	Network Call Correlation Identifier v1.0	af-cs-0140.000 (291kb)	March, 2000
	PNNI Addendum for Path and Connection Trace, Version 1.0	af-cs-0141.000. (506kb)	March, 2000

Operation of the Bearer Independent call control (BICC) Protocol with SIG 4.0/PNNI 1.0-AINI	af-cs-vmoa-0146.000 (87kb)	July, 2000
UBR with MDCR Addendum to UNI 4.0/PNNI 1.0 AINI	af-cs-0147.000 (232kb)	July, 2000
Modification of Traffic Descriptor for an Active Connection	af-cs-0148.000 (179kb)	July, 2000
Modification of Traffic Paramenters for an Active Connection Signalling Specification (PNNI, AINI, and UNI) Version 2.0	af-cs-0148.001 (172kb)	May, 2001
Behavior Class Selector Signalling Version 1.0	af-cs-0159.000 (222kb)	October, 2000
Guaranteed Frame Rate(GFR) Signalling(PNNI,AINI, and UNI) Version 1.0	af-cs-0167.000 (112kb)	August, 2001
Domain-based rerouting for active point-to-point calls, Version 1.0	af-cs-0173.000 (656kb)	August, 2001
Loop Detection, Version 1.0	af-cs-0176.000 (88kb)	April, 2002
Signalling Congestion Control, Version 1.0	af-cs-0181.000 (142kb)	April, 2002
Call Processing, Version 1.0	af-cs-0182.000 (201kb)	April, 2002
Private Network-Network Interface Specification v.1.1	af-pnni-0055.001 (2.4mb) **Contents are identical to af-pnni-0055.002 with revision marks**	April, 2002
Private Network-Network Interface Specification v.1.1	af-pnni-0055.002 (2.2mb) April, 2002Contents are identical to af-pnni-005.001 without revision marks Text versions of MIB documents below: PNNI-MIB (214kb) SPVC-MIB (48kb)	
ATM User Network Interface (UNI) Signalling Specification version 4.1	af-sig-0061.001 436(kb) **Identical to af-sig-0061.002 with revision marks**	April, 2002
ATM User Network Interface(UNI) Signalling Specification Version 4.1	af-sig-0061.002 (351kb) **Identical to af-sig-0061.001 without revision marks**	April, 2002

	Policy Routing, Version 1.0	af-cs-0195.000 (652kb)	April, 2003
	ATM-MPLS Network Interworking Signalling Specification 1.0	af-cs-0197.000 360(kb)	August, 2003
	PNNI Addendum for Path and Connection Trace Version 1.1 (PACT 1.1)	af-cs-0141.001 420(kb)	February, 2004
	Addendum to Policy Routing V1.0 for a Policy Constraint MIB	af-cs-0198.000 42(kb)	February, 2004
Data Exchange Interface	Data Exchange Interface version 1.0	af-dxi-0014.000 (98kb)	Aug, 1993
Directory and Naming Services	ATM Named System v2.0	af-dans-0152.000 (208kb)	July, 2000
Frame-based ATM	Frame-based ATM Transport over Ethernet (FATE)	af-fbatm-0139.000 (66kb)	Mar, 2000
	Frame-based ATM Transport over Ethernet (FATE)	af-fbatm-0139.001 (44kb)	July, 2002
	Frame Based ATM over Sonet/SDH	af-fbatm-0151.000	July, 2000
ILMI (Integrated Local Mgmt. Interface)	ILMI 4.0	af-ilmi-0065.000 (281kb)	Sep, 1996
Network Management	Customer Network Management (CNM) for ATM Public Network Service	af-nm-0019.000 (51kb)	Oct, 1994
	M4 Interface Requirements and Logical MIB	af-nm-0020.000 (233kb)	Oct, 1994
	M4 Interface Requirements and Logical MIB: ATM Network Element View	af-nm-0020.001 (838kb)	Oct, 1998
	CMIP Specification for the M4 Interface	af-nm-0027.000 (164kb)	Sep, 1995
	CMIP Specification for the M4 Interface: ATM Network Element View, Version 2	af-nm-0027.001 (726kb)	July, 1999
	M4 Public Network view	af-nm-0058.000 (344kb)	Mar, 1996
	M4 Interface Requirements and Logical MIB: ATM Network View, Version 2	af-nm-0058.001 (1.0mb)	May, 1999
	M4 "NE View"	af-nm-0071.000 (133kb)	Jan, 1997
	Circuit Emulation Service Interworking Requirements, Logical and CMIP MIB	af-nm-0072.000 (51kb)	Jan, 1997
	M4 Network View CMIP MIB Spec v1.0	af-nm-0073.000 (172kb)	Jan, 1997
	M4 Network View Requirements & Logical MIB Addendum	af-nm-0074.000 (74kb)	Jan, 1997

	ATM Remote Monitoring SNMP MIB	af-nm-test-0080.000 (81kb)	July, 1997
	SNMP M4 Network Element View MIB	af-nm-0095.001 (198kb)	July, 1998
	Network Management M4 Security Requirements and Logical MIB	af-nm-0103.000 (203kb)	Jan, 1999
	Auto-configuration of PVCs	af-nm-0122.000 (68kb)	May, 1999
	Requirements and Logical MIB for Management of Path and Connection Trace	af-nm-0153.000 (81kb)	April, 2001
	ATM Usage Measurement Requirements	af-nm-0154.000 (390kb)	November, 2000
	Addendum to te ILMI Autoconfiguration Extension	af-nm-0165.000 (299kb) MIB document (57kb)	April, 2002
	M4 Interface: ATM Network View, CORBA MIB, Version 2	af-nm-0185.000 (425kb)	August, 2002
	ATM Performance Management Bulk Data File Structure	af-nm-0194.000 (58kb)	April, 2003
Physical Layer	Issued as part of UNI 3.1: 44.736 DS3 Mbps Physical Layer 100 Mbps Multimode Fiber Interface Physical Layer 155.52 Mbps SONET STS-3c Physical Layer 155.52 Mbps Physical Layer	af-uni-0010.002 (1.6mb) **Currently Available to Members Only**	
	ATM Physical Medium Dependent Interface Specification for 155 Mb/s over Twisted Pair Cable	af-phy-0015.000 (74kb)	Sep, 1994
	DS1 Physical Layer Specification	af-phy-0016.000 (17kb)	Sep, 1994
	Utopia	af-phy-0017.000 (45kb)	Mar, 1994
	Mid-range Physical Layer Specification for Category 3 UTP	af-phy-0018.000 (121kb)	Sep, 1994
	6,312 Kbps UNI Specification	af-phy-0029.000 (22kb)	June, 1995
	E3 UNI	af-phy-0034.000 (23kb)	Aug, 1995
	Utopia Level 2	af-phy-0039.000 (986kb)	June, 1995
	Physical Interface Specification for 25.6 Mb/s over Twisted Pair	af-phy-0040.000 (199kb)	Nov, 1995
	A Cell-based Transmission Convergence Sublayer for Clear Channel Interfaces	af-phy-0043.000 (14kb)	Jan, 1996
	622.08 Mbps Physical Layer	af-phy-0046.000 (79kb)	Jan, 1996
	155.52 Mbps Physical Layer Specification for Category 3 UTP (See also UNI 3.1, af-uni-0010.002)	af-phy-0047.000 (124kb)	Nov, 1995

	120 Ohm Addendum to ATM PMD Interface Spec for 155 Mbps over TP	af-phy-0053.000 (14kb)	Jan, 1996
	DS3 Physical Layer Interface Spec	af-phy-0054.000 (41kb)	Mar, 1996
	155 Mbps over MMF Short Wave Length Lasers, Addendum to UNI 3.1	af-phy-0062.000 (99kb)	July, 1996
	WIRE (PMD to TC layers)	af-phy-0063.000 (46kb)	July, 1996
	E-1 Physical Layer Interface Specification	af-phy-0064.000 (41kb)	Sep, 1996
	155 Mbps over Plastic Optical Fiber (POF) Version 1.0	af-phy-0079.000 (40kb)	May, 1997
	155 Mb/s Plastic Optical Fiber and Hard Polymer Clad Fiber PMD Specification Version 1.1	af-phy-0079.001 (96kb)	Jan, 1999
	Inverse ATM Mux Version 1.0	af-phy-0086.000 (406kb)	July, 1997
	Inverse Multiplexing for ATM (IMA) Specification Version 1.1	af-phy-0086.001 (943kb)	March, 1999
	Physical Layer High Density Glass Optical Fiber Annex	af-phy-0110.000 (108kb)	Feb, 1999
	622 and 2488 Mbit/s Cell-Based Physical Layer	af-phy-0128.000 (182kb)	July, 1999
	ATM on Fractional E1/T1	af-phy-0130.000 (25kb)	Oct, 1999
	2.4 Gbps Physical Layer Specification	af-phy-0133.000 (62kb)	Oct, 1999
	2.5 and 10 Gbps Physical Layer Specification	af-phy-0133.001 (303kb)	Oct, 2003
	Physical Layer Control	af-phy-0134.000 (150kb)	Oct, 1999
	Utopia 3 Physical Layer Interface	af-phy-0136.000 (1.3mb)	Nov, 1999
	Specification of the Device Control Protocol (DCP) Version 1.0	af-phy-0138.000 (223kb)	Mar, 2000
	Multiplexed Status Mode (MSM3)	af-phy-0142.000 (120kb)	March, 2000
	Frame-Based ATM Interface (Level 3)	af-phy-0143.000 (175kb)	March, 2000
	UTOPIA Level 4	af-phy-0144.001 (173kb)	March, 2000
	Cell-Based 1000 Mbit/s (CB1G)Physical Layer Specification over Single-mode or Multi-mode Fiber and Category 6 Twisted pair Copper Cabling	af-phy-0162.000 (192kb)	April, 2001
P-NNI	Interim Inter-Switch Signaling Protocol	af-pnni-0026.000 (127kb)	Dec, 1994
	Private Network-Network Interface Specification V. 1.0	af-pnni-0055.000 (1.3mb)	Mar, 1996

	PNNI 1.0 Addendum (soft PVC MIB)	af-pnni-0066.000 (43kb)	Sep, 1996
	PNNI ABR Addendum	af-pnni-0075.000 (8kb)	Jan, 1997
	PNNI v1.0 Errata and PICs	af-pnni-0081.000 (1.5mb)	July, 1997
	(See PNNI 1.1, af-pnni-0055.001, af-pnni-0055.002)		
Routing and Addressing	PNNI Augmented Routing (PAR) Version 1.0	af-ra-0104.000 (243kb)	Jan, 1999
	ATM Forum Addressing: User Guide Version 1.0	af-ra-0105.000 (115kb)	Jan, 1999
	ATM Forum Addressing: Reference Guide	af-ra-0106.000 (152kb)	Feb, 1999
	PNNI Addendum for Mobility Extensions Version 1.0	af-ra-0123.000 (232kb)	May, 1999
	ATM Bi-Level Addressing Document, Version 1.0	af-ra-0164.000 (176kb)	April, 2001
	Addendum to PNNI, Version 1.0-Secure Routing	af-ra-0171.000 (65kb)	Nov, 2001
Residential Broadband	Residential Broadband Architectural Framework	af-rbb-0099.000 (161kb)	July, 1998
	RBB Physical Interfaces Specification	af-rbb-phy-0101.000 (250kb)	Jan, 1999
Service Aspects and Applications	Frame UNI	af-saa-0031.000 (57kb)	Sep, 1995
	Circuit Emulation	af-saa-0032.000 (142kb)	Sep, 1995
	Native ATM Services: Semantic Description	af-saa-0048.000 (554kb)	Feb, 1996
	Audio/Visual Multimedia Services: Video on Demand v1.0	af-saa-0049.000 (168kb)	Jan, 1996
	Audio/Visual Multimedia Services: Video on Demand v1.1	af-saa-0049.001 (237kb)	Mar, 1997
	ATM Names Service	af-saa-0069.000 (80kb)	Nov, 1996
	FUNI 2.0	af-saa-0088.000 (160kb)	July, 1997
	Native ATM Services DLPI Addendum Version 1.0	af-saa-api-dlpi-0091.000 (133kb)	February, 1998
	API Semantics for Native ATM Services	af-saa-0108.000 (566kb)	Feb, 1999
	FUNI Extensions for Multimedia	af-saa-0109.000 (51kb)	Feb, 1999
	H.323 Media Transport over ATM	af-saa-0124.000 (319kb)	July, 1999

Security	ATM Security Framework Version 1.0	af-sec-0096.000 (71kb)	February, 1998
	ATM Security Specification Version 1.0	af-sec-0100.001 (1.6mb)	Feb, 1999
	ATM Security Specification Version 1.1	af-sec-0100.002 (1.3mb)	March, 2001
	Security Specification Version 1.1 Protocol Implementation Conformance Statement (PICS) Proforma Specification	af-sec-0163.000.pdf (49kb)	March, 2001
	Control Plane Security	af-sec-0172.pdf (108kb)	Nov, 2001
	Methods of Securely Managing ATM Network Elements-Implementation Agreements,Version 1.1	af-sec-0179.000 (60kb)	April, 2002
	Security Services Renegotation Addendum to Security, Verson 1.1	af-sec-0180.000 (255kb)	March, 2002
	Addendum to Security Specification v1.1 – In-Band Security for Simplex Connections	af-sec-0187.000 (197kb)	August, 2002
	ATM Connection Filtering MIB and Audit Log	af-sec-0188.000 (69kb)	July, 2002
	Addendum to Sec 1.1 Secure CBR Traffic in a Policed Network	af-sec-0189.000 (21kb)	July, 2002
Signaling	(See UNI 3.1, af-uni-0010.002)		
	UNI Signalling 4.0	af-sig.0061.000 (470kb)	July, 1996
	Signaling ABR Addendum	af-sig-0076.000 (8kb)	Jan, 1997
	(See UNI 4.1, af-sig-0061.001, af-sig-0061.002)		
Testing			
	Introduction to ATM Forum Test Specifications	af-test-0022.000 (48kb)	Dec, 1994
	PICS Proforma for the DS3 Physical Layer Interface	af-test-0023.000 (25kb)	Sep, 1994
	PICS Proforma for the SONET STS-3c Physical Layer Interface	af-test-0024.000 (28kb)	Sep, 1994
	PICS Proforma for the 100 Mbps Multimode Fibre Physical Layer Interface	af-test-0025.000 (23kb)	Sep, 1994
	PICS Proforma for the ATM Layer (UNI 3.0)	af-test-0028.000 (48kb)	Apr, 1995
	Conformance Abstract Test Suite for the ATM Layer for Intermediate Systems (UNI 3.0)	af-test-0030.000 (158kb)	Sep, 1995
	Interoperability Test Suite for the ATM Layer (UNI 3.0)	af-test-0035.000 (.mp version)	Apr, 1995

Interoperability Test Suites for Physical Layer: DS-3, STS-3c, 100 Mbps MMF (TAXI)	af-test-0036.000 (63kb)	Apr, 1995
PICS Proforma for the DS1 Physical Layer	af-test-0037.000 (23kb)	Apr, 1995
Conformance Abstract Test Suite for the ATM Layer (End Systems) UNI 3.0	af-test-0041.000 (154kb) (.mp version) (128kb)	Jan, 1996
PICS for AAL5 (ITU spec)	af-test-0042.000 (38kb)	Jan, 1996
PICS Proforma for the 51.84 Mbps Mid-Range PHY Layer Interface	af-test-0044.000 (37kb)	Jan, 1996
Conformance Abstract Test Suite for the ATM Layer of Intermediate Systems (UNI 3.1)	af-test-0045.000 (139kb) (.mp version) (136kb)	Jan, 1996
PICS for the 25.6 Mbps over Twisted Pair Cable (UTP-3) Physical Layer	af-test-0051.000 (30kb)	Mar, 1996
Conformance Abstract Test Suite for the ATM Adaptation Layer (AAL) Type 5 Common Part (Part 1)	af-test-0052.000 (79kb) (.mp version) (63kb)	Mar, 1996
PICS for ATM Layer (UNI 3.1)	af-test-0059.000 (47kb)	July, 1996
Conformance Abstract Test Suite for the UNI 3.1 ATM Layer of End Systems	af-test-0060.000 (121kb) (.mp version) (120kb)	June, 1996
Conformance Abstract Test Suite for the SSCOP Sub-layer (UNI 3.1)	af-test-0067.000 (539kb) (.mp version) (741kb)	Sep, 1996
SSCOP Conformance Abstract Test Suite, Version 1.1	af-test-0067.001 (539kb) (.mp version) (777kb)	May, 1999
PICS for the 155 Mbps over Twisted Pair Cable (UTP-5/STP-5) Physical Layer	af-test-0070.000 (68kb)	Nov, 1996
PICS for Direct Mapped DS3	af-test-0082.000 (31kb)	July, 1997
Abstract Test Suite for Signalling (UNI 3.1) for the Network Side	af-test-0090.000 (4.8mb) (.mp version) (3.9mb)	September, 1997
ATM Test Access Function (ATAF) Specification Version 1.0	af-test-nm-0094.000 (99kb)	February, 1998

	PICS for Signalling (UNI v3.1) – User Side	af-test-0097.000 (132kb)	April, 1998
	Interoperability Test for PNNI Version 1.0	af-test-csra-0111.000 (425kb)	Feb, 1999
	PICS Proforma for UNI 3.1 Signalling (Network Side)	af-test-csra-0118.000 (313kb)	May, 1999
	ATM Forum Performance Testing Specification	af-test-tm-0131.000 (653kb)	Oct, 1999
	Implementation Conformance Statement (ICS) Proforma Style Guide	af-test-0137.000 (122kb)	Mar, 2000
	Conformance ATS for PNNI Routing	af-test-0155.000 (905kb) (.mp version) (1.5mb)	October, 2000
	Conformance ATS for PNNI Signalling	af-test-0156.000 (1.5m) (.mp version) (2.0mb)	October, 2000
	Conformance ATS for ABR Source and Destination Behaviors	af-test-tm-0157.000 (572kb) (.mp version) (90kb)	January, 2001
	UNI Signalling Performance Test Suite	af-test-0158.000 (601kb)	October, 2000
	ATM Inter-Network Interface (AINI) Specification	af-test-0158.001 (2,5mb)	July, 2002
	Introduction to ATM Forum Test Specifications, Version 2.0	af-test-0177.000 (97kb)	Oct, 2001
	Abstract Test Suite for UNI 3.1 Network side	af-test-0090.002 (10.0mb)	August, 2002
	Abstract Test Suite for UNI 4.0 Network Side	af-test-0190.000 (22.8mb) (.mp zip version) (225kb) (.fm version) (651kb)	October, 2002
Traffic Management	(See UNI 3.1, af-uni-0010.002)		
	Traffic Management 4.0	af-tm-0056.000 (454kb)	Apr, 1996
	Traffic Management ABR Addendum	af-tm-0077.000 (8kb)	Jan, 1997
	Traffic Management 4.1	af-tm-0121.000 (710kb)	March, 1999
	Addendum to TM 4.1:Differentiated UBR	af-tm-0149.000 (48kb)	July, 2000

	Addendum to Traffic Management v4.1 optional minimum desired cell rate indication for UBR	af-tm-0150.000 (33kb)	July, 2000
Voice & Telephony over ATM	Circuit Emulation Service 2.0	af-vtoa-0078.000 (306kb)	Jan, 1997
	Voice and Telephony Over ATM to the Desktop	af-vtoa-0083.000 (126kb)	May, 1997
	Voice and Telephony over ATM to the Desktop	af-vtoa-0083.001 (142kb)	Feb, 1999
	(DBCES) Dynamic Bandwith Utilization in 64 KBPS Time Slot Trunking Over ATM – Using CES	af-vtoa-0085.000 (80kb)	July, 1997
	ATM Trunking Using AAL1 for Narrow Band Services v1.0	af-vtoa-0089.000 (93kb)	July, 1997
	ATM Trunking Using AAL2 for Narrowband Services	af-vtoa-0113.000 (184kb)	Feb, 1999
	Low Speed Circuit Emulation Service	af-vtoa-0119.000 (62kb)	May, 1999
	ICS for ATM Trunking Using AAL2 for Narrowband Services	af-vtoa-0120.000 (136kb)	May, 1999
	Low Speed Circuit Emulation Service (LSCES) Implementation Conformance Statement Proformance	af-vtoa-0132.000 (59kb)	Oct, 1999
	Loop Emulation Service Using AAL2	af-vmoa-0145.000 (264kb)	July, 2000
	Loop Emulation Service Using AAL2 File Transfer Addendum	af-vmoa-0174.000 (104kb)	Oct, 2001
	Loop Emulation Service Using AAL2	af-vmoa-0175.000 (290kb)	Oct, 2001
	Loop Emulation Service Using AAL2 Rev 1	af-vmoa-0145.001 (386kb)	February, 2003
User-Network Interface (UNI)	ATM User-Network Interface Specification V2.0	af-uni-0010.000	June, 1993
	ATM User-Network Interface Specification V3.0	af-uni-0010.001 (1.9mb)	Sep, 1993
	ATM User-Network Interface Specification V3.1 (See UNI 3.1, af-uni-0010.002)		

5.5 Rendimiento y flexibilidad

Conmutación por hardware

ATM está diseñado de tal forma que se emplean simples elementos de lógica hardware en cada nodo para realizar la conmutación. En un enlace de 1 Gbps llega una nueva celda, y se transmite una celda cada 0.43 microsegundos. El tiempo de conmutación es mínimo.

Conexión virtual VC

ATM proporciona un entorno conmutado VC("Virtual Connection"). El VC se puede establecer bien a partir de un PVC("Permanent Virtual Connection") o de un SVC("Switched Virtual Connection") dinámico. La gestión de SVC hace con implementaciones del protocolo Q.93B.

Celdas

Todo tipo de información(voz, imágenes, vídeo, datos, etc.) se transporta a través de la red en bloques muy pequeños(48 bytes de datos más una cabecera de 5 bytes) llamados celdas.

Encaminamiento

El flujo de información se produce a lo largo de rutas(llamadas "canales virtuales") establecidas como una serie de punteros por la red. La cabecera de una celda contiene un identificador que vincula la celda al camino correcto que debe tomar para llegar a su destino.

Las celdas de un canal virtual particular siempre siguen el mismo camino y se entregan en el destino en el mismo orden en el que llegaron al canal.

Sean como sean las redes de la siguiente generación tendrán que ser rentables, las nuevas tecnologías están redefiniendo la curva de costes de la red la tecnología óptica dobla cada 9 meses su capacidad de transmisión mientras que los microprocesadores doblan su densidad cada 18 meses, la expectativa de una futura red híbrida opto electrónica es muy atractiva ya que para los operadores representa una oportunidad de bajar los costes drásticamente.

Además las redes deben construirse para generar ingresos sostenidos esto se basa en la generación de servicios básicos y de valor añadido fiables escalable y adaptada al cliente que aproveche mejor la curva de eficiencia de costes de las nuevas redes tanto de paquetes como ópticas ,no será sorpresa que la arquitectura multi servicio de ATM este en la mayoría de planes de los operadores para establecer la red de la próxima generación ésta probada tecnología proporciona en la actualidad el 40% de los ingresos por servicios generado por la red de operadores globales

La influencia de Internet y el dinámico mercado de las telecomunicaciones ha generado una gran demanda en la flexibilidad asi como un mayor control sobre

los costes, las redes de la próxima generación se van a basar en 4 conceptos fundamentales convergencia, migración, consolidación y simplificación.

La capa de servicio de la red del futuro será una red IP con capacidad multi servicio, la ubicuidad, simplicidad y estructura abierta de IP lo convierte en un catalizador natural para nuevas aplicaciones, servicios trafico y consecuentemente los nuevos ingresos. Los desarrollos de estándares de calidad de servicio (QoS) IP, la ingeniería de tráfico basado en la conmutación de etiquetas multi protocolo (MPLS) proporcionará las capacidades técnicas para ser viables estas redes IP multi servicio

La red núcleo esta evolucionando a lo que se llama red óptica de paquetes, la principal característica de esta evolución del coste por megabit (ancho de banda) en esta parte de la red.

Debido a la abundancia de Fibra óptica y los avances de la tecnología óptica, la capa de transporte del núcleo de la red se está convirtiendo en una red de conmutación óptica, es mas barato establecer capacidad de transporte en términos de fibras dedicadas (fibra oscura) o alternativamente con lamdas, que extraer anchos de banda de una red óptica sincronía tradicional (SDH).

La anterior proporciona un mayor ancho de banda (de varias ordenes de magnitud) es más flexible y rápida de establecer y mas sencilla de gestionar, los nuevos equipos del núcleo consistirán pues en nuevos equipos de multiplexación óptica densa (DWDM) y los crosconectores ópticos (OXC), Estos elementos están evolucionando rápidamente hacia un sistema de conmutación óptico puro en la que una trayecto óptico lamda entre 2 nodos puede ser establecido o eliminado automáticamente de forma similar a un trayecto de conmutación de etiquetas (LSP) en una red MPLS

Aunque la red de transporte del núcleo se basará en la conmutación de lamdas con nodos de alta capacidad de canales seguirá la necesidad de servicios de menor capacidad, esto es especialmente cierto en la periferia de la red donde los servicios tradicionales de multiplexación en el tiempo TDM son todavía importantes para soportar las estructuras de red existentes y los servicios de líneas alquiladas el equipamiento SDH tendrá simplemente una nueva función en la periferia de la red óptica

Las redes IP-MPLS serán capaces de soportar tráfico tradicional multi servicio, la optimización de costes y el refinamiento de las tareas de operación han hecho que las redes tradicionales sean altamente rentables y fiables probablemente las redes IP-MPLS ofrecerán estos nuevos servicios sobre nuevos métodos de acceso tales como las redes privadas virtuales y telefonía en paquetes sobre línea de abonados digitales asimétrica ADSL y el Gigabit ethernet. Estos nuevos servicios irán extrayendo en forma gradual el tráfico de los nodos tradicionales.

Los operadores de red encaran grandes desafíos a la hora de construir nuevas redes de núcleo óptico, los productos y servicios IP estan todavía viendo la oportunidad de trabajar con portadora clásica.

Los productos de la próxima generación tendrán que introducir la portadora grande ya existente en conmutadores ATM, tales como gestión de tráfico, ingeniería de tráfico y una robustez de auténtica plataforma.

Cada vez mas la tecnología IP-MPLS está revalidando las cualidades de la tecnología ATM emulando muchas de sus características y funciones hasta tal punto que la convergencia IP-MPLS y ATM en el núcleo es ahora técnicamente factible uno de los primeros ejemplos de esta convergencia es la introducción de la mediación ATM sobre un núcleo de red IP-MPLS en el cual el tráfico ATM puede ser encapsulado de forma transparente a través del núcleo IP-MPLS usando túneles LSP. Ésta convergencia en el núcleo de red acomodará todo tipo de tráfico viejo y nuevo estando bien posesionado para proporcionar opciones de ahorro de coste significativo para los operadores de red que buscan la consolidación de redes troncales.

¿Dónde queda ATM? La tecnología ATM continua soportando muchas y muy amplias redes multi servicio en el mundo. Desde las redes criticas Frame Relay hasta la nueva generación de redes de voz sobre paquetes.

ATM ha demostrado ser la tecnología preferida en el mundo de los negocios y de los operadores de red, vendidos por su flexibilidad y fiabilidad los servicios basados en ATM todavía no tienen equivalentes fiables en las redes IP actuales. Pasarán algunos años antes que los operadores de red puedan preferir redes IP a los servicios de datos ATM y frame relay.

Los proveedores de servicios ATM tienen 2 razones para para revolucionar sus redes.

- Escalar el núcleo ATM existente integrándolo directamente con el nuevo núcleo de conmutación óptica en vez de multiplexar sobre el núcleo de transporte SDH, mas caro.
- El núcleo ATM puede ser superpuesto sobre el núcleo óptico IP-MPLS usando técnicas de mediación ATM /IP-MPLS , para consolidar el ancho de banda de ambas redes en un solo núcleo, es muy atractivo para operadores con núcleo IP con rápida expansión

La evolución de las redes ATM a permitido consolidar diferentes tipos de redes a bajo costo, la demanda explosiva de nuevos servicios, la exigencia del aumento de ancho de banda además de proteger la fuente de ingresos de los servicios actuales de voz y datos donde escasea el financiamiento, el objetivo es construir fuentes de ingresos de alta rentabilidad, el mensaje clave es RESULTADOS.

Con respecto a la calidad de servicio (QoS, *Quality of Service*). Protocolo de reserva de recursos (RSVP, *Resource Reservation Protocol*), Servicios diferenciados (DiffServ, *Differentiated Services*), IEEE 802.1p, QoS de ATM, etc

La API de GQoS es una extensión de la interfaz de programación Winsock Incluye API y componentes del sistema que proporcionan a las aplicaciones un método para reservar ancho de banda de red entre el cliente y el servidor mecanismos QoS como RSVP, Diffserv, 802.1p o QoS de ATM. RSVP es un protocolo de señalización de capa 3 que se utiliza para reservar ancho de banda para flujos individuales en una red. RSVP es un mecanismo QoS por flujo porque configura una reserva para cada flujo. Diffserv es otro mecanismo QoS de capa 3. Diffserv define 6 bits del encabezado IP que determinan cómo se da prioridad a los paquetes IP[3]. Al tráfico Diffserv se puede dar prioridad en 64 clases posibles denominadas Comportamientos por salto (PHB, *Per Hop Behaviors*). Por otro lado, 802.1p es un mecanismo QoS de capa 2 que define cómo deben dar prioridad al tráfico los dispositivos de capa 2 como los conmutadores Ethernet. 802.1p define 8 clases de prioridad entre 0 y 7. DiffServ y 802.1p se denominan mecanismos QoS agregados porque clasifican el tráfico en un número finito de clases de prioridad.

La secuencia de sucesos siguiente caracteriza la interacción de una aplicación con GQoS:

1. La aplicación solicita QoS en términos abstractos mediante GQoS.
2. La solicitud de la aplicación se traduce a mensajes de señalización RSVP. Los mensajes de señalización RSVP salen a la red y reservan ancho de banda en todos los nodos habilitados para RSVP en la ruta de acceso de red.
3. Además de configurar las reservas, los mensajes RSVP están sometidos al análisis de los servidores de directivas de la red. Los servidores de directivas pueden rechazar la solicitud RSVP si infringe la directiva de la red. Esto da al administrador de la red un medio para decidir quién obtiene QoS.
4. Una vez que se ha instalado la reserva RSVP, Windows 2000 comienza a marcar todos los paquetes salientes de ese flujo con la clase DiffServ correcta y la prioridad 802.1p.
5. Cuando el tráfico del flujo atraviesa la red, recibe la ventaja del orden de prioridad 802.1p en los conmutadores Ethernet habilitados para 802.1p, la ventaja de las reservas RSVP en los enrutadores habilitados para RSVP y las ventajas del orden de prioridad DiffServ en las nubes preparadas para DiffServ de la red.

Hay varios otros mecanismos QoS, como Servicios integrados sobre ATM (ISATM, *Integrated Services over ATM*) que asigna automáticamente las solicitudes GQoS a QoS de ATM en Redes IP sobre ATM clásicas. Servicios integrados con una tasa de bits baja (ISSLOW, Integrated Services Over Low Bit Rate) es otro mecanismo QoS que mejora la latencia del tráfico con prioridad en vínculos WAN lentos. Además de la API de GQoS, una aplicación de control o administración tiene acceso a la funcionalidad de control de tráfico mediante la API de Control de tráfico (TC, *Traffic Control*). La API de TC permite que una aplicación de control o administración ayude a proporcionar cierto grado de calidad de servicio para las aplicaciones que no están preparadas para QoS. Windows 2000 proporciona también un servidor de directivas denominado Servicio de control de admisión de QoS (QoS ACS, *QoS Admission Control Service*). El ACS de QoS permite que los administradores de red controlen quién recibe QoS en la red. El servicio ACS de QoS expone también una API

denominada API de Módulo de directiva local (LPM, *Local Policy Module*). La API de LPM permite a los fabricantes independientes de software (ISV) crear módulos de directivas personalizados que contribuyen a la funcionalidad de aplicación de directivas del ACS de QoS.

En la figura 5.2 se ilustran los componentes del sistema implicados en QoS y RSVP. GQoS es un proveedor de QoS que puede invocar señalización RSVP, desencadenar control de tráfico y proporcionar notificación de sucesos a la aplicación. **Rsvp.exe** es responsable de la señalización RSVP a o desde la red, y de llamar a la biblioteca **Traffic.dll** para agregar flujos y filtros a la pila. El clasificador de paquetes es responsable de clasificar los paquetes según los filtros de paquetes indicados por **Traffic.dll**. El programador de paquetes mantiene colas independientes para cada clasificación del tráfico e incluye un analizador de compatibilidad, un conformador y un secuenciador de paquetes. El conformador administra los flujos en las colas de paquetes a la velocidad acordada y el secuenciador envía paquetes a la interfaz de red en el orden de prioridad de las colas que administra. El tráfico que no tiene especificación QoS se coloca en la cola de mejor esfuerzo, que es la que tiene la prioridad más baja.

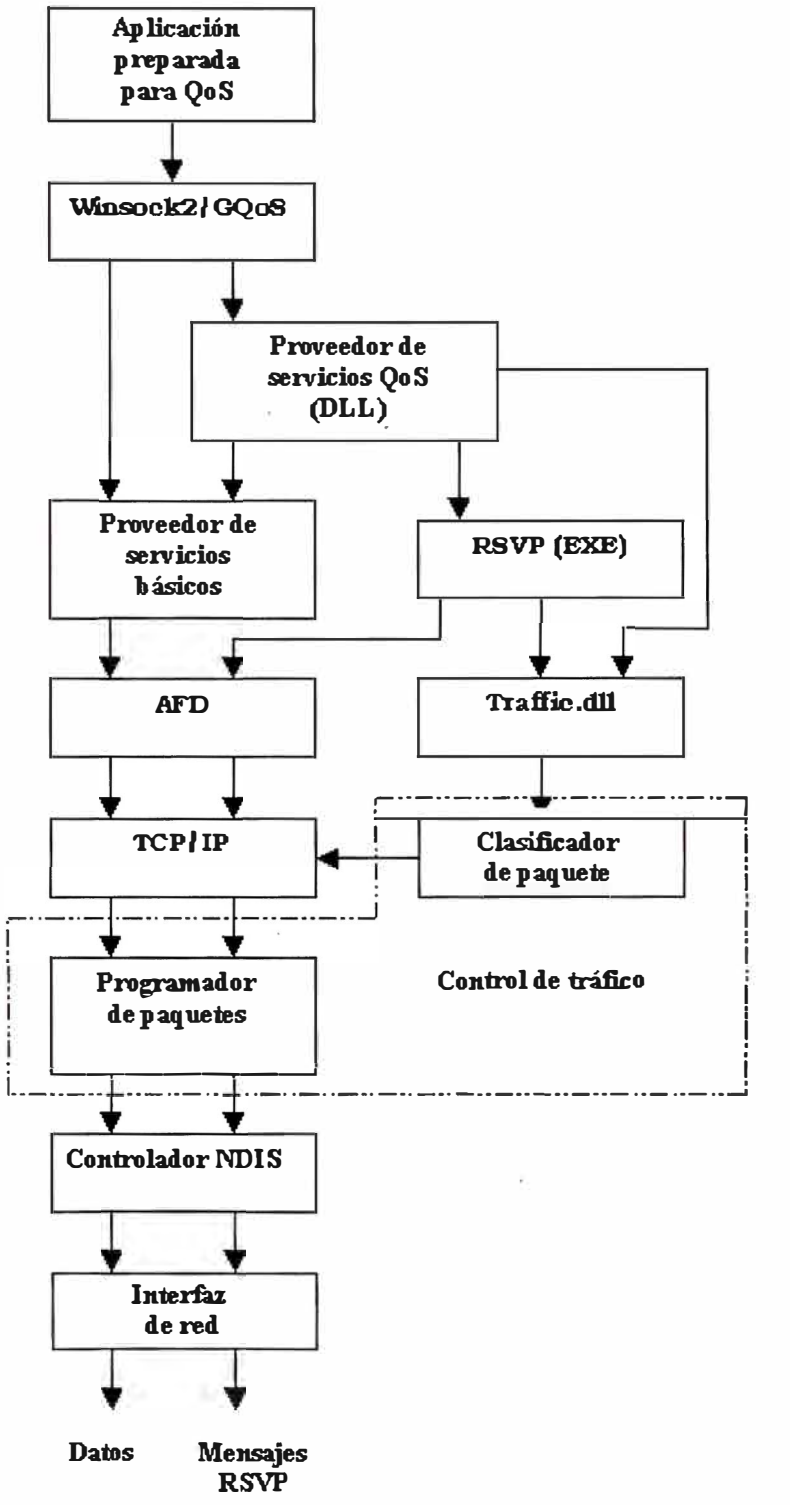


Figura 5.2 Aplicación RSVP de QoS.

En el diagrama de la figura se muestra cómo utiliza una aplicación RSVP de QoS para entregar un flujo de datos a uno o varios clientes. La aplicación es un servidor de audio y necesita 1 megabit por segundo de ancho de banda confiable para suministrar al cliente

una calidad de audio aceptable. RSVP admite flujos tanto de unidifusión como de multidifusión. Este ejemplo utiliza un flujo de unidifusión a un único cliente.

La aplicación se inicializa y completa una estructura que se debe proporcionar a GQoS. Esta estructura incluye una especificación de flujo de envío y recepción. Las especificaciones de flujo incluyen parámetros como el ancho de banda punta, la latencia, la variación de retardo, el tipo de servicio, etc

En Windows 2000 , la aplicación llama a **WSAConnect** para conectar con el cliente. La llamada a esta función desencadena una serie de sucesos. Se invoca a RSVP para enviar una señal a la red mediante el envío de mensajes especiales de ruta de ruta de acceso. Un mensaje de ruta de acceso se envía a la misma dirección IP de destino a la que se dirige el flujo, pero su objetivo es configurar los enrutadores del flujo e identificarlo. Un enrutador que recibe un mensaje de ruta de acceso inserta su propia dirección IP en el último salto del mensaje de ruta de acceso y reenvía el mensaje al siguiente enrutador de la ruta hasta que llega al cliente. Esto da al cliente la capacidad de entender la ruta entre el remitente y él mismo, y reservar ancho de banda a lo largo de esa ruta para la aplicación. El cliente devuelve una solicitud de reserva (que vuelve a describir el flujo deseado) por la misma ruta. Los enrutadores que se encuentran en la ruta tienen la responsabilidad de examinar los recursos disponibles para ellos y determinar si pueden aceptar la reserva. Si todos los enrutadores a lo largo de la ruta aceptan la reserva, la aplicación puede contar con disponer del ancho de banda de red deseado y otras características.

Como las redes son dinámicas y el servidor o el cliente podrían haber abandonado por error los recursos sin notificarlo a la red, se deben actualizar con frecuencia tanto los mensajes de ruta como las solicitudes de reserva. Si no hubo cambios en la red, los mensajes de ruta adicionales y las reservas actualizan sólo la ruta actual. No obstante, si aparece una nueva ruta, la ruta seguida por el flujo podría cambiar sobre la marcha cuando la red realice ajustes.

Cuando se utiliza una aplicación de servidor para hacer multidifusión a muchos clientes, ocurre una secuencia de sucesos similar. Una diferencia interesante es

que cuando los enrutadores reciben solicitudes de reserva de diversos clientes que hacen referencia al mismo flujo, pueden combinar las solicitudes de reserva en lugar de mantener reservas individuales para el mismo flujo de información.

CAPITULO VI
CONEXIONES AVANZADAS DE TCP/IP PARA ATM

6.1 Introducción

La única forma para que un protocolo de nivel superior se comunique sobre una red ATM es por medio de la capa ATM AAL("ATM Adaptation Layer"). La función de esta capa es realizar el mapeado entre las PDUs y las celdas. Hay cuatro tipos diferentes de AAL, AAL1, AAL2, AAL3/4 Y AAL5. Estos AALs ofrecen distintos servicios a los protocolos de nivel superior. Aquí se muestran las características de AAL5, usado para TCP/IP:

- Modo mensaje y modo flujo
- Entrega garantizada
- Entrega no garantizada(usada por TCP/IP)
- Fragmentación de los datos en bloques y segmentos
- Operación multipunto

AAl5 proporciona las mismas funciones que una LAN en el nivel MAC("Medium Access Control"). Los extremos del VC saben el tipo de AAL por medio del mecanismo de configuración de la celda, por lo que la cabecera de la celda no ha de llevarlo. Para los PVCs el tipo AAL se configura administrativamente en los extremos cuando se establece la conexión. ara los SVCs, el tipo de AAL se comunica por el canal vía 0.93B como parte de la solicitud de establecimiento y definición de la conexión y los extremos usan las señales de control para configurarse. Los conmutadores ATM no suelen preocuparse del tipo de AAL de los VCs. El formato AAL5 especifica un formato de paquete con un tamaño máximo de 64KB - 1 byte de usuario. Las "primitivas" que ha de usar el protocolo de nivel superior como interfaz con la capa AAL(en el SAP de AAL("Service

Access Point") están definidas rigurosamente. Cuando un protocolo de nivel superior envía datos, estos son procesados primero por la capa de adaptación, luego por ATM y por último la capa física se encarga de enviar los datos por la red ATM. Las celdas se transportan y las recibe el otro extremo de la conexión en su capa física, que las pasa a ATM, que tras procesarlas las pasa al AAL receptor, que a su vez devuelve los datos al nivel superior. La función total que ha realizado la red ATM ha sido un transporte no garantizado de información(se podría haber perdido una parte). Desde un punto de vista más conservador del proceso de datos, todo lo que ha hecho la red ATM ha sido sustituir un enlace físico por otro tipo de conexión física - todos los protocolos de alto nivel siguen teniendo que efectuarse(por ejemplo IEEE 802.2).

Direccionamiento

Una dirección ATM de un extremo de la conexión se codifica bien como una dirección de 20 bytes basada en OSI NSAP(utilizada para direccionamiento en redes privadas, con tres formatos posibles) o como una dirección E.164 Public UNI(del estilo de los números telefónicos, usados para redes TM públicas).

Broadcast, Multicast

En la actualidad no hay funciones de broadcast similares a las de las LANs. Pero sí existe una función de multicast. El término ATM para multicast es "conexión punto - multipunto".

6.2 Interfase y Funciones del NDIS

NDIS (, Network Driver Interface Specification) ofrece un conjunto de normas para la comunicación entre protocolos y controladores de tarjetas adaptadoras. Así, en cualquier estación de trabajo podrá utilizarse cualquier combinación de controladores de protocolo compatibles con NDIS junto con cualquier controlador de tarjeta adaptadora de red compatible con NDIS.

Es probable que las computadoras existentes en su red tengan distintos tipos de tarjetas adaptadoras de red, por lo que necesitará distintos controladores de tarjetas adaptadoras de red. Gracias al estándar NDIS, podrá utilizar exactamente el mismo controlador de protocolo en todas sus estaciones de trabajo, sin

necesidad de disponer de una versión diferente del protocolo para cada tarjeta adaptadora de red, como sucedería si utilizase pilas de protocolos monolíticas
Además, NDIS permite que varios protocolos utilicen una misma tarjeta de red. Normalmente, cuando se utiliza un protocolo monolítico con una tarjeta adaptadora de red, dicho protocolo monopoliza la tarjeta de red, impidiendo la utilización de otros protocolos con dicha tarjeta.
Cuando una computadora incorpora varios protocolos, la computadora transmite los datos utilizando primero un protocolo, después el siguiente protocolo y así sucesivamente. Cuando instale varios protocolos en una misma computadora, designará el orden en que la computadora los utilizará. El primer protocolo de esta serie suele conocerse como protocolo principal.
Cuando elija una tarjeta adaptadora de red, debe asegurarse de que la tarjeta elegida admita la arquitectura de su red (por ejemplo, Ethernet o Token-Ring) y su sistema de cableado (por ejemplo, coaxial delgado o par trenzado). Además de estos factores, debe tenerse en cuenta tanto la velocidad como el costo, así como los compromisos entre ambos parámetros.
En las tarjetas adaptadoras de red, la velocidad depende principalmente del ancho del bus y de la memoria que incorpore la tarjeta. El ancho del bus de una tarjeta de red es el número de contactos que se utilizan para conectar la tarjeta al bus de la computadora. Se obtendrá mayor rendimiento cuanto más se aproxime el ancho del bus de la tarjeta al ancho del bus interno de la computadora. La memoria incorporada en la propia tarjeta permite a ésta almacenar temporalmente las tramas que entran y salen por la red. Sin embargo, no siempre una tarjeta con más memoria constituye la opción óptima, ya que a partir de un cierto punto, las ventajas asociadas a la mayor cantidad de memoria disminuyen y es la velocidad máxima de otros componentes de la red lo que limita el rendimiento, impidiendo mejoras adicionales.
Algunas tarjetas incorporan también procesadores integrados (estas tarjetas suelen conocerse como tarjetas inteligentes).
Antes de invertir en un determinado tipo de tarjeta de red, asegúrese de que exista un controlador conforme al estándar NDIS para dicha tarjeta. Además, cerciórese de que el fabricante dispone de infraestructura suficiente para atender las necesidades de su empresa

Si está tratando con un distribuidor, asegúrese de que éste posea una vía de comunicación adecuada con el fabricante de la tarjeta.

El controlador de tarjeta de red se enlazará automáticamente a todos los protocolos NDIS que se estén ejecutando en ese momento en la computadora; si posteriormente se agregan otros protocolos, también quedarán enlazados de forma automática al controlador de la tarjeta de red.

Para la comunicación entre los niveles de sesiones y de transporte del modelo de referencia OSI, Microsoft ha desarrollado y admite la Interfaz de controlador de transporte (TDI).

Al igual que NDIS, TDI aumenta la versatilidad de conexión en red al permitir que distintos protocolos de transporte y componentes de red de niveles superiores (como el servidor y el redirector) puedan comunicarse a través de una interfaz común. Varios protocolos diferentes que se ajusten al estándar TDI podrán cooperar con distintos componentes de niveles superiores que también admitan TDI. Cuando un redirector o un servidor realice una llamada a un transporte, se utilizará la interfaz TDI para realizar la llamada, por lo que no será necesario conocer nada acerca de los protocolos de transporte que se estén utilizando.

Aunque TDI es ahora la interfaz de comunicación entre los protocolos de transporte y elementos de software de nivel superior como el redirector o el servidor

NDIS 3.1 admite servicios básicos que permiten a un módulo de protocolos enviar paquetes sin procesar a través de un dispositivo de red y que se notifique a ese mismo módulo acerca de los paquetes entrantes recibidos por un dispositivo de red.

En NDIS 4.0 se agregaron las siguientes características nuevas con respecto a NDIS 3.1:

- Compatibilidad con datos fuera de banda (requerido para PC de difusión)
- Extensión para medios WirelessWAN
- Envío y recepción de paquetes a gran velocidad (una mejora significativa en el rendimiento)
- Extensión para medios IrDA rápidos

- Detección de medios (requerido para el logotipo Diseñado para Windows en la Guía de diseño de hardware PC 97 y posteriores). La pila de TCP/IP en Microsoft Windows 2000 utiliza información de detección de medios, que se describe en la sección "Configuración automática del cliente" de este documento.
- Filtro de todos los paquetes locales (impide que Monitor de red monopolice la CPU)
- Numerosas funciones nuevas de sistema NDIS (requerido para la compatibilidad binaria de minipuertos en Windows 95, Windows 98, Windows NT y Windows 2000)

NDIS 5.0 incluye toda la funcionalidad definida en NDIS 4.0 más las extensiones siguientes:

- Administración de energía NDIS (requerido para Administración de energía de red y Activación de red)
- Plug and Play. (NDIS en Windows 95 ya admitía Plug and Play; por lo tanto, este cambio sólo se aplica a los controladores de red de Windows 2000.)
- Compatibilidad para Instrumentación de administración de Windows (WMI, *Windows Management Instrumentation*), que proporciona instrumentación compatible con Web-based Enterprise Management (WBEM) de minipuertos NDIS y sus adaptadores asociados
- Compatibilidad con un único formato INF en todos los sistemas operativos Windows. El nuevo formato INF está basado en el formato INF de Windows 98.
- Minipuerto deserializado para mejorar el rendimiento
- Mecanismos de descarga de tareas, como la suma de comprobación de TCP y UDP, y Reenvío rápido de paquetes
- Extensión para medios de difusión (necesario para Servicios de difusión para Windows)
- Compatibilidad con Calidad de servicio (QoS)

- Compatibilidad con controladores intermedios (requerido para PC de difusión, redes de área local virtuales, programación de paquetes para QoS y compatibilidad de NDIS para dispositivos de red IEEE 1394)

NDIS puede apagar los adaptadores de red cuando el sistema solicita un cambio en el nivel de energía. Esta solicitud puede iniciarla el usuario o el sistema. Por ejemplo, puede que el usuario desee poner el equipo en modo de reposo o que el sistema solicite un cambio en el nivel de energía en función de la inactividad del teclado o del *mouse* (ratón). Además, la desconexión del cable de red puede iniciar una solicitud de apagado si la tarjeta de interfaz de red (NIC) admite esta funcionalidad. En este caso, el sistema espera durante un período que se puede configurar antes de apagar la tarjeta de interfaz de red porque la desconexión podría deberse a cambios temporales en el cableado de la red y no a la desconexión de un cable del propio dispositivo de red.

La directiva de administración de energía de NDIS se basa en la *falta de actividad de red*. Esto significa que todos los componentes de red superpuestos deben aceptar la solicitud antes de que se pueda apagar la tarjeta de interfaz de red. Si hay sesiones activas o archivos abiertos en la red, cualquiera de los componentes implicados o todos ellos pueden rechazar la solicitud de apagado.

El equipo también puede activarse desde un estado de bajo consumo de energía en función de los sucesos de la red. Una señal de activación puede ser provocada por:

- La detección de un cambio en el estado de los vínculos de la red (por ejemplo, la reconexión de cables)
- La recepción de una trama de activación de red

Al inicializarse los controladores, NDIS consulta las funciones del minipuerto para determinar si admite elementos como el paquete Magic, coincidencia de modelos activación ante cambios en los vínculos, y para determinar el menor estado de energía requerido para cada método de activación. Después, los protocolos de red consultan las funciones del minipuerto. En el momento de la ejecución, el

protocolo establece la directiva de activación mediante identificadores de objeto (OID), como **Enable Wakeup**, **Set Packet Pattern** y **Remove Packet Pattern**.

Actualmente, Microsoft TCP/IP es la única pila de protocolos de Microsoft que permite la administración de energía de las redes. Registra los siguientes modelos de paquetes en la inicialización de los minipuertos:

- Paquete IP dirigido
- Difusión ARP para la dirección IP de la estación
- Difusión NetBIOS sobre TCP/IP para el nombre de equipo asignado de la estación

Hay disponibles controladores compatibles con NDIS para una amplia variedad de tarjetas de interfaz de red de muchos fabricantes. La interfaz NDIS permite que varios controladores de protocolo de distintos tipos enlacen a un único controlador de tarjeta de interfaz de red y permite que un único protocolo enlace a varios controladores de tarjetas. La especificación NDIS describe el mecanismo de multiplexado utilizado para lograrlo. Los enlaces se pueden ver o cambiar desde la carpeta Conexiones de red de Windows.

TCP/IP en Windows 2000 es compatible con:

- Ethernet (y SNAP 802.3)
- FDDI
- Token Ring (802.5)
- ATM (LANE y CLIP)
- ARCnet
- Vínculos de redes de área extensa (WAN) dedicados, como el Servicio de datáfono digital (DDS, *Dataphone Digital Service*) y portadora T (T1, T1 fraccionaria y T3)
- Servicios de redes de área extensa de acceso telefónico circuitos permanentes, como teléfono analógico, RDSI (ISDN) y xDSL
- Servicios WAN de conmutación de paquetes, como X.25, Frame Relay y ATM

Algunos objetivos de estas nuevas características son:

- Aumentar la facilidad de uso y reducir el costo total de propiedad
- Mejorar el rendimiento
- Habilitar nuevos tipos de medios, servicios y aplicaciones
- Mejorar la flexibilidad en la arquitectura de los controladores

La funcionalidad de la capa de vínculo está dividida entre la combinación de tarjeta de interfaz de red y controlador, y el controlador de la pila de protocolos de bajo nivel. Los filtros de la combinación de tarjeta de red y controlador se basan en la dirección de control de acceso al medio (MAC, *Media Access Control*) de destino de cada trama.

Normalmente, el hardware filtra todas las tramas entrantes excepto aquellas que contienen una de las direcciones de destino siguientes:

- La dirección del adaptador
- La dirección de difusión con todo unos (FF-FF-FF-FF-FF-FF)
- Las direcciones de multidifusión en las que un controlador de protocolo de este host ha registrado interés, mediante una primitiva NDIS

Como esta primera decisión de filtrado la toma el hardware, la tarjeta de interfaz de red descarta todas las tramas que no cumplan los criterios del filtro sin que la CPU realice ningún procesamiento. Todas las tramas (incluidas las difusiones) que atraviesan el filtro del hardware se pasan después hacia arriba al controlador de la tarjeta de interfaz de red a través de una interrupción de hardware[2]. El controlador de la tarjeta es software que se ejecuta en el equipo, por lo que las tramas que llegan hasta aquí requieren tiempo de la CPU para procesarlas. El controlador de la tarjeta de interfaz de red carga la trama en la memoria del sistema desde la tarjeta de interfaz. Después, la trama es dirigida (transmitida) al controlador o controladores de transporte enlazados correspondientes. La especificación NDIS 5.0 proporciona más detalles acerca de este proceso.

Las tramas se pasan hacia arriba a todos los controladores de transporte enlazados en el orden en que se enlazan.

Cuando un paquete atraviesa una red o una serie de redes, la dirección de control de acceso al medio de origen siempre es la de la tarjeta de interfaz de red que la colocó en el medio y la dirección de control de acceso al medio de destino es la de la tarjeta de interfaz de red que debe recogerla de allí. Esto significa que, en una red enrutada, las direcciones de control de acceso a los medios de origen y de destino cambian con cada salto a través de un dispositivo de capa de red (enrutador o conmutador de capa 3).

Cada tipo de medio tiene un tamaño máximo de trama que no se puede superar. La capa de vínculo es responsable de descubrir esta unidad máxima y comunicarla a los protocolos que están por encima. La pila de protocolos puede consultar a los controladores NDIS la unidad máxima de transmisión local. El conocimiento de la unidad de una interfaz la utilizan los protocolos de capas superiores, como TCP, que optimizan automáticamente el tamaño de los paquetes para cada medio

Si un controlador de tarjeta de interfaz de red, por ejemplo un controlador ATM, utiliza el modo de emulación de red de área local, puede informar de que tiene una unidad MTU mayor de lo que se espera para ese tipo de medio. Por ejemplo, puede emular Ethernet, pero informar de una unidad MTU de 9180 bytes. Windows NT y Windows 2000 aceptan y utilizan el tamaño de MTU comunicado por el adaptador, aunque sobrepase la MTU normal de un tipo de medio determinado.

En ocasiones, la MTU comunicada a la pila de protocolos puede ser menor de lo que se esperaría para un tipo de medio determinado. Por ejemplo, el uso del estándar 802.1p para QoS sobre Ethernet suele reducir (esto depende del hardware) en 4 bytes la MTU comunicada debido a encabezamientos mayores de la capa de vínculo.

Funciones

Administrador de energía

- Plug and Play
- Instrumental de administración de Windows
- Compatibilidad para el formato INF sencillo

Minipuertos paralelizados

Mecanismos de descarga

Filtro para todos los paquetes locales

NDIS4.1

NDIS5

6.3 NDIS orientado a la conexión

NDIS orientado a la conexión, requerido para admitir el Modo de transferencia asincrónica (ATM, *Asynchronous Transfer Mode*), Línea digital asimétrica de abonado (ADSL, *Asymmetric Digital Subscriber Line*) y Modelo de controladores de Windows–Arquitectura de secuencia de conexiones (WDM-CSA, *Windows* Driver Model–Connection Streaming Architecture)

6.4 Calculo de comprobación en TCP/IP

El tamaño de la ventana de recepción de TCP es la cantidad de datos de recepción (en bytes) que se pueden almacenar en el búfer a la vez en una conexión. El host remitente sólo puede enviar esa cantidad de datos antes de esperar confirmación y actualización de la ventana del host receptor. La pila de TCP/IP en Windows 2000 se diseñó para ajustarse a sí misma en la mayor parte de los entornos y utiliza tamaños de ventana mayores que las versiones anteriores. En lugar de utilizar un tamaño de ventana de recepción predeterminado no modificable, TCP se ajusta a incrementos homogéneos del tamaño de segmento máximo (MSS, *Maximum Segment Size*) negociado durante la configuración de la conexión. La coincidencia de la ventana de recepción con los incrementos homogéneos del tamaño de segmento máximo aumenta el porcentaje de segmentos TCP de tamaño total utilizados durante la transmisión de cantidades masivas de datos.

El tamaño de la ventana de recepción tiene un valor predeterminado que se calcula de la siguiente manera:

1. La primera solicitud de conexión enviada a un host remoto anuncia un tamaño de ventana de recepción de 16 KB (16384 bytes).

2. Después de establecer la conexión, el tamaño de la ventana de recepción se redondea a un incremento del tamaño de segmento máximo (MSS) de TCP que se negoció durante la configuración de la conexión.
3. Si no es al menos cuatro veces este tamaño, se ajusta a 4 * MSS, con un tamaño máximo de 64 KB, salvo que esté vigente una opción de escalado de ventanas (RFC 1323).

Para Ethernet, la ventana se establece normalmente a 17520 bytes (16 KB redondeado a doce segmentos de 1460 bytes.) Hay dos métodos para configurar el tamaño de la ventana de recepción con valores determinados:

- El parámetro *TcpWindowSize* del Registro (consulte el apéndice A)
- La función **setsockopt** de Windows Sockets (para cada socket)

Para mejorar el rendimiento en redes con ancho de banda y retraso elevados, se ha introducido en Windows 2000 la compatibilidad con ventanas escalables (RFC 1323). Este documento RFC describe un método para admitir ventanas escalables al permitir a TCP negociar un factor de escalado para el tamaño de la ventana al establecer la conexión. Esto permite una ventana de recepción real de hasta 1 gigabyte (GB). La sección 2.2 del documento RFC 1323 lo describe muy bien:

"Un TCP puede enviar la opción de escala de ventana de tres bytes en un segmento SYN. Tiene dos objetivos: 1. indicar que TCP está preparado para realizar escalado de ventana de envío y recepción y 2. comunicar un factor de escala que se debe aplicar a la ventana de recepción. Así, un TCP que está preparado para escalar ventanas debe enviar la opción, incluso si su propio factor de escala es 1. El factor de escala está limitado a una potencia de dos y codificado mediante logaritmos, con lo que se puede implementar mediante operaciones de desplazamiento de bits.

Opción Escala de ventana (Wsopt) de TCP:
Clase: 3 Longitud: 3 bytes
+--------+--------+--------+

| Clase=3 |Longitud=3|shift.cnt|

+---------+---------+---------+

"Esta opción es una oferta, no una promesa; ambas partes deben enviar opciones de escala de ventana en sus segmentos SYN para habilitar el escalado de ventanas en los dos sentidos. Si está habilitado el escalado de ventanas, el TCP que envió esta opción desplazará a la derecha los valores de recepción de ventana real mediante la operación 'shift.cnt' de los bits para transmitir en SEG.WND. El valor *shift.cnt* puede ser cero (ofrece escalar, pero aplica un factor de escala de 1 a la ventana de recepción).

"Esta opción se puede enviar en un segmento <SYN> inicial (es decir, un segmento con el bit SYN activado y el bit ACK desactivado). También se puede enviar en un segmento <SYN,ACK>, pero sólo si se recibió una opción de escala de ventana en el segmento <SYN> inicial. Una opción de escala de ventana en un segmento sin un bit SYN debe pasarse por alto.

"El campo Ventana del propio segmento SYN (es decir, <SYN> o <SYN,ACK>) nunca se escala".

Cuando se leen trazas de red de una conexión establecida por dos equipos que admiten ventanas escalables, hay que tener en cuenta que los tamaños de ventana anunciados en la traza deben escalarse según el factor de escala negociado. El factor de escala se puede observar en los paquetes de establecimiento de la conexión (protocolo de enlace de tres vías), como se ilustra en la siguiente captura de Monitor de red:

Dirección de origen Dirección de destino Protocolo Descripción
THEMACS1 NTBUILDS TCP ....S., len:0, seq:725163-725163,
ack:0, win:65535, src:1217 dst:139
+ IP: ID = 0xB908; Proto = TCP; Len: 64
TCP: ....S., len:0, seq:725163-725163, ack:0, win:65535,
src:1217 dst:139 (Sesión NB+ TRAMA: propiedades de trama base
+ ETHERNET: ETYPE = 0x0800 : Protocolo = IP: DOD de Internet
Protocolo

T)
TCP: Puerto de origen = 0x04C1
TCP: Puerto de destino = Servicio de sesión NETBIOS
TCP: Número de secuencia = 725163 (0xB10AB)
TCP: Número de confirmación = 0 (0x0)
TCP: Desplazamiento de datos = 44 (0x2C)
TCP: Reservado = 0 (0x0000)
+ TCP: Indicadores = 0x02 : ....S.
TCP: Ventana = 65535 (0xFFFF)
TCP: Suma de comprobación = 0x8565
TCP: Puntero urgente = 0 (0x0)
TCP: Opciones
+ TCP: opción Tamaño de segmento máximo
TCP: Option Nop = 1 (0x1)
TCP: opción Escala de ventana
TCP: Tipo de opción = Escala de ventana
TCP: Longitud de opción = 3 (0x3)
TCP: Escala de ventana = 5 (0x5)
TCP: Option Nop = 1 (0x1)
TCP: Option Nop = 1 (0x1)
+ TCP: opción Marcas de tiempo
TCP: Option Nop = 1 (0x1)
TCP: Option Nop = 1 (0x1)
+ TCP: opción SACK permitido
00000: 8C 04 C8 BD A3 82 00 00 50 7D 83 80 08 00 45 00 ....P}....E.
00010: 00 40 B9 08 40 00 80 06 A7 1A 9D 36 15 FD AC 1F .@..@......6....
00020: 3B 42 04 C1 00 8B 00 0B 10 AB 00 00 00 00 B0 02 ;B..............
00030: FF FF 85 65 00 00 02 04 05 B4 01 03 03 05 01 01 ...e............
00040: 08 0A 00 00 00 00 00 00 00 00 01 01 04 02 ....

El equipo que envía el paquete anterior ofrece la opción de escala de ventana, con un factor de escalado de 5. Si el equipo de destino responde y acepta la opción de escala de ventana en el SYN-ACK, se entiende que cualquier ventana

de TCP anunciada por este equipo tiene que desplazarse a la izquierda cinco bits desde este punto hacia adelante (el propio SYN no se escala). Por ejemplo, si el equipo anunció una ventana de 32 KB en su primer envío de datos, este valor debería desplazarse a la izquierda (introduciendo ceros desde la derecha) 5 bits, como se muestra a continuación:

32Kbytes = 0x7fff = 111 1111 1111 1111
Desplazar a la izquierda 5 bits = 1111 1111 1111 1110 0000 = 0xffffe (1.048.544 bytes)
Como un número desplazado a la izquierda 5 bits es equivalente a multiplicarlo por 25 o 32. 32767 * 32 = 1.048.544

El factor de escala no es necesariamente simétrico, por lo que puede ser distinto para cada una de las direcciones del flujo de datos.

Windows 2000 utiliza el escalado de ventanas automáticamente si el parámetro *TcpWindowSize* del Registro se establece a un valor mayor de 64 KB y el parámetro *Tcp1323Opts* del Registro se configura correctamente. Consulte el apéndice A para obtener detalles acerca de cómo configurar este parámetro.

Como se especifica en el documento RFC 1122, TCP utiliza retardo en la confirmación (ACK) para reducir el número de paquetes enviados a los medios. La pila TCP/IP de Microsoft utiliza una solución común para implementar el retardo en la confirmación. Cuando TCP recibe datos en una conexión, sólo devuelve una confirmación si se cumple una de las condiciones siguientes:

- No se envió confirmación para el segmento anterior recibido.
- Se recibe un segmento, pero no llegan más segmentos antes de 200 milisegundos para esa conexión.

normalmente se envía una confirmación por cada dos segmentos TCP recibidos en una conexión, salvo que caduque el temporizador de retardo en la confirmación (200 milisegundos). Se puede ajustar el temporizador de retardo en la confirmación mediante el parámetro *TcpDelAckTicks* del Registro

Por ejemplo Windows 2000 admite una importante característica de rendimiento conocida como *Confirmación selectiva* (SACK, *Selective Acknowledgement*). SACK es especialmente importante para conexiones que utilizan tamaños grandes de ventana TCP. Antes de utilizar SACK, un receptor sólo podía confirmar el número de secuencia más reciente de datos contiguos que se habían recibido o el borde izquierdo de la ventana de recepción. Cuando SACK está habilitado, el receptor sigue utilizando el número ACK para confirmar el borde izquierdo de la ventana de recepción, pero también puede confirmar individualmente otros bloques no contiguos de datos recibidos. SACK utiliza opciones de encabezado TCP.

Opción Sack permitido

Esta opción de dos bytes puede ser enviada en un SYN por un TCP que ha sido extendido para recibir (y, supuestamente, procesar) la opción SACK una vez abierta la conexión. NO SE DEBE enviar en segmentos que no sean SYN.

Opción Sack permitido de TCP:
Clase: 4

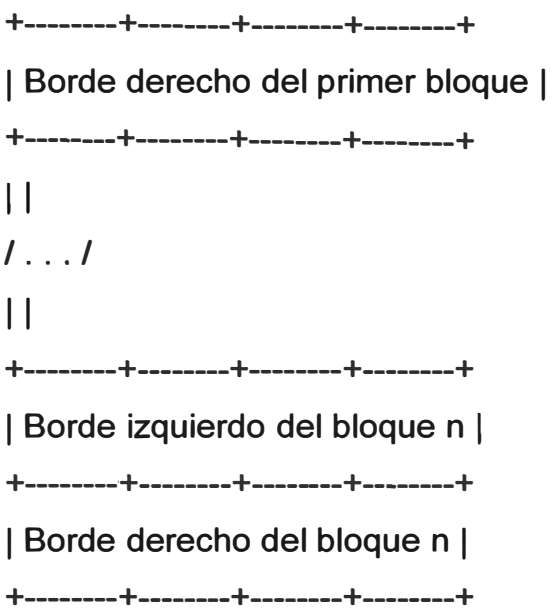
```
+---------+---------+
| Clase=4 | Longitud=2|
+---------+---------+
```

Formato de opción Sack

La opción SACK se ha de utilizar para comunicar información de confirmación extendida desde el receptor al remitente sobre una conexión TCP establecida
Opción SACK de TCP:
Clase: 5
Longitud: variable

```
+--------+--------+
| Clase=5 | Longitud |

+--------+--------+--------+--------+
| Borde izquierdo del primer bloque |
```



Cuando SACK está habilitado (opción predeterminada), se puede descartar un paquete o una serie de ellos y el destinatario puede informar al remitente de qué datos se han recibido exactamente y qué datos faltan. Después, el remitente puede volver a enviar selectivamente los datos perdidos sin necesidad de retransmitir los bloques de datos que se han recibido correctamente. SACK está controlado por el parámetro *SackOpts* del Registro. En la siguiente captura de Monitor de red se ilustra un host que confirma todos los datos hasta el número de secuencia 54857341, más los datos del número de secuencia 54858789-54861685.

+ TRAMA: propiedades de trama base
+ ETHERNET: ETYPE = 0x0800 : Protocolo = IP: Protocolo Internet DOD
+ IP: ID = 0x1A0D; Proto = TCP; Len: 64
TCP: .A...., len:0, seq:925104-925104, ack:54857341, win:32722,
src:1242 dst:139
TCP: Puerto de origen = 0x04DA
TCP: Puerto de destino = Servicio de sesión NETBIOS
TCP: Número de secuencia = 925104 (0xE1DB0)
TCP: Número de confirmación = 54857341 (0x3450E7D)
TCP: Desplazamiento de datos = 44 (0x2C)
TCP: Reservado = 0 (0x0000)
+ TCP: Indicadores = 0x10 : .A....

TCP: Ventana = 32722 (0x7FD2)
TCP: Suma de comprobación = 0x4A72
TCP: Puntero urgente = 0 (0x0)
TCP: Opciones
TCP: Nop opción = 1 (0x1)
TCP: Nop opción = 1 (0x1)
+ TCP: opción Marcas de tiempo
TCP: Nop opción = 1 (0x1)
TCP: Nop opción = 1 (0x1)
TCP: opción SACK
TCP: Tipo de opción = 0x05
TCP: Longitud de opción = 10 (0xA)
TCP: Borde izquierdo del bloque = 54858789 (0x3451425)
TCP: Borde derecho del bloque = 54861685 (0x3451F75)

Otra característica del documento RFC 1323 introducida en Windows 2000 es la compatibilidad con las marcas de tiempo de TCP. Al igual que SACK, las marcas de tiempo son importantes para las conexiones que utilizan tamaños de ventana grandes. Las marcas de tiempo se concibieron para ayudar a TCP a medir con precisión el tiempo de trayecto de ida y vuelta (RTT, *Round-trip Time*) para ajustar el tiempo de espera de las retransmisiones.

Clase: 8
Longitud: 10 bytes

```
+-------+-------+--------------------+--------------------+
|Clase=8 | 10 | Valor TS (TSval) |Respuesta de eco TS (TSecr)|
+-------+-------+---------------------+--------------------+
```

1 1 4 4

La opción de marcas de tiempo lleva dos campos de marcas de tiempo de cuatro bytes. El campo de valor de la marca de tiempo (Tsval, *Time-stamp Value*) contiene el valor actual del reloj de marcas de tiempo del TCP que envía la opción.

El campo Respuesta de eco de marcas temporales (Tsecr, *Timestamp Echo Reply*) sólo es válido si está configurado el bit ACK en el encabezado de TCP; si es válido, refleja un valor de marca de tiempo que fue enviado por el TCP remoto en el campo TSval de una opción de marca de tiempo. Cuando TSecr no es válido, su valor debe ser cero. El valor de TSecr será, generalmente, el de la opción de marca temporal recibida más recientemente; no obstante, hay excepciones que se explican más adelante.

Un TCP puede enviar la opción de marcas de tiempo (Tsopt, Timestamps Option) en un segmento <SYN> inicial (es decir, un segmento que contiene un bit SYN y ningún bit ACK) y puede enviar un valor TSopt en otros segmentos sólo si recibió un valor TSopt en el segmento <SYN> inicial de la conexión.

El campo de la opción de marcas de tiempo se puede ver en una captura de Monitor de red si se expande el campo de opciones de TCP, como se muestra a continuación:

TCP: opción Marcas de tiempo
TCP: Tipo de opción = Marcas de tiempo
TCP: Longitud de opción = 10 (0xA)
TCP: Marca de tiempo = 2525186 (0x268802)
TCP: Marca de tiempo de respuesta = 1823192 (0x1BD1D8)

El uso de marcas de tiempo está deshabilitado de forma predeterminada. Se puede habilitar mediante el parámetro *Tcp1323Opts* del Registro.

CAPITULO VII
MODELO DE CONEXIÓN

7.1 Introducción

La tecnología de transporte que conduce el tráfico de la red de una empresa a otra y de las empresas a los clientes, no es una consideración apremiante para la mayoría de los administradores y gerentes de redes. a los gerentes de redes no les preocupa cuál de las telefónicas lleva sus paquetes de datos de un lugar a otro.

Sin embargo, las telefónicas de cualquiera de las generaciones de tecnología de redes pronto estarán comenzando a promocionar los beneficios de adelantos en la forma de llevar los paquetes de un extremo al otro, por lo que los clientes potenciales deben comenzar a familiarizarse con las ventajas y desventajas de cada una de estas propuestas.

Aunque protocolos como SNA, IPX y DECnet seguirán durante unos cuantos años, TCP/IP es el conjunto de protocolos de mayor crecimiento para las capas de Transporte y de Red, y hoy en día abarca las porciones de tráfico de los protocolos de capa 3 y capa 4. Por tanto, se analizará los diferentes métodos que existen para transmitir tráfico TCP/IP y los protocolos de capa enlace y física que se adaptan a la transmisión de TCP/IP . Los principales competidores en esta carrera son IP sobre ATM sobre SONET, IP sobre PPP e IP sobre fotones.

La capa 1, la capa física, se preocupa por el flujo de bits sobre el medio y define los impulsos eléctricos, electromagnéticos u ópticos que serán interpretadas como bits, así como los medios, conectores y otros aspectos físicos del camino de datos. La capa 2, la capa de enlace de datos, , se preocupa por los flujos de datos

lógicos entre dos puntos, y controla el acceso al vínculo físico y generalmente incluyen alguna forma de detección y corrección de errores de forma tal que las capas superiores puedan contar con un enlace confiable.

Uno de los principales objetivos es dar solución a la congestión en los conmutadores ATM , aunque hay diversos mecanismos de control de errores , de control de flujo y control de congestión , no hay que olvidar que es de extremo a extremo debido a la característica orientada a la conexión esto acaba perjudicando el rendimiento de la red o a los propios extremos de la comunicación, la congestión de los conmutadores en la red son impredecibles y cuando aparecen pueden acabar afectando negativamente a toda la red.
Una cuestión básica es el modelo de compresión arquitectónico ATM
El modelo ATM se apoya en la capa física, ATM y de adaptación ATM , me centraré en la capa de adaptación ATM por que permite adaptar los diferentes flujos de información a ATM .
La capa ALL está compuesta por 2 sub capas SC (sub capa de convergencia) y SAR (sub capa de segmentación y reensamblado)
La sub capa SC se encarga de la corrección de errores, control de flujo y la sincronización de extremo a extremo.
La sub capa SAR encargado de segmentar el tráfico continuo de tramas en celdas de 48 bytes las cuales son recibas por la capa ATM, además detecta posibles celdas erradas o perdidas .
Si observamos la siguiente figura donde se muestra las capas de la arquitectura tanto en los nodos de la red como en los nodos locales y en los de usuario, se puede ver que los nodos de la red o conmutadores no necesitan de la capa ALL ya que el trafico de la red es completamente nativo ATM y lo que se transmite entre conmutadores son celdas, los nodos o conmutadores locales si necesitan de la capa ALL por que necesitan ajustar el flujo de trafico entre los terminales de usuario que como se puede observar puede hacerse desde cualquier protocolo de comunicación como el TCP/IP por ejemplo en realidad las celdas no son visibles por los usuarios que lo que hacen es generar las tramas propias de los protocolos que emplean estas tramas son las que se segmentan en la capa ALL y luego pasan a la capa ATM que se encarga del transporte de las celdas a la capa física

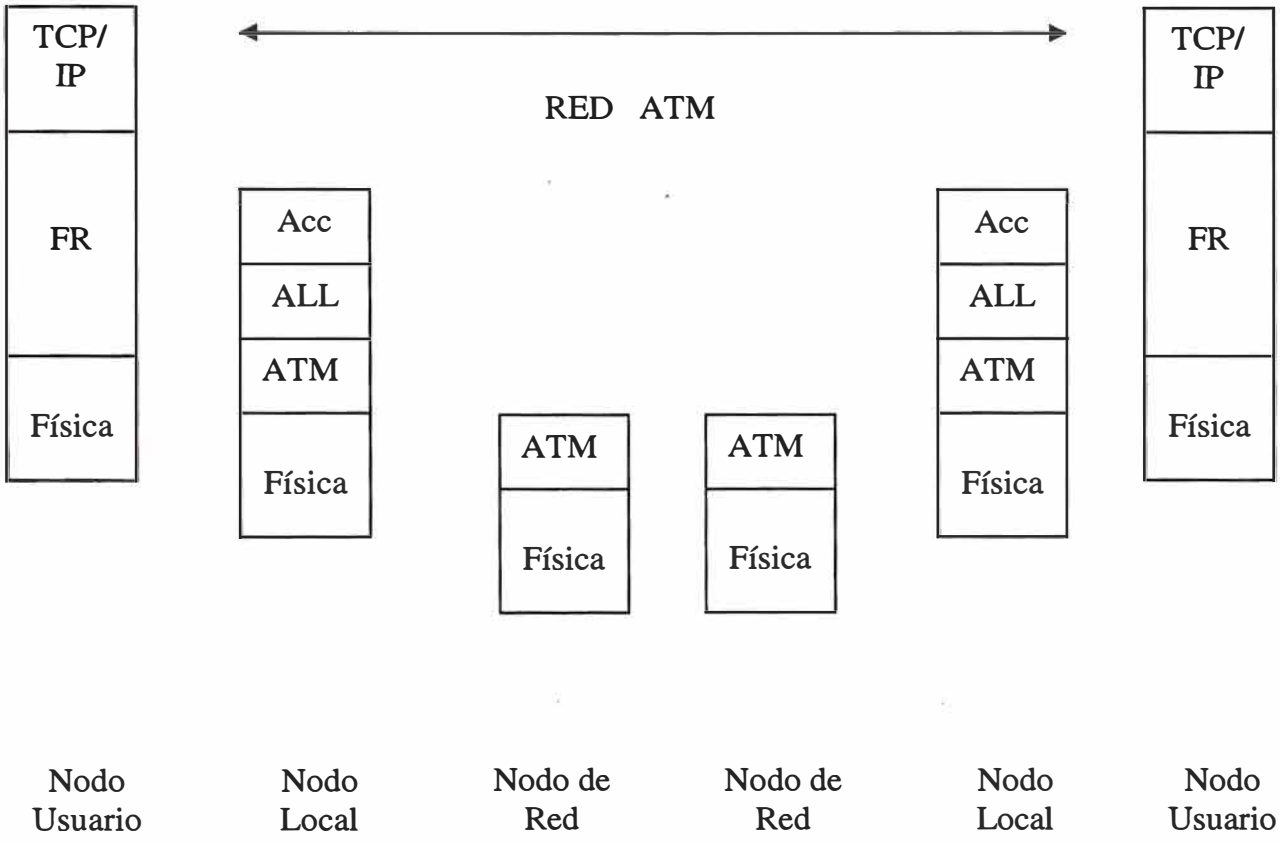


Figura 7.1 Conexión ATM.

Con respecto al servicio está tecnología esta destinada a soportar diversos tipos de servicios en tiempo real o diferido.

Una conexión es establecida cono resultado de una negociación entre el usuario y la red, como por ejemplo la clase de servicio a adoptar CoS , la cual esta definida por los parámetros de tráfico de la conexión y por sus parámetros de calidad de servicio QoS la ITU a propuesto varias CoS definidas como ATC (ATM transfer capability) cada ATC especifica un conjunto de parámetros y procedimientos de la capa ATM para sustentar un modelo de servicio y un conjunto de valores QoS asociados. Cada ATC es especificada en términos de un modelo de servicio, un descriptor de tráfico, unos procedimientos específicos,

una definición de conformidad y los compromisos de QoS demandados por cada conexión los cuales la red se compromete a cumplir.
La ITU propone los siguientes ATC

- DBR (deterministic bit rate) la velocidad de conexión la fuente es constante y la tolerancia CDV (cell delay variation) es reducida, este tipo de servicio se emplea en aplicaciones en tiempo real (voz, audio, video) la capacidad de transferencia deterministica esta pensada para tráfico CBR (constant bit rate) y por lo tanto para responder a compromisos de QoS en términos de perdidas de celdas, retardo de transferencia de celdas y variación del retardo adaptado a CBR, DBR no se limita exclusivamente a aplicaciones CBR puede usarse con requisitos QoS menos rigurosos.

PROBLEMAS EN ATM:

En el pasado los protocolos de comunicaciones de datos evolucionaron en respuesta a circuitos poco confiables. Los protocolos en general detectan errores en bits y tramas perdidas, luego retransmiten los datos.

Los usuarios puede que jamás vean estos errores reportados, la degradación de respuesta o de caudal (through put) serían los únicos síntomas.

A diferencia de los mecanismos de control extremo a extremo que utiliza TCP en internerworking, la capacidad de Gbit/seg de la red ATM genera un juego de requerimientos necesarios para el control de flujo. Si el control del flujo se hiciese como una realimentación del lazo extremo a extremo, en el momento en que el mensaje de control de flujo arribase a la fuente, ésta habría transmitido ya algunos Mbytes de datos en el sistema, exacerbando la congestión. Y en el momento en que la fuente reaccionase al mensaje de control, la condición de congestión hubiese podido desaparecer apagando innecesariamente la fuente. La constante de tiempo de la realimentación extremo a extremo en las redes ATM (retardo de realimentación por producto lazo - ancho de banda) debe ser lo

suficientemente alta como para cumplir con las necesidades del usuario sin que la dinámica de la red se vuelva impractica.

Las condiciònes de congestión en las redes ATM están previstas para que sean extremadamente dinámicas requiriendo de mecanismos de hardware lo suficientemente rápidos para llevar a la red al estado estacionario, necesitando que la red en sí, éste activamente involucrada en el rápido establecimiento de este estado estacionario. Sin embargo, esta aproximación simplista de control reactivo de lazo cerrado extremo a extremo en condiciones de congestión no se considera suficiente para las redes ATM.

El consenso entre los investigadores de este campo arroja recomendaciones que incluyen el empleo de una colección de esquemas de control de flujo, junto con la colocación adecuada de los recursos y dimensionamiento de las redes, para que aunados se pueda tratar y evadir la congestión ya sea:

Detectando y manipulando la congestión que se genera tempranamente monitoreando de cerca las entradas/salidas que están dentro de los conmutadores ATM y reaccionando gradualmente a medida que vaya arribando a ciertos niveles prefijados.

Tratando y controlando la inyección de la conexión de datos dentro de la red en la UNI (unidad interfaz de red) de tal forma que su tasa de inyección sea modulada y medida allí primero, antes de tener que ir a la conexión de usuario a tomar acciones mas drásticas.

El estado de la red debe ser comunicado a la UNI, generando rápidamente una celda de control de flujo siempre que se vaya a descartar una celda en algún nodo debido a congestión. La UNI debe entonces manejar la congestión, cambiando su tasa de inyección o notificándola a la conexión de usuario para que cese el flujo dependiendo del nivel de severidad de la congestión.

El mayor compromiso durante el control de congestión es el de tratar y afectar solo a los flujos de conexión que son responsables de la congestión y actuar de forma transparente frente a los flujos que observan buen comportamiento. Al mismo tiempo, permitir que el flujo de conexión utilice tanto ancho de banda como necesite sino hay congestión.

La recomendación UIT - T I. 371 especifica un contrato de tráfico que define como el tráfico del usuario seria administrado. El contrato que existe para cada conexión virtual (virtual path o virtual channel), es básicamente un acuerdo entre el usuario y la red con respecto a la Calidad de Servicio (Quality Of Service - Q o S) y los parámetros que regulan el flujo de celdas. Estos descriptores de trafico dependen de una particular clase de servicio y pueden incluir bajo la especificación del ATM Forum UNI / a cinco Q o S referenciados en los AALS. El objetivo de estas sub clases de servicio es agrupar características de servicio como requerimiento de ancho de banda similares, sensibilidad a la perdida de datos y retardos para un correcto manejo de los datos en los puertos de acceso ATM, etc. Estos parámetros pueden incluir el Sustained Cell Rate (SCR), el Mínimum Cell Rate (MCR), el Peak Cell Rate (PCR) y/o el Burst Tolerance (BT). Para soportar todas las diferentes clases de servicios definidos por los estándares el switch ATM debe ser capaz de definir éstos parámetros en base a cada VC o cada VP y debe proveer amortiguadores (buffers) para absorber las ráfagas de tráfico.

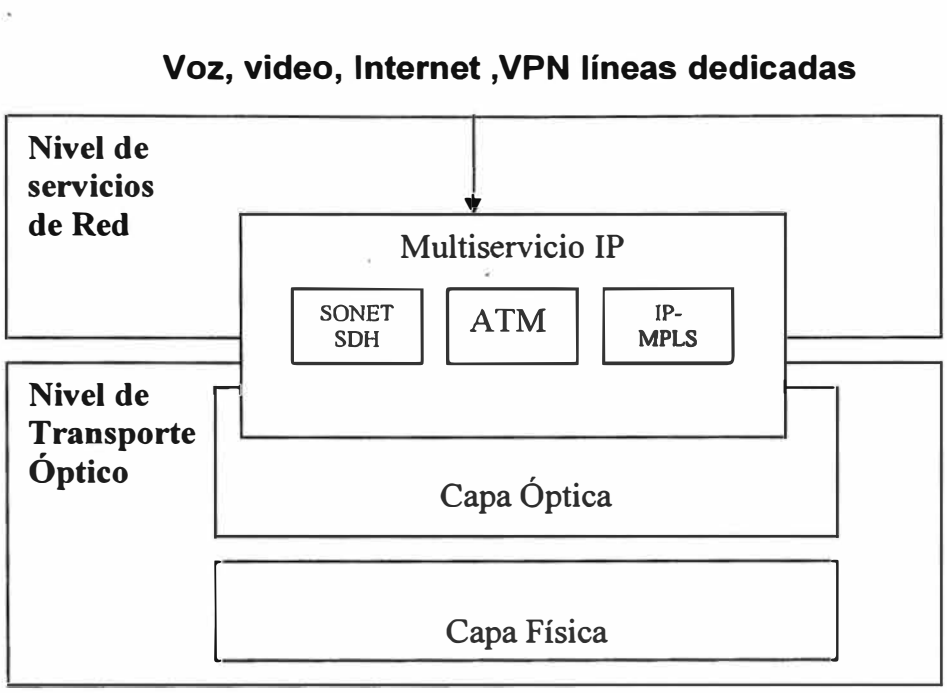


Figura 7.2 Redes de la próxima Generación.

CONCLUSIONES

1. En todas las redes existen gran variedad de elementos hardware (*switchs*, *routers*, *bridges*, *brouters*, *hubs*, *ETDs*, etc.) que realizan muy diversas funciones (conmutación, *routing*, puenteo, controles de congestión y de flujo, garantía de *QoS*, ejecución de aplicaciones, etc.). En la actualidad la red es mayormente un canal de comunicación para transferir paquetes entre equipos finales (ayudada por los elementos hardware antes citados). Pero también se están realizando importante esfuerzos para equipar a los elementos hardware con elevadas prestaciones aportadas por diversas técnicas software. Esto dota a la red de características activas (*active networks*) en el sentido de que los elementos hardware que la componen computan, modifican u operan los contenidos de los paquetes y también serán capaces de transferir o propagar código. Por consiguiente, una red activa es una red programable.
2. Existen dos planteamientos en la realización de redes activas, la idea del conmutador programable y la de la cápsula. Una red es activa si en sus árboles de distribución *multicast* existen nodos activos con capacidad para ejecutar programas y/o capaces de implementar mecanismos de propagación de código.
3. Algunas de las ventajas de los protocolos activos son conseguidos instalando nodos activos en puntos estrategicos de la red. Los *protocol boosters* son una nueva contribución a las redes activas o programables. Una ventaja de los *boosters* es que pueden ser fácilmente "inyectados" en los sistemas actuales sin provocar cambios en la infraestructura de red.
4. El concepto de agente de comunicaciones para servicios de comunicaciones multimedia en redes de área extensa.

Este papel introduce una arquitectura software orientada a agente y propone el concepto de agente de comunicación para servicio de comunicación multimedia. Los servicios multimedia son expresados como agentes.

5. Los conceptos como redes activas, protocolos *boosters* o agentes software han sido propuestos y desarrollados para redes *IP*, sin embargo, la actividad empieza a notarse en las redes *ATM*,
6. Los agentes software móviles usados para implementar operaciones robustas y funciones de mantenimiento en redes *ATM* desempeñan un rol similar al de las células *OAM* en *ATM* estándar, pues son transmitidos entre entidades de control a intervalos regulares usando recursos predefinidos.
7. La diferencia entre los agentes móviles y las células *OAM* reside en que éstos pueden contener código.
8. Para concluir destacar que la integración del tráfico IP sobre tecnología ATM es también uno de los campos más activos en la actualidad. En esta línea pueden destacarse los siguientes protocolos: IP-over-ATM, IP Switching, Tag Switching, NHRP, MPOA, IMSS, CSR, ARIS AREQUIPA RED y EPD.
9. Aunque se ha hablado mucho sobre la "saturación de ancho de banda" en los años recientes, el costo real de la capacidad de red no ha caído substancialmente. Los proveedores de servicio que comercializan ancho de banda a precios altos desean obtener los mayores beneficios, y detestan el hecho de que cada vez que un paquete IP es convertido en una celda ATM, 5 bytes de la celda de 53 bytes se pierden como sobrecargo.
10. Además, el tamaño de los marcos que no coincide perfectamente con el de las celdas genera una transmisión no óptima. Por ejemplo, un marco de 40 bytes de datos (un tamaño bastante común en la Internet pues los paquetes de sincronización y confirmación normalmente tienen el tamaño mínimo, es decir, 64 bytes) cabrá en una sola celda, lo que permitirá un sufijo de corrección de errores de 8 bytes en la capacidad de carga de 48 bytes. Pero un marco de 41 bytes requerirá dos celdas, y la segunda celda estará completamente vacía, será casi exclusivamente sobrecargo. Lo que

es más, un paquete voluminoso tendrá que ser dividido en muchas celdas, hasta un máximo de 38 para un paquete de 1.518 bytes, y la pérdida de cualquiera de estas celdas implicará la retransmisión de todo el paquete.

11. El sobrecargo de la conversión de tráfico IP a ATM comúnmente se conoce como "impuesto de celda". Dependiendo de sus parámetros sobre tamaño de marco y tasa de pérdida de paquetes, el impuesto a pagar puede constituir alrededor del 13% o más del caudal total de datos.
12. Es difícil para los proveedores de servicio percibir algún beneficio de ATM y la conversión del tráfico en celdas, particularmente si su tráfico consiste exclusivamente de tráfico IP. Los paquetes IP no tienen requerimientos particulares en cuanto a calidad de servicio, al menos para redes IP tradicionales.
13. La función de combinación estadística de ATM puede ser realizada perfectamente por un enrutador, si no están presenten datos con requerimientos críticos de tiempo. En estas circunstancias, los fabricantes de enrutadores y conmutadores, comenzaron a escuchar las demandas de "IP sobre SONET" o "Paquete sobre SONET".
14. La idea sería la eliminación de la capa ATM para acabar con el impuesto de las celdas. Sin embargo, no es posible eliminar por decreto la capa Data-link, aunque es posible sustituir el protocolo PPP (Point-to-Point Protocol: Protocolo Punto a Punto).
15. Las especificaciones RFC1619, PPP sobre SONET/SDH; RFC1661, PPP, RFC1662 y PPP en esquemas HDLC, son normas relevantes para emplear PPP sobre ATM en la capa 2.
16. PPP es un mecanismo sencillo que sólo conoce ambos extremos de un enlace, mientras que ATM establece circuitos virtuales que pueden cruzar varios enlaces de un extremo al otro.
17. PPP no fue diseñado para mezclar o combinar varios flujos de datos, tal como lo hace ATM.
18. Con PPP será necesario instalar controles de calidad de servicio en la capa 3 y superiores, pues PPP no puede establecer prioridades o responder coherentemente a la congestión. Pero si todo lo que el usuario necesita es

un enlace de datos entre dos puntos, libre de errores, PPP puede ser suficiente.

19. Una de las fortalezas de ATM usted paga solamente por la carga de celdas que es efectivamente transportada y conmutada para usted.

20. Hoy día los accesos conmutados a Internet están creando "Cuellos de Botella" en la infraestructura. Para soluciomar este problema los fabricantes no solo han desarrollado sistemas de acceso sino aplicaciones para soluciones de fin a fin con conmutadores ATM, con solventes sistemas de administración de la red (Network Management).

21.- Conbinando la simplicidad de la multiplexación por división en el tiempo (Time Division Multiplex TDM) encontrado en la conmutación de circuitos, con la eficiencia de las redes de conmutación de paquetes con multiplexación estadística es la caracteristica fundamental de la tecnología ATM.

RECOMENDACIONES

1. Se recomienda emplear TCP/IP sobre ATM en redes LAN, para poder aprovechar la comunicación por ráfagas de una LAN y la QoS de una tecnología ATM.
2. Para Redes con gran tráfico es recomendable este tipo de enlace pues la parte de ATM se encarga de la realimentación extremo a extremo evitando un Xoff o CTS innecesario que podría darse en TCP.
3. Se recomienda este tipo de manejo de información pues el trato de la transferencia de información depende del tipo de información a transferir por ende los costos no son iguales para todos (bajo demanda).
4. Lo interesante de ATM es que es una tecnología que se adapta a cualquier sistema SONET, ISDN. LAN, permitiendo manejo de información de voz, video y datos con la calidad de servicio que el sistema requiera
5. Respecto a los costos es importamte tomar en cuenta que los equipos involucrados en este tipo de conexión es alto pero a larga justificable por la calidad de servicio que ofrecen.

APÉNDICE A
COSTOS EN UNA RED ATM

IP VPN con conexión ATM

Conexión de 1 a 19 puertos

Contrato libre (es decir se puede cancelar el contrato en cualquier momento).

	Costo	
Velocidad 9.6 Kbps	$595 $159.46	instalación. mensualidad.
Velocidad 19.2 Kbps	$595 $199.92	instalación. mensualidad.

Contrato 1 año

Velocidad 9.6 Kbps	$595 $153.51	instalación. mensualidad.
Velocidad 19.2 Kbps	$595 $193.97	instalación. mensualidad.

Contrato 2 años

Velocidad 9.6 Kbps	$595 $138.04	instalación. mensualidad.
Velocidad 19.2 Kbps	$595 $173.74	instalación. mensualidad.

Se necesita para la instalación de un Router brindado por la Empresa proveedora del servico (Telefónica) o por el usuario.

Además de un MODEM brindado por la Empresa proveedora del servico (Telefónica).

BIBILIOGRAFÍA

1.- Comunicación entre Computadoras y Tecnologías de Redes ″Michael A. Gallo Willian M. Hancock″

2.- TCP/IP ″John Ray″ PRENTICE HALL, Madrid 1999

3.- Redes para Todos ″Mark Gibbs″ Segunda Edición México 1995

4.- Andrew S. Tanenbaum ″ Redes de Computadoras″.Tercera Edición 1997

5.- Iain McLean ″TCP/IP″ (Ed. Anaya Multimedia)

6.- www.tlmat.unican.es/inteca/ inteca/users/aeg/ryst/Material/CAP1RED.pdf.

7.- arcos.inf.uc3m.es/~juange/ro/992000/99intro.pdf.

8.- www.ra-ma.es/down/Glosario.pdf

9.- http://www.rediris.es/rediris/boletin/46-47/ponencia10.html

10.- http://www.monografias.com/trabajos/atm/atm.shtml

11.- http://www.comunicaciones.unitronics.es/tecnologia/atm.htm

12.- http://ciberhabitat.gob.mx/museo/cerquita/redes/ethernet/02.htm