UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

"IMPLEMENTACIÓN DE MULTICAST EN UNA RED DE DATOS IP"

TESIS

QUE PARA OBTENER EL TÍTULO DE INGENIERO EN TELECOMUNICACIONES

PRESENTAN:

JAVIER ORTIZ VILLASEÑOR
JOSÉ LUIS SANDOVAL ESPITIA

DIRECTOR:

ING. RODOLFO ARIAS VILLAVICENCIO

MÉXICO, D.F. MARZO DEL 2004

Por fin después de tanto tiempo hoy llego a la culminación de este ciclo en mi vida, por ello quiero agradecer a todos aquellos que de alguna forma me han apoyado en el transcurso de mi carrera.

A mis padres y mis hermanas por su apoyo y su cariño, por ser parte esencial de mi vida.

A mis tíos, Jorge y Yola, y a mis primos, por su apoyo durante mi estancia en la facultad.

A todos mis compañeros y amigos de la facultad, gracias por todos esos momentos tan gratos.

A mis compañeros de trabajo Adriana, Arturo y Gerardo por su apoyo y comprensión para que pudiera sacar adelante este proyecto.

A mi compañero en este proyecto José Luis, quien me brindó la oportunidad de participar con él en la realización de este trabajo.

A nuestro asesor Rodolfo (Fito), por sus enseñanzas y su apoyo brindados.

A todos aquellos que han formado parte de mi vida...

Gracias
Javier Ortiz Villaseñor

Este trabajo es la culminación de mucho tiempo de esfuerzo, no solamente mío, si no de mucha gente que de alguna forma me acompaño y me apoyo en el transcurso de mi carrera, de tal forma que quiero hacer participes y agradecer de la culminación de este ciclo en mi vida a todas esas personas que de alguna u otra razón formaron parte de mi vida.

Muy en especial a mi **mamá** que es la principal responsable de que yo este aquí tratando de redactar una hoja de agradecimiento, después de muchos desvelos te agradezco las esperas por la madrugada, tu apoyo incondicional y solo quiero que sepas que te amo y que siempre estamos juntos ya que todo el tiempo estas en mi pensamiento. Te quiero mamá

Gracias papá por apoyarme en todos los sentidos y creer en mí al igual que yo creo en ti.

A mis hermanas y a mi hermano, los cuales me acompañan en los momentos malos y buenos, por comprenderme y apoyarme en todos aquellos momentos de tensión en mi carrera.

A mi tío Javier el cual me inculco desde muy pequeño el amor por el estudio y por el cual siento una gran admiración.

A mi novia Lilia por caminar junto a mí en toda esta trayectoria y estar conmigo todo el tiempo apoyándome y comprendiendo los tiempos difíciles.

Y a todos mis compañeros y amigos que me acompañaron por el paso de la facultad, Jorge, Gris, Juan, Liceth, Carlos, Angélica, Fernando y Ana y todos los que me faltan, gracias por hacer de este viaje un momento inolvidable en mi vida.

A mis amigos Gerardo, Arturo, Adriana quienes me abrieron un espacio para poder desarrollar esta tesis, compartiendo con migo sus experiencias y conocimientos.

A mi compañero de trabajo Javier quien tuvo la paciencia de trabajar conmigo y del cual sigo aprendiendo muchas cosas.

Y por ultimo a mi asesor Rodolfo, por permitirme la oportunidad de trabajar con el en este proyecto

Gracias

José luis Sandoval Espitia

ÍNDICE

Lista de figuras

ANEXO B Tablas de enrutamiento

ANEXO C Configuración de las aplicaciones

ANEXO E Calidad de servicio de redes de datos IP

ANEXO F Aplicaciones Multimedia

Lista de tablas

CAPÍTULO 1. Introducción

Hoy en día, Internet y en muchos casos, las redes internas corporativas que funcionan bajo el concepto de redes unicast, han crecido en tamaño, alcance y en términos de usuarios conectados. Gran cantidad de usuarios de estas redes frecuentemente desean acceder a la misma información al mismo tiempo. La aparición de nuevas tecnologías ha permitido que existan aplicaciones que accedan esta información al mismo tiempo, es decir, que una sola fuente requiera de enviar la misma información a varios puntos destinos, no importando en donde se localicen estos a lo largo de toda la red, o bien en otras redes que mantengan conexión con la nuestra.

1.1 Definición del problema

En casos como el mencionado anteriormente, se origina un incremento considerable en los parámetros que permiten caracterizar la funcionalidad de una red de datos. Y con este incremento, se reduce la calidad de servicio que una red puede ofrecer a sus usuarios.

Los parámetros de los que hablamos son: ancho de banda (*BW*), carga del servidor (*server load*) y carga de la red (*network loading*). Estos parámetros, sobre una red de datos unicast, según la dimensión original de la red, pueden soportar algunos incrementos, pero siempre es necesario fijar un porcentaje de utilización máxima que nuestra red puede permitir, para con esto evitar la perdida de la información o la indisponibilidad de la red.

Analizando como se ve afectado cada uno de estos parámetros en el caso de varios usuarios accesando la misma información, podemos decir que el ancho de banda de la red se ve afectado de manera directamente proporcional conforme se incremente la cantidad de usuarios unicast conectados y que estén haciendo peticiones de paquetes al servidor. Ejemplificando, una charla por Internet, con una técnica de compresión de audio, requiere 8 kbps; un servicio de video comprimido requiere de 120 kbps, esto con una baja calidad; y que un video con formato MPEG-2 (Moving Picture Expert Group 2) requiere de 1.5 Mbps. Imaginando que estas aplicaciones están siendo requeridas por un usuario a un mismo servidor de servicios de Internet, esto nos generaría un consumo de 1.628 Mbps de ancho de banda. Ahora bien, si lo multiplicamos por la cantidad total de usuarios que pudieran estar haciendo las mismas peticiones, resultaría una cantidad enorme de ancho de banda que muy probablemente saturaría los enlaces disponibles, y por otra parte el servidor tendría que intentar atender todas las peticiones de los usuarios por separado, lo que finalmente originaria adicionalmente una saturación en el servidor, y por ende, una reducción en la calidad de servicio.

De manera ligada, podemos ver que la carga del servidor se vería afectada de manera lineal conforme se incremente el número de usuarios conectados. El servidor debe atender por separado las peticiones que cada usuario realice, llegando al extremo en que el servidor este saturado y exceda su capacidad tecnológica o la infraestructura de la red para cubrir la creciente demanda de servicios. Una solución no viable sería el incremento en la capacidad del CPU del servidor e incrementar el ancho de banda de

las interfaces de red, para poder cubrir esta demanda. Finalmente, tendría que haber varios servidores para cubrir estos requerimientos.

Ahora bien, la carga de uso de la red también se ve afectada. En el ambiente de una red IP unicast donde una fuente envía paquetes a un destino especifico, la dirección destino que viaja en el paquete transmitido es la de un simple y único destino. Estos paquetes son reenviados a lo largo de la red desde la fuente hasta el destino por los enrutadores. Si son varias las peticiones que esta atendiendo este servidor, entonces, la carga de trabajo que está sufriendo cada uno de los enrutadores de cada trayectoria servidor-destino, se incrementa proporcionalmente.

Dados los inconvenientes derivados de usar IP unicast [33] en situaciones donde varios usuarios desean acceder a la misma información al mismo tiempo, se desarrollo un nuevo paradigma que supliera la necesidad mencionada sin afectar el ancho de banda y uso de la red, permitiendo además desarrollos escalables y sencillos de implementar. El desarrollo concluyo con las especificaciones de IP multicast [47].

1.2 Objetivos y metas de la tesis

La presente tesis tiene como objetivo la implementación de multicast en una red IP que ya está en operación cuya topología es de tipo jerárquica con tres niveles: dorsal, distribución y acceso.

En nuestro caso, el hecho de que la red tenga esta topología nos facilitará la implementación de multicast ya que podremos identificar fácilmente los flujos de tráfico generados por las aplicaciones multicast que se introduzcan a la red, facilitando así algunas consideraciones en el diseño de la implementación.

Una vez implementado multicast sobre la red, siguiendo los lineamientos derivados de esta tesis, se realizarán pruebas de la implementación utilizando dos aplicaciones con características diferentes. La primera de ellas es Windows Media Player la cual es utilizada como un reproductor multimedia y se encuentra instalada en la mayoría de las computadoras con sistemas *Windows* actualmente por lo que su uso es muy común. La segunda aplicación fue un mensajero instantáneo: Prochat, la cual como su nombre lo indica es una aplicación de conversación en tiempo real (Chat Room). Este tipo de aplicación resulta interesante para el estudio de nuestra tesis ya que cada usuario conectado a una sesión es receptor y fuente de tráfico multicast al mismo tiempo. Con lo cual podremos observar el comportamiento de la red ante la presencia de múltiples fuentes alrededor de la misma.

1.3 Estructura de la tesis

El desarrollo del presente trabajo se expone en los siguientes cinco capítulos, los cuales referiremos brevemente a continuación:

En el **capítulo dos** describimos de manera general conceptos y definiciones básicas para el entendimiento de las redes de datos, tomando en consideración la importancia del modelo OSI (Open Systems Interconection) y del modelo TCP/IP [33] (Transmission Control Protocol/Internet Protocol) y sus capas fundamentales para el entendimiento de la estructura lógica de una red, así como para el aislamiento de problemas en la red, en este punto se explica detalladamente las capas que constituyen el modelo OSI:

Aplicación, Presentación Sesión, Transporte, Red, Enlace de Datos y Física. Así como las del modelo TCP/IP el cual es el protocolo más utilizado hoy en día ya que por medio de este se hace posible la comunicación de datos a cualquier parte del mundo.

En cuanto a las redes LAN y WAN, en este capítulo también se definen características lógicas y físicas por separado así como dispositivos y tecnologías como Token Ring, FDDI (Fiber Distributed Data Interface), Ethernet [45] y 802.3 [16] para redes LAN (Local Area Network); y PPP (Point to Point Protocol) y HDLC (High-level Data Link Control) para redes WAN (Wide Area Network).

En el **capítulo tres** nos enfocamos a la tarea de explicar de manera amplia el desarrollo de la tecnología multicast hasta nuestros días, la implementación de MBONE (**M**ulticast **B**ackb**ONE**) mundial la cual es la columna vertebral de las aplicaciones multicast en Internet, ubicamos de manera clara en que capas de modelo OSI se encuentra multicast, esto es de suma importancia ya que como sabemos OSI marca el parámetro a seguir en cuanto se refiere a la separación relativa de las tecnologías ofreciéndonos una visión clara de entorno que rodea a multicast con respecto a otras tecnologías y su interacción con ellas. Se explica de manera clara el concepto de grupo multicast ya que en este concepto descansa la filosofía multicast, esto es, Multicast tomó como base la creación de grupos, los cuales están formados por todos aquellos hosts que requieren de cierta información común entre ellos, cada grupo está identificado por una dirección en específico a la cual dichos hosts se referirán para conseguir la información deseada. Se establece el direccionamiento lógico asignado por IANA (Internet Assigned Names Authority) la cual está capacitada para asignar los rangos de direcciones IP propios de la tecnología multicast.

Dentro de este capítulo se describe la forma en la cual esta tecnología crea rutas para poder llegar a los hosts y los métodos que ésta utiliza, llamadas árboles en base a su filosofía de grupos. De acuerdo a los requerimientos en la implementación de una red multicast, se determina que tipo de árbol se utilizará por lo cual es interesante el estudio de los distintos árboles que se pueden crear, para tomar la mejor decisión en la implementación. La interacción entre los hosts y el grupo esta determinada por los protocolos IGMP (Internet Group Management Protocol) [48] y PIM (Protocol Independent Multicast) [49] por lo cual se estudia paso a paso en este capítulo, proporcionando una visión lógica y coherente a nivel mensajes y tramas entre los equipos.

En el **capítulo cuatro** nos enfocamos a multicast en la capa dos del modelo OSI. En las redes donde se considere la implementación de multicast es importante tomar en cuenta el impacto de la tecnología en los equipos de la capa dos, como son switch y concentradores ya que multicast fue creado con la finalidad de llegar a los hosts que requieran una información específica y no a todos los usuarios, en este capítulo se recopila la información necesaria para tomar en consideración dicho impacto y como minimizarlo utilizando distintas tecnologías como IGMP *Snooping* o CGMP (Cisco Group Management Protocol) [10], explicando de manera sencilla y profunda el intercambio de paquetes entre el host y el switch así como con el enrutador, de esta manera podemos determinar que tanto del desempeño del equipo estamos dispuestos a sacrificar tomando como base las aplicaciones que el usuario requerirá. En este capítulo encontraremos

ejemplos claros de cómo se realizan las actualizaciones de las tablas tanto en los enrutadores como en los equipos de capa dos y como se realiza el intercambio de solicitudes de acceso o de abandono a los grupos.

En el **capítulo cinco** tratamos el papel que juega multicast en la capa tres del modelo OSI. Se expondrán las principales características de los distintos tipos de protocolos de enrutamiento que existen; posteriormente trataremos a cada protocolo por separado ubicándolos en alguna de las dos clases existentes ("modo denso" y "modo esparcido"), y para cada protocolo se mostrarán sus principales ventajas o deficiencias.

Los protocolos que tratamos en este capítulo son DVMRP (Distance Vector Multicast Routing Protocol) [12], PIM-DM (PIM Dense Mode) [9], PIM-SM (PIM Sparse Mode) [8] y CBT (Core Based Trees) [1], al final de cada protocolo se concentran sus ventajas y su capacidad de escalabilidad facilitando de esta forma la decisión de utilizar uno u otro protocolo de acuerdo a las características de una red donde se implementará.

En el **capítulo seis,** analizaremos el caso de una empresa que desea ofrecer capacitación a distancia para sus principales clientes implementando algunas aplicaciones del tipo *e-learning*. Para ello crearemos una red prototipo sobre la cual, nos basaremos para determinar las posibilidades de implementación de multicast en la red real. En este prototipo de la red ejemplificaremos el proceso a seguir para la implementación de multicast sobre una red IP ya en operación. Se determinarán las condiciones iniciales de la red y su capacidad para poder soportar esta tecnología, definiendo características de los equipos, aplicaciones que actualmente se corren en la red, capacidad en los enlaces y en las computadoras que correrán las aplicaciones sugeridas.

Determinaremos además el papel que jugaran los enrutadores con respecto a multicast, definiendo los protocolos multicast que se utilizarán y las razones por las que se escogieron. Una vez definidos los aspectos lógicos, se pasará a la implementación de multicast sobre la red, definiéndose las configuraciones de los equipos para la activación de esta tecnología. Una vez implementada, se realizarán una serie de pruebas para verificar su correcta operación, mostrándose los resultados de las mismas, así como una comparación entre unicast y multicast, en lo que a uso de ancho de banda se refiere para poder apreciar mejor las ventajas adquiridas con esta tecnología.

En el **Anexo A** se muestran las configuraciones finales en los enrutadores y *switches* para la implementación de multicast en la red prototipo de nuestro proyecto.

En el **Anexo B** se muestran las tablas de enrutamiento, necesarias para verificar el correcto funcionamiento del protocolo de enrutamiento, en nuestro caso PIM-SM.

En el **Anexo C** tratamos de forma completa la configuración de la aplicación por medio de pantallas.

En el **Anexo D**, mostraremos una implementación alternativa a la usada en el capítulo 6, utilizando el modo *PIM Sparse-Dense-Mode,* propio de los equipos Cisco.

En el **Anexo E**, explicaremos de forma breve la Calidad de Servicio (QoS) en Redes de Datos IP, su implementación y los beneficios que esta ofrece.

En el **Anexo F**, mencionaremos algunas aplicaciones multimedia donde multicast puede ser aplicado, por ejemplo: video conferencia y *streaming,* entre otros.

CAPITULO 2. Fundamentos de redes de datos.

2.1 El modelo OSI

Al principio de su desarrollo las redes LAN, MAN y WAN eran en cierto modo caóticas. A principios de la década de los 80 se produjeron tremendos aumentos en la cantidad y el tamaño de las redes. A medida que las empresas se dieron cuenta de que podrían ahorrar mucho dinero y aumentar la productividad con la tecnología de *networking*, comenzaron a agregar redes y a expandir las redes existentes casi simultáneamente con la aparición de nuevas tecnologías y productos de red. A mediados de los 80, estas empresas enfrentaron gradualmente problemas más serios, debido a la necesidad de comunicarse entre si.

Para enfrentar el problema de incompatibilidad de las redes y su imposibilidad de comunicarse entre sí, la *Organización Internacional para la Normalización* (***ISO***) estudió esquemas de red como DECNET (Digital Equipment Corporation Networking), SNA (Systems Network Arquitecture) y TCP/IP [33] a fin de encontrar un conjunto de reglas. Como resultado de esta investigación, la ISO desarrolló un modelo de red que ayudaría a los fabricantes a crear redes que fueran compatibles y que pudieran operar con otras redes.

El *Modelo de Referencia OSI*, lanzado en 1984, fue el esquema descriptivo que crearon. Este modelo proporcionó a los fabricantes un conjunto de estándares que aseguraron una mayor compatibilidad e interoperabilidad entre los distintos tipos de tecnología de red utilizados por las empresas a nivel mundial.

El modelo de referencia OSI es el modelo principal para las comunicaciones por red. Aunque existen otros modelos, en la actualidad la mayoría de los fabricantes de redes relacionan sus productos con el modelo de referencia OSI, especialmente cuando desean enseñar a los usuarios cómo utilizar sus productos. Los fabricantes consideran que es la mejor herramienta disponible para enseñar a enviar y recibir datos a través de una red. Este modelo permite a los usuarios observar las funciones de red que se producen en cada capa. Más importante aún, el modelo de referencia OSI es un marco que se puede utilizar para comprender cómo viaja la información a través de una red. Además, puede usar el modelo de referencia OSI para visualizar cómo la información o los paquetes de datos viajan desde los programas de aplicación (hojas de cálculo, documentos, etc.), a través de un entorno de red (por ej., cables, etc.), hasta otro programa de aplicación ubicado en otra computadora de la red, aún cuando el remitente y el receptor tengan distintas tecnologías de red.

2.1.1 El modelo en capas

En el modelo de referencia OSI, hay siete capas numeradas, cada una de las cuales ilustra una función de red particular. Esta división de las funciones de red se denomina *división en capas*.

Si se realiza una división en capas para entender el funcionamiento de la red, obtenemos las siguientes ventajas:

- Divide el proceso de la comunicación de red en partes más pequeñas y sencillas.
- Normaliza los componentes de red para permitir el desarrollo y el soporte de los productos de diferentes fabricantes.
- Permite a los distintos tipos de hardware y software de red comunicarse entre sí.
- Impide que los cambios en una capa puedan afectar las demás capas, de manera que se puedan desarrollar con más rapidez e independencia.
- Simplifica el aprendizaje.

El modelo OSI define 7 capas como se muestra en la figura 2.1

Figura 2.1 Modelo OSI

Cada capa individual del modelo OSI tiene un conjunto de funciones que debe realizar para que los paquetes de datos puedan viajar en la red desde el origen hasta el destino. Es por esto que presentaremos una breve descripción de cada capa del modelo de referencia OSI tal como aparece en la figura 2.1. Teniendo claro cada una de las funciones que ocurren en cada capa, nos permitirá una comprensión más exacta del funcionamiento de una red de datos.

2.1.1.1 Funciones de las capas

La capa física

La capa física define las especificaciones eléctricas, mecánicas, de procedimiento y funcionales para activar, mantener y desactivar el enlace físico entre sistemas finales. Las características tales como niveles de voltaje, temporización de cambios de voltaje, velocidad de datos físicos, distancias de transmisión máximas, conectores físicos y otros atributos similares se definen a través de las especificaciones de la capa física.

La capa de enlace de datos

La capa de enlace de datos proporciona un tránsito de datos confiable a través de un enlace físico. Al hacerlo, la capa de enlace de datos se ocupa del direccionamiento físico,

la topología de red, el acceso al medio de transmisión, la notificación de errores, entrega ordenada de tramas y control de flujo así como la multiplexación, la cual se refiere al proceso en el que varios canales de datos se combinen en un solo canal físico para su transporte. Los puentes (bridges) actúan en este nivel en el grupo de protocolos. A continuación se presenta una lista de protocolos que ocupan este nivel.

- Control de enlace de datos de alto nivel (High-level Data Link Control HDLC). Manejadores y tecnologías LAN, como Ethernet o Token Ring.
- ATM para redes de área extensa WAN de transmisión rápida.
- Network Driver Interface Specification (NDIS) de Microsoft.
- Open Data link Interface (NODI) de Novell

La capa de red

La capa de red es una capa compleja que proporciona conectividad y selección de ruta entre dos sistemas de *host* que pueden estar ubicados en redes geográficamente distintas. Está relacionado con los procedimientos de conmutación y transmisión de datos y oculta dichos procedimientos a los niveles superiores. Los enrutadores actúan en este nivel. Este nivel vela por que los paquetes sean dirigidos a su destino en la red. Si está dirigido a un segmento de la red, este nivel lo envía a un dispositivo de enrutamiento el cual lo reenvía a su destino. Para recordar de manera sencilla esta capa, pensemos en selección de ruta, conmutación, direccionamiento lógico y enrutamiento. A continuación mostramos una lista de algunos protocolos que ocupan este nivel:

- Protocolo de Internet (IP)
- Protocolo X.25 [22]
- Internet work Packet Exchange (IPX) de Novell [30]
- Virtual Networking System (VINES) Internet Protocol (VIP) [55]
- Connection Less Network Protocol (CLNP) [19]

La capa de transporte

La capa de transporte segmenta los datos originados en el *host* emisor y los reensambla en una corriente de datos dentro del sistema del *host* receptor. El límite entre la capa de sesión y la capa de transporte puede imaginarse como el límite entre los protocolos de capa de medios y los protocolos de capa de *host*. Mientras que las capas de aplicación, presentación y sesión están relacionadas con aspectos de las aplicaciones, las tres capas inferiores se encargan del transporte de datos. La capa de transporte intenta suministrar un servicio de transporte de datos que aísla las capas superiores de los detalles de implementación del transporte. Específicamente, temas como la confiabilidad del transporte entre dos *host* es responsabilidad de la capa de transporte. Al proporcionar un servicio de comunicaciones, la capa de transporte establece, mantiene y termina adecuadamente los circuitos virtuales. La capa de transporte debe optimizar el empleo de los recursos de transmisión disponibles, a fin de asegurar lo más económicamente

posible el nivel de rendimiento requerido por cada usuario del servicio de transporte. Esta optimización se realizará por la toma en consideración del conjunto de peticiones formuladas por todos los usuarios simultáneos, en los límites de los recursos, puesto a disposición de la capa de transporte. En particular, para optimizar el uso de conexiones de redes, la capa de transporte podría ser dedicada a efectuar un multiplexado, es decir, a utilizar una conexión de red para soportar varias conexiones de transporte. Esta multiplexación es transparente para la capa de sesión.

Por el contrario, la capa de TRANSPORTE podría utilizar varias conexiones de red para soportar una conexión de transporte. Es el caso cuando existe mucho flujo de información entre dos terminales.

Al proporcionar un servicio confiable, se utilizan dispositivos de detección y recuperación de errores de transporte. Si se pierden datos del paquete, el protocolo del nivel de transporte coordina con el nivel de trasporte de origen para la retransmisión del paquete. Este nivel asegura que se reciban los datos en el orden apropiado. Los siguientes protocolos pueden estar en este nivel:

- Transmission Control Protocol (TCP) [34]
- User Datagram Protocol (UDP) [38]
- Sequenced Packed Exchange (SPX) [50]
- NetBios/NetBEUI [43]

La capa de sesión

Como su nombre lo implica, la capa de sesión establece, administra y finaliza las sesiones entre dos *host* que se están comunicando. La capa de sesión proporciona sus servicios a la capa de presentación. También sincroniza el diálogo entre las capas de presentación de los dos *host* y administra su intercambio de datos. Además de regular la sesión, esta capa ofrece disposiciones para una eficiente transferencia de datos, clase de servicio y un registro de excepciones acerca de los problemas de la capa de sesión, presentación y aplicación. Para recordar la Capa 5 en cuanto a su función, pensemos en diálogos y conversaciones.

La capa de presentación

La capa de presentación garantiza que la información que envía la capa de aplicación de un sistema pueda ser leída por la capa de aplicación de otro. De ser necesario, la capa de presentación traduce entre varios formatos de datos utilizando un formato común. También son interpretados los códigos dentro de los datos, como tabuladores y caracteres especiales. Asimismo es en este nivel donde se lleva a cabo el cifrado de datos y traducción desde otros juegos de caracteres. Para recordar de manera más rápida la función de esta capa, pensemos en un formato de datos común.

La capa de aplicación

La capa de aplicación es la capa del modelo OSI más cercana al usuario; suministra servicios de red a las aplicaciones del usuario. Difiere de las demás capas debido a que no proporciona servicios a ninguna otra capa del modelo OSI, sino solamente a aplicaciones que se encuentran fuera del modelo. Algunos ejemplos de dichos procesos de aplicación son los programas de hojas de cálculo, de procesamiento de texto y los de las terminales bancarias. La capa de aplicación establece la disponibilidad de los potenciales socios de comunicación, sincroniza y establece acuerdos sobre los procedimientos de recuperación de errores y control de la integridad de los datos. Una manera fácil de recordar las funciones de la Capa 7 es pensar en la utilidad de los navegadores de *Web*.

2.1.2 Encapsulado de datos.

Todas las comunicaciones de una red parten de un origen y se envían a un destino, y la información que se envía a través de una red se denomina datos. Si una maquina (*host* A) desea enviar datos a otra (*host* B), en primer término los datos deben empaquetarse a través de un proceso denominado encapsulamiento el cual se describe a continuación.

El encapsulamiento rodea los datos con la información de control necesaria antes de que se una al tránsito de la red. Por lo tanto, a medida que los datos se desplazan a través de las capas del modelo OSI, se va añadiendo la información de control necesaria para cada capa en forma de encabezados y colas.

Para ver cómo se produce el encapsulamiento, podemos examinar la forma en que viajan los datos a través de las capas, como lo ilustra la figura 2.2. Primero pasan a través de la capa de aplicación y recorren todas las demás capas en sentido descendente. Como puede verse, el empaquetamiento y el flujo de los datos que se intercambian experimentan cambios a medida que las redes ofrecen sus servicios a los usuarios finales.

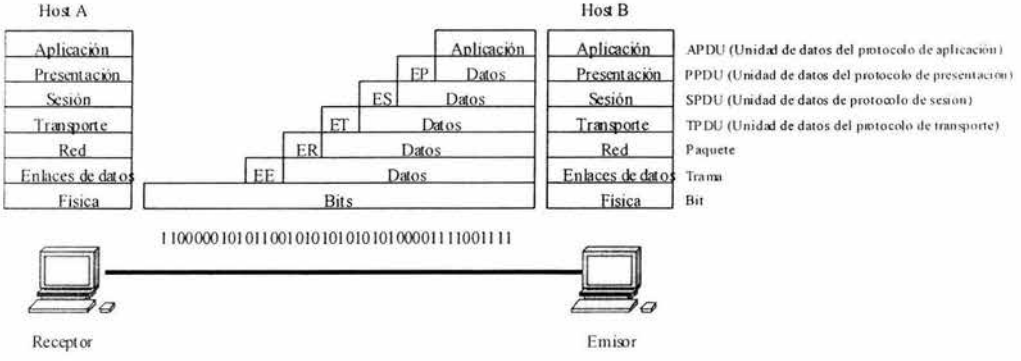


Figura 2.2 Encapsulado de datos en el modelo OSI

Los paquetes de datos de una red parten de un origen y se envían a un destino. Cada capa depende de la función de servicio de la capa OSI que se encuentra debajo de ella como se muestra en la figura 2.3. Para brindar este servicio, la capa inferior utiliza el encapsulamiento para colocar la PDU (*Packet Data Unit*) de la capa superior en su campo de datos, luego le puede agregar cualquier encabezado y cola que la capa necesite para ejecutar su función. Posteriormente, a medida que los datos se desplazan hacia abajo a través de las capas del modelo OSI, se agregan encabezados y colas adicionales. Después de que las Capas 7, 6 y 5 han agregado su información, la Capa 4 agrega su propia información de control. A este agrupamiento de datos, la PDU de la Capa 4, se le denomina ***segmento.***

Por ejemplo, la capa de red presta un servicio a la capa de transporte y la capa de transporte presenta datos al subsistema de red. La tarea de la capa de red consiste en trasladar esos datos a través de la red. Ejecuta esta tarea encapsulando los datos y agregando un encabezado, con lo que crea un **paquete** (PDU de la Capa 3). Este encabezado contiene la información necesaria para completar la transferencia, como por ejemplo, las direcciones lógicas origen y destino.

La capa de enlace de datos suministra un servicio a la capa de red. Encapsula la información de la capa de red en una ***trama*** (la PDU de la Capa 2) el encabezado de la trama contiene información (por ej., direcciones físicas) que es necesaria para completar las funciones de enlace de datos. La capa de enlace de datos suministra un servicio a la capa de red encapsulando la información de la capa de red en una **trama.**

La capa física también suministra un servicio a la capa de enlace de datos. La capa física codifica los datos de la trama de enlace de datos en un **patrón de unos y ceros (bits)** para su transmisión a través del medio de transmisión (generalmente un cable) en la Capa 1.

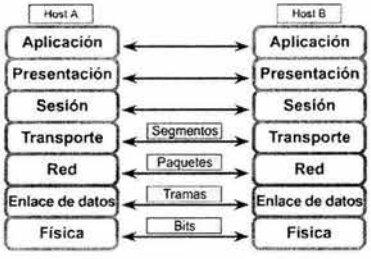


Figura 2.3 Relación de las capas en un host A origen y en un host B destino

2.2 El modelo TCP/IP

Aunque el modelo de referencia OSI sea universalmente reconocido, el estándar abierto de Internet desde el punto de vista histórico y técnico es el *Protocolo de control de Transmisión/Protocolo Internet* (TCP/IP) [33].

El modelo de referencia TCP/IP mostrado en la figura *2.4* y la pila de protocolos TCP/IP hacen que sea posible la comunicación entre dos máquinas, desde cualquier parte del mundo, a casi la velocidad de la luz. El modelo TCP/IP tiene importancia histórica, al igual que las normas que permitieron el desarrollo de la industria telefónica, de energía eléctrica, el ferrocarril, la televisión, etc.

El Departamento de Defensa de EE.UU. (*DoD*) creó el modelo TCP/IP porque necesitaba una red que pudiera sobrevivir ante cualquier circunstancia, incluso una guerra nuclear. Para brindar un ejemplo más amplio, supongamos que el mundo está en estado de guerra, atravesado en todas direcciones por distintos tipos de conexiones: cables, microondas, fibras ópticas y enlaces satelitales. Imaginemos entonces que se necesita que fluya la información o los datos (organizados en forma de paquetes), independientemente de la condición de cualquier nodo o red en particular de Internet (que en este caso podrían haber sido destruidos por la guerra). El DoD desea que sus paquetes lleguen a destino siempre, bajo cualquier condición, desde un punto determinado hasta cualquier otro. Este problema de diseño de difícil solución fue lo que llevó a la creación del modelo TCP/IP, que desde entonces se transformó en el estándar a partir del cual se desarrolló Internet.

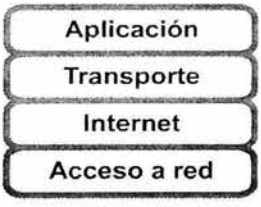


Figura 2.4 Modelo TCP/IP

El modelo TCP/IP tiene cuatro capas: la capa de aplicación, la capa de transporte, la *capa de Internet* y la capa de acceso de red. Es importante observar que algunas de las capas del modelo TCP/IP poseen el mismo nombre que las capas del modelo OSI. Es importante no confundir las capas de los dos modelos, porque la capa de aplicación tiene diferentes funciones en cada modelo.

2.2.1 Funciones de las capas

Capa de aplicación

Los diseñadores de TCP/IP sintieron que los protocolos de nivel superior deberían incluir los detalles de las capas de sesión y presentación. Simplemente crearon una capa de aplicación que maneja protocolos de alto nivel, aspectos de representación, codificación

y control de diálogo. El modelo TCP/IP combina todos los aspectos relacionados con las aplicaciones en una sola capa y garantiza que estos datos estén correctamente empaquetados para la siguiente capa.

Capa de transporte

La capa de transporte se refiere a los aspectos de calidad del servicio con respecto a la confiabilidad, el control de flujo y la corrección de errores. Uno de sus protocolos, el protocolo para el control de la transmisión (TCP), ofrece maneras flexibles y de alta calidad para crear comunicaciones de red confiables, sin problemas de flujo y con un nivel de error bajo. TCP es un protocolo orientado a la conexión. Mantiene un diálogo entre el origen y el destino mientras empaqueta la información de la capa de aplicación en unidades denominadas segmentos. Orientado a la conexión no significa que el circuito exista entre las maquinas que se están comunicando (esto sería una conmutación de circuito). Significa que los segmentos de Capa 4 viajan de un lado a otro entre dos *hosts* para comprobar que la conexión exista lógicamente para un determinado período. Esto se conoce como **conmutación de paquetes.**

Capa de Internet

El propósito de la *capa de Internet* es enviar paquetes origen desde cualquier red en Internet y que estos paquetes lleguen a su destino independientemente de la ruta y de las redes que recorrieron para llegar hasta allí. El protocolo específico que rige esta capa se denomina Protocolo Internet (IP). En esta capa se produce la determinación de la mejor ruta y la conmutación de paquetes.

Capa de acceso de red

El nombre de esta capa es muy amplio y se presta a confusión. También se denomina capa de *host* a red. Es la capa que se ocupa de todos los aspectos que requiere un paquete IP para transportarse a través de un enlace físico. Esta capa incluye los detalles de tecnología LAN y WAN y todos los detalles de las capas físicas y de enlace de datos del modelo de referencia OSI.

2.2.2 Gráfico de protocolo de TCP/IP

El diagrama que aparece en la figura 2.5 se denomina *gráfico de protocolo*. Este gráfico ilustra algunos de los protocolos comunes especificados por el modelo de referencia TCP/IP. En la capa de aplicación, aparecen distintas tareas de red que probablemente no sean muy conocidas, pero como usuario de Internet, probablemente sean usadas todos los días.

Estas aplicaciones incluyen las siguientes:

- *FTP*: Protocolo de Transferencia de Archivos. [35]
- *HTTP*: Protocolo de Transferencia de Hipertexto. [52]
- *SMTP*: Protocolo de Transferencia de Correo Simple. [37]
- *DNS*: Sistema de Nombres de Dominio. [36]
- *TFTP*: Protocolo Trivial de Transferencia de Archivo. [41]

El modelo TCP/IP enfatiza la máxima flexibilidad, en la capa de aplicación, para los creadores de software. La capa de transporte involucra dos protocolos: el Protocolo de Control de Transmisión (TCP) y el Protocolo de Datagrama de Usuario (UDP).

La capa inferior, la capa de acceso de red, se relaciona con la tecnología específica de LAN o WAN que se utiliza.

En el modelo TCP/IP existe solamente un protocolo de red: el Protocolo Internet, o IP, independientemente de la aplicación que solicita servicios de red o del protocolo de transporte que se utiliza. Esta es una decisión de diseño deliberada. IP sirve como protocolo universal que permite que cualquier maquina en cualquier parte del mundo pueda comunicarse en cualquier momento.

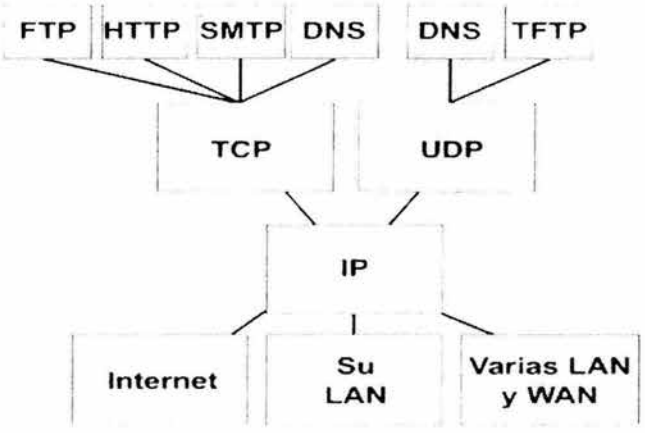


Figura 2.5 Gráfico de protocolo: TCP/IP

2.2.3 Direcciones IP

La dirección IP contiene la información necesaria para enrutar un paquete a través de la red. Cada dirección origen y destino contiene una dirección de 32 bits. El campo de dirección origen contiene la dirección IP del dispositivo que envía el paquete. El campo destino contiene la dirección IP del dispositivo que recibe el paquete.

Las direcciones IP se expresan como números de notación decimal: se dividen los 32 bits de la dirección en cuatro *octetos*. El valor decimal máximo de cada octeto es 255 (el número binario de 8 bits más alto es 11111111, y esos bits, de derecha a izquierda, tienen valores decimales de 1, 2, 4, 8, 16, 32, 64 y 128, lo que suma 255 en total).

Hay tres clases de direcciones IP como se muestra en la figura 2.6 que pueden ser asignadas a una organización: Clases A, B y C. Normalmente las direcciones de clase A fueron asignadas a empresas de gran envergadura como, por ejemplo, Hewlett Packard) y las direcciones de Clase B para las medianas empresas. Se otorgan direcciones de Clase C para todos los demás solicitantes.

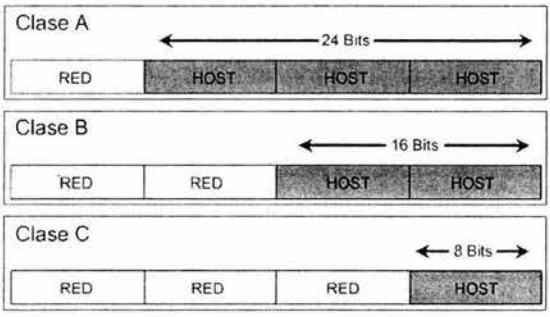


Figura 2.6 Clases de direcciones IP

Clase A

Cuando está escrito en formato binario, el primer bit (el bit que está ubicado más a la izquierda) de la dirección de Clase A siempre es 0. Un ejemplo de una dirección IP de clase A es 124.95.44.15. El primer octeto, 124, identifica el número de red asignado a la organización. Los administradores internos de la red asignan los 24 bits restantes. Una manera fácil de reconocer si un dispositivo forma parte de una red de Clase A es verificar el primer octeto de su dirección IP, cuyo valor debe estar entre 0 y 126. (127 *comienza* con un bit 0, pero está reservado para fines especiales).

Todas las direcciones IP de Clase A utilizan solamente los primeros 8 bits para identificar la parte de la red de la dirección. Los tres octetos restantes se pueden utilizar para la parte del *host* de la dirección. A cada una de las redes que utilizan una dirección IP de Clase A se les pueden asignar hasta (2^{24}-2) ó 16,777,214 direcciones IP posibles para los dispositivos que están conectados a la red.

Clase B

Los primeros 2 bits de una dirección de Clase B siempre son 10 (uno y cero). Un ejemplo de una dirección IP de Clase B es 151.10.13.28. Los dos primeros octetos identifican el número de red. Los administradores internos de la red asignan los 16 bits restantes. Una manera fácil de reconocer si un dispositivo forma parte de una red de Clase B es verificar el primer octeto de su dirección IP, el cual debe estar comprendido entre 128 y 191.

Todas las direcciones IP de Clase B utilizan los primeros 16 bits para identificar la parte de la red de la dirección. Los dos octetos restantes de la dirección IP se encuentran reservados para la porción del *host* de la dirección. Cada red que usa un esquema de direccionamiento IP de Clase B puede tener asignadas hasta (2^{16}-2), ó 65,534 direcciones IP posibles a dispositivos conectados a su red.

Clase C

Los 3 primeros bits de una dirección de Clase C siempre son 110 (uno, uno y cero). Un ejemplo de dirección IP de Clase C es 201.110.213.28. Los tres primeros octetos identifican el número de red.

Los administradores internos de la red asignan los 8 bits restantes. Una manera fácil de reconocer si un dispositivo forma parte de una red de Clase C es verificar el primer octeto de su dirección IP, el cual está comprendido entre 192 y 223.

Todas las direcciones IP de Clase C utilizan los primeros 24 bits para identificar la porción de red de la dirección. Sólo se puede utilizar el último octeto de una dirección IP de Clase C para la parte de la dirección que corresponde al *host*. A cada una de las redes que utilizan una dirección IP de Clase C se les pueden asignar hasta (2^8-2), ó 254 direcciones IP posibles para los dispositivos que están conectados a la red.

Dirección de subred

El campo de subred siempre se ubica inmediatamente a continuación del número de red (figura 2.7). Es decir, los bits que se pidieron prestados deben ser los primeros *n* bits del campo de *host* por defecto, donde *n* es el tamaño deseado del nuevo campo de subred.

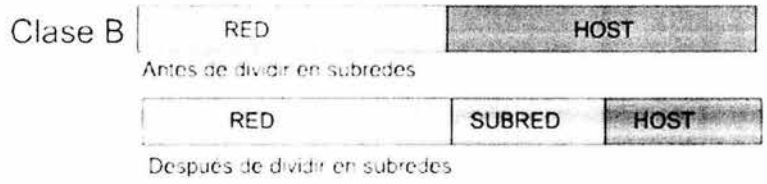


Figura 2.7 Asignación de las subredes

La mascara de subred es la herramienta que utiliza el enrutador para determinar cuáles son los bits que corresponden a los bits de enrutamiento y cuáles son los bits que corresponden a los bits de *host*, (figura2.8)

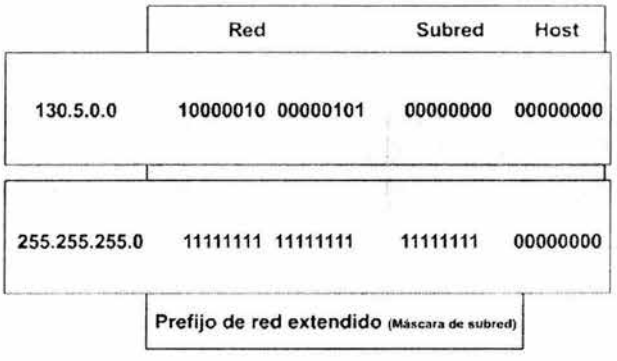


Figura 2.8 Mascaras de subred

Las máscaras de subred usan el mismo formato que las direcciones IP. Tienen una longitud de 32 bits y están divididas en cuatro octetos, escritos en formato decimal

separado por puntos. Las máscaras de subred tienen todos unos en las posiciones de bit de red (determinadas por la clase de dirección) así como también las posiciones de bit de subred deseadas, y tienen todos ceros en las posiciones de bit restantes, designándolas como la porción de *host* de una dirección.

Por defecto, si no se pide ningún bit prestado, la máscara de subred para una red de Clase B sería 255.255.0.0, que es el equivalente en notación decimal punteada de los 1s en los 16 bits que corresponden al número de red de Clase B.

Si se pidieran prestados 8 bits para el campo de subred, la máscara de subred incluiría 8 bits 1 adicionales y se transformaría en 255.255.255.0.

Por ejemplo, si la máscara de subred 255.255.255.0 se asociara con la dirección de Clase B 130.5.2.144 (8 bits que se han pedido prestados para la división en subredes), el enrutador sabría que debe enrutar este paquete hacia la subred 130.5.2.0 en lugar de hacerlo simplemente a la red 130.5.0.0

2.2.4 Datagrama IP

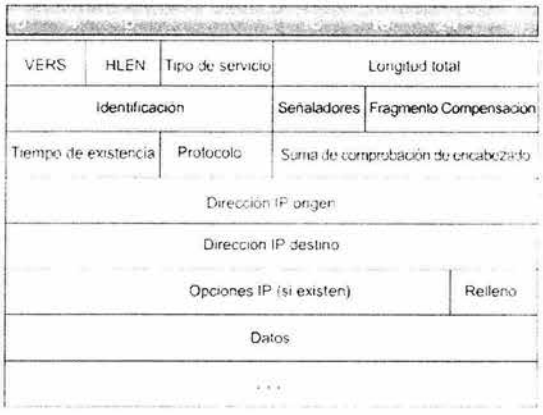


Figura 2.9 Datagrama IP

- *Versión:* Indica la versión de IP que se usa en el momento (4 bits)
- *Longitud del encabezado IP (HLEN):* Indica la longitud del encabezado del datagrama en palabras de 32 bits (4 bits)
- *Tipo de servicio:* Especifica el nivel de importancia que le ha sido asignado por un protocolo de capa superior en particular (8 bits)
- *Longitud total:* Especifica la longitud de todo el paquete IP, incluyendo datos y encabezado, en bytes (16 bits)
- *Identificación:* Contiene un número entero que identifica el datagrama actual (16 bits)

- *Señaladores:* Un campo de 3 bits en el que los dos bits de orden inferior controlan la fragmentación; un bit que especifica si el paquete puede fragmentarse y el segundo si el paquete es el último fragmento en una serie de paquetes fragmentados (3 bits)
- *Compensación de fragmentos:* El campo que se utiliza para ayudar a reunir los fragmentos de datagramas (16 bits)
- *Tiempo de existencia:* Indica el período en el cual se considera válido el paquete o datagrama. Mantiene un contador cuyo valor decrece en cada enrutador por donde pasa el paquete hasta llegar a cero. Cuando se llega a ese punto se descarta el datagrama, impidiendo así que los paquetes entren en un loop interminable (8 bits)
- *Protocolo:* Indica cuál es el protocolo de capa superior que recibe los paquetes entrantes después de que se ha completado el procesamiento IP (8 bits)
- *Suma de comprobación del encabezado:* Ayuda a garantizar la integridad del encabezado IP (16 bits)
- *Dirección origen:* Especifica el nodo emisor (32 bits)
- *Dirección destino:* Especifica el nodo receptor (32 bits)
- *Opciones:* Permite que IP soporte varias opciones, como la seguridad (longitud variable)
- *Datos:* Contiene información de capa superior (longitud variable, máximo 64 kb)
- *Relleno*: se agregan ceros adicionales a este campo para garantizar que el encabezado IP siempre sea un múltiplo de 32 bits

2.3 Comparación entre el modelo OSI y el modelo TCP/IP

Si comparamos el modelo OSI y el modelo TCP/IP, podemos observar que ambos presentan similitudes y diferencias. Para poder apreciar estas características, listemos los siguientes ejemplos:

Similitudes

- Ambos se dividen en capas
- Ambos tienen capas de aplicación, aunque incluyen servicios muy distintos
- Ambos tienen capas de transporte y de red similares
- La tecnología es de conmutación por paquetes (no de conmutación por circuito)
- Los profesionales en el área de redes deben conocer ambos

Diferencias

- TCP/IP combina las funciones de la capa de presentación y de sesión en la capa de aplicación
- TCP/IP combina la capa de enlace de datos y la capa física del modelo OSI en una sola capa
- TCP/IP parece ser más simple porque tiene menos capas
- Los protocolos TCP/IP son los estándares en torno a los cuales se desarrolló Internet, de modo que la credibilidad del modelo TCP/IP se debe en gran parte a sus protocolos. En comparación, las redes típicas no se desarrollan normalmente a partir del protocolo OSI, aunque el modelo OSI se usa como guía de referencia, figura 2.10.

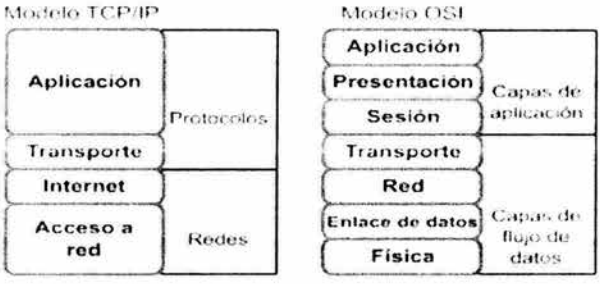


Figura 2.10 Comparación entre TCP/IP y OSI

2.4 Redes LAN

2.4.1 Descripción general y sus variantes

Las redes LAN son redes de datos de alta velocidad y bajo nivel de errores que abarcan un área geográfica relativamente pequeña (1-1000m). Las LAN conectan estaciones de trabajo, dispositivos periféricos, terminales y otros dispositivos que se encuentran en un solo edificio u otra área geográfica limitada.

Se usan para conectar computadoras personales o estaciones de trabajo, con objeto de compartir recursos e intercambiar información. Están restringidas en tamaño, lo cual significa que el tiempo de transmisión, en el peor de los casos, se conoce, lo que permite cierto tipo de diseños (deterministas) que de otro modo podrían resultar ineficientes. Además, simplifica la administración de la red. Suelen emplear tecnologías de difusión mediante un cable sencillo al que están conectadas todas las máquinas.

Operan a velocidades entre 10 y 100 Mbps. Además, tienen bajo retardo y experimentan pocos errores.

A su nivel más elemental, una LAN no es más que un medio compartido (como un cable coaxial al que se conectan todas las computadoras y las impresoras) junto con una serie

de reglas que rigen el acceso a dicho medio. La LAN más difundida y utilizada es la de tipo Ethernet.

Además de proporcionar un acceso compartido, las redes LAN modernas también proporcionan al usuario multitud de funciones avanzadas. Hay paquetes de software de gestión y monitoreo para controlar la configuración de los equipos en la LAN, la administración de los usuarios, y el control de los recursos de la red. Una estructura muy utilizada consiste en varios servidores a disposición de distintos usuarios. Los primeros, por lo general máquinas más potentes, proporcionan servicios como control de impresión, archivos compartidos y correo a los últimos, por lo general computadoras personales.

Hay topologías de red muy diversas (bus, estrella, anillo) y diferentes protocolos de acceso. A pesar de esta diversidad, todas las LAN comparten la característica de poseer un alcance limitado (normalmente abarcan un edificio) y de tener una velocidad suficiente para que la red de conexión resulte invisible para los equipos que la utilizan.

2.4.2 Topologías físicas

La *topología* define la estructura de una red. La definición de topología está compuesta por dos partes, la topología física, que es la disposición real de los cables (los medios) y la topología lógica, que define la forma en que los *hosts* acceden a los medios.

Las topologías físicas que se utilizan comúnmente son de bus, de anillo, en estrella, en estrella extendida, jerárquica y en malla. Estas topologías se indican en la figura 2.11.

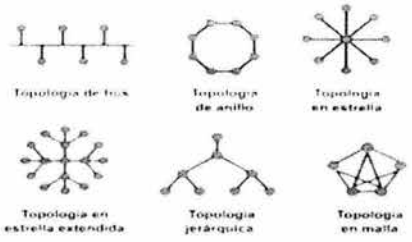


Figura 2.11 Topologías físicas

- La topología de bus utiliza un único segmento al que todos los *hosts* se conectan de forma directa.
- La topología de anillo conecta un *host* con el siguiente y al último *host* con el primero. Esto crea un anillo físico de cable.

- La topología en estrella conecta todos los cables con un punto central de concentración. Por lo general, este punto es un *hub* o un *switch*.
- La topología en estrella extendida se desarrolla a partir de la topología en estrella. Esta topología enlaza estrellas individuales enlazando los *hubs/switches*. Esto permite extender la longitud y el tamaño de la red.
- La topología jerárquica se desarrolla de forma similar a la topología en estrella extendida pero, en lugar de enlazar los *hubs/switches*, el sistema se enlaza con una computadora que controla el tráfico de la topología.
- La topología en malla se utiliza cuando no puede existir absolutamente ninguna interrupción en las comunicaciones, por ejemplo, en los sistemas de control de una central nuclear. De modo que, como puede observar en el gráfico, cada *host* tiene sus propias conexiones con los demás *hosts*. Esto también se refleja en el diseño de la Internet, que tiene múltiples rutas hacia cualquier ubicación.

2.4.3 Topologías lógicas

La topología lógica de una red es la forma en que los *hosts* se comunican a través del medio. Los dos tipos más comunes de topologías lógicas son broadcast y transmisión de *tokens*.

La topología de broadcast simplemente significa que cada *host* envía sus datos hacia todos los demás *hosts* del medio de red. Las estaciones no siguen ningún orden para utilizar la red, el orden es el primero que entra, el primero que la utiliza. Esta es la forma en que funciona Ethernet y 802.3.

El segundo tipo es transmisión de *tokens*. La transmisión de *tokens* controla el acceso a la red al transmitir un *token* electrónico de forma secuencial a cada *host*. Cuando un *host* recibe el *token*, eso significa que el *host* puede enviar datos a través de la red. Si el *host* no tiene algún dato para enviar, transmite el *token* hacia el siguiente *host* y el proceso se vuelve a repetir.

2.4.4 Dispositivos LAN

Host

Los dispositivos que se conectan de forma directa a un segmento de red se denominan *hosts*. Los dispositivos *host* no forman parte de ninguna capa. Tienen una conexión física con los medios de transmisión por medio de una tarjeta de interfaz de red (NIC) y las otras capas del modelo de referencia OSI se ejecutan en el software ubicado dentro del *host*, ejecutando todo el proceso de encapsulamiento y desencapsulamiento para realizar la tarea de enviar mensajes de correo electrónico, imprimir informes, escanear figuras o acceder a las bases de datos.

Medios de transmisión

Las funciones básicas de los medios de transmisión consisten en transportar un flujo de información, a través de una LAN. Salvo en el caso de las LAN inalámbricas (que usan la

atmósfera, o el espacio, como el medio), por lo general, los medios de transmisión son un cable o fibra. Los medios de transmisión se consideran componentes de Capa 1 de las LAN.

Se pueden desarrollar redes informáticas con varios tipos de medios distintos. Cada medio tiene sus ventajas y desventajas; lo que constituye una ventaja para uno de los medios (costo de la categoría 5) puede ser una desventaja para otro de los medios (costo de la fibra óptica). Algunas de las ventajas y las desventajas son las siguientes:

- Longitud del cable
- Costo
- Facilidad de instalación
- Cantidad total de computadoras en los medios.

El cable coaxial, la fibra óptica o incluso el espacio abierto pueden transportar señales de red, sin embargo, el medio principal utilizado en las redes de datos tipo LAN se denomina cable de par trenzado sin blindaje categoría 5 (UTP CAT 5).

Equipos de interconexión entre redes LAN

- Si es necesario extender la red más allá de este límite definido por los medios de transmisión, se debe agregar un dispositivo a la red, figura 2.12. Este dispositivo se denomina *repetidor*. El propósito de un repetidor es regenerar y retemporizar las señales de red a nivel de señales eléctricas, de tal forma que los bits representados por estas, puedan viajar a mayor distancia a través de los medios de transmisión.

- El propósito de un *hub* es regenerar y retemporizar las señales de red. Esto se realiza para un gran número de *host* (por ej., 4, 8 o incluso 24) utilizando un proceso denominado concentración. Se puede observar que esta definición es muy similar a la del repetidor, es por ello que el hub también se denomina repetidor multipuerto. La diferencia es la cantidad de cables que se conectan al dispositivo. Las razones por las que se usan los *hubs* son crear un punto de conexión central para los medios de cableado y aumentar la confiabilidad de la red. La confiabilidad de la red se ve aumentada al permitir que cualquier cable falle sin provocar una interrupción en toda la red.

- Un puente es un dispositivo de capa 2 diseñado para conectar dos segmentos de LAN. El propósito de un puente es filtrar el tráfico de una LAN, permitiendo bloquear el tráfico local de un segmento hacia el otro segmento, aumentando así la eficiencia de la red.

Aunque los enrutadores y los *switches* han adoptado muchas de las funciones del puente, estos siguen teniendo importancia en muchas redes. Para comprender la conmutación y el enrutamiento, primero debe comprender cómo funciona un puente.

- Un *switch*, al igual que un puente, es un dispositivo de la capa 2. De hecho, el *switch* se denomina puente multipuerto, así como el hub se denomina repetidor multipuerto. La diferencia entre el hub y el *switch* es que los *switches* toman decisiones basándose en las direcciones MAC y los *hubs* no toman ninguna decisión. Como los *switches* son capaces de tomar decisiones, hacen que la LAN sea mucho más eficiente. Los *switches* hacen esto "conmutando" datos sólo desde el puerto al cual está conectado el *host* correspondiente. A diferencia de esto, el hub envía datos a través de todos los puertos de modo que todos los *host* deban ver y procesar (aceptar o rechazar) todos los datos.

- El enrutador está ubicado en la capa de red del modelo OSI, o capa 3. Al trabajar en la capa 3, esto permite que el enrutador tome decisiones basándose en grupos de direcciones de red en lugar de las direcciones MAC individuales. Los enrutadores también pueden conectar distintas tecnologías de la capa 2 como, por ejemplo, Ethernet, Token-ring y FDDI. Dada su capacidad para enrutar paquetes basándose en la información de la Capa 3, los enrutadores se han transformado en el núcleo de Internet, ejecutando el protocolo IP. El propósito de un enrutador es examinar los paquetes entrantes (datos de la capa 3), elegir cuál es la mejor ruta para ellos a través de la red y luego conmutarlos hacia el puerto de salida adecuado. Los enrutadores son los dispositivos de regulación de tráfico más importantes en las redes de gran tamaño e importancia. Permiten que prácticamente cualquier tipo de *host* se pueda comunicar con otro *host* en cualquier parte del mundo.

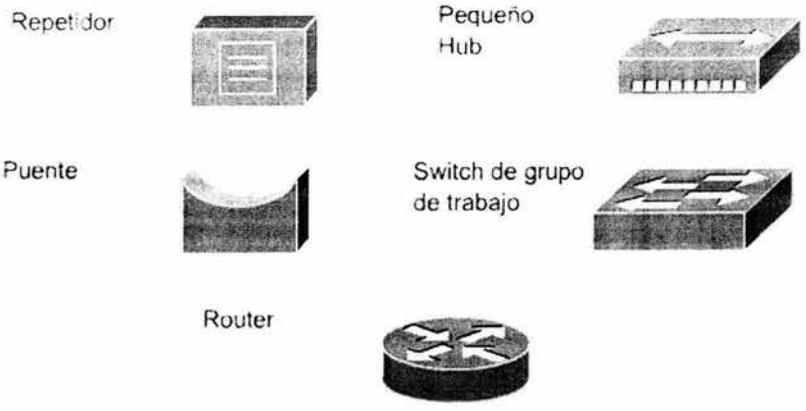


Figura 2.12 Equipos de interconexión de LAN´s

2.4.5 Tecnologías LAN

2.4.5.1 Token Ring

Descripción general y sus variantes

IBM desarrolló la primera red Token Ring en los años setenta. Todavía sigue siendo la principal tecnología LAN de IBM y desde el punto de vista de implementación de LAN ocupa el segundo lugar después de Ethernet (IEEE 802.3). La especificación **IEEE 802.5** es prácticamente idéntica a la red Token Ring de IBM, y absolutamente compatible con ella. La especificación IEEE 802.5 se basó en el Token Ring de IBM y se ha venido evolucionando en paralelo con este estándar. El término Token Ring se refiere tanto al Token Ring de IBM como a la especificación 802.5 del IEEE. En la figura 2.13 se destacan las similitudes y diferencias principales entre los dos estándares.

	Red Token Ring de IBM	IEEE 802.5
Velocidad de los datos	4 o 16 Mbps	4 o 16 Mbps
Estaciones/segmentos	260 (Par trenzado blindado) 72 (Par trenzado sin blindaje)	250
Topología	Estrella	No especificado
Medios	Par trenzado	No especificada
Señalización	Banda base	Banda base
Metodo de acceso	Transmisión de tokens	Transmisión de tokens
Codificación	Diferencial Manchester	Diferencial Manchester

Figura 2.13 Comparación entre la red Token Ring de IBM e IEEE 802.5

Token Ring emplea una topología lógica de anillo y una topología física de estrella. La NIC de cada computadora se conecta a un cable que, a su vez, se enchufa a un hub central llamado unidad de acceso a multiestaciones (MAU).

Para el acceso al medio, Token Ring se basa en un esquema de paso de señales (*token* passing), es decir que pasa un *token* (o señal) a todas las computadoras de la red. La computadora que esté en posesión del *token* tiene autorización para transmitir su información a otra computadora de la red. Cuando termina, el *token* pasa a la siguiente computadora del anillo. Si la siguiente computadora tiene que enviar información, acepta el *token* y procede a enviarla.

En caso contrario, el *token* pasa a la siguiente computadora del anillo y el proceso continúa. La MAU se salta automáticamente un nodo de red que no esté encendido. Sin embargo, dado que cada nodo de una red Token Ring examina y luego retransmite cada *token* (señal), un nodo con mal funcionamiento puede hacer que deje de trabajar toda la red. Token Ring tiende a ser menos eficiente que CSMA/CD (de Ethernet) en redes con poca actividad, pues requiere una sobrecarga adicional. Sin embargo, conforme aumenta la actividad de la red, Token Ring llega a ser más eficiente que CSMA/CD.

Formato de trama de Token Ring

Las redes Token Ring definen dos tipos de tramas: ***data/command*** y ***tokens***. Ambos formatos se muestran en la figura 2.14.

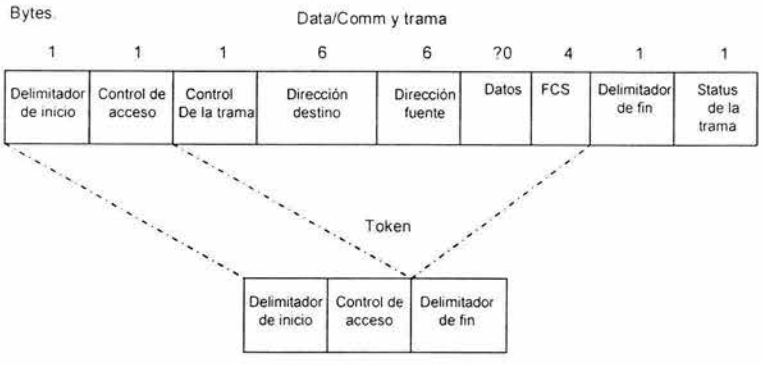


Figura 2.14 Frame de Token Ring

Tramas Data/Command

- Las tramas *Data/command* varían en tamaño, dependiendo del tamaño del campo de datos. Llevan información hacia protocolos de otro nivel. Las tramas *command* contienen información de control y no contienen datos para llevar a otros protocolos.

- En las tramas *Data/command*, hay un byte de control de trama después del byte de control de acceso. El byte de control de trama indica cuando la trama contiene datos o información de control.

- Seguido del byte de control de trama hay dos campos de direcciones los cuáles identifican las estaciones destino y fuente.

- El campo de datos se encuentra después de los campos de direcciones. La longitud de este campo está limitado por el *ring token holding time*, el cuál define el máximo tiempo que una estación puede tener el *token*.

- Seguido del campo de datos está el campo *frame check sequence* (FCS). Este campo es llenado por la terminal fuente con un valor calculado dependiendo del contenido de la trama. La estación de destino recalcula este valor para determinar si la trama tuvo algún daño durante el tiempo que se movió, si sí, la trama es descartada.

- El delimitador del final indica el final de un *token* o trama de data/command, este también contiene bits para señalar una trama dañada e identificar que esta trama sea la última en una secuencia lógica.

- Campo Frame Status, Es un campo de 1 byte que se utiliza para terminar una trama de data/command. Este campo incluye el indicador de confirmación de dirección y el indicador del copiado de la trama.

Token

- Los *token* son de 3 bytes de longitud y consisten en un delimitador de inicio, un byte de control de acceso y un delimitador final.

- El delimitador de inicio, alerta a cada estación de la llegada de un *token* (o una trama *data/command*). Este campo incluye señales que distinguen este byte del resto de la trama por una violación al esquema usado en la misma.

- El byte de control de acceso contiene los campos de prioridad y reservación, como un *token* bit (usado para diferenciar un *token* de una trama *data/command*) y un bit monitor (usado por el monitor activo para determinar cuando una trama está circulando en el anillo a baja velocidad.

- Finalmente, las señales finales de delimitación señalan el final del *token* o de la trama *data/command*. Aquí también están contenidos los bits que muestran si el *token* está dañado.

Transmisión de token

Los *token* se transportan en una pequeña trama a través de la red como se muestra en la figura 2.15. La posesión del *token* otorga el derecho a transmitir datos. Si un nodo que recibe un *token* no tiene información para enviar, transfiere el *token* a la siguiente estación terminal. Cada estación puede mantener al *token* durante un período de tiempo máximo determinado, según la tecnología específica que se haya implementado.

Cuando una estación tiene información para transmitir, toma el *token* el cual en su campo de control de acceso compuesto por bits de *token*, bits de monitor, bits de prioridad y bits de reserva, le modifica a 1 el bit de *token*. Por lo cual el *token* se transforma en una secuencia de inicio de trama. A continuación, la estación agrega la información para transmitir al *token* y envía estos datos a la siguiente estación del anillo. No hay ningún *token* en la red mientras la trama de información gira alrededor del anillo. En este momento, las otras estaciones del anillo no pueden realizar transmisiones. Deben esperar a que el *token* esté disponible, las redes Token Ring no tienen colisiones.

Si el anillo acepta el envío anticipado del *token*, se puede emitir un nuevo *token* cuando se haya completado la transmisión de la trama.

A diferencia de las redes CSMA/CD (Carrier Sense Multiple Access with Collision Detection), como Ethernet, las redes de transmisión de *tokens* son deterministicas. Esto significa que se puede calcular el tiempo máximo que transcurrirá antes de que cualquier estación terminal pueda realizar una transmisión. Esta característica, y varias características de confiabilidad, hacen que las redes Token Ring sean ideales para las aplicaciones en las que cualquier demora deba ser predecible y en las que el funcionamiento sólido de la red sea importante. Los entornos de automatización de fábricas son ejemplos de operaciones de red que deben ser sólidas y predecibles.

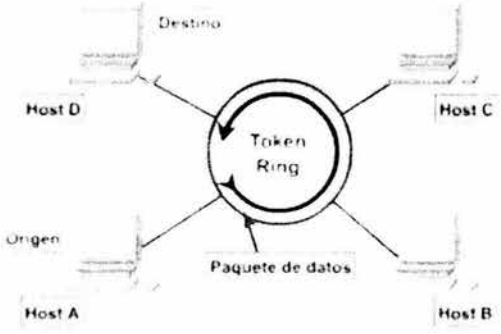


Figura 2.15 Transmisión de tokens de Token Ring

La topología física en estrella de la red Token Ring de IBM también contribuye a la confiabilidad general de la red. Las *MSAU* (*unidades de acceso de estación múltiple*) activas pueden ver toda la información de una red Token Ring, lo que les permite verificar si existen problemas y, de ser necesario, eliminar estaciones del anillo de forma selectiva. *Beaconing*, una de las fórmulas Token Ring, detecta e intenta reparar las fallas de la red. Cuando una estación detecta la existencia de un problema grave en la red (por ejemplo, un cable roto), envía una trama *beacon*. La trama *beacon* define un *dominio de error*. Un dominio de error incluye la estación que informa acerca del error, su *vecino corriente arriba activo más cercano (NAUN)* y todo lo que se encuentra entre ellos. El *beaconing* inicia un proceso denominado *autoreconfiguración*, en el que los nodos situados dentro del dominio de error automáticamente ejecutan diagnósticos. Este es un intento de reconfigurar la red alrededor de las áreas en las que hay errores. Físicamente, las MSAU pueden lograrlo a través de la reconfiguración eléctrica.

2.4.5.2 Interfase de Datos Distribuidos por Fibra (FDDI)

A mediados de los años ochenta, las estaciones de trabajo de alta velocidad para uso en ingeniería habían llevado las capacidades de las tecnologías Ethernet y Token Ring existentes hasta el límite de sus posibilidades. Los ingenieros necesitaban una LAN que pudiera soportar sus estaciones de trabajo y las nuevas aplicaciones. Al mismo tiempo, los administradores de sistemas comenzaron a ocuparse de los problemas de confiabilidad de la red ya que se implementaban aplicaciones críticas de las empresas en las redes de alta velocidad.

Para solucionar estos problemas, la comisión normalizadora ANSI X3T9.5 creó el estándar *Interfase de Datos Distribuida por Fibra (FDDI)*. Después de completar las especificaciones, el ANSI envió la FDDI a la Organización Internacional de Normalización (ISO), la cual creó entonces una versión internacional de dicha interfaz que es absolutamente compatible con la versión estándar del ANSI.

En la actualidad las implementaciones de FDDI no son tan comunes como Ethernet o Token Ring.

Especificaciones de FDDI

FDDI tiene cuatro especificaciones:

1. *Control de acceso al medio (MAC)*: Define la forma en que se accede al medio, incluyendo:
   - Formato de Trama
   - Tratamiento del Token
   - Direccionamiento
   - Código de Redundancia Cíclica y mecanismos de corrección de errores.
2. *Protocolo de capa física (PHY)*: define los procedimientos de codificación o decodificación, incluyendo:
   - Requisitos de reloj
   - Entramado
   - Otras funciones
3. *Medio de capa física (PMD)*: Define las características del medio de transmisión, incluyendo:
   - Enlace de fibra óptica
   - Niveles de energía
   - Tasas de bit en error
   - Componentes ópticos
   - Conectores

4. *Administración de estaciones(SMT)*: define la configuración de la estación FDDI, incluyendo:
   - Configuración del anillo
   - Características de control del anillo
   - Inserción y eliminación de una estación
   - Inicialización
   - Aislamiento y recuperación de fallas
   - Programación
   - Recopilación de estadística

FDDI utiliza una estrategia de transmisión de *tokens* sobre una topología lógica tipo anillo similar a la de Token Ring. No se producen colisiones en las redes FDDI. Si se soporta el envió anticipado del *token*, se puede emitir un nuevo *token* cuando se haya completado la transmisión de la trama.

La trama de información gira alrededor del anillo hasta que llega la estación destino establecida, que copia la información para su procesamiento. La trama de información gira alrededor del anillo hasta que llega a la estación emisora y entonces se elimina. La estación emisora puede verificar en la trama que retorna si la trama se recibió y se copió en el destino.

Formato de trama de FDDI

Los campos de una trama FDDI se muestran en la figura 2.16 y son los siguientes:

- *Preámbulo*: Prepara cada estación para recibir la trama entrante
- *Delimitador de inicio*: indica el comienzo de una trama, y está formado por patrones de señalización que lo distinguen del resto de la trama
- *Control de trama*: indica el tamaño de los campos de dirección, si la trama contiene datos asíncronos o síncronos y otra información de control
- *Dirección destino*: contiene una dirección unicast (singular), multicast (grupal) o broadcast (cada estación); las direcciones destino tienen 6 bytes (por ejemplo, Ethernet y Token Ring)
- *Dirección origen*: identifica la estación individual que envió la trama. Las direcciones origen tienen 6 bytes (como Ethernet y Token Ring)
- *Datos*: información de control, o información destinada a un protocolo de capa superior

- *Secuencia de verificación de trama (FCS)*: la estación origen la completa con una verificación por redundancia cíclica (CRC) calculada, cuyo valor depende del contenido de la trama (como en el caso de Token Ring y Ethernet). La estación destino vuelve a calcular el valor para determinar si la trama se ha dañado durante el tránsito. La trama se descarta si está dañada.
- *Delimitador de fin*: contiene símbolos que no son datos que indican el fin de la trama
- *Estado de la trama*: permite que la estación origen determine si se ha producido un error y si la estación receptora reconoció y copió la trama

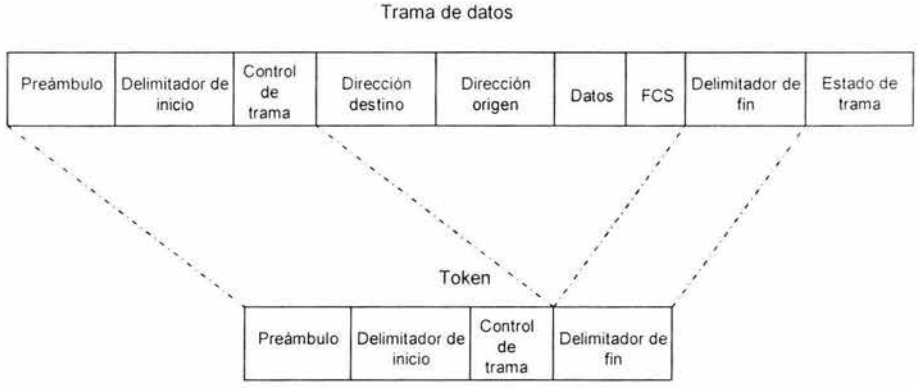


Figura 2.16 Formato de trama de FDDI

Clasificación de tráfico en FDDI

FDDI acepta la asignación en tiempo real del ancho de banda de la red, lo que la hace ideal para varios tipos de aplicación. La FDDI proporciona esta ayuda mediante la definición de dos tipos de tráfico: síncrono y asíncrono.

Síncrono

- El tráfico síncrono puede consumir una porción del ancho de banda total de 100 Mbps de una red FDDI, mientras que el tráfico asíncrono puede consumir el resto.
- El ancho de banda síncrono se asigna a las estaciones que requieren una capacidad de transmisión continua. Esto resulta útil para transmitir información de voz y vídeo. El ancho de banda restante se utiliza para las transmisiones asíncronas.
- La especificación SMT de FDDI define un esquema de subasta distribuida para asignar el ancho de banda de FDDI.

Asíncrono

- El ancho de banda asíncrono se asigna utilizando un esquema de prioridad de ocho niveles. A cada estación se asigna un nivel de prioridad asíncrono.
- FDDI también permite diálogos extendidos, en los cuales las estaciones pueden usar temporalmente todo el ancho de banda asíncrono.
- El mecanismo de prioridad de la FDDI puede bloquear las estaciones que no pueden usar el ancho de banda síncrono y que tienen una prioridad asíncrona demasiado baja.

Nodos de FDDI

FDDI especifica el uso de anillos dobles para las conexiones físicas como se muestra en la figura 2.17. El tráfico de cada anillo viaja en direcciones opuestas. Físicamente, los anillos están compuestos por dos o más conexiones punto a punto entre estaciones adyacentes. Los dos anillos de la FDDI se conocen uno como primario y el segundo como secundario. El anillo primario se usa para la transmisión de datos, mientras que el anillo secundario se usa generalmente como respaldo.

Las estaciones Clase B, o *estaciones de una conexión (SAS),* se conectan a un anillo, mientras que las de Clase A, o *estaciones de doble conexión (DAS),* se conectan a ambos anillos. Las SAS se conectan al anillo primario a través de un concentrador que suministra conexiones para varias SAS. El concentrador garantiza que si se produce una falla o interrupción en el suministro de alimentación en algún SAS determinado, el anillo no se interrumpa. Esto es particularmente útil cuando se conectan al anillo PC o dispositivos similares que se encienden y se apagan con frecuencia. Una configuración FDDI típica que cuenta tanto con DAS como con SAS aparece ilustrada en la figura. Cada DAS de la FDDI está provista de dos puertos, designados A y B.

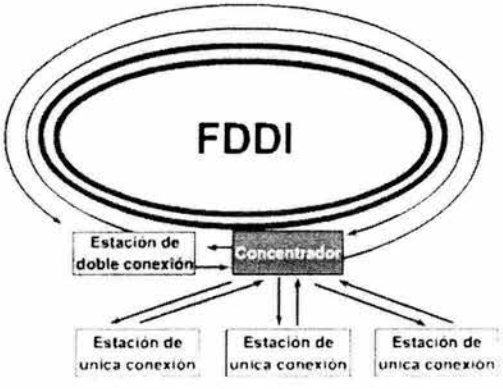


Figura 2.17 Nodos de FDDI: DAS, SAS y concentrador

Estos puertos conectan la estación con el anillo doble FDDI y, por lo tanto, cada puerto suministra una conexión tanto para el anillo primario como con para el secundario.

FDDI usa un esquema de codificación denominado ***4B/5B***. Cada grupo de 4 bits de datos se envía como un código de 5 bits. Las fuentes de señales de los transmisores de la FDDI son LEDs (diodos emisores de luz) o láseres.

2.4.5.3 Ethernet y 802.3

Descripción general y sus variantes

Ethernet es la tecnología de red de área local (LAN) de uso más generalizado. El diseño original de Ethernet representaba un punto medio entre las redes de larga distancia y baja velocidad y las redes especializadas de las aulas de cómputo, que transportaban datos a altas velocidades y a distancias muy limitadas. Ethernet se adecua bien a las aplicaciones en las que un medio de comunicación local debe transportar tráfico esporádico y ocasionalmente pesado, a velocidades muy elevadas.

La arquitectura de red Ethernet tiene su origen en la década de los 60's en la Universidad de Hawai, donde se desarrolló el método de acceso utilizado por Ethernet, o sea, el CSMA/CD (Acceso Múltiple con Detección de Portadora y Detección de Colisiones). El centro de investigaciones PARC (Palo Alto Research Center) de Xerox Corporation desarrolló el primer sistema Ethernet experimental a principios de la década de los 70's. Este sistema sirvió como base de la especificación 802.3 publicada en 1980 por el Instituto de Ingeniería Eléctrica y Electrónica (IEEE).

Poco después de la publicación de la especificación IEEE 802.3 en 1980, Digital Equipment Corporation, Intel Corporation y Xerox Corporation desarrollaron y publicaron conjuntamente una especificación Ethernet denominada "Versión 2.0" que era sustancialmente compatible con la IEEE 802.3. En la actualidad, Ethernet e IEEE 802.3 retienen en conjunto la mayor parte del mercado de protocolos de LAN. Hoy en día, el término Ethernet a menudo se usa para referirse a todas las LAN de acceso múltiple con detección de portadora y detección de colisiones (CSMA/CD), que generalmente cumplen con las especificaciones Ethernet, incluyendo IEEE 802.3.

Ethernet e IEEE 802.3 especifican tecnologías similares; ambas son LAN de tipo CSMA/CD. Las estaciones de una LAN de tipo CSMA/CD pueden acceder a la red en cualquier momento. Antes de enviar datos, las estaciones CSMA/CD escuchan a la red para determinar si se encuentra en uso. Si lo está, entonces esperan. Si la red no se encuentra en uso, las estaciones comienzan a transmitir. Una colisión se produce cuando dos estaciones escuchan para saber si hay tráfico de red, no lo detectan y, acto seguido transmiten de forma simultánea. En este caso, ambas transmisiones se dañan y las estaciones deben volver a transmitir más tarde. Los algoritmos de postergación determinan el momento en que las estaciones que han tenido una colisión pueden volver a transmitir. Las estaciones CSMA/CD pueden detectar colisiones, de modo que saben que su información ha sido dañada y que requerirán volverla a transmitir.

Tanto las LAN Ethernet como las LAN IEEE 802.3 son redes de broadcast. Esto significa que cada estación puede ver todas las tramas, aunque una estación determinada no sea el destino propuesto para esos datos. Cada estación debe examinar las tramas que recibe para determinar si corresponden al destino. De ser así, la trama pasa a una capa de protocolo superior dentro de la estación para su adecuado procesamiento.

Existen diferencias sutiles entre las LAN Ethernet e IEEE 802.3, figura 2.18. Ethernet proporciona servicios que corresponden a las Capas 1 y 2 del modelo de referencia OSI. IEEE 802.3 especifica la capa física, la Capa 1 y la porción de acceso al canal de la capa de enlace de datos, la Capa 2, pero no define un protocolo de Control de Enlace Lógico. Tanto Ethernet como IEEE 802.3 se implementan a través del hardware. Normalmente, el componente físico de estos protocolos es una tarjeta de interfaz en una maquina *host* o son circuitos de una placa de circuito impreso dentro de una maquina *host*.

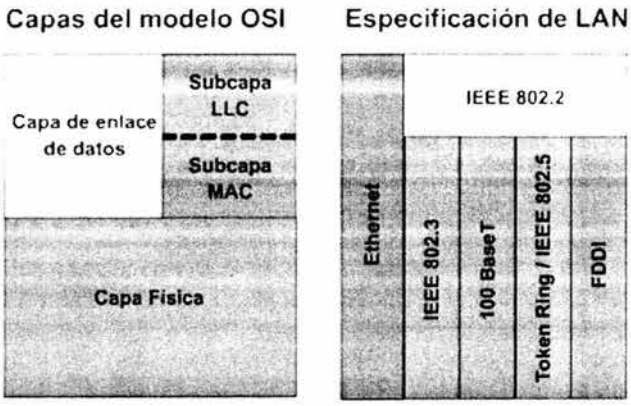


Figura 2.18 Similitudes y diferencias entre las capas 1y 2 del modelo OSI

El formato de la trama de Ethernet y 802.3 consta de diversos campos los cuales se detallan a continuación.

Formato de trama de Ethernet e IEEE 802.3

Los campos de trama Ethernet e IEEE 802.3 se muestran en la figura 2.19 y se describen en los siguientes resúmenes:

Ethernet						
7	1	6	6	2	46-1500	4
Preámbulo	Inicio de delimitador de trama	Dirección destino	Dirección origen	Tipo	Datos	Secuencia de verificación de trama

IEEE 802.3						
7	1	6	6	2	64-1500	4
Preámbulo	Inicio de delimitador de trama	Dirección destino	Dirección origen	Longitud	Encabezado y datos 802.2	Secuencia de verificación de trama

Figura 2.19 Formatos de trama Ethernet e IEEE 802.3

- *Preámbulo*: El patrón de unos y ceros alternados les indica a las estaciones receptoras que una trama Ethernet o IEEE 802.3 esta por iniciar. La trama Ethernet incluye un byte adicional que es el equivalente al campo Inicio de trama (SOF) de la trama IEEE 802.3.

- *Inicio de trama (SOF)*: El byte delimitador de IEEE 802.3 finaliza con dos bits 1 consecutivos, que sirven para sincronizar las porciones de recepción de trama de todas las estaciones de la LAN. SOF se especifica explícitamente en Ethernet.

- *Direcciones destino y origen*: Los primeros 3 bytes de las direcciones son especificados por IEEE según el proveedor o fabricante. El proveedor de Ethernet o IEEE 802.3 especifica los últimos 3 bytes. La dirección origen siempre es una dirección unicast (de nodo único). La dirección destino puede ser unicast, multicast (grupo de nodos) o de broadcast (todos los nodos).

- *Tipo (Ethernet)*: El tipo especifica el protocolo de capa superior que recibe los datos una vez que se ha completado el procesamiento Ethernet.

- *Longitud (IEEE 802.3)*: La longitud indica la cantidad de bytes de datos que sigue este campo.

- *Datos (Ethernet)*: Una vez que se ha completado el procesamiento de la capa física y de la capa de enlace, los datos contenidos en la trama se envían a un protocolo de capa superior, que se identifica en el campo tipo. Aunque la versión 2 de Ethernet no especifica ningún relleno, al contrario de lo que sucede con IEEE 802.3, Ethernet espera por lo menos 46 bytes de datos.

- *Datos (IEEE 802.3)*: Una vez que se ha completado el procesamiento de la capa física y de la capa de enlace, los datos se envían a un protocolo de capa superior, que debe estar definido dentro de la porción de datos de la trama. Si los datos de la trama no son suficientes para llenar la trama hasta una cantidad mínima de 64

bytes, se insertan bytes de relleno para asegurar que por lo menos haya una trama de 64 bytes.

- *Secuencia de Verificación de Trama (FCS)*: Esta secuencia contiene un valor de verificación CRC de 4 bytes, creado por el dispositivo emisor y recalculado por el dispositivo receptor para verificar la existencia de tramas dañadas.

Operación de Ethernet

Ethernet es una tecnología de broadcast de medios compartidos que se resume en la figura 2.20.

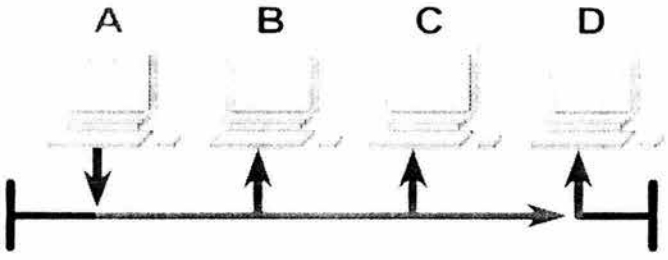


Figura 2.20 Operación de Ethernet

El método de acceso CSMA/CD que se usa en Ethernet ejecuta tres funciones:

1. Transmitir y recibir paquetes de datos
2. Decodificar paquetes de datos y verificar que las direcciones sean válidas antes de transferirlos a las capas superiores del modelo OSI
3. Detectar errores dentro de los paquetes de datos o en la red

En el método de acceso CSMA/CD, los dispositivos de red que tienen datos para transmitir a través de los medios de la red funcionan según el modo "escuchar antes de transmitir". Esto significa que cuando un dispositivo desea enviar datos, primero debe verificar si los medios de la red están ocupados. El dispositivo debe verificar si existen señales en los medios de red. Una vez que el dispositivo determina que los medios de red no están ocupados, el dispositivo comienza a transmitir los datos. Mientras transmite los datos en forma de señales, el dispositivo también escucha. Esto lo hace para comprobar que no haya ninguna otra estación que esté transmitiendo datos a los medios de red al mismo tiempo. Una vez que ha terminado de transmitir los datos, el dispositivo vuelve al modo de escucha.

Los dispositivos de red pueden detectar cuando se ha producido una colisión porque aumenta la amplitud de la señal en el medio de red. Cuando se produce una colisión, cada dispositivo que está realizando una transmisión continúa transmitiendo datos durante un período breve.

Esto se hace para garantizar que todos los dispositivos puedan detectar la colisión. Una vez que todos los dispositivos de una red detectan que se ha producido una colisión, cada dispositivo invoca a un algoritmo de espera para transmisión. Después de que todos los dispositivos de una red han sufrido una postergación durante un período determinado de tiempo (que es distinto para cada dispositivo), cualquier dispositivo puede intentar obtener acceso al medio de transmisión nuevamente. Cuando se reanuda la transmisión de datos en la red, los dispositivos involucrados en la colisión no tienen prioridad para transmitir datos. En la figura 2.21 se presenta un resumen del proceso CSMA/CD.

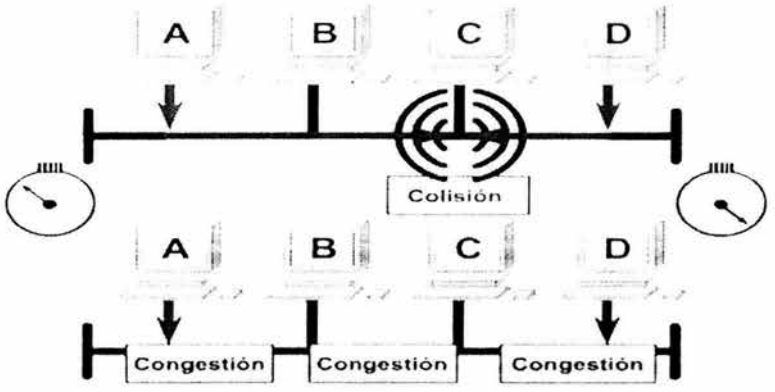


Figura 2.21 Confiabilidad de Ethernet

Ethernet trasmite información bajo un esquema tipo *broadcast*. Esto significa que todos los dispositivos de una red pueden ver todos los datos que pasan a través del medio de transmisión. Sin embargo, no todos los dispositivos de la red procesan los datos. Solamente el dispositivo cuya dirección MAC concuerda con la dirección MAC que transporta los datos copiará la información.

Una vez que el dispositivo ha verificado que la dirección MAC que trasporta la trama es la suya, entonces la verifica para ver si hay errores. Si el dispositivo detecta que hay errores, se descarta la trama. El dispositivo destino no enviará ninguna notificación al dispositivo origen, sin tener en cuenta si la trama ha llegado a su destino con éxito o no. Ethernet es una arquitectura de red no orientada a conexión considerada como un sistema de entrega de "máximo esfuerzo".

2.5 Redes WAN

2.5.1 Descripción general

Una WAN (red de área amplia) opera en la capa física y la capa de enlace de datos del modelo de referencia OSI. Interconecta las LAN (redes de área local) que normalmente se encuentran separadas por grandes distancias físicas.

Las características principales de las WAN son las siguientes:

- Operan dentro de un área geográfica mayor que la de las LAN locales. Utilizan los servicios de proveedores de servicios de telecomunicaciones tales como los operadores Regional Bell (RBOC), Sprint y MCI (en los EE.UU.) y TELMEX en el caso de México.
- Usan conexiones seriales de diversos tipos para acceder al ancho de banda dentro de áreas geográficas extensas.
- Por definición, las WAN conectan dispositivos separados por áreas geográficas extensas, figura 2.22. Entre estos dispositivos se incluyen:
  - *Enrutadores* : ofrecen varios servicios, entre ellos funciones de redes internas y puertos de interfaz de WAN
  - *Switches WAN*: utilizan al ancho de banda de las WAN para la comunicación de voz, datos y video
  - *Módems:* Es un dispositivo que permiten a las computadoras comunicarse entre sí a través de líneas telefónicas, esta comunicación se realiza a través de la modulación y demodulación de señales electrónicas que pueden ser procesadas por computadoras, las señales analógicas se convierten en digitales y viceversa. Los modems pueden ser externos o internos dependiendo de su ubicación física en la red.
  - *Servidores de comunicaciones*: concentran la comunicación de usuarios de servicios de acceso con marcación (dial-up).

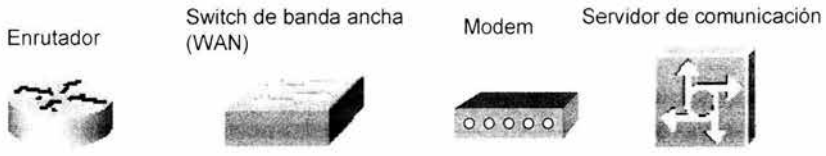


Figura 2.22 Redes y dispositivos de área amplia

2.5.2 Protocolos y estándares para WAN

Los protocolos de la capa física de las WAN describen cómo suministrar conexiones eléctricas, mecánicas, operacionales y funcionales para los servicios de WAN. Estos servicios a menudo se obtienen de proveedores de servicios de WAN, proveedores de servicio alternos y las empresas de servicios postales, telefónicos y telegráficos (PTT).

Los protocolos de enlace de datos WAN describen cómo se transportan las tramas entre sistemas a través de un solo enlace de datos. Incluyen protocolos diseñados para operar a través de servicios conmutados, dedicados punto a punto, multipunto y múltiple acceso, como Frame Relay. Los estándares de las WAN son definidos y administrados por una serie de autoridades reconocidas, tales como las siguientes:

- Sector de Normalización de las Telecomunicaciones de la Unión Internacional de Telecomunicaciones (UIT-T), antiguamente denominado Comité Consultivo Internacional Telegráfico y Telefónico (CCITT)
- Organización Internacional de Normalización (ISO)
- Fuerza de Tareas de Ingeniería de Internet (IETF)
- Asociación de Industrias Electrónicas (EIA)

Normalmente los estándares de WAN describen los requisitos de la capa física y de la capa de enlace de datos. La capa física de las WAN describe la interfaz entre el equipo terminal de datos (DTE) y el equipo de terminación de circuito de datos (DCE). Normalmente el DCE es el proveedor del servicio, mientras que el DTE es el dispositivo conectado.

Varios estándares de capa física se especifican en la siguiente lista.

- EIA/TIA-232
- EIA/TIA-449
- V.24
- V.35
- X.21
- G.703
- EIA-530

En tanto que algunos protocolos de capa 2 son los siguientes:

- Control de Enlace de Datos de Alto Nivel (HDLC) [17]: un estándar IEEE que probablemente no sea compatible con los distintos proveedores, ya que cada proveedor puede haberlo implementado de diferentes maneras. HDLC soporta configuraciones punto a punto y multipunto con un gasto mínimo

- *Frame Relay* [27]: Usa instalaciones digitales de alta calidad y entramado simplificado sin mecanismos de corrección de errores, lo que significa que puede enviar información de Capa 2 mucho más rápidamente que otros protocolos de WAN.
- *Protocolo Punto a Punto (PPP)* [11]: Contiene un campo de protocolo para identificar el protocolo de capa de red que transporta.
- *Control de Enlace de Datos Síncrono (SDLC)* [51]: Protocolo de enlace de datos de WAN diseñado por IBM para los entornos SNA. Ha sido reemplazado en gran parte por el más versátil: HDLC.
- *Protocolo Internet de Enlace Serial (SLIP)* [54]: Protocolo de enlace de datos de WAN sumamente popular para transportar paquetes IP. Ha sido reemplazado en varias aplicaciones por el más versátil PPP.
- *Procedimiento de Acceso al Enlace Balanceado (LAPB)* [18]: Protocolo de enlace de datos utilizado por X.25. Posee amplias capacidades de verificación de errores.
- *Procedimiento de Acceso al Enlace en el Canal D (LAPD)* [25]: Protocolo de enlace de datos de WAN utilizado para señalización y para la configuración de llamada de Canal D de RDSI. Las transmisiones de datos tienen lugar en los canales B de RDSI
- *Trama de Procedimiento de Acceso a Enlaces (LAPF)* [26]: Para Servicios de Portadora en Modo de Trama, un protocolo de enlace de datos de WAN, similar a LAPD, utilizado con tecnologías Frame Relay

2.5.3 Tecnologías de WAN

A continuación ofrecemos una breve descripción de las tecnologías de WAN más comunes. Estas tecnologías se dividen en servicios conmutados por circuito, conmutados por celdas, digitales dedicadas y analógicas.

Servicios de conmutación de circuitos

- *POTS (Servicio telefónico analógico):* No es un servicio informático de datos, pero se incluye por dos motivos: (1) muchas de sus tecnologías forman parte de la creciente infraestructura de datos, (2) es un modelo sumamente confiable, de fácil uso para una red de comunicaciones de área amplia; los medios típicos son el cable de cobre de par trenzado y el cable coaxial.
- *RDSI (Red Digital de Servicios Integrados) de banda angosta* [20]: Una tecnología versátil, de amplio uso e históricamente importante. Fue el primer servicio con marcación totalmente digital. Es de uso bastante generalizado, aunque varía considerablemente de un país a otro. El ancho de banda máximo es de 128 kbps para la BRI (Interfaz de Acceso Básico) de menor costo y de aproximadamente 2Mbps para la PRI (Interfaz de Acceso Principal). El medio típico es el cable de cobre de par trenzado

Servicios de conmutación por paquetes

- *X.25:* Tecnología más antigua pero todavía ampliamente utilizada, que posee amplias capacidades de verificación de errores desde la época en que los enlaces de las WAN eran más susceptibles a los errores, lo que hace que su confiabilidad sea muy grande, pero al mismo tiempo limita su ancho de banda. El ancho de banda puede ser de 2 Mbps como máximo. Es ampliamente utilizada, y su costo es moderado. El medio típico es el cable de cobre de par trenzado.
- *Frame Relay*: Versión conmutada por paquetes de la RDSI (Red Digital de Servicios Integrados). Se ha transformado en una tecnología de WAN sumamente popular por derecho propio. Es más eficiente que X.25, con servicios similares.

  El ancho de banda máximo es de 44,736 Mbps. En los EE.UU. son muy populares los anchos de banda de 56kbps y 384kbps. Es de uso generalizado, el costo es de moderado a bajo. Entre los medios típicos se incluyen el cable de cobre de par trenzado y el cable de fibra óptica.

Servicios de conmutación por celdas

- *ATM (Modo de Transferencia Asíncrona)* [21]*:* Tiene una cercana relación con la RDSI. Es una tecnología de WAN (e inclusive de LAN) cuya importancia va en aumento. Utiliza tramas pequeñas, de longitud fija (53 bytes) para transportar los datos. El ancho de banda máximo es actualmente de 622 Mbps, aunque se están desarrollando velocidades mayores. Los medios típicos son el cable de cobre de par trenzado y el cable de fibra óptica. Su uso es generalizado y está en aumento; el costo es elevado.

Servicios digitales dedicados

- *T1, T3* [2]*, E1, E3* [29]: La serie T de servicios en los EE.UU. y la serie E de servicios en Latinoamérica y Europa son tecnologías de WAN sumamente importantes. Usan la multiplexación por división de tiempo para "dividir" y asignar ranuras de tiempo para la transmisión de datos; el ancho de banda es:
  - T1: 1,544 Mbps
  - T3: 44,736 Mbps
  - E1: 2,048 Mbps
  - E3: 34,368 Mbps
  - Además de otros anchos de banda disponibles

  Los medios utilizados son normalmente el cable de cobre de par trenzado y el cable de fibra óptica. Su uso es muy generalizado; el costo es moderado.

- *xDSL (DSL por Digital Subscriber Line (Línea Digital del Suscriptor) y x por una familia de tecnologías)* [5]: Tecnología WAN nueva y en desarrollo para uso doméstico. Su ancho de banda disminuye a medida que aumenta la distancia desde el equipo de las compañías telefónicas. Las velocidades máximas de 51,84

Mbps son posibles en las cercanías de una central telefónica; son más comunes los anchos de banda mucho menores (desde 100 kbps hasta varios Mbps). Su uso es limitado pero en rápido aumento; el costo es moderado y se reduce cada vez más. x indica toda la familia de tecnologías DSL, entre ellas:

- *HDSL*: DSL de alta velocidad de bits (↑2.048Mbps, ↓2.048Mbps)
- *SDSL*: DSL de línea única (↑768kbps, ↓768kbps)
- *ADSL*: DSL asimétrica (↑640kbps, ↓8Mbps)
- *VDSL*: DSL de muy alta velocidad de bits (↑2.3Mbps, ↓52Mbps)
- *RADSL*: DSL adaptable a la velocidad

- *SONET (Red Óptica Síncrona)* [4] / *SDH (Jerarquía Digital Síncrona)* [28]: Conjunto de tecnologías de capa física de muy alta velocidad, diseñadas para cables de fibra óptica, pero que también pueden funcionar con cables de cobre.

  SONET fue definido por la Institución Americana de Estándares Nacionales (ANSI), soporta varias velocidades implementadas a diferentes niveles de OC (portadora óptica) desde los 51,84 Mbps (OC-1) hasta los 9,952 Mbps (OC-192). Es usada principalmente en Estados Unidos.

  SDH fue definido por el Instituto de Estándares Europeos de Telecomunicaciones (ETSI) y es el estándar más utilizado en el mundo. Soporta diferentes velocidades implementadas en módulos de transferencia síncrona (STM) desde 155 Mbps (STM-1) hasta 9920 Mbps (STM-64).

  Pueden alcanzar estas impresionantes velocidades de datos mediante el uso de multiplexación por división de longitud de onda (WDM), en la que láseres configurados para colores ligeramente diferentes (longitudes de onda) envían enormes cantidades de datos ópticamente; su uso es generalizado en las redes dorsales de Internet. Su costo es elevado, no es una tecnología que se pueda usar a nivel doméstico.

Otros servicios de WAN

- *Módems de marcación analógica (conmutación analógica)*: Su velocidad es limitada, pero son muy versátiles. Funcionan con la red telefónica existente. El ancho de banda máximo aproximado es de 56 kbps. El costo es bajo. Su uso es muy generalizado. El medio típico es la línea telefónica de par trenzado.

- *Módems por cable (analógico compartido)*: Colocan señales de datos en el mismo cable que las señales de televisión. Es cada vez más popular en regiones donde hay gran cantidad de cable coaxial de TV instalado. El ancho de banda máximo disponible puede ser de 30 Mbps [7], aunque esto se degrada a medida que más usuarios se conectan a un segmento determinado de la red (comportándose como LAN no conmutadas). El costo es relativamente bajo. Su uso es limitado pero está en aumento. El medio es cable coaxial.

- *Inalámbrico*: No se necesita un medio porque las señales son ondas electromagnéticas. Existen varios enlaces de WAN inalámbricos, dos de los cuales son:
  - Terrestre: Anchos de banda normalmente dentro del intervalo de Mbps (Ejemplo: microondas). El costo es relativamente bajo. Normalmente se requiere línea de vista. El uso es moderado.
  - *Satélite*: Puede servir a los usuarios móviles (Ej., red telefónica celular) y usuarios remotos.
  - *Broadband Wireless Access* [15]: es una tecnología de red área metropolitana inalámbrica (WMAN). Soporta una arquitectura punto-multipunto para proveer accesos inalámbricos fijos a alta velocidad para hogares, pequeños negocios, edificios comerciales, etc.

2.5.4 Formatos de encapsulamiento de WAN

Los dos encapsulamientos WAN punto a punto más comunes son HDLC y PPP. Todos los encapsulamientos de línea serial comparten un formato de trama común, con los siguientes campos, tal como se indica en la figura 2.23.

PPP

Preámbulo	Dirección	Control	Protocolo	Datos	FCS	Señalador

HDCL

Preámbulo	Dirección	Control	Propietario	Datos	FCS	Señalador

Figura 2.23 Formato de encapsulamiento de trama WAN

Descripción de los campos

- Preámbulo: Indica el comienzo y fin de la trama y usa el modelo hexadecimal 7E.
- Dirección: Campo de 1 ó 2 bytes para dirigir la estación final en entornos multipunto.
- Control: Indica si la trama es de información, supervisión o sin numerar. También contiene códigos de función específicos.
- Datos: Datos encapsulados.
- FCS: Secuencia de verificación de trama (FCS).

Cada tipo de conexión WAN utiliza un protocolo de Capa 2 para encapsular el tráfico mientras atraviesa el enlace WAN. Para asegurarse de que se utiliza el protocolo de encapsulamiento adecuado, es necesario configurar el tipo de encapsulamiento de Capa 2 que se debe utilizar para cada interfaz serial en el enrutador. La elección del protocolo de encapsulamiento depende de la tecnología WAN y del equipo de comunicación.

2.5.4.1 Encapsulamiento PPP

PPP es un método de encapsulamiento de línea serial estándar (que se describe en RFC 1332 y RFC 1661). Este protocolo puede, entre otras cosas, verificar la calidad del enlace durante el establecimiento de la conexión. Además, tiene soporte para autenticación a través del protocolo de autenticación de contraseña (PAP) y el protocolo de autenticación de saludo (CHAP), el encapsulamiento se muestra en la figura 2.24.

PPP

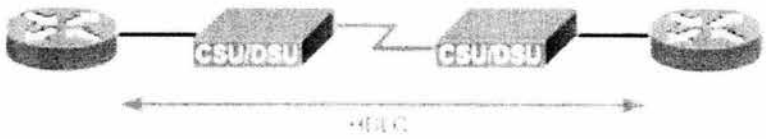
Señalador	Dirección	Control	Protocolo	Datos	FCS	Señalador

Figura 2.24 Encapsulamiento PPP

2.5.4.2 Encapsulamiento HDLC

HDLC es un protocolo de la capa de enlace de datos que se deriva del protocolo de encapsulamiento de control de enlace de datos síncrono (SDLC). HDLC es el encapsulamiento por defecto de Cisco para las líneas seriales. Esta implementación es muy simplificada; no usa ventanas ni control de flujo y sólo se permiten las conexiones punto a punto. El campo de dirección siempre se establece en todos unos. En el caso de equipos Cisco, se inserta además un código propietario de 2 bytes después del campo de control, lo que significa que el entramado HDLC no puede operar con equipos de otros proveedores.

Si ambos extremos de una conexión de línea dedicada son enrutadores con sistema operativo de red (IOS) de Cisco, normalmente se utiliza el encapsulamiento HDLC, el encapsulamineto se muestra en la figura 2.25.



HDLC

Señalador	Dirección	Control	Propietario	Datos	FCS	Señalador

Figura 2.25 Encapsulamiento HDLC de Cisco

CAPÍTULO 3. Fundamentos de IP multicast

En este capítulo se determinarán de manera inicial la importancia y ventajas de multicast, así como los conceptos e ideas que serán de gran utilidad para la comprensión de lo que es la tecnología multicast y las diferencias que existen con respecto a otras.

3.1 Primeros desarrollos

En los años 80´s, en la Universidad de Stanford, el doctor Deering y Cheriton [46] desarrollaron un Sistema Operativo Distribuido, este sistema fue llamado *Vsystem*, el cual consistía de algunas computadoras conectadas vía un segmento Ethernet simple creando un sistema multiproceso no muy eficaz. Estas computadoras se comunicaban a través de mensajes especiales enviados sobre la red Ethernet. El sistema operativo permitia a una computadora enviar mensajes a otro grupo de computadoras en el mismo segmento Ethernet usando multicast en la capa MAC.

El proyecto creció y necesitó más computadoras, pero las únicas disponibles se localizaban en otro sitio del campus, interponiéndose algunos enrutadores entre ambas redes. Para que las computadoras del otro sitio pudieran funcionar como parte del sistema multiproceso, los mensajes multicast a nivel de capa MAC debían ahora extenderse y funcionar también en la capa de Red (capa 3 del modelo OSI). Esta tarea fue responsabilidad en primera instancia del Dr Deering [46].

Después de estudiar los protocolos de enrutamiento IP: OSPF (*Open Shortest Path First*) [32] y RIP (*Routing Information Protocol*) [6], Dr Deering concluyó que el mecanismo de estado-enlace de OSPF podría ser extendido para soportar multicast, así como también los mecanismos básicos de RIP podrían ser la base de un nuevo protocolo de enrutamiento multicast basado en el algoritmo vector-distancia.

Estas ideas dieron la base para más investigaciones en el campo del IP multicast y conformaron la tesis doctoral del Dr deering "Multicast Routing in a Datagram Network" publicada en Diciembre de 1991.

Esta tesis también describía el protocolo "*Host Membership Protocol*", el cual es la base para IGMP (*Internet Group Membership Protocol*) de hoy en día. También sus ideas basadas en el estudio de RIP se convirtieron en la base de DVMRP (Distance Vector Multicast Routing Protocol) desarrollado más tarde por él mismo autor. Estos dos protocolos conformaron las primeras bases para que una red de paquetes IP soportara multicast en la capa 3. Desde entonces, los avances en la tecnología IP multicast han continuado y otros protocolos como PIM (*Protocol Independent Multicasting*) y extensiones multiprotocolo para BGP (*Border Gateway Protocol*) [39] se han desarrollado. Estos protocolos permiten que el IP multicast escale mas allá de los límites iniciales, soportándose en amplias redes empresariales y eventualmente también será soportado completamente en Internet.

3.1.1 Multicast Backbone

A principios de los 90´s varios miembros de la comunidad de investigadores se quejaron con la agencia DARPA (Defense Advanced Research Projects Agency), cuerpo perteneciente al gobierno para asuntos de Internet en aquellos tiempos, porque el Internet se había convertido en una red pública y por lo tanto ya no estaba disponible para la experimentación e investigación de nuevas tecnologías. Como resultado el gobierno de Estados Unidos creó la DARPA Tested Network (DARTNet) para dar a los investigadores una red experimental en la cual ellos pudieran realizar pruebas y evaluaciones de nuevas herramientas y tecnologías sin afectar el funcionamiento de la Internet.

DARTNet estaba compuesto inicialmente por enlaces T1 interconectando varios sitios como Xerox PARC, Lawrence Berkley Labs, SRI, ISI, BBN, MIT y la Universidad de Delaware. Estos sitios usaban Sun SPARCstations corriendo "*routed*" como demonio de enrutamiento unicast y "*mrouted*" como demonio de enrutamiento multicast DVRMP. Por lo tanto, DARTNet soportaba IP multicast nativo en todos los sitios. Las audio conferencias semanales entre investigadores de varios sitios DARTNet alrededor de los Estados Unidos fueron muy pronto, prácticas normales.

En 1992, la IETF (Internet Engineering Task Force) realizó planes para realizar un siguiente encuentro en San Diego, California. Desafortunadamente, uno de los investigadores de DARTNet no podía asistir al evento, de esta situación nació la idea de crear un túnel DVMRP que se configurara entre una SPARCstation de la IETF y el backbone DARTNet. También se enviaron invitaciones para participar, en esta convención de la IETF, a otros investigadores de los Estados Unidos, Australia, Suecia e Inglaterra. Junto con las invitaciones se envió también la información de cómo configurar un túnel DVRMP entre una SPARCstation y el backbone DARTNet a través de Internet. Varios sitios respondieron a la invitación y el resultado fue la primer audio conferencia multicast de la IETF a través de Internet alrededor del mundo.

Después de esta convención, que resultó todo un éxito, se desconfiguraron los túneles por lo que todo volvió a la normalidad en la DARTNet. Para la siguiente convención de la IETF, que tuvo lugar en Washington el verano del mismo año, se planeó no sólo enviar audio sino video hacia todos los participantes alrededor del mundo, por lo que se tuvieron que construir nuevamente los túneles hacia la DARTNet. Después de esta convención los administradores de la DARTNet y los participantes en los otros sitios decidieron dejar los túneles de manera permanente para las futuras conferencias a través de Internet. Así nació el corazón de la red multicast que pronto fue llamada "**Mbone**" (Multicast Backbone). A partir de la fecha de su creación el Mbone no ha dejado de crecer.

En resumen podemos decir que aunque IP multicast empezó a tener presencia a principios de los 90´s, su verdadero potencial apenas está siendo observado. Las corporaciones están empezando a ver las ventajas en el la utilización del ancho de banda y las capacidades para entregar contenido a varios receptores a la vez. Los proveedores de servicio de Internet también están viendo ventajas en el ofrecimiento de IP multicast como servicio a sus clientes, muchos de los cuales están dispuestos a pagar

este servicio. El MBone por sí mismo ha gozado de crecimiento rápido en los últimos años, y todo indica que esta tendencia continuará.

3.2 Multicast en el modelo OSI

Multicast se ubica en la segunda y tercera capa del modelo OSI, figura 3.4.

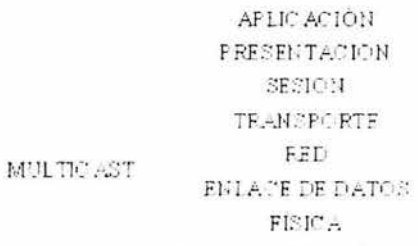


Figura 3.1 Multicast en el modelo OSI

Ethernet y FDDI, por ejemplo, soportan direcciones unicast, multicast y broadcast. Una computadora individual puede recibir direcciones unicast, algunas direcciones multicast y direcciones broadcast. Token Ring también soporta el concepto de direccionamiento multicast pero utiliza diferentes técnicas. Token Ring tiene direcciones funcionales que se puedan utilizar para tratar grupos de receptores.

Si las aplicaciones están limitadas a una simple LAN, es suficiente el uso de una técnica multicast a nivel de capa 2 "Enlace de Datos". Sin embargo, muchas aplicaciones multipunto tienen un valor preciso porque no están limitadas a una simple LAN. Cuando una aplicación se extiende a la Internet lo cual consiste en diferentes tipos de medios tales como Ethernet, Token ring, FDDI, ATM, Frame Relay, SMDS y otras tecnologías, es necesario implementar multicast en la capa 3.

Hay varios parámetros que la capa de red debe definir en orden para soportar las comunicaciones multicast:

- *Addressing*. Debe haber una dirección de la capa de red que se utilicé para comunicarse con un grupo de receptores
- *Dynamic registration*. Debe haber un mecanismo para que la computadora se comunique con la red, indicando que es un miembro de un grupo multicast en

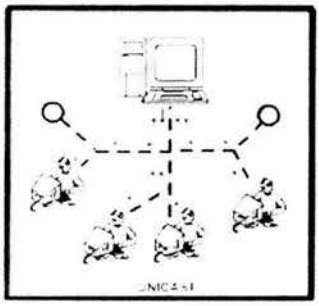
particular. Sin esta capacidad, la red no puede saber qué redes necesitan recibir el tráfico para cada grupo multicast.

- *Multicast routing*. La red debe poder construir los árboles de distribución de paquetes que permiten que las fuentes envíen los paquetes a todos los receptores. Una meta fundamental de estos árboles de distribución es asegurarse de que existe una sola copia de cada paquete a la vez en cualquier red dada (es decir, si hay múltiples receptores en un rama dada, en esa rama debe haber solamente una copia de los paquetes).

La IETF ha estado desarrollando los estándares que tratan cada uno de los parámetros descritos arriba.

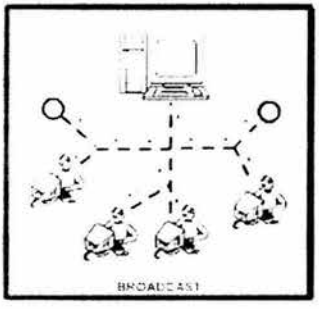
- *Addressing*. El espacio de direcciones IP está dividido en cuatro piezas: Clase A, Clase B, Clase C, y Clase D. Las clases A, B y C son utilizadas para el tráfico Unicast. La clase D esta reservada para el tráfico multicast.
- *Dynamic registration*. RFC 1112 define a Internet Group Membership Protocol (IGMP). IGMP especifica cómo el *host* debe informar a la red que es un miembro de un grupo particular de multicast.
- *Multicast routing*. Hay varios estándares disponibles para el enrutamiento del tráfico IP multicast:
  - DVMRP (Distance Vector Multicast Routing Protocol): RFC 1075. [12]
  - MOSPF (Multicast Open Shortest Path First) [31], extensión de OSPF para permitir el soporte de IP multicast: RFC 1584.
  - PIM (Protol-Independent Multicast) es un protocolo multicast que puede ser usado en conjunto con protocolos de ruteo IP unicast. Definido por dos documentos titulados PIM: Motivation and Architecture y PIM: Protocol Especification.

3.3 Multicast vs. Unicast



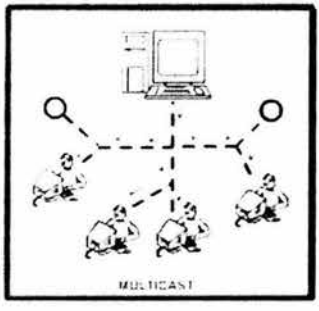
Unicast: Con un diseño unicast, el tráfico de las aplicaciones es enviado a cada uno de los miembros del grupo como se muestra en la figura 3.2. Esta técnica es muy simple de poner en ejecución, pero tiene restricciones significativas de escalamiento, si el grupo es grande. Además, requiere de un ancho de banda adicional, porque la misma información tiene que ser llevada a múltiples tiempos -- incluso en acoplamientos compartidos.

Figura 3.2 Unicast



Broadcast: Esta técnica es mucho más simple de poner en ejecución que unicast, se envía una copia de la aplicación a una sola dirección broadcast. Sin embargo, si esta técnica es usada, se debe tener cuidado en tener un límite broadcast en la frontera de la LAN para evitar tráfico innecesario enviando el broadcast a todos lados, como se observa en la figura 3.3. Esto ocasiona que los recursos de la red se utilicen innecesariamente, si sólo se necesita enviar dicho tráfico a un grupo pequeño.

Figura 3.3 Broadcast



Multicast. Con el diseño multicast, se envía una copia de la aplicación al grupo de computadoras que deseen recibirla. Esta técnica direcciona los paquetes a un grupo de receptores, y depende de la red que se remitan los paquetes solamente a las redes que necesitan recibirlas, como se muestra en la figura 3.4

Figura 3.4 Multicast

3.3.1 Ventajas de IP multicast

IP multicast entrega información a múltiples receptores sin añadir carga adicional a la fuente o a los receptores, utilizando el menor ancho de banda posible. Los paquetes multicast son reproducidos por los enrutadores en los que se encuentra habilitada esta

funcionalidad. Otras tecnologías requieren que la fuente envíe más de una copia de la información (algunas requieren que la fuente envíe una copia individual a cada receptor), por lo que en presencia de miles de receptores también se ven beneficiadas las aplicaciones que utilizan poco ancho de banda. Con aplicaciones con alto consumo de ancho de banda (como en el caso de transmisión de video MPEG), el uso de multicast se hace casi indispensable cuando se tiene más de un receptor.

3.4 Multicast básico

3.4.1 Concepto de grupo multicast

Multicast se basa en el concepto de grupo. Un grupo arbitrario de receptores expresa el interés común de recibir un flujo de datos en particular. Este grupo no tiene ninguna frontera física o geográfica (los *hosts* pueden localizarse en cualquier lugar de la red). Los *hosts* interesados en recibir flujos de datos dirigidos a un grupo en particular, deben unirse a éste utilizando IGMP.

3.4.2 Direcciones IP multicast

Las direcciones multicast especifican un grupo arbitrario de *hosts* IP que se han unido al grupo y desean recibir tráfico enviado a este grupo.

3.4.2.1 Direcciones IP clase D

La IANA ha asignado el espacio de direcciones de la clase D para ser utilizado para IP multicast, las direcciones en este espacio son denotadas por el prefijo binario 1110 en los primeros cuatro bits más significativos del primer octeto como se muestra en la figura 3.5.

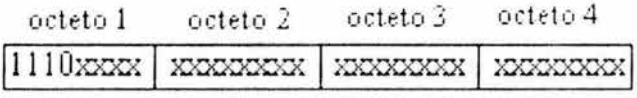


Figura 3.5 Direcciones clase D

Lo que significa que las direcciones de los grupos IP multicast caerán en el rango:

- 224.0.0.0 – 239.255.255.255

Nota: Este rango de direcciones es únicamente para la dirección del grupo (dirección destino) del tráfico multicast. La dirección fuente para los datagramas multicast es siempre la dirección unicast fuente.

3.4.2.2 Direcciones multicast de enlaces locales

La IANA a reservado el rango 224.0.0.0 a la 224.0.0.255 para uso de los protocolos de segmentos de áreas locales. Los paquetes con IP destino dentro de este rango, no serán reenviados por los enrutadores (sin importar su tiempo de vida TTL) por lo tanto no

saldrán de la red local. A los enrutadores que no permiten la salida a estos paquetes se les llama *"broken routers"*.

La **Tabla 3.1** muestra algunas de las direcciones multicast reservadas tomadas de la base de datos de la IANA. La tabla lista las direcciones reservadas para enlaces locales. Los protocolos de red utilizan estas direcciones para descubrimiento automático de enrutadores y para comunicar información importante para el enrutamiento. Por ejemplo, OSPF utiliza 224.0.0.5 y 224.0.0.6 para intercambiar información de estado de enlace.

Dirección	Uso	Referencia
224.0.0.1	All Hosts	RFC 1112, JBP
224.0.0.2	All Multicast Routers	JBP
224.0.0.3	Unassigned	JBP
224.0.0.4	DVMRP Routers	RFC 1075, JBP
224.0.0.5	OSPF Routers	RFC 1583, JXM1
224.0.0.6	OSPF Designated Routers	RFC 1583, JXM1
224.0.0.7	ST Routers	RFC 1190, KS14
224.0.0.8	ST Hosts	RFC 1190, KS14
224.0.0.9	RIP2 Routers	RFC 1723, SM11
224.0.0.10	IGRP Routers	Farinacci

Tabla 3.1 Direcciones multicast de enlace local

3.4.2.3 Direcciones globales.

Al rango de direcciones de la 224.0.1.0 a la 238.255.255.255 se les llama *Globally Scoped Address*. Se pueden utilizar para enviar datos a través de multicast entre organizaciones y a través de la Internet.

Algunas de estas direcciones han sido reservadas para ser utilizadas por aplicaciones multicast por la IANA. Por ejemplo la dirección 224.0.1.1 ha sido reservada para NTP (*Network Time Protocol*).

Para obtener mayor información sobre direcciones multicast reservadas, se sugiere se visite la siguiente dirección: http://www.iana.org/assignments/multicast_addresses

3.4.2.4 Direccionamiento glop (de revoltijo).

El RFC-2770 propone que el rango de direcciones 233.0.0.0/8 sea reservado para direcciones definidas estáticamente por organizaciones que ya cuentan con un número AS (*Autonomous System*) reservado. El número AS del dominio se incrusta dentro del segundo y tercer octetos del rango 233.0.0.0/8.

Por ejemplo, el AS 62010 se escribe en hexadecimal (base 16) como F23A. Al separar los dos octetos F2 y 3A se obtienen los números 242 y 58 en decimal (base 10). Esto nos daría una subred de 233.242.58.0 que sería reservada globalmente para ser utilizada por el AS 62010

3.4.3 Direcciones multicast de capa 2.

Normalmente las Tarjetas de Red (NIC's) en un segmento LAN sólo recibirán paquetes destinados para su dirección MAC o a la dirección MAC de broadcast. Por lo anterior se ideó una forma en la que múltiples *hosts* pudieran recibir el mismo paquete y aún así ser capaces de diferenciar entre grupos multicast.

La norma 802.3 usa el bit 0 del primer octeto para indicar una trama broadcast y/o multicast. La figura 3.6 muestra la localización del bit Broadcast/Multicast en una trama Ethernet.

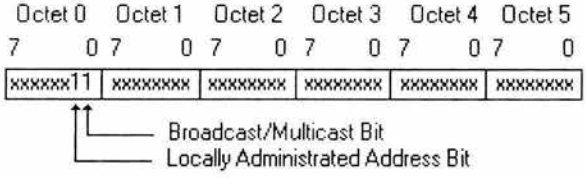


Figura3.6 Formato de la dirección MAC para la norma IEEE 802.3

3.4.3.1 Mapeo de direcciones MAC Ethernet

La IANA posee un bloque de direcciones MAC Ethernet que comienza con 01:00:5E en hexadecimal. La mitad de este bloque está asignado para direcciones multicast. Esto define al rango que va de la dirección 0100.5e00.0000 a la 0100.5e7f.ffff como el rango válido para direcciones Ethernet MAC.

Esta asignación permite usar 23 bits en la dirección Ethernet para corresponder al grupo de direcciones IP multicast. El mapeo coloca a los últimos 23 bits de la dirección IP del grupo multicast en estos 23 bits disponibles en la dirección Ethernet (como se muestra en la Figura 3.7).

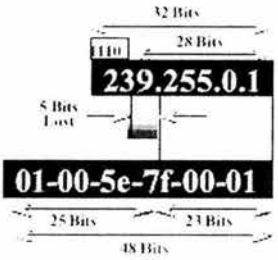


Figura 3.7 Mapeo de IP multicast a Dirección MAC Ethernet

Como se mostró previamente, en la sección 3.4.2.1, en el rango de direcciones asignado a multicast, los primeros 4 bits siempre son 1110, por lo que la variación de los restantes 28 son los que realmente identifican a cada uno de los grupos. Sin embargo, como sólo se cuenta con 23 bits a utilizar en la dirección Ethernet, los primeros 5 bits de la dirección IP multicast son descartados en este mapeo. Esto provoca que la dirección resultante no sea única. De hecho coinciden 32 distintos identificadores de grupos de capa 3 con la misma dirección Ethernet multicast, figura 3.8.

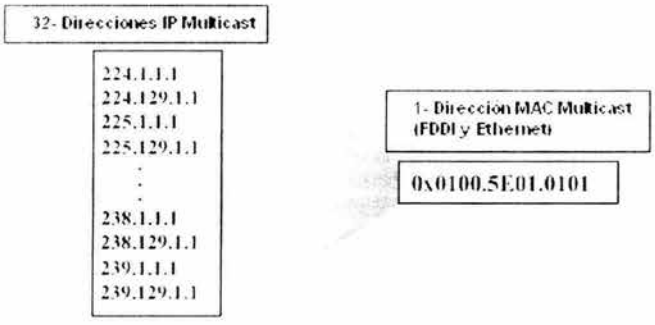


Figura 3.8 Ambigüedades en direcciones MAC multicast

3.4.4 Árboles de distribución multicast

Los árboles de distribución son usados para describir las rutas que tomará el tráfico multicast a través de la red.

Los dos tipos básicos de árboles de distribución multicast son:

- Árboles Fuente
- Árboles Compartidos

3.4.4.1 Árboles fuente

La forma más simple de un árbol de distribución es un Árbol Fuente con su raíz en la fuente y sus ramas formando un árbol que se extiende a través de la red hasta los receptores. Debido a que este árbol utiliza la ruta más corta a través de la red, también se le conoce como un *"shortest path tree"* (SPT).

En la figura 3.9 se muestra un ejemplo de SPT para el grupo 244.1.1.1 enraizado en la fuente (*Host* A) y conectando dos receptores (*Hosts* B y C).

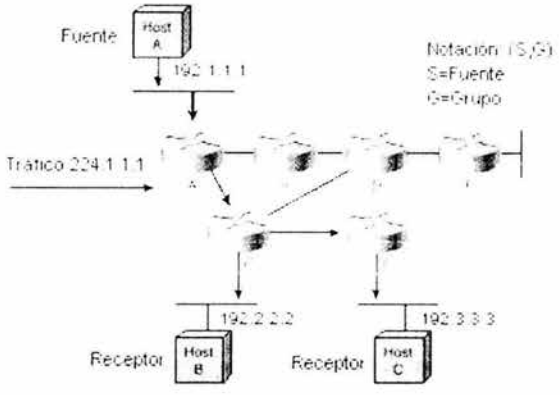


Figura 3.9 Árbol fuente

La notación especial de (S,G) enumera un SPT donde S es la dirección IP unicast de la fuente y G es la dirección IP multicast del grupo. Utilizando esta notación, el SPT de la Figura 3.12 sería (192.1.1.1,224.1.1.1).

La notación (S,G) implica que existe un SPT para cada fuente y para cada grupo destino.

3.4.4.2 Árboles compartidos

A diferencia de los Árboles Fuente que tienen sus raíces en la fuente, los Árboles Compartidos utilizan una raíz común localizada en algún punto de la red al que se le llama Punto de Reunión (RP - Rendezvous Point).

La Figura 3.9 muestra un árbol compartido (*Shared Tree*) para el grupo 224.2.2.2 con la raíz localizada en el enrutador D.

Cuando se utiliza un árbol compartido, las fuentes deben enviar su tráfico a la raíz y luego el tráfico se reenvía, a través del árbol compartido, a los receptores.

Como en este ejemplo todas las fuentes de información comparten el árbol, se utiliza la siguiente notación (*,G) para representar el árbol. En este caso * significa todas las fuentes y G representa el grupo multicast. Por lo anterior el caso que se muestra en la Figura 3.10 podría ser escrito como (*,224.2.2.2).

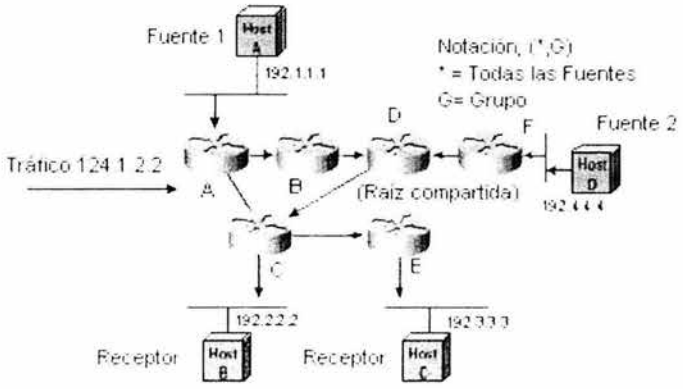


Figura 3.10 Árbol compartido

Los miembros de los grupos multicast pueden unirse o salir en cualquier momento, por lo que los árboles de distribución deben ser actualizados dinámicamente. Cuando todos los receptores activos en una cierta rama dejan de requerir el tráfico para un grupo multicast, los enrutadores remueven dicha rama y le dejan de enviar tráfico. Si un receptor en esa rama pide tráfico multicast, el enrutador modificará dinámicamente el árbol de distribución y volverá a enviar tráfico.

Los Árboles Compartidos pueden ser divididos en: Bidireccionales y Unidireccionales.

Árboles compartidos Bidireccionales

En este caso el tráfico multicast puede fluir arriba y abajo del árbol compartido para alcanzar los receptores. En la figura 3.11 se nota que el tráfico multicast del *host* E se comienza a reenviar en su primer salto, en este caso el enrutador B, hacia arriba del árbol y hacia la raíz del árbol compartido la cual reenvía el tráfico hacia abajo para llegar a los demás receptores.

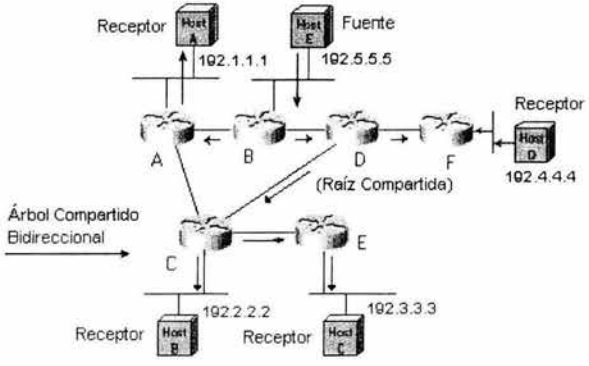


Figura 3.11 Árbol Compartido Bidireccional

Árboles compartidos unidireccionales

Solo permite el flujo de tráfico multicast hacia abajo del árbol, de la raíz a los receptores. Consecuentemente las fuentes deben utilizar un método para enviar el tráfico multicast primeramente a la raíz, como se muestra en la figura 3.12. Uno de los métodos que se pueden utilizar para enlazarse con la raíz es SPT enraizado a las fuentes y empujando el tráfico desde la raíz (compartida) hacia los receptores abajo del árbol.

En la figura 3.12 se muestra como primero el *Host* E se enlaza por medio de SPT a la raíz, en este caso el *Host* D. Cuando la raíz recibe el tráfico éste lo reenvía hacia abajo del árbol compartido hacia los receptores. PIM utiliza este método para enviar el tráfico multicast a la raíz o RP.

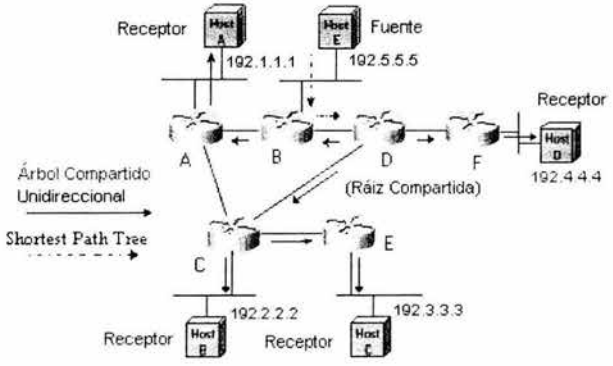


Figura 3.12 Árbol compartido unidireccional, utilizando SPT para enviar el tráfico a la raiz

Otro método para enviar directamente el tráfico multicast a la raíz, en el primer salto, es enviar tráfico unicast directamente a la raíz, figura 3.13.

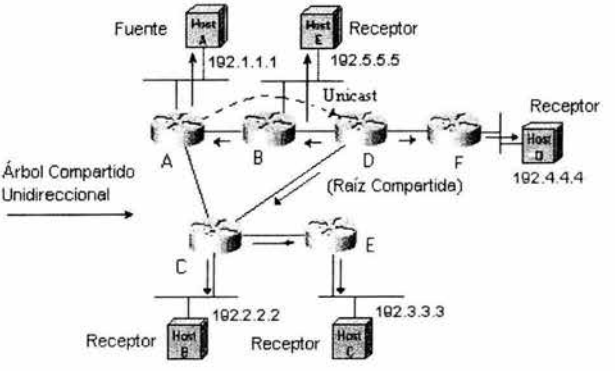


Figura 3.13 Árbol compartido unidireccional utilizando unicast para enviar el tráfico a la raiz

CBT utiliza este método cuando solo una sola fuente envía tráfico a un grupo. En este ejemplo (figura 3.16) el *Host* A es la fuente y el *Host* E es ahora el receptor. El enrutador

B encapsula el tráfico multicast recibido del *Host* A y lo reenvía de forma unicast directamente a la raíz vía túnel IP – IP. La raíz desencapsula los paquetes y los envía hacia abajo del árbol compartido.

3.4.4.3 Ventajas y desventajas de los árboles de distribución

Los SPT tienen la ventaja de crear la ruta óptima entre la fuente y los receptores. Esto garantizará la cantidad mínima de latencia en la red al enviar tráfico multicast. Esta optimización trae consigo algunos inconvenientes. Los enrutadores deben mantener información de la ruta para cada fuente. En una red que tiene miles de fuentes de tráfico multicast y miles de grupos, puede provocar que se agoten los recursos en los enrutadores. El consumo de memoria del tamaño de la tabla de enrutamiento multicast es un factor que debe ser tomado en cuenta al momento del diseño.

Los árboles compartidos tienen la ventaja de requerir la mínima cantidad de espacio en cada enrutador. Esto permite disminuir los requerimientos totales de memoria para una red que sólo permite árboles compartidos. La desventaja de los árboles compartidos se debe a que en ciertas circunstancias las rutas entre la fuente y los receptores pueden no ser las óptimas, lo cual produce algo de latencia en la entrega de paquetes. Por lo anterior es muy importante escoger cuidadosamente la colocación del RP cuando se implemente en un ambiente donde sólo se permitan árboles compartidos.

3.4.5 Reenvío de tráfico multicast

En enrutamiento unicast el tráfico se enruta a través de la red siguiendo una ruta única de la fuente al *host* destino. A un enrutador unicast no le interesa realmente la dirección fuente, sino sólo la destino y cómo reenviar el tráfico hacia esta última.

El enrutador busca en su tabla de enrutamiento y luego reenvía una copia del paquete unicast por la interfaz por la que puede llegar a su destino.

En el enrutamiento multicast, la fuente se encuentra enviando tráfico a un grupo arbitrario de *hosts* que se encuentran representados por una dirección de grupo multicast. El enrutador multicast debe determinar cuál dirección es "*upstream*" (hacia la fuente de tráfico) y cuál o cuáles direcciones son "*downstream*" (hacia el o los destinos). Al concepto de *Reverse Path Forwarding* (RPF) se le puede definir como "enviar tráfico multicast lejos de la fuente" (en vez de "hacia el receptor"), esto quiere decir que los enrutadores evitarán reenviar tráfico multicast a su dirección "upstream" y desecharán el tráfico multicast que provenga de su(s) dirección(es) "downstream".

3.4.6 Reverse Path Fowarding

RPF es un concepto fundamental en el enrutamiento multicast, que permite a los enrutadores enviar correctamente el tráfico multicast al árbol de distribución. RPF hace uso de la tabla de enrutamiento unicast para determinar a los vecinos "*upstream*" y "*downstream*". Un enrutador solo enviará un paquete multicast si se recibe sobre la

interfaz "*upstream*". Esta revisión RPF ayuda a garantizar que el árbol de distribución se encuentra libre de "*loops*".

El mecanismo de verificación de RPF es como sigue:

- El enrutador busca la dirección fuente en la tabla de enrutamiento unicast para determinar si el paquete llegó por la interfaz que es la ruta de regreso a la fuente.
- Si el paquete llegó por la interfaz que lleva a la fuente, la revisión RPF es exitosa y el paquete se reenvía.
- Si la revisión RPF del paso 2 falla, el paquete se descarta.

Las siguientes figuras 3.14 y 3.15, muestran el proceso de verificación RPF.

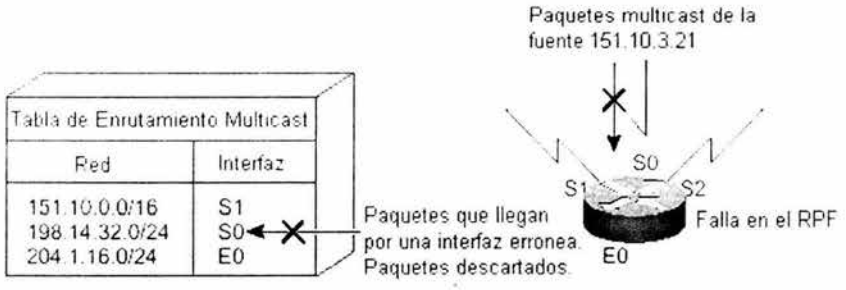


Figura 3.14 Proceso de verificación RPF fallido (paquete descartado)

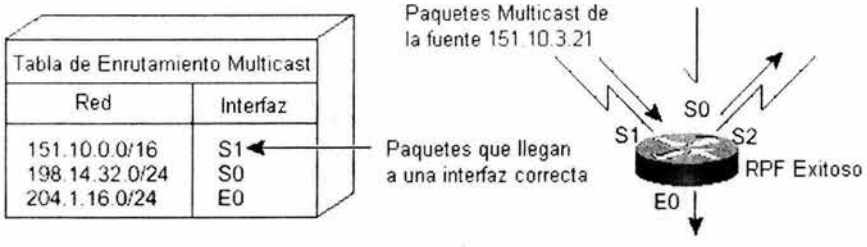


Figura 3.15 Proceso de verificación RPF exitoso (paquete reenviado)

3.4.7 Umbrales TTL

Cada vez que un paquete IP multicast es reenviado por un enrutador, el valor TTL del encabezado del paquete es disminuido en uno. Si este valor TTL es disminuido hasta cero, el enrutador descartará el paquete.

Los umbrales de TTL se pueden aplicar a las interfaces individuales de un enrutador multicast para evitar que los paquetes multicast con un TTL menor que el del umbral sean reenviados fuera de la interfaz, figura 3.16. De esta forma se puede tener un cierto control y administración del reenvió multicast.

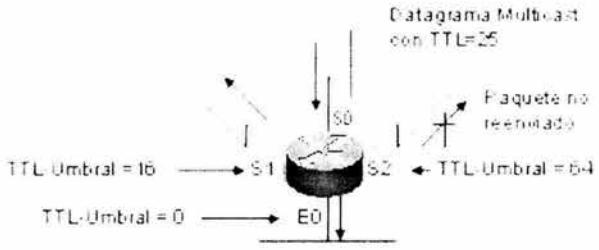


Figura 3.16 Umbral TTL

3.5 *Internet Group Management Protocol (IGMP)*

El uso de multicast en redes TCP/IP está definido como estándar en el RFC 1112. Además de definir las extensiones de direcciones y *hosts* para la compatibilidad de los *hosts* IP con multicast, también define la versión 1 del Protocolo de Administración del Grupo Internet (IGMP). El RFC 2236, define la versión 2 de IGMP. Ambas versiones de IGMP proporcionan un protocolo para intercambiar y actualizar información acerca de la pertenencia de *hosts* a grupos multicast específicos.

Algunos aspectos importantes de multicast son los siguientes:

- La pertenencia a grupos es dinámica, lo que permite a los *hosts* unirse al grupo o abandonarlo en cualquier momento.
- La capacidad de los *hosts* de unirse a grupos multicast se realiza mediante el envío de mensajes IGMP.
- Los grupos no tienen límite de tamaño y los miembros pueden estar repartidos en diversas redes IP (si los enrutadores de conexión admiten la propagación del tráfico multicast y la información de pertenencia a grupos).
- Un *host* puede enviar tráfico IP a la dirección IP del grupo aunque no pertenezca al grupo correspondiente.

3.5.1 Mensajes IGMP

IGMP se utiliza para intercambiar información acerca del estado de pertenencia entre enrutadores IP que admiten multicast y miembros de grupos multicast. Los *hosts* miembros informan acerca de su pertenencia al grupo multicast y los enrutadores multicast sondean periódicamente el estado de las pertenencias a grupos. Los tipos de mensajes IGMP se describen en la Tabla 3.2

Tipo de mensaje IGMP	Descripción
Informe de pertenencia de host	Cuando un host se une a un grupo multicast, envía un mensaje de informe de pertenencia de host IGMP, en el que declara su pertenencia a un grupo de hosts específico. También se envían mensajes de informe de pertenencia de host IGMP en respuesta a una consulta de pertenencia de host IGMP enviada por un enrutador
Consulta de pertenencia de host	Los enrutadores multicast utilizan esta consulta para sondear periódicamente la red en busca de los miembros del grupo
Dejar grupo	Un host envía este mensaje cuando abandona un grupo de hosts si es el último miembro de ese grupo en el segmento de red

Tabla 3.2 Mensajes IGMP

3.5.2 IGMP versión 1

La RFC 1112 define la especificación para IGMP Versión 1. En la figura 3.17 se muestra un diagrama del formato del paquete utilizado en esta versión.

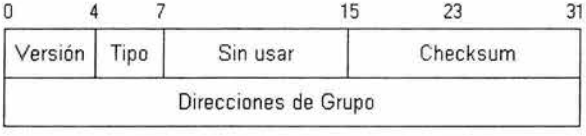


Figura 3.17 Formato del mensaje IGMPv1

Descripción de los Campos

- Campo Versión: El campo versión contiene el identificador de la versión en este caso es (1).
- Campo Tipo: En la versión 1 de IGMP, los dos tipos de mensajes que son utilizados entre el *host* y el enrutador son:
  - Indagación de membresía
  - Reporte de membresía
- Campo *Checksum*: El campo *checksum* es de 16 bits, es el complemento a uno de la suma del complemento a uno del mensaje IGMP. El campo *checksum* es cero cuando se realiza el cálculo de *checksum*.
- El campo Direcciones de Grupo contiene la dirección del grupo multicast cuando es enviado el Reporte de Membresía. Este campo es cero cuando se utiliza dentro del mensaje de Indagación de Membresía, y es ignorado por el *host*.

Los *hosts* envían *Reportes de Membresía* IGMP correspondientes a un grupo multicast en particular para indicar que están interesados en unirse a ese grupo.

El enrutador envía periódicamente una *Indagación de Membresía* IGMP para verificar que al menos un *host* en la subred está interesado aún en recibir el tráfico dirigido a ese grupo. Cuando no hay respuesta a tres Indagaciones de Membresía IGMP el enrutador expirará al grupo y no seguirá enviando tráfico dirigido a ese grupo.

En la figura 3.18 se muestra un ejemplo de cómo ocurre este proceso. El cual permite a todos los enrutadores multicast determinar cuales grupos multicast están activos (esto es, cuales *hosts* están interesados en un grupo multicast) en un red local.

En este ejemplo, los *Hosts* H1 y H2 desean recibir tráfico multicast del grupo 224.1.1.1 además, el *Host* H3 desea recibir tráfico multicast del grupo 224.2.2.2.

Enrutador A es un Indagador IGMP para la subred y es responsable de la administración de las indagaciones. EL enrutador B no es un indagador éste simplemente escucha y graba las respuestas de los *hosts*.

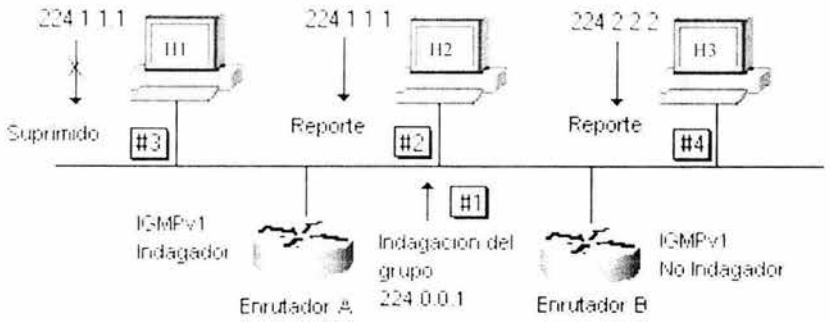


Figura 3.18 Indagación-Respuesta de IGMPv1

3.5.2.1 Proceso IGMPv1 indagación-respuesta.

El mecanismo es el siguiente:

El Enrutador A periódicamente (por defecto 60 segundos) indaga las membresías multicast de IGMPv1 de todos los *hosts* usando el grupo multicast 224.0.0.1 en la subred local. Todos los *hosts* que desean escuchar a este grupo en tanto que estén habilitados pueden recibir esta indagación.

Cuando los *hosts* reciben este mensaje, el primero en contestar en este caso es el *host* H2 el cual envía un Reporte de Membresía del grupo 224.1.1.1 del cual es miembro. En este reporte informa al enrutador de la subred que ese *host* esta interesado en recibir tráfico multicast de dicho grupo.

En el caso del *host* H1, el cual escucha al grupo multicast 224.1.1.1 y el Reporte de Membresía del *host* H2, suprime el reporte que enviaría al enrutador ya que el *host* H2 se ha encargado de esto. Este mecanismo ayuda a reducir el aumento de tráfico en la red local.

Cuando el *host* H3 recibe el Mensaje de Indagación del enrutador, el *host* responde con un Reporte de Membresía pero en este caso para el grupo multicast 224.2.2.2 del cual es miembro. Este reporte informa al enrutador de la subred que está interesado en recibir el tráfico multicast del grupo mencionado.

El resultado es que ahora el enrutador conoce que receptores están activos para los grupos multicast 224.1.1.1 y 224.2.2.2 de la subred local, de forma simultánea, el enrutador B conoce la misma información.

3.5.2.2 Mecanismo de supresión de reportes

Cuando el *host* recibe la indagación de membresía IGMP, el *host* comienza una cuenta descendente del *report-time* para cada grupo multicast con el que desea enlazarse. Cada *report-time* es inicializado con un valor aleatorio entre cero y un intervalo de respuesta

máximo. Por defecto son 10 segundos. Estos 10 segundos corresponden a la espera aleatoria de los usuarios para enviar un reporte y los 60 segundos de arriba corresponden a la indagación del enrutador.

Si el *report-timer* expira, el *host* multicast prepara el reporte de membresía para activar el grupo asociado con un *report-time*.

Si el *host* escucha a otro *host* enviando un reporte de membresía este cancela el *report-time* asociado con el reporte de membresía recibido de tal modo que se suprimen los reportes enviados al grupo

3.5.2.3 Elección del enrutador indagador

Si existen múltiples enrutadores multicast en una subred, más de uno enviará mensajes de indagación, ocupando ancho de banda. Para solventar lo anterior, se escoge a un enrutador de entre todos los conectados a una subred para enviar los mensajes de indagación. En este caso el enrutador responsable de enviar todos los mensajes de indagación se convierte en el enrutador esencial. Desafortunadamente el RFC 1112 no especifica la elección de este enrutador, si no que deja todo este proceso a los Protocolos de Ruteo Multicast (PIM, DVMRP, etc.)

3.5.2.4 Proceso de enlace de IGMPv1

En el caso de que un *host* esté interesado en recibir tráfico de un grupo, éste no espera los mensajes de indagación del enrutador, si no que él mismo envía un reporte de solicitud de membresía para el grupo multicast al cual desea enlazarse.

3.5.2.5 Proceso de abandono de IGMPv1

Desafortunadamente en IGMPv1 no existe un mensaje de abandono, simplemente el *host* detiene el proceso de tráfico del grupo multicast y cesa las respuestas a las indagaciones, el resultado es que el enrutador deja de percibir actividad de un grupo en particular en la subred y es cuando detiene el envío de tráfico.

3.5.3 IGMP versión 2

IGMP Versión 2 trabaja básicamente de la misma forma que la versión 1. La diferencia principal es que existe un mensaje de Abandono de Grupo. Los *hosts* ahora pueden comunicarle activamente al enrutador multicast local su intención de abandonar al grupo.

El enrutador luego envía una Indagación específica de grupo y determina si existe algún *host* interesado en recibir el tráfico.

Si no hay respuesta el enrutador expirará al grupo y no continuará enviando el tráfico. Esto puede reducir en gran medida la latencia de abandono comparándolo con IGMP versión 1.

El RFC 2236 define la especificación para IGMP Versión 2. En la figura 3.19 se muestra un diagrama del formato del paquete de esta versión.

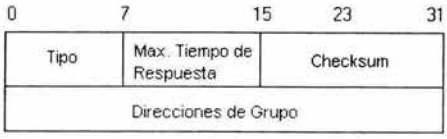


Figura 3.19 Formato del mensaje IGMPv2

Descripción de los campos

- Campo Tipo: En este campo se coloca el código que representa a cada tipo de mensaje en IGMPv2 existen los siguientes tipos de mensajes.
  - Indagación de Membresía (Código = 0x11)
  - Reporte de Membresía versión 1 (Código = 0x12)
  - Reporte de Membresía versión 2 (Código = 0x16)
  - Abandono de Grupo (Código = 0x17)
- Campo Máximo Tiempo de Respuesta: Este campo es nuevo en esta versión, el cual es utilizado solo en el mensaje de Indagación de Membresía Específica y es el tiempo que espera para que el *host* responda al mensaje. Este campo es utilizado para la supresión de reportes ya que se coloca en él, el valor del límite superior del tiempo para poder descartar mensajes. Además este campo puede ser cambiado de tal forma que se puedan controlar y limitar los niveles de latencia de las respuestas de los miembros.
- Campo *Checksum*: El campo *checksum* es de 16 bits, es el complemento a uno de la suma del complemento a uno del mensaje IGMP. El campo *checksum* es cero cuando se realiza el cálculo de *checksum*.
- Campo Direcciones de Grupo: Cuando una Indagación General es enviada, el campo de Dirección de Grupo es fijado a cero para distinguirlo de una indagación de Grupo Específico, el cuál contiene el grupo multicast que es indagado.

  Cuando un Reporte de Membresía o un mensaje de Abandono de Grupo es enviado, este campo contiene la dirección del grupo multicast.

3.5.3.1 Afinación de indagación-respuesta

El campo de Tiempo Máximo de Respuesta permite que el tiempo de respuesta pueda ser configurado sobre la indagación IGMP, la cual informa a todos los *host* el límite superior del retardo de la respuesta a la indagación.

La afinación del valor del Tiempo Máximo de Respuesta controla el umbral del proceso de respuesta.

Esta característica es importante cuando un número grande de grupos son activos en la subred y desean esparcir las respuestas sobre un periodo largo de tiempo. Por ejemplo en la figura 3.20 se describe un diagrama de tiempo en donde se muestra la Indagación General y las respuestas de IGMPv2 de los tiempos ajustados por defecto para la subred con 18 grupos activos esparcidos a través de 18 diferentes *hosts*.

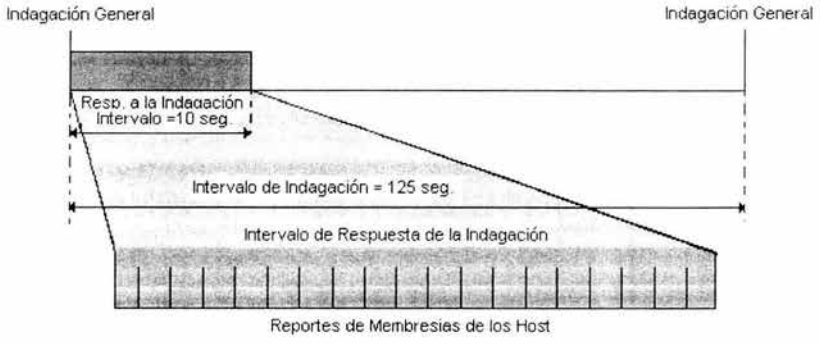


Figura 3.20 Equilibrio entre la Indagación y Respuesta IGMPv2

Se puede ver en este ejemplo que los 18 reportes tienden a expandirse en todo el intervalo de respuesta, esta tendencia se debe a la selección aleatoria de los valores de *report-timer* de los *host* en el proceso de Supresión de Reportes. Esta distribución es afectada por la aleatoriedad de estos números en los *hosts* en la implementación de IGMPv2. Las respuestas, sin embargo generalmente son expandidas a través de la mayoría de los intervalos de Indagación-Respuesta.

Aumentando el valor del Tiempo de Respuesta Máximo, como se muestra en la siguiente figura, el periodo de las respuestas de cada *host* puede esparcirse en el aumento de la Indagación General, de tal modo que se disminuye la frecuencia de las respuestas. Reduciendo frecuencia se crean otros problemas ya que al incrementar el intervalo de Indagación–Respuesta utilizando un valor grande de Tiempo de Respuesta Máximo también disminuye la latencia de abandono porque el enrutador de indagación desea ahora un tiempo más largo para cerciorarse de que no hay *hosts* para el grupo en la subred, figura 3.21.

Por lo tanto, el diseño de la red debe lograr un equilibrio razonable entre la frecuencia de respuestas y la latencia de abandono.

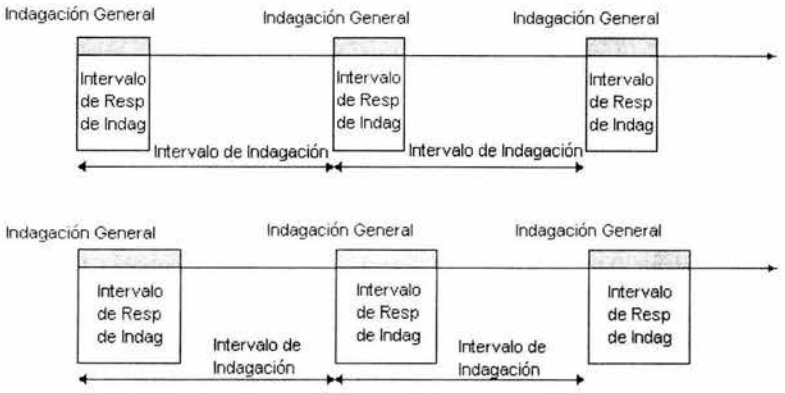


Figura 3.21 Decremento de los Umbrales de Respuesta

3.5.3.2 Mensaje de abandono de grupo

IGMPv2 define un nuevo grupo de mensajes que el *host* envía cuando desea abandonar un grupo. El RFC dice "Cuando un *host* abandona un grupo multicast, debe enviar un mensaje de Abandono de Grupo a todos los enrutadores multicast del la red (Grupo 224.0.0.2)", es importante mencionarlo ya que en IGMP Snooping para la aplicación en *switches* LAN no siempre es bueno enviar mensaje de abandono. Esto se explicara más adelante.

3.5.3.3 Mensajes de indagación de grupos específicos

Este es otro tipo nuevo de mensaje implementado en esta versión, esto es enviar un mensaje de indagación a un grupo especifico y no a todos, al recibir este mensaje el *host* responde de la misma manera como lo haría cuando responde a una indagación general.

3.5.3.4 Proceso de abandono IGMPv2

Este proceso disminuye de manera considerable el tiempo de latencia y se lleva acabo como se muestra en la figura 3.22.

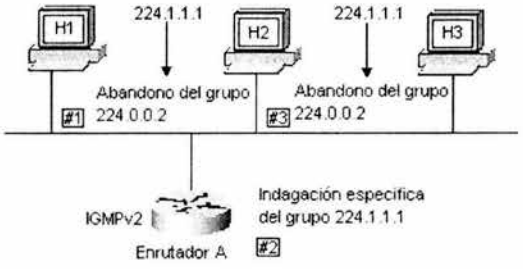


Figura 3.22 Proceso de abandono utilizando IGMPv2

En este ejemplo el *host* H2 y H3 son miembros actuales del grupo 224.1.1.1 aunque H2 desea abandonar el grupo. La secuencia de eventos es como sigue:

- El *host* H2 envía un mensaje de abandono de grupo, a todos los enrutadores multicast de la red (224.0.0.2).
- El enrutador A (asumiendo que este es el enrutador de indagación) escucha este mensaje. Sin embargo éste mantendrá en la lista solo los miembros que estén activos para este grupo en la subred. El enrutador enviará un mensaje de indagación específica para saber cuantos de los *hosts* permanecen en el grupo 224.1.1.1 por lo que solo los miembros a este grupo responderán.
- EL *host* H3 responde con un Reporte de Membresía, de esta forma le indica al enrutador que hay un miembro presente en ese grupo.

3.5.3.5 Proceso de elección del indagador

IGMPv2 utiliza la dirección IP dentro de los mensajes de indagación general para elegir el enrutador indagador de acuerdo al siguiente procedimiento.

Cuando los enrutadores son encendidos envían un mensaje general de indagación al grupo **Todos los Sistemas Multicast** (224.0.0.1) con la dirección de su interfaz dentro del campo "dirección fuente". Cuando los enrutadores reciben los mensajes, comparan esa dirección con la propia. El enrutador con la dirección IP más baja es el elegido para ser el indagador.

Todos los demás enrutadores inician un temporizador (250 segundos por omisión, que es el doble del tiempo de indagación), el cual se reinicia cada vez que se recibe un mensaje general de indagación. Si no se recibe ningún mensaje durante 250 segundos se asume que el indagador está abajo y el proceso se inicia de nuevo.

3.5.4 Compatibilidad de IGMPv1 e IGMPv2

3.5.4.1 *Host* V2 / Enrutador V1

En esta relación los reportes de membresía versión 2 enviados por los *hosts*, son ignorados por los enrutadores. Por lo tanto los *hosts* deben enviar reportes de membresía versión 1 siempre que detecten indagaciones de versión 1, además también deben dejar de enviar mensajes de Abandono de Grupo en esta situación.

Los *hosts* pueden detectar las diferencias en las indagaciones IGMPv1 e IGMPv2 examinando el octeto correspondiente al campo "*Maximum Response Time*". En las indagaciones IGMPv1 este campo es cero, mientras que en las indagaciones IGMPv2 este campo siempre es diferente de cero.

Para mantenerse este estado, el *host* versión 2 inicia un temporizador de 400 segundos cuando detecta una indagación IGMPv1. Este temporizador se reinicia a 400 segundos

cuando otra indagación IGMPv1 es recibida. Si este temporizador expira entonces el *host* comienza a enviar de nuevo mensajes de membresía IGMPv2.

3.5.4.2 *Host* V1 / Enrutador V2

Los *hosts* versión 1 responden de manera normal a las indagaciones tanto de IGMPv1 e IGMPv2 porque, esencialmente, las dos tienen el mismo formato. Sólo el segundo octeto es diferente en los mensajes IGMPv2 y es ignorado por los *hosts* versión 1.

De cualquier forma los reportes IGMPv1 siempre serán recibidos por el enrutador. Como el *host* V1 no entiende los reportes de la versión 2 los ignora, por lo tanto no se activa la supresión de reportes.

Siempre que exista un *host* versión 1 miembro de un grupo, el enrutador debe ignorar los mensajes de abandono (de los *hosts* versión 2) para dicho grupo. Además el enrutador debe establecer un temporizador para notar que aún existe un *host* versión 1 miembro del grupo.

3.5.4.3 Mezcla IGMPv1 e IGMPv2 en enrutadores

Si existe un enrutador IGMPv1 sobre una subred, los demás enrutadores presentes deberán operar entonces también con IGMPv1. Para ello es recomendable configurar manualmente IGMPv1 en el resto de enrutadores que se encuentren dentro de la subred.

3.5.5 IGMP versión 3

IGMP Versión 3 (IGMPv3) es el siguiente paso en la evolución de IGMP. IGMPv3 permite a un *host* receptor de multicast señalar a un enrutador los grupos de los cuales quiere recibir tráfico multicast y de cuáles fuentes se espera este tráfico. Esta información de membresía permite al enrutador reenviar el tráfico únicamente de aquéllas fuentes de las cuales los receptores pidieron el tráfico.

En IGMPv3, existen los siguientes tipos de mensajes:

- Escrutinio de membresía Versión 3
- Reporte de membresía Versión 3

Un diagrama del formato que tiene el paquete de escrutinio para un mensaje IGMPv3 se muestra en la figura 3.23.

Figura 3.23 Formato del mensaje de escrutinio de IGMPv3

La Tabla 3.3 describe los campos más importantes en el mensaje de escrutinio IGMPv3.

Campo	Descripción
Tipo = 0x11	Indagación IGMP.
Codigo de Respuesta Máxima	Código de respuesta máxima (en segundos). Si el código es menor que 128, entonces es igual al tiempo de respuesta máximo. Si el código es mayor o igual a 128, es representado por un valor de punto flotante (en mantisa y formato del exponente).
Direcciones de Grupo	La dirección es 0.0.0.0 para Indagaciones Generales.
S	Bandera S. Esta bandera indica cuando el procesamiento del enrutador es suprimido.
QRV	Querier Robustness Value. Este valor afecta contadores de tiempo y el número de recomprobaciones.
DIC	Querier's Query Interval Code (en segundos). Si el código es menor que 128, entonces es igual a Querier's Query Interval. Si el código es mayor o igual a 128, entonces representa un valor de punto flotante (en mantisa y formato del exponente).
Número de Fuentes [N]	Numero de Fuentes presentes en la Indagación. Este número es distinto a cero para una indagación del grupo-fuente.
Dirección de Fuentes [1...N]	Dirección de la fuente

Tabla 3.3 Campos del mensaje de escrutinio

La figura 3.24 muestra un diagrama del formato del paquete de reporte para un mensaje IGMPv3.

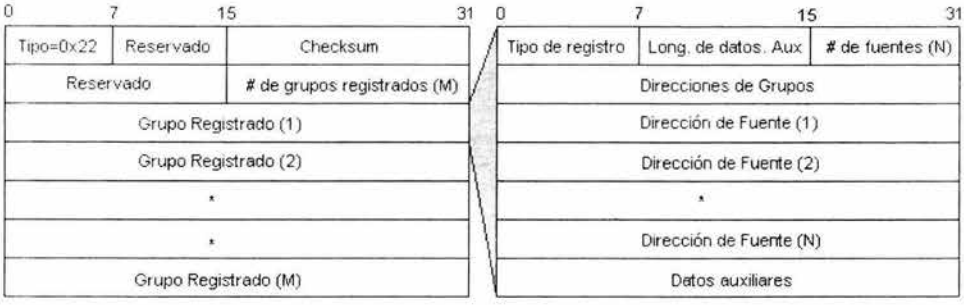


Figura 3.24 Formato del paquete de reporte para un mensaje IGMPv3

Descripción de los campos del mensaje de reporte IGMPv3, en la Tabla 3.4

Campo	Descripción
# de grupos registrados [M]	Numero de registros de grupo presentados en el reporte.
Grupo registrado [1...M]	Bloque de los campos que contienen la información con respecto a la calidad de los miembros de las Fuentes con un solo grupo multicast de la interfaz de la cual el informe fue enviado.
Tipo de Registro	Tipo de registro de grupo.
# de Fuente [N]	Numero de Fuentes presentadas en el registro.
Dirección Fuente [1...N]	Dirección de la fuente.

Tabla 3.4. Descripción de los campos del mensaje de reporte IGMPv3.

IGMPv3 soporta aplicaciones que señalan explícitamente fuentes de las cuales desean recibir tráfico.

Con IGMPv3, los receptores indican su membresía a un grupo multicast de *hosts* en los siguientes dos modos:

- Modo de INCLUSIÓN. En este modo el receptor anuncia su membresía a un grupo y le entrega una lista de direcciones fuente (la lista de INCLUSIÓN) de la cual desea recibir el tráfico.
- Modo de EXCLUSIÓN. En este modo el receptor anuncia su membresía a un grupo multicast y le entrega una lista de direcciones fuente (la lista de EXCLUSIÓN) de las cuales no desea recibir tráfico. El *host* sólo recibirá tráfico de las fuentes cuya dirección IP no esté listada en la lista de EXCLUSIÓN

Para recibir tráfico de todas las fuentes (de la misma forma en la que se comporta IGMPv2), el *host* utiliza un modo de membresía de EXCLUSIÓN con una lista de EXCLUSIÓN vacía.

3.6 Aplicaciones multimedia

Para mucha gente, en lo primero que piensan con respecto al término IP multicast es en video conferencia recordando las partes que lo componen, el video y el audio simultáneamente sin embargo estos términos por si solos son muy simples de comprender no así el proceso que lleva unirlos y trasladarlos de un lugar a otro no solo a un receptor sino a muchos en tiempo real.

3.6.1 RTP (Real-Time Protocol)

RTP (*Real Time* Protocol) [14] está documentado en el RFC 1889, este protocolo se creó atendiendo la necesidad de cubrir las deficiencias de TCP/IP ya que éste servía para casi todo excepto para enviar información en tiempo real (audio y video).

Sin embargo el encapsulamiento de tráfico en tiempo real no es suficiente ya que RTP no presenta ninguna calidad de servicio, de tal forma se crea el protocolo RTCP (Real Time Control Protocolo) de esta forma al Protocolo RTP es conocido como la unión de dos componentes.

1. RTP: Componente que lleva los datos en tiempo real.
2. RTCP: Componente de Control el cual provee la información acerca, de los participantes de las sesiones y monitoréa la entrega de los datos utilizando algunas simples medidas de calidad de servicio QoS como la edad de los paquetes etc.

De tal manera que podemos nombrar las características de RTP

- Datos: Temporización, Detección de Perdidas, Etiquetados de Contenidos.
- Control: Realimentación de QoS, Estimación de miembros y detección de bucles.

Funcionalidad

- Segmentación realizada por UDP o IP.
- Resecuenciación de paquetes.
- Detección de perdidas para estimación posterior
- Identificación de la fuente.

Para comprender un poco más acerca de RTP y RTCP se explicara un ejemplo de Audio Conferencia.

Las aplicaciones tipicas de multimedia multicast se les asigna dos puertos uno para RTP y otro para RTCP en le primero para la cadena de datos en este caso el audio y el otro para la cadena de control.

El audio entrante es muestreado en pequeños ranuras de tiempo o slots (por ejemplo 40ms) para una aplicación de audio. EL audio de estas ranuras de tiempo son codificadas utilizando algunas de las técnicas conocidas, como lo es PCM (*Pulse Code Modulation*), ADPCM (*Adaptive Differential Pulse Code Modulation*) entre otras. Los datos ya codificados son almacenados dentro del paquete RTP. El encabezado del paquete RTP contiene la secuencia de los números y las etiquetas de tiempo que indicarán el esquema de codificación utilizado.

Cuando el paquete RTP de la aplicación de audio es recibido, se utilizan los números y las etiquetas de tiempo para determinar cuantos paquetes se han perdido, la codificación de cada paquete es almacenada en el buffer del destino ordenado según los números y las etiquetas. Para después ser decodificado y poder recobrar el audio.

Los problemas que se llegan a presentar son con respecto al tamaño del buffer y al congestionamiento de la red, esto provoca que el audio se escuche entrecortado. Al utilizar un buffer grande al tratar de buscar los datos en él se puede llegar a tener un retardo, lo cual afectaría si se estuviera transmitiendo en conjunto con una videoconferencia.

De tal forma que es de suma importancia conocer quienes participaran en la videoconferencia y quienes recibirán la transmisión de la aplicación de audio. Los receptores multicast periódicamente introducen reportes (RR) dentro del paquete RTCP de control desde el puerto. Estos reportes contienen los nombres de los usuarios y la información del número de paquetes perdidos y del intervalo de espera de cada fuente dentro de la conferencia. Las fuentes utilizan esta información para saber cual de las transmisiones esta siendo recibida por cada receptor y en algunos casos cambiar a algún otro método de codificación y mejorar la recepción.

Las fuentes, o en ocasiones llamados remitentes, también envían reportes multicast (SRs) dentro de los paquetes RTCP de control por el mismo puerto. Estos reportes contienen la misma información que los reportes de los receptores sin embargo se les agrega 20 bytes en la sección que contenían las marcas de tiempo. Los miembros del grupo utilizan esta información para calcular el *round-trip-time* (RTT) y otras estadísticas sobre el flujo de tráfico.

3.6.2 Protocolo de control RTCP

Todas las aplicaciones basadas en RTP utilizan RTCP para transmitir periódicamente sesiones de control de información hacia todos los participantes de la conferencia, este proceso se describe a continuación.

Cada flujo RTP cuenta con información adicional proporcionada por RTCP utilizada para realizar un seguimiento de la calidad de la transmisión como son los detalles sobre participantes y estadísticas de rendimiento y pérdidas, que permiten realizar cierto control de flujo y congestión, permitiendo la determinación de congestiones locales o generalizadas.

Mediante CNAME (*Canonical Name*) se puede identificar sin equivocaciones a un emisor RTP y asociar la información a los receptores, por otra parte relaciona el reloj del emisor y las marcas de tiempo, lo cual es necesario para sincronizar entre flujos, pues las marcas de tiempo de diferentes flujos no son comparables por ser relativas, además de proveer de codificación adaptable, esto es, codificar la diferencia con el valor anterior de la muestra.

Al utilizar RTP sobre IP multicast se puede permitir que la aplicación escale a un número grande de participantes.

De tal forma que podemos decir que: toda estación terminal dentro de una sesión multicast multimedia basada en RTP es una fuente de tráfico multicast.

3.6.3 Protocolo de anuncio de sesión (SAP)

SAP es un protocolo de anuncio para sesiones de conferencias multicast y fue desarrollado por Multiparty Multimedia Session Control (MMUSIC) en conjunto con Internet Engineering Task Force (IETF). La versión actual SAPv1 es descrita en IETF-Draft: draft-ietf-mumusic-sap-00.txt.

Los clientes SAP se anuncian periódicamente en la sesiones de conferencia mediante paquetes multicast que contienen información de las sesiones (direcciones multicast y puertos), esta información es determinada por el protocolo SDP (Protocolo de Descripción de Sesiones), el cual esta basado en un formato de texto ASCII donde cada línea de texto corresponde a un tipo de anuncio diferente y corresponde a una descripción de sesión en particular .Los anuncios SAP tiene un rango límite con respecto al ancho de banda consumido, cuando el límite es manejado por medio de TTL por cada sesión y del total de números de anuncios enviados por un cliente SAP. La tabla 3.5 muestra según las especificaciones de SAP este rango de AB límite.

TTL	Ancho de Banda
1 a 15	2kbps
16 a 63	1kbps
64 a 127	1 kbps
128 a 255	200 bps

Tabla 3.5 Relación TTL- ancho de banda

Cuando se determina de forma manual los límites de anuncio, es recomendado asignar un ancho de banda no mayor a 500bps.

CAPÍTULO 4. Multicast en capa 2

En los últimos años, los equipos de capa 2: "LAN *Switches*", han ido de una tecnología muy costosa, la cual era implementada solamente en el *backbone* de la red, a una tecnología relativamente madura, rentable y que en la actualidad se encuentra ya formando parte de la infraestructura normal de prácticamente cualquier red LAN.

En este capitulo trataremos algunas de las implementaciones IP multicast en este tipo de *switches* LAN.

4.1 Multicast sobre redes LAN

4.1.1 Características de los *switches* LAN

El funcionamiento de un *switch* es relativamente simple, al llegar una trama al *switch* en lo primero que éste se fija es en la dirección MAC destino. Esta dirección es buscada en una tabla de direcciones del *switch* para saber por qué puerto debe ser reenviada dicha trama. Para esto el *switch* debe aprender la dirección MAC de la estación terminal y el puerto donde ésta está conectada. Esto lo logra con el intercambio de tramas que realizan la estación y el *switch* al conectarlos o de forma más precisa al configurar el *switch*.

Para crear su tabla el *switch* registra estos datos, tanto la MAC y el puerto, en una memoria de contenido direccional llamada CAM (*Content-Addressable Memory*) de tal forma que el proceso mencionado es más rápido.

En la figura 4.1 se muestra el diagrama de bloques de un *switch* típico de alto desempeño, en esta figura se muestra que el Centro de Conmutación utiliza la información dentro de la tabla CAM para tomar decisiones de reenvió. Por su parte el CPU esta conectado a los puertos por medio del Centro de Conmutación.

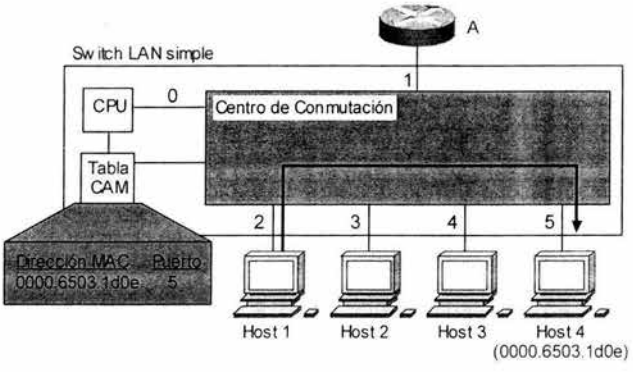


Figura 4.1 Arquitectura de Switch LAN simple.

En el diagrama se muestra como el *switch* utiliza la tabla CAM para enviar una trama del *Host* 1 al *Host* 4.

Tratándose de tráfico multicast, el comportamiento por defecto de un *switch* de capa 2 es el reenviar todo el tráfico a cada puerto que pertenece a la misma LAN dentro del *switch*. Esto entorpece el propósito del *switch*, el cual tiene como función limitar el tráfico sólo a los puertos donde existen receptores interesados en recibirlo.

Para tratar este problema existen hasta la fecha tres métodos.

- IGMP Snooping
- Cisco Group Management Protocol (CGMP)
- IEEE's Generic Attribute Resolution Protocol (GARP)

Los dos primeros se tratarán de forma completa, el tercero es relativamente nuevo y requiere cambios drásticos, principalmente en las estaciones terminales como lo es nuevo *hardware* y *software* para implementar la solución.

4.1.2 IGMP Snooping

El proceso de IGMP *Snooping* requiere que el *switch* LAN examine o "fisgoneé" parte de la información de capa 3 en los paquetes IGMP enviados entre los *hosts* y el enrutador.

Cuando el *switch* escucha un Reporte IGMP de un *host* para un grupo multicast en particular, el *switch* añade el número del puerto del *host* a la tabla CAM asociada a multicast. Cuando el *switch* escucha el mensaje IGMP de abandono de grupo de un host, éste remueve el puerto de la tabla de entradas.

Al ser transmitidos los mensajes de control de IGMP como paquetes multicast, éstos son indistinguibles de datos multicast de capa 2. Un *switch* que corre IGMP Snooping debe examinar cada uno de los paquetes de datos multicast para revisar si contiene cualquier tipo de información de control IGMP. Si IGMP Snooping se implementa en un *switch* con un CPU lento podría provocar un grave impacto especialmente cuando los datos se transmiten a altas tasas. Si se va a implementar IGMP Snooping se necesitarán *switches* con procesadores poderosos y con ASIC's (Application-Specific Integrated Circuit) especiales que puedan realizar las revisiones de IGMP en *hardware*.

A continuación se explicarán algunos casos y ejemplos de la implementación de IGMP Snooping.

En la figura 4.2 se muestra lo que ocurre en un caso típico cuando un par de *hosts* se quieren enlazar a un grupo multicast y son dados de alta en la lista CAM para construir el flujo de tráfico multicast hasta ellos.

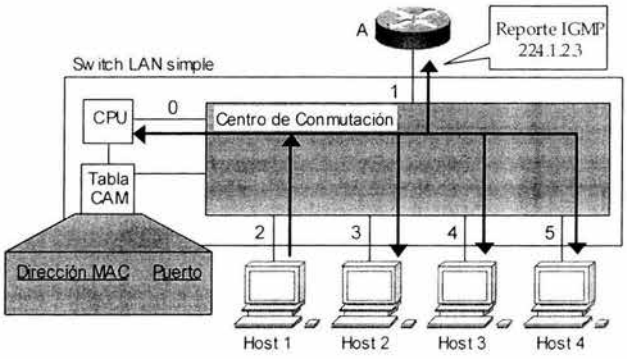


Figura 4.2 Uniéndose a un grupo con IGMP Snooping – Paso 1

1. El *host* 1 (conectado en el puerto 2) desea enlazarse al grupo 224.1.2.3 y envía un reporte de solicitud de membresía IGMP con una dirección MAC destino 0x0100.5E01.0203. Inicialmente, como se muestra en la figura 4.2, la tabla CAM se encuentra vacía, este reporte fluye a través de todos los puertos incluyendo el puerto interno del *switch* conectado al CPU (Puerto 0).

2. Cuando el CPU recibe el reporte, utiliza la información para registrar en la tabla tanto el puerto del *host* (puerto 2), como el puerto del enrutador (puerto 1) y el interno (puerto 0), como se muestra en la figura 4.3.

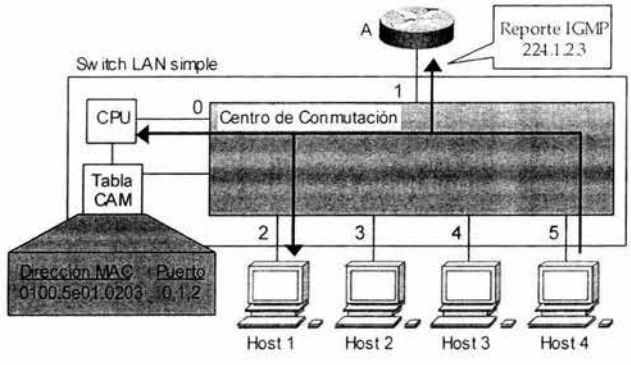


Figura 4.3 Uniéndose a un grupo con IGMP Snooping – Paso 2

Si otro nuevo *host* quiere agregarse al mismo grupo (e.g. el *host* 4 en el puerto 5, ver figura 4.4), entonces la tabla solo será modificada en la sección de puertos ya que se agregará al que desea unirse, siguiendo el procedimiento anterior.

4.1.2.1 Impacto en el desempeño con IGMP Snooping

En el ejemplo anterior no se presentó ningún problema ya que el CPU sólo tenía que verificar los mensajes de membresía de los *hosts* que querían tráfico de ese grupo, de esta manera no se afecta el desempeño del *switch*.

Pero que pasa si ahora el *host* 1 envía un video a 1.5 Mbps (ver anexo E para conocer algunas técnicas de codificación de video) hacia el grupo 224.1.2.3, como en la figura 4.4, todas las tramas de video deberán también ser analizadas por el CPU, es decir que todo el tráfico multicast debe ser analizado trama por trama por el *switch* para poder determinar los mensajes IGMP que van del enrutador a los *hosts* y viceversa.

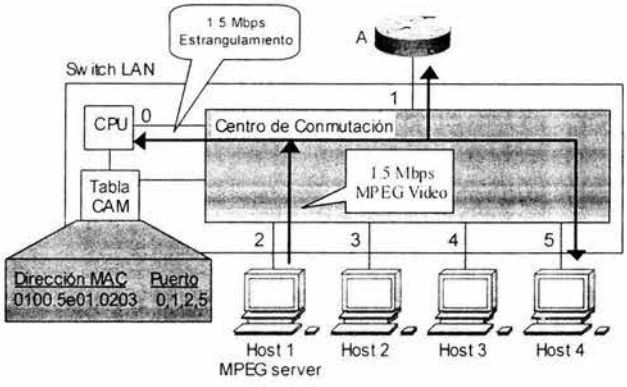


Figura 4.4 Tráfico Multicast sobrecargando el CPU del Switch.

En esta situación el trabajo que realiza el CPU se incrementa de forma considerable, llegando a afectar el desempeño del *switch* a grados que pueden llegar a ser catastróficos para el equipo y la subred. Mientras más tráfico se genere, el *switch* responderá de forma lenta, de tal manera que los enlaces multicast son afectados con respecto a la latencia y puede llegar a cortarse la comunicación con el grupo.

Para resolver este problema es necesario reducir el trabajo del CPU utilizando ASICs (*Application Specific Integrated Circuits*) y las tablas CAM e introducir la verificación de tramas a algo más profundo llegando a analizar la información de capa 3 para que el *switch* tome decisiones.

La tabla CAM es programada para que sólo reenvíe las tramas que contengan mensajes IGMP para ser procesadas por el CPU.

En la figura siguiente se muestra la versión simplificada de los *Switch* LAN después de que el Centro de Conmutación fue rediseñado con un nuevo ASIC para entender la capa 3.

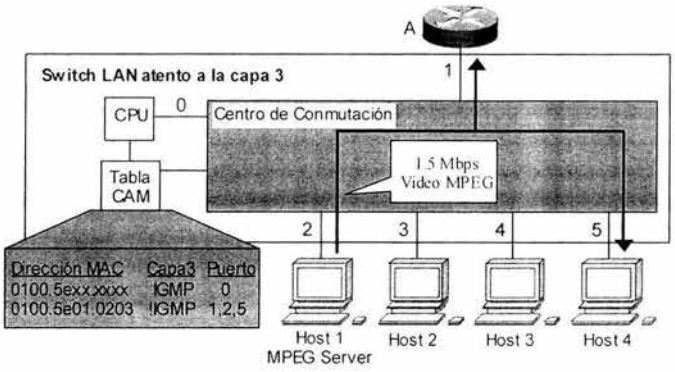


Figura 4.5 Switch con conocimiento de capa 3.

En la figura 4.5 se puede ver como ahora el CPU no se encarga de todo el tráfico que cruza al *switch*, se reduce a unas cuantas tramas IGMP de tráfico por segundo.

Ahora el proceso de enlace de los *hosts* con un grupo es prácticamente igual que en un *switch* LAN simple. A continuación se presentará la descripción de abandono de grupo con IGMP *Snooping*, tomando en cuenta para las siguientes figuras que los *hosts* 1 y 4 son miembros del grupo.

4.1.2.2 Abandono de grupo con IGMP Snooping

Asumiendo que el *host* 1 desea abandonar el grupo, se presentan los siguientes eventos.

1. El *host* 1 envía un mensaje de abandono de grupo al grupo de "Todos los Enrutadores Multicast", grupo 224.0.0.2 (MAC 0x0100.5e00.0002), este mensaje entra en la tabla CAM y también es interceptado por el CPU, el cual no lo reenvía por ningún otro puerto, como se muestra en la figura 4.6.

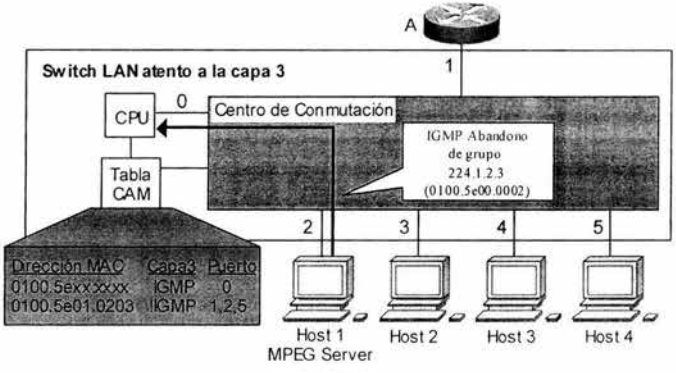


Figura 4.6 IGMP Snooping: Abandono de Grupo – Paso 1

2. El CPU del *switch* en respuesta al mensaje de abandono, envía una indagación General IGMP de regreso al puerto 2 para saber si existen otros *hosts* que sean miembros del grupo sobre ese puerto (esto es cuando múltiples *hosts* son conectados al *switch* por medio de un concentrador), figura 4.7.

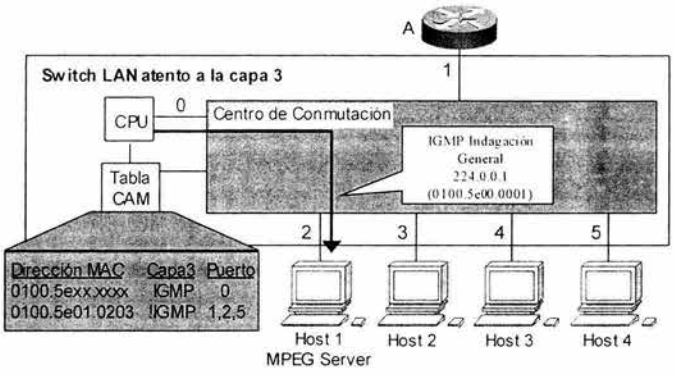


Figura 4.7 IGMP Snooping: Abandono de Grupo – Paso 2

3. Si otro mensaje es recibido por algún *host* conectado en el puerto 2, entonces el CPU rápidamente descarta el mensaje original de abandono de grupo del *host* 1, si por otra parte, el reporte IGMP no es recibido en este puerto (que es el caso de nuestro ejemplo), el CPU borra el puerto de la tabla CAM (ver el resultado en la figura 4.8) de tal forma que no se le envían mensajes al enrutador.

4. Ahora asumamos que el *host* 4 abandona el grupo y envía un mensaje de abandono de grupo IGMP. Una vez más el mensaje de abandono es interceptado por el CPU del *switch* como se muestra en la figura 4.8

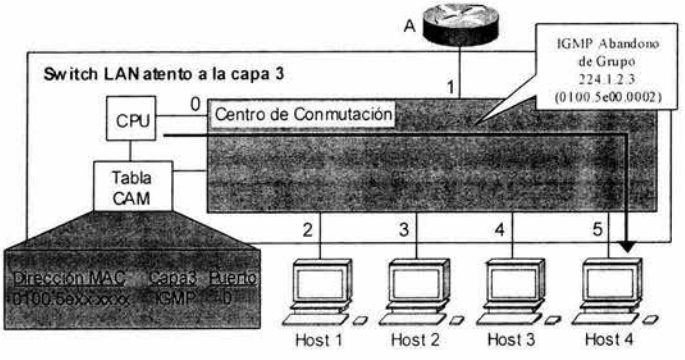


Figura 4.8 IGMP Snooping: Abandono de Grupo – Paso 3

5. El CPU responde enviando una Indagación General al puerto 5, como se ve en la figura 4.9, para saber si existe algún otro *host* que pertenezca al grupo.

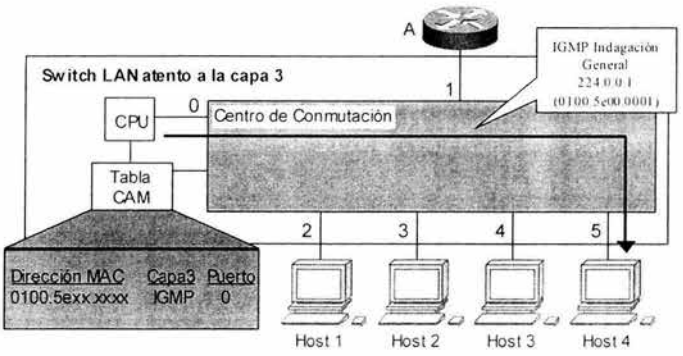


Figura 4.9 IGMP Snooping: Abandono de Grupo – Paso 4

6. Si no existe otro *host* en ese puerto, como es en nuestro ejemplo, y no se recibe ningún mensaje IGMP entonces el puerto es borrado de la tabla CAM y con él la dirección MAC, de tal forma que el CPU borra las entradas para este grupo y reenvía un Mensaje de Abandono de Grupo al enrutador siguiendo el procedimiento normal antes visto en el capitulo 3, figura 4.10.

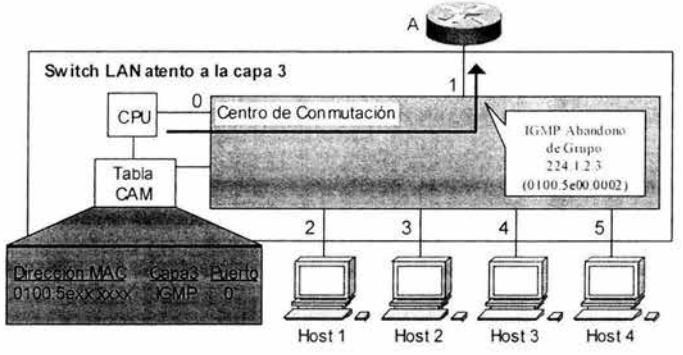


Figura 4.10 IGMP Snooping: Abandono de Grupo – Paso 5

4.1.2.3 Mantenimiento del grupo con IGMP Snooping

Asumiendo que el *host* 1 y 4 nuevamente están enlazados al grupo multicast 224.1.2.3 con los resultados en la tabla CAM mostrados en la figura 4.11 se describirá el procedimiento de mantenimiento.

1. El enrutador A envía periódicamente indagaciones generales a todos los *hosts* usando el grupo 224.0.0.1 (dirección MAC 0x0100.5E00.0001). El CPU del *switch* intercepta estas indagaciones y las transmite a todos los puertos del *switch*.

2. Cada *host* que es miembro de un grupo, en este caso los *host* 1 y 4, envían un reporte IGMP en respuesta a la indagación. Para esto el CPU intercepta todos los mensajes IGMP, por lo tanto cada *host* no puede escuchar los reportes que envían los demás miembros del grupo. De esta manera no hay supresión de mensajes, ya que si existiera, el CPU no podría determinar que puertos quieren recibir el tráfico multicast.

3. Para guardar el estado de miembros del grupo en el enrutador, el *switch* debe enviar uno o más (de preferencia solo uno) reportes IGMP al enrutador A, figura 4.12.

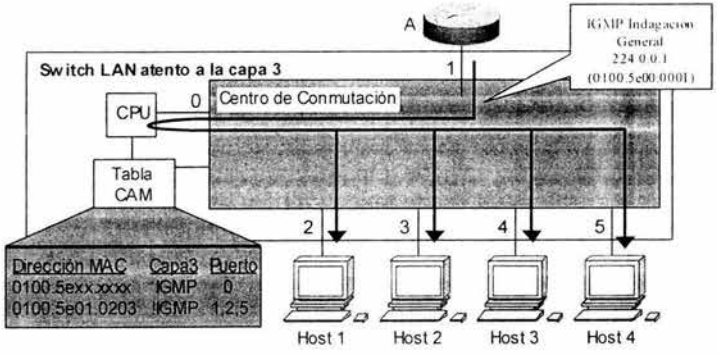


Figura 4.11 IGMP Snooping: Mantenimiento de Grupo – Paso 1

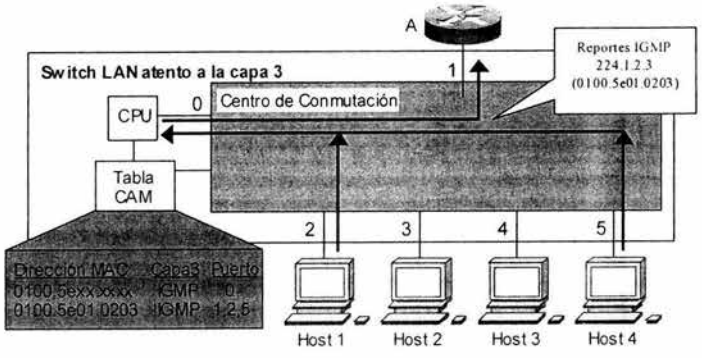


Figura 4.12 IGMP Snooping: Mantenimiento de Grupo – Paso 2

4.1.2.4 IGMP Snooping y fuentes de envió solamente

Las fuentes multicast no requieren de un enlace de grupo multicast al cual le envían información, por lo tanto no necesitan enviar reportes de miembro IGMP, esto representa un problema para el *switch*. Considerando la situación siguiente, supongamos que el *host* 1 desea tomar el papel de fuente y enviar un video a 1.5 Mbps al grupo multicast 224.1.2.3 y asumiendo que ninguno de los *host* esta enlazado con ese grupo, figura 4.13.

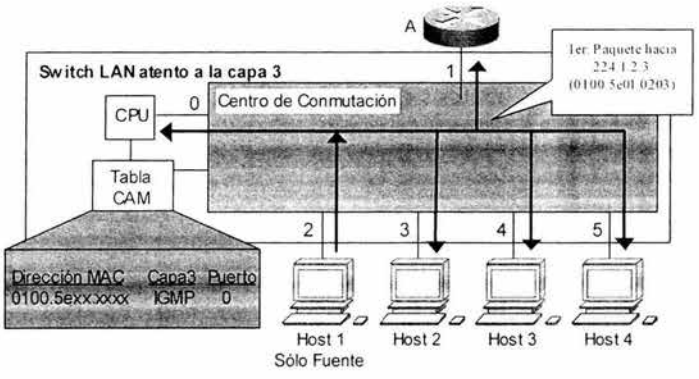


Figura 4.13 IGMP Snooping y Fuentes de envío solamente – Paso 1

Cuando ocurre esta situación, el CPU necesita indiscriminadamente escuchar todas las tramas multicast que fluyen a través de él, por lo cual los *switches* que no tengan un *hardware* especial como ASICs para ayudar a reducir la carga que tiene el CPU, sufren drásticamente en la degradación de su performance. En muchos casos, en los *switch* simples optan por permitir a las fuentes de solo envió, continúen con el flujo a todos los puertos de los cuales algunos o todos pueden enviar reportes de membresía para el grupo.

Hipotéticamente los *switches* más actuales con la facilidad de ver datos de la capa tres, cuando detectan este tipo de circunstancias, responden actualizando la tabla CAM como se muestra en la figura 4.14 de esta forma construye un flujo multicast de solo-fuente solo en el puerto del enrutador. Este paso depende en gran medida de cómo la capa 3 ASICs es implementada dentro del *switch*.

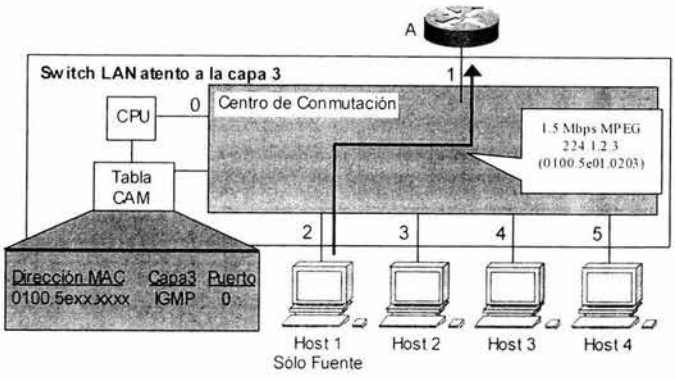


Figura 4.14 IGMP Snooping y Fuentes de envío solamente – Paso 2

4.1.2.5 Detección de enrutadores con IGMP Snooping

Asumiendo que en el *switch* se encuentra conectado más de un enrutador, pero sólo uno funciona como indagador IGMP, los enrutadores restantes conectados al *switch* no enviarán indagaciones generales a diferencia del indagador elegido. En este caso, el *switch* no puede detectar que otros enrutadores están conectados.

La mejor aproximación es, no sólo escuchar las indagaciones IGMP, sino escuchar un tipo especial de protocolo de enrutamiento que pueda facilitar la detección de los enrutadores que están conectados al *switch*. Los *switches* de Cisco están implementados con IGMP Snooping, el cual puede leer paquetes que contienen saludos OSPF PIMv1 y PIMv2 DVMRP, Indagaciones IGMP, CGMP, HSRP (*Hot Standby Router Protocol*) [53], etc. mensajes que podría estar enviando el enrutador periódicamente.

4.1.3 Cisco Group Management Protocol (CGMP)

CGMP es un protocolo desarrollado por Cisco que permite a los *switch* Catalyst utilizar información IGMP en enrutadores Cisco para hacer decisiones de reenvío de capa 2. CGMP tiene que ser configurado tanto en enrutadores multicast como en *switches* de capa 2. El resultado es que con CGMP el tráfico IP multicast se entrega sólo a esos puertos del *switch* Catalyst que están interesados en el tráfico. Todos los demás puertos que no han pedido explícitamente el tráfico no lo recibirán.

El concepto básico de CGMP se muestra en la Figura 4.15. Cuando un *host* se une a un grupo multicast (inciso a), éste envía un mensaje de Reporte de Membresía al grupo destino (para este caso, la dirección 224.1.2.3) El reporte IGMP se pasa a través del *switch* al enrutador para el procesamiento IGMP normal. El enrutador (que debe tener habilitado CGMP en esta interfaz) recibe el reporte IGMP y lo procesa normalmente, pero además también crea un mensaje de CGMP-*Join* y lo envía al *switch*.

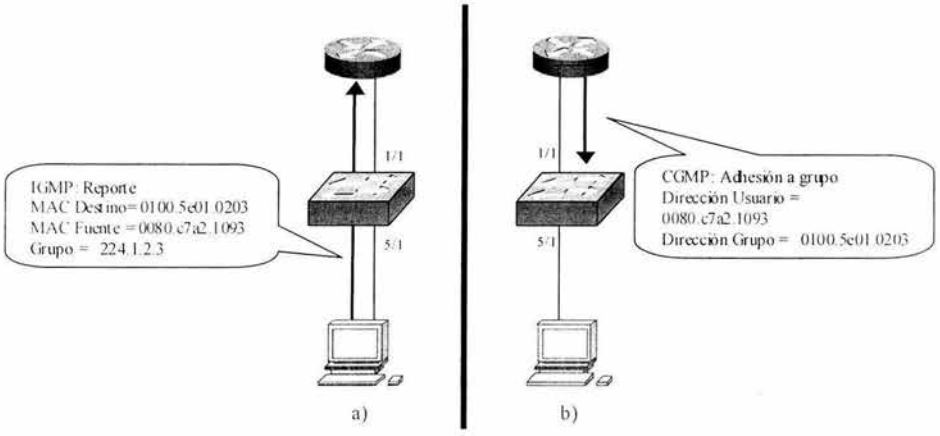


Figura 4.15 Operación Básica de CGMP

El enrutador reenvía los reportes IGMP a través de CGMP hacia el *switch* de la siguiente manera:.

1. Copia la dirección destino MAC 0x0100.5e01.0203 (que corresponde a la dirección IP del grupo multicast 224.1.2.3) del reporte IGMP.

2. Copia la dirección MAC de la fuente, 0x0080.c7a2.1093 (la cual es la dirección MAC unicast del *host* que quiere enlazarse al grupo.

3. Con esta información se construye un mensaje "CGMP-*Join*", el cual se envía al *switch* por medio de una dirección MAC multicast bien conocida: 0x0100.0CDD.DDDD.

Con esto se habilita al *switch* para recibir mensajes CGMP, después ocurren los siguientes eventos.

1. Se busca en la tabla CAM para una entrada de un grupo multicast específico que se encuentra en el campo GDA (*Group Destination Address*) del mensaje CGMP-Join.

2. Si no es encontrado este grupo, el *switch* crea la entrada y agrega todos los puertos de los enrutadores. Nuevamente crea una tabla CAM listando los puertos entrantes (la entrada es creada si todos los enrutadores conectados al *switch* pueden recibir direccionamiento multicast de este grupo).

3. Se busca en la tabla CAM la dirección MAC unicast específica en el campo USA (*Unicast Source Address*) del mensaje CGMP-Join. Copia el número de puerto de la tabla entrante CAM (el puerto conectado al host) a la tabla CAM multicast de entrada encontrada, creada en el paso anterior.

4.1.3.1 Manteniendo el grupo con CGMP

Los puertos individuales únicamente son borrados de la tabla CAM de entrada si se recibe un mensaje de borrar el puerto desde el enrutador. La tabla CAM además tiene un tiempo de expiración el cual es limpiado cada vez que se recibe un mensaje CGMP join para el grupo, esto ocurre siempre que el enrutador envía Indagaciones Generales. Sin embargo la tabla CAM puede ser borrada en las siguientes circunstancias.

- Siempre que la topología de spanning tree de la VLAN cambia (estos cambios pueden ocurrir cuando un puerto de la VLAN cambia en su estado de receptor a transmisor (*learning-forwarding*)
- Cuando el enrutador envía un mensaje de borrado de grupo o un mensaje de borrado de todos los grupos.
- Cuando una de las tarjetas del *switch* es removida.

Siempre que la tabla CAM del *switch* es borrada, el *switch* automáticamente aprende el estado de los puertos a través del mecanismo normal de las Indagaciones Generales IGMP. Durante este periodo de aprendizaje, el *host* envía reportes IGMP en respuesta a las indagaciones del enrutador. Estos reportes alternados, son trasladados por CGMP join en el enrutador, esto causa que el *switch* regenere las entradas en la tabla CAM (Este aprendizaje toma de 1 a 1.5 intervalos completos de indagación).

4.1.3.2 Abandono de grupo con CGMP

Cuando un *host* IGMPv2 desea abandonar un grupo, normalmente reenvía un mensaje de abandono de grupo IGMP al grupo de "Todos los Enrutadores Multicast", 224.0.0.2. Cuando CGMP es habilitado sobre el enrutador y el *switch* LAN, el enrutador puede simplemente colocar el mensaje de abandono de grupo dentro del mensaje de abandono de CGMP utilizando el mismo método utilizado para el enlace, visto anteriormente. Desafortunadamente IGMPv2 no siempre requiere que se envíen mensajes de abandono de grupo, además no siempre se puede utilizar IGMPv2, el sistema *Windows95* solo corre IGMPv1 y en esta versión no existen los mensajes de abandono.

Proceso de abandono local de CGMP

El proceso de abandono local por sí mismo asegura que ningún otro receptor permanezca en el mismo segmento después de que un mensaje de abandono de grupo sea recibido. Si sigue existiendo un miembro, el borrado del puerto es cancelado ya que el flujo debe seguir hacia ese puerto, si ya no hay miembros que respondan sobre el puerto, entonces el *switch* checa si alguno de los miembros sobre otro puerto del *switch*, es miembro, si un miembro es encontrado, entonces no sucede nada, si no existe otro miembro, entonces el *switch* envía un mensaje de abandono de grupo IGMP al enrutador fuente. El enrutador de esta forma se asegura que no existe ningún otro miembro en la LAN.

La figura 4.16 se muestra el proceso local de abandono de CGMP sobre el *switch* cuando múltiples *hosts* son conectados a los puertos del *switch* a través de un medio compartido (concentrador).

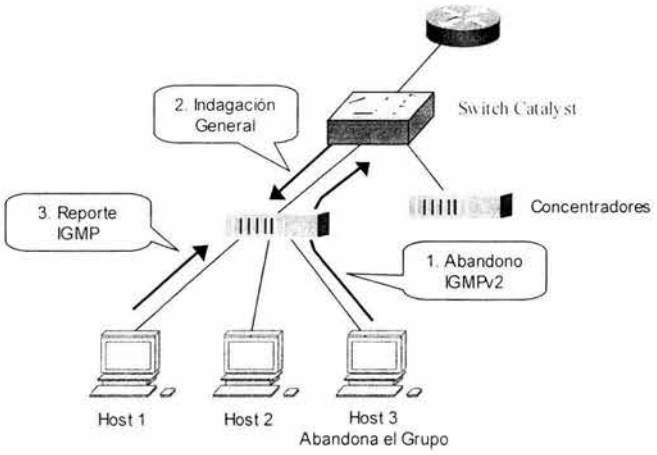


Figura 4.16 CGMP Procesamiento de abandono local – Paso 1

Inicialmente, el *host* 1, 2 y 3 son miembros del mismo grupo multicast. Los tres pasos en la figura se describen a continuación:

1. El *host* 3 desea abandonar el grupo y envía un mensaje de abandono de grupo IGMPv2 al grupo de "Todos los Enrutadores Multicast" (224.0.0.2). Este mensaje es localmente procesado por el *switch* y no es reenviado al enrutador.

2. El *switch* envía una Indagación General IGMP al puerto por donde se recibió el mensaje de abandono para determinar si existe algún miembro perteneciente al grupo (Es importante notar que la contestación es una indagación general IGMP y no una indagación a un grupo especifico de la versión 2 esto es, para que un *host* que solo maneja la versión 1 pueda contestar).

3. El *host* 1 (en este ejemplo) responde con un reporte IGMP para el grupo, el cual le comunica al *switch* que existe un miembro fijo para el grupo en ese puerto por lo cual dicho puerto no es removido de la tabla CAM.

En la figura 4.17 se muestra un proceso de abandono local CGMP cuando un solo *host* abandona el grupo.

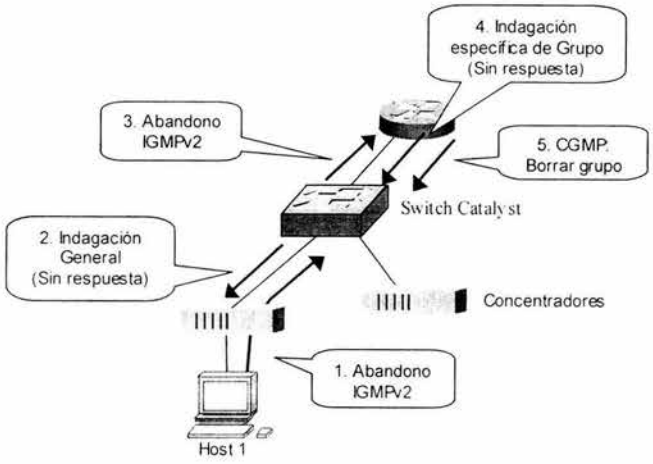


Figura 4.17 CGMP Procesamiento de abandono Local – Paso 2

Los siguientes 5 pasos que muestra la figura son descritos a continuación.

1. El *host* 1 envía un mensaje de abandono de grupo IGMP al *switch* el cual no lo reenvía al enrutador.

2. El *switch* envía una indagación general IGMP al puerto por el cual recibió el mensaje de abandono de grupo.

3. A causa de que no recibe respuesta a la indagación general, el *switch* envía un mensaje de abandono de grupo IGMP al enrutador.

4. El enrutador recibe el mensaje y realiza de forma normal el proceso de abandono de grupo enviando una indagación específica de grupo para cerciorarse de que no exista ningún otro miembro del grupo.

5. A causa de la falta de respuesta, el enrutador borra el estado IGMP del grupo y envía un mensaje de borrado de grupo CGMP al *switch*, esto causa que el *switch* borre de su tabla CAM el grupo.

4.1.3.3 Impacto en desempeño por CGMP

El impacto en el desempeño por la implementación de CGMP en el *switch* es muy bajo comparado con el impacto que tiene con IGMP Snooping. La razón es que el *switch* recibe y procesa un bajo número de tramas del enrutador a diferencia de todas las tramas multicast que deben ser procesadas cuando se tiene IGMP Snooping. Por lo tanto, CGMP puede ser implementado a un bajo costo en los *switches* LAN.

El desempeño en el enrutador por la implementación de CGMP es también muy bajo. En muchos casos los gastos indirectos de CPU adicionales por CGMP en el enrutador son demasiado pequeños y no es un problema para los ingenieros de la red.

4.1.3.4 CGMP y fuentes de solo envío

A diferencia de IGMP Snooping, CGMP no necesita de un procesamiento especial para hacer eficiente el manejo del caso de una fuente de envió solamente, si existen otros miembros del grupo sobre la LAN. En este caso es deseable que el *switch* CGMP construya un tráfico de la fuente que no fluya a todos los otros *hosts* sobre el *switch* que no sean miembros del grupo. Solo un puerto del *switch* necesita recibir el tráfico de la fuente: el puerto conectado al enrutador.

Para un enrutador, detectar esta situación es relativamente simple. Por ejemplo, si un enrutador está recibiendo tráfico de un grupo multicast desde una fuente por una cierta interfaz y no hay estados de membresía para dicho grupo sobre esta interfaz, entonces el enrutador sabe que existe una fuente de solo-envío . En este caso el enrutador responde enviando un mensaje CGMP-Join para sí mismo, de modo que el *switch* crea una nueva tabla CAM multicast de entrada que contiene solo el puerto del enrutador. Si la fuente para de enviar tráfico al grupo, el estado multicast del enrutador queda en tiempo de espera eventualmente. Esto causa que el enrutador envíe un mensaje de borrado de grupo al *switch* para remover la tabla CAM multicast de entradas para el grupo.

4.1.3.5 Detección de enrutadores con CGMP

Aunque los *switch* Catalyst Cisco tienen un comando para designar manualmente un puerto al enrutador, no es necesario ya que cuando un enrutador Cisco es conectado a un *switch*, CGMP asigna un puerto al enrutador habilitando de manera automática CGMP sobre la interfaz por la cual se conectaron. Estos mensajes le comunican al *switch* por cual puerto está conectado al enrutador para que posteriormente sea agregado en las nuevas entradas de la tabla CAM.

4.1.3.6 Otros tipos de problemas en el *switch*eo LAN

Aunque IGMP Snooping y CGMP resuelven en gran medida problema del control de tráfico multicast en un *switch*, quedan algunos problemas fuera de su alcance

- IGMPv1 Leave Latency
- Enlaces entre *switches*
- Enrutamiento basado en *switches*

IGMPv1 Leave Latency

Considerando el ejemplo de la figura 4.18, donde dos *hosts* IGMPv1 están enlazados al mismo grupo. En este caso el enrutador está listo para enviar mensajes CGMP de enlace al *switch* LAN, el cual en respuesta, ha instalado una tabla CAM para limitar las tramas multicast de las direcciones de grupo a solo aquellos puertos donde están conectados el enrutador y los dos *hosts*.

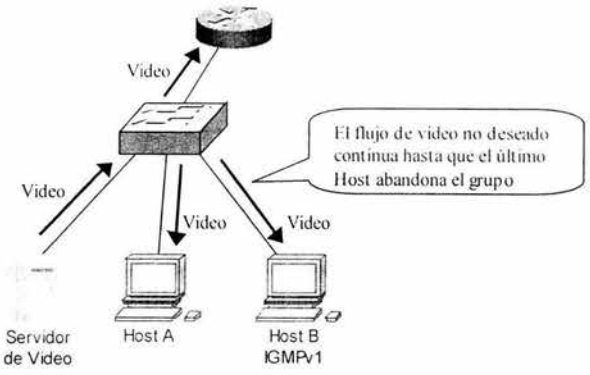


Figura 4.18 IGMPv1 Problema de la Latencia de Abandono

Ahora, si el *host* B abandona el grupo, simplemente lo hace sin enviar algún tipo de mensaje. Esto da lugar a que el *switch* no pueda remover los puertos de los *hosts* de la tabla CAM, pués no se recibió ningún mensaje de abandono CGMP. (Lo mismo ocurre con IGMP Snooping) de cualquier forma el *switch* LAN continuaría enviando el flujo de tráfico multicast al puerto donde está conectado el *host* B.

En este punto, el *host* B continua recibiendo tráfico no deseado, este problema no se puede resolver, ya que si existe un receptor activo sobre el *switch* LAN para los grupos de video, entonces el *host* B seguirá enlazado.

Problema de enlace entre switches

Dentro de redes con múltiples *switches* conectados, se consume innecesariamente ancho de banda sobre las interconexiones entre *switches* en el tráfico mutlicast enviado al enrutador. En la figura 4.19 se muestra una topología jerárquica común de *switch* LAN la cual consiste en un *switch* central LAN de alta velocidad y una fuente de tráfico multicast conectada a uno de los *switches* de las capas inferiores. El enrutador conectado al *switch* central provee de conexión al resto de la red y corre, al igual que los *switches*, CGMP. Asumiendo que la fuente multicast es un servidor de IP/TV y que reenvía 1.5 Mbps en un video en formato MPEG el cual en este momento no esta siendo recibido por ningún *host* en ninguna parte de la red.

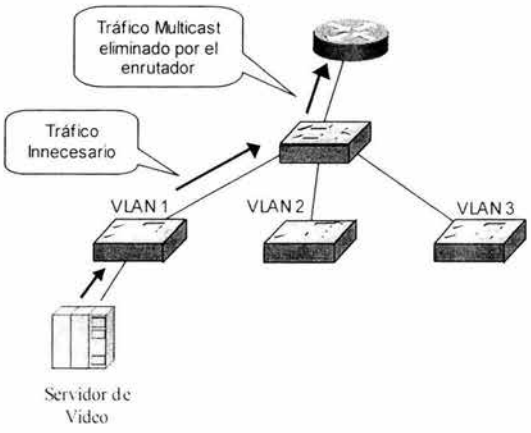


Figura 4.19 Problema de enlaces entre switches

Recordando que, los enrutadores multicast deben recibir indiscriminadamente todo los paquetes multicast de las estaciones en la red LAN local en el siguiente orden:

- Recibir y procesar los Reportes de Membresía IGMP
- Recibir todos los datos multicast para poderlos encaminar como es necesario a otra parte de la red.

Estos dos requerimientos se aplican a los 1.5 Mbps del flujo de video sobre las interconexiones de los *switches* en el enrutador que se encuentra conectado en el *switch* central, esto ocurre incluso si no existen miembros del grupo multicast en toda la red. En el caso de que el ancho de banda no se utilice, no existe ningún problema, pero en el caso de que este ancho de banda se utilice para darle servicios a una red de área metropolitana, existe un gran problema.

Una solución parcial de este problema se muestra en la figura 4.20. En esta figura la red se ha rediseñado para colocar la fuente multicast conectándola al *switch* principal, Bajo esta condición CGMP se encarga de que se creé la tabla CAM de entrada en el *switch* central, esto obligará al flujo de 1.5 Mbps ser distribuido por el enrutador al resto de la red donde se requiera. Desafortunadamente, si no existen receptores en ningún lugar en la red, los datos que fluyen al enrutador siguen siendo ancho de banda perdido. Lo cual es otro problema.

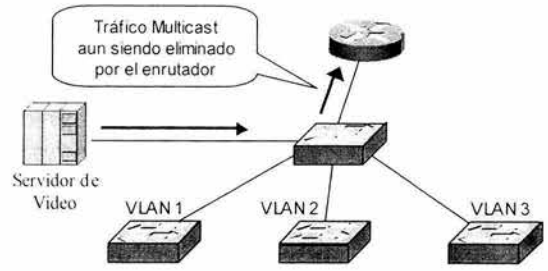


Figura 4.20 Solución parcial, enlace entre switches

4.2 Multicast sobre redes NBMA

Antes de empezar debemos tener bien claro que es una Red NBMA (Non-Broadcast Multi Acess). Un ejemplo de una Red NBMA es una Red Frame Relay o ATM que conecta varios enrutadores como se muestra en la figura 4.21.

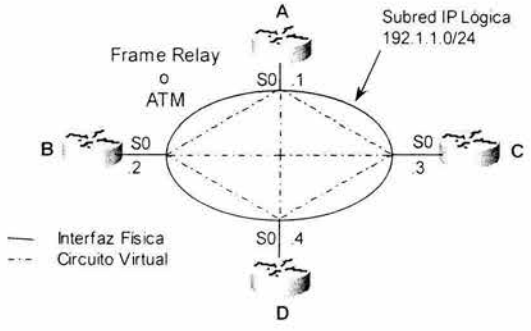


Figura 4.21 Red NBMA de malla completa

4.2.1 Redes NBMA en la capa 3

Cuando una red NBMA es configurada como una subred IP lógica, como en la figura anterior, el enrutamiento aparece de la misma manera que en una red Ethernet, como se muestra en la figura 4.22.

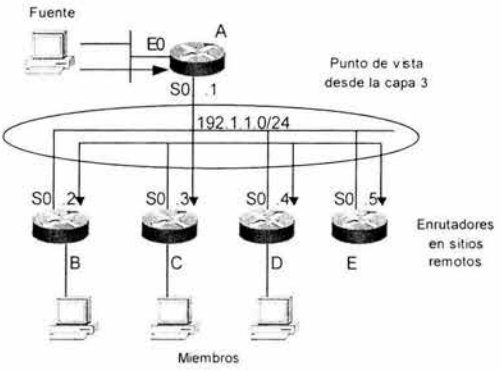


Figura 4.22 Punto de vista desde la capa 3 de un enrutador

Como se puede observar, desde el punto de vista del enrutador A en la capa 3 en esta red todos los vecinos PIM están directamente conectados a la red vía broadcast, sin embargo el enrutador A desea enviar una sola copia de un paquete multicast para ser recibido por todos.

4.2.2 Redes NBMA en la capa 2

La topología real de esta red NBMA en la capa 2 se muestra a continuación, figura 4.23.

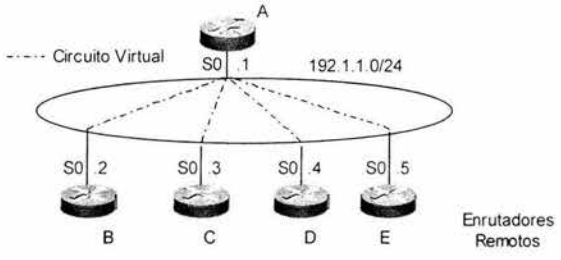


Figura 4.23 Realidad en capa 2

El resultado en la realidad bajo la capa 2 es la incapacidad del enrutador A para poder enviar una sola copia a todos los demás enrutadores, por lo cual el enrutador A debe utilizar una técnica llamada *pseudobroadcast* para enviar una copia de un paquete multicast sobre cada circuito virtual para cada enrutador dentro de la subred.

4.2.2.1 Pseudobroadcast

Para entender el concepto de pseudobroadcast, imagine siete localizaciones WAN que corran OSPF (Open Shortest Path First). Cuando un enrutador envía un paquete *hello* OSPF a una dirección de grupo multicast IP, la capa de enlace de datos del enrutador

repliega el paquete *hello*, enviando una copia a cada vecino WAN. En este ejemplo, seis copias del paquete hello se crean y son enviados sobre el enlace de multiacceso WAN.

En este caso Pseudobroadcast trata a los paquetes multicast como tráfico broadcast y lo replica a todos los vecinos (creando una copia por cada vecino) en la WAN sin importar si requieren este tráfico o no.

Obviamente la combinación del impacto en el desempeño del enrutador y el ineficiente uso de ancho de banda al replicar tráfico a otros puntos de la red donde no es necesario, hacen de Pseudobroadcast una muy mala opción en el diseño de redes multicast.

4.2.2.2 PIM modo NBMA

Las características de PIM permiten configurar al enrutador de tal forma que a diferencia del pseudobroadcast que envía a todos el tráfico, este solo envía el flujo de tráfico a los vecinos miembros del grupo.

De tal forma que se obtienen los siguientes beneficios

- El tráfico es rápidamente conmutado
- Los enrutadores solo reciben tráfico de los grupos a los cuales se han enlazado

Sin embargo existen problemas en algunos procedimientos como lo es el podado de los árboles en acoplamientos parciales en redes NBMA y PIM-SM

4.2.2.3 PIM y acoplamiento parcial de las redes NBMA

El acoplamiento parcial de las redes NBMA es cuando cada enrutador no tiene un circuito virtual a cada uno de los enrutadores en la red. Esto es un problema en la función apropiada de PIM, debido al hecho de que los paquetes multicast son enviados por un enrutador dentro una nube NBMA parcialmente acoplada y no puede alcanzar a todos los miembros de la red. Ciertos mecanismos de la operación de PIM dependen de esto y no funcionaran si este no es el caso. Uno de los mecanismos que fallan es *PIM-Dense Mode Pruning*.

Para poder solucionar estos problemas es necesario que el código de PIM esté enterado de la topología real de capa dos de la nube NBMA, para esto es necesario que el código PIM trate a la nube como una colección de circuitos punto-punto en vez de un medio de difusión, como se muestra en la figura 4.24.

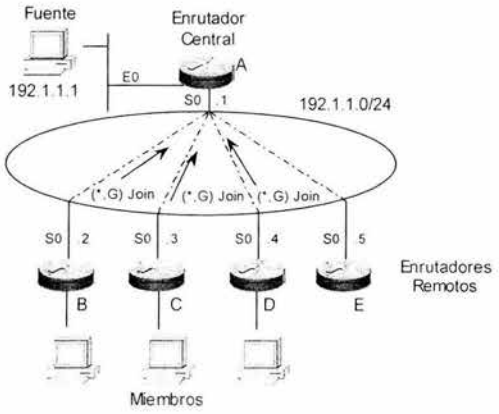


Figura 4.24 PIM en modo NBMA

Suponiendo que el enrutador B, C y D desean enlazarse al grupo, éstos envían reportes al enrutador A. De esta forma se tiene lo siguiente, figura 4.25.

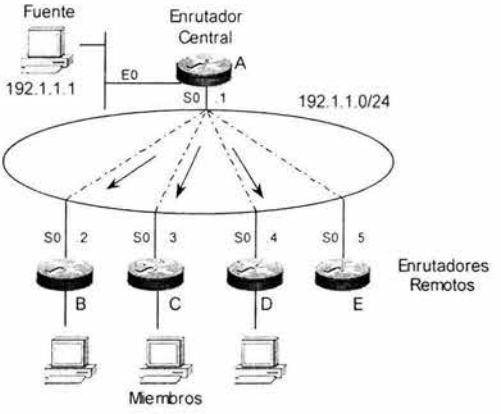


Figura 4.25 Detalles del modo NBMA en capa 2

De esta forma el tráfico solo es enviado a los enrutadores que lo solicitan. Solucionando el acoplamiento parcial que se presentaba anteriormente, y si un enrutador no tiene miembros es removido de la lista sin afectar a los demás enrutadores.

4.2.2.4 Auto-RP sobre redes NBMA

Cuando otro enrutador desea ser ahora el punto de reunión y se configura el *Auto-RP*, *Candidate-RP* o *Mapping Agent*, existe un problema ya que los mensajes de

anunciamiento no los escucharan los demás enrutadores, como se muestra en la figura 4.26

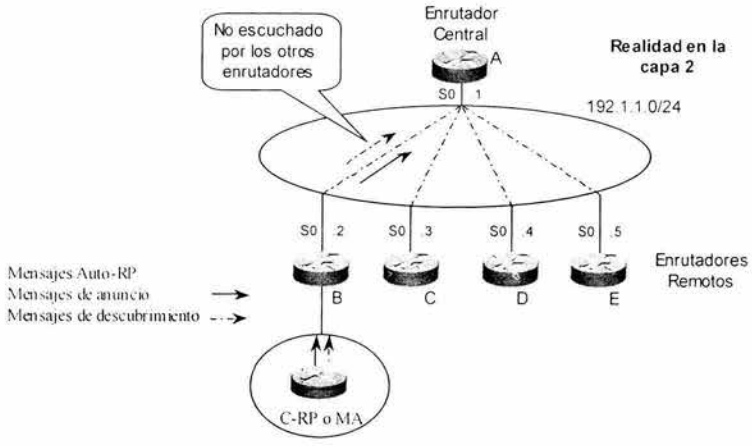


Figura 4.26 Problema de inundación de mensajes con PIM-DM Auto-RP

Problema de inundación de mensajes con PIM-DM Auto-RP

En una red de acoplamiento parcial el enrutador B no tiene circuitos virtuales a todos los demás enrutadores dentro de la nube NBMA, por lo cual los mensajes de Auto-RP del enrutador B no alcanzan a los otros enrutadores. Este problema es debido al hecho de que los mensajes de Auto-RP (en los grupos 224.0.1.39 y 224.0.1.40) normalmente fluyen en modo denso, a no ser que se les configure un RP estático a cada uno de los enrutadores. En esta situación ningún mensaje de informe de Auto-RP o mensaje de descubrimiento que llegue al enrutador A será reenviado a través de la interfaz serial 0.

La mejor forma para resolver este problema es habilitar **ip pim nbma-mode** y mover las funciones del *Mapping Agent* del sitio central de la red y dejar al candidato-RP donde está como se muestra en la figura 4.27

Ahora, los mensajes de descubrimiento de Auto-RP pueden alcanzar a todos los enrutadores dentro de la red y los mensajes de Candidato-RP pueden simultáneamente alcanzar los *Mapping Agent(s)*. Todos los *Mapping Agents* están dentro del sitio central de la red.

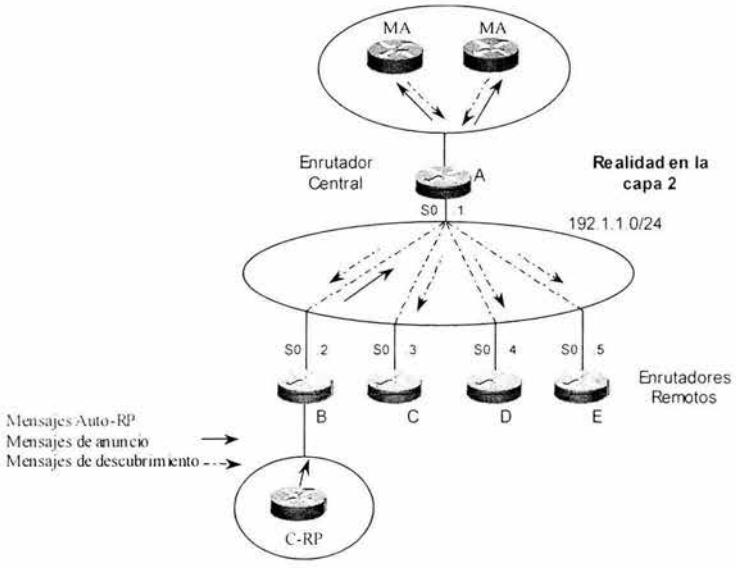


Figura 4.27 Colocando adecuadamente los Mapping Agents

En resumen la regla general, es que todo Auto-RP *Mapping Agents* se debe configurar dentro de la red del sitio central para prevenir problemas en los mecanismos de Auto-RP. Por otra parte, si no se desea diseñar de esta manera y se quiere tener un Auto-RP *Mapping Agent* en un lugar remoto de la red, será necesario agregar circuitos virtuales a la red NBMA, por ejemplo si se quiere tener un *Mapping Agent* dentro de la red, detrás de la enrutador B se requiere agregar circuitos virtuales del enrutador B a todos los demás enrutadores como se muestra en la figura 4.28.

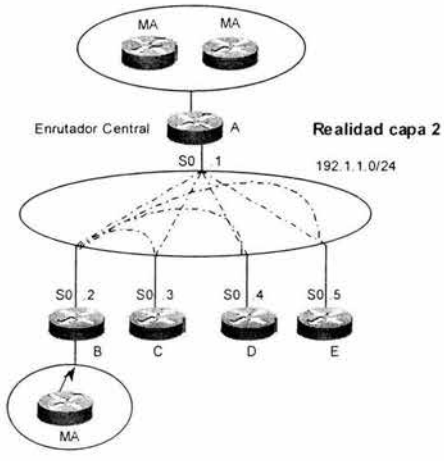


Figura 4.28 Agregando Circuitos Virtuales para resolver el problema

4.2.3 Multicast sobre la nube NBMA de ATM

Las redes ATM aceptan la implementación de Circuitos Virtuales Permanentes (PVC) y utilizan una infraestructura similar a la de Frame Relay. En este caso, el proveedor del servicio configura manualmente las rutas de los circuitos virtuales a través de ATM o Frame Relay. Sin embargo algunas redes ATM tienen la capacidad de soportar la creación de Circuitos Virtuales Conmutados (SVCs) por medio del estándar ATM (UNI) esta señalización permite a los enrutadores establecer nuevas rutas a través de ATM en la marcha. La señalización ATM (UNI) no solo permite la creación de circuitos virtuales punto-punto, si no que también puede establecer circuitos virtuales punto-multipunto en los cuales puede ser sustancialmente incrementada la eficiencia utilizando multicast en el flujo de tráfico sobre ATM.

4.2.3.1 Circuitos virtuales punto-multipunto en ATM

En la figura 4.29 muestra un ejemplo de *broadcast* en un enlace de circuitos virtuales punto-multipunto de un enrutador a todos los demás enrutadores en la nube ATM NBMA. El enrutador A utiliza este esquema para entregar por medio de *broadcast* el tráfico multicast a los enrutadores B, C y D, aunque solo se muestran los circuitos virtuales entre estos enrutadores, la realidad es que se crean 4 ya que cada enrutador crea un circuito virtual.

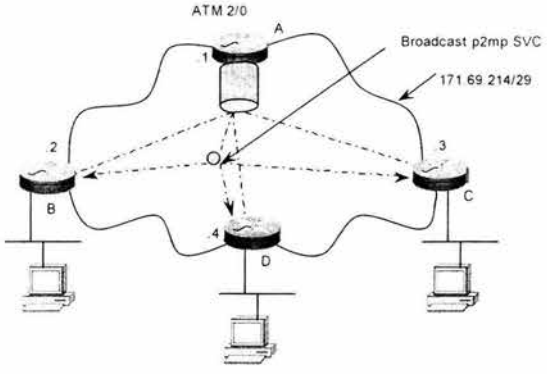


Figura 4.29 Circuito Virtual ATM Broadcast

4.2.3.2 Circuito virtual punto-multipunto por grupo

El uso del broadcast especial de p2mp VC ciertamente reduce la carga de trabajo en el enrutador por los paquetes de réplica, sin embargo el problema comienza cuando la red se fija de esta manera y se desea ahora eliminar el tráfico en algunos enrutadores.

Por ejemplo, asumiendo que sólo el enrutador A envía tráfico por medio de broadcast p2mp VC, aunque el enrutador A envía solo un paquete en la nube de ATM este llegará a los enrutadores C y D los cuales no desean este tráfico.

Para resolver este problema, existe la separación de los VC's p2mp de esta forma se distribuye el tráfico multicast solo a los enrutadores que lo requieren.

En la figura 4.30 se muestra la solución anterior.

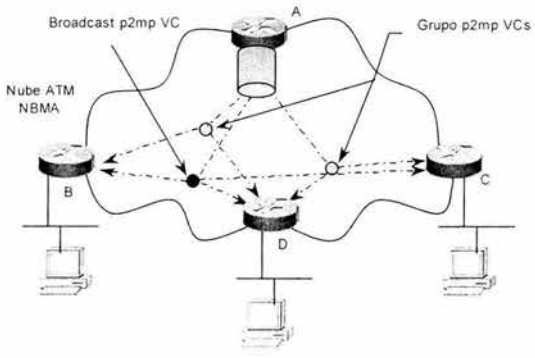


Figura 4.30 Circuito Virtual ATM P2MP por Grupo

En la figura se observa que el *p2mp VC broadcast* aún se usa para tráfico broadcast en general (Por ejemplo: "Hello" de OSPF). Sin embargo, ahora estamos usando un VC p2mp de grupo para cada subconjunto de enrutadores en la red que se hayan agregado a un grupo multicast en particular. Esto permite al enrutador A enviar un único paquete multicast al interior de la nube ATM y contar con que este paquete llegará sólo a aquellos enrutadores que tienen necesidad del tráfico. Con este esquema se obtiene el máximo rendimiento multicast que se puede obtener en una red ATM sin la necesidad de contar con *switches* ATM con conocimiento de multicast.

CAPÍTULO 5. Protocolos de enrutamiento multicast

5.1 Categorías de protocolos de enrutamiento multicast

Los protocolos de enrutamiento multicast se pueden dividir en dos categorías básicas

- Protocolos de Modo Denso (DVMRP y PIM-DM)
- Protocolos de Modo Esparcido (PIM-SM y CBT)

5.1.1 Protocolos de modo denso

Este protocolo emplea solo SPT para entregar tráfico multicast utilizando el principio de *push*. El principio de *push* asume que en todas las subredes dentro de la red tienen por lo menos un receptor del tráfico multicast (S,G), y por lo tanto el tráfico es empujado inundando todos los puntos de la red, este proceso es análogo al *broadcast* de la radio o de la televisión la cual es transmitido por todo el aire y llega a todas las casas . Para recibir la señal basta con sintonizar para recibir el programa.

5.1.2 Protocolos de modo Esparcido

Los protocolos de modo Esparcido hacen uso de los árboles compartidos y ocasionalmente, en algunos casos se utiliza SPT para distribuir tráfico multicast a receptores multicast dentro de la red. Este protocolo utiliza el modelo *pull* esto quiere decir que el tráfico multicast no será enviado a menos que se solicite utilizando un mecanismo explícito de enlace. Utilizando el ejemplo anterior de la televisión el modo esparcido es muy parecido a pago por evento.

5.2 Distance Vector Multicast Routing Protocol (DVMRP)

DVMRP es muy similar en muchos aspectos a RIP (*Routing Information Protocol*).

Algunas características claves de DVMRP son:

- Esta basado en vectores de distancia.
- Realiza actualizaciones de enrutamiento periódicas.
- Tiene un límite Infinito (32 saltos).
- El envenenamiento inverso tiene un especial significado.
- *Classless* (esto es, las actualizaciones de rutas incluyen máscaras).

5.2.1 Descubrimiento de vecinos DVMRP

El descubrimiento de vecinos lo realiza en cada una de sus interfaces, de tal forma que periódicamente reenvía mensajes a todos los enrutadores DVMRP como se muestra en la figura 5.1, utilizando la dirección de grupo 224.0.0.4

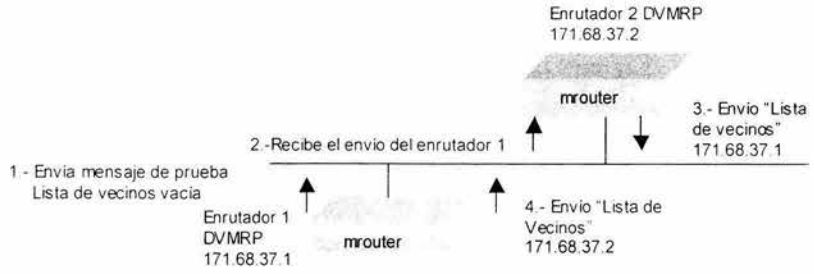


Figura 5.1 Descubrimiento de Vecinos

1. El enrutador 1 envía un primer paquete de prueba, tomando en cuenta que este enrutador no tienen ninguna otra información más que su propia dirección, es decir, la lista de vecinos en el mensaje de prueba está vacía.
2. El enrutador 2 escucha el mensaje enviado de prueba por el enrutador 1 y agrega las direcciones IP del enrutador 1 dentro de la lista interna de vecinos sobre esta interfaz.
3. El enrutador 2 envía un mensaje de prueba con las direcciones que tiene en su lista incluyendo la de él y las que ha recibido.
4. El enrutador 1 escucha el mensaje de prueba enviado por el enrutador 2 y agrega la dirección de éste a su lista.

5.2.2 Intercambio de reportes de ruta DVMRP

En la figura 5.2 se muestra una porción de una red multicast, dos enrutadores DVMRP conectados a una misma red. Asuma que el enrutador 2 envía primero los reportes.

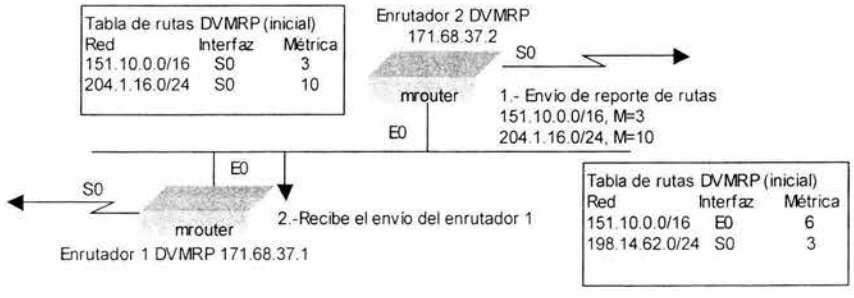


Figura 5.2 Intercambio de rutas paso 1 y 2

Este reporte contiene dos rutas anunciadas, las cuales son recibidas por el enrutador 1 (paso 2) quien las compara con las que tiene, si las encuentra, compara sus métricas, si la que tiene es mayor a la que recibió entonces la actualiza con el nuevo valor y modifica la interfaz asociada, si no, la ignora. En el caso que no la encuentre en su tabla, entonces la agrega, como es el caso de la red 204.1.16.0/24.

Ahora el enrutador 1 responde enviando su propio reporte de enrutamiento (paso 3), como se muestra en la figura 5.3, al enrutador 2, en este paso "envenena" el regreso de las dos rutas recibidas del enrutador 2, esto es, aumentando su métrica a infinito (32). Estos cambios dan a conocer al enrutador 2 que el enrutador 1 esta por debajo de él para estas redes. El enrutador actualiza su tabla de rutas DVMRP agregando la red 198.14.32.0/24 (paso 4), como se muestra en la figura 5.4

Ahora el enrutador 2 envía otro reporte al enrutador 1 y envenena la red 198.14.32.0/24 agregándole una métrica de 32, este cambio informa al enrutador 1 que el enrutador 2 está por debajo del enrutador 1 para esta red, esto se muestra en la figura 5.4.

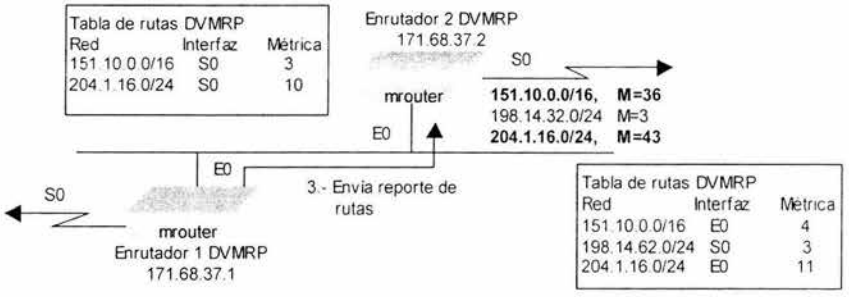


Figura 5.3 Intercambio de rutas paso 3

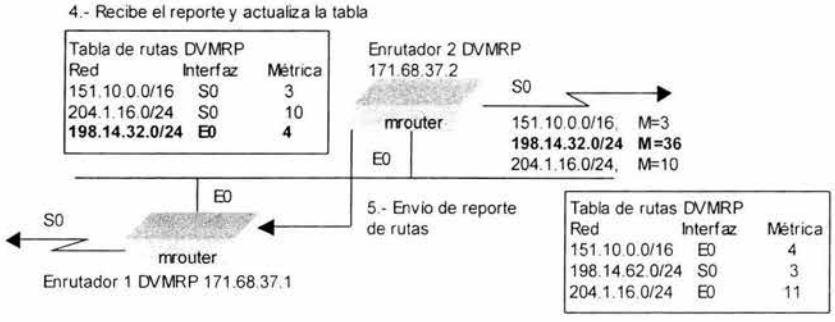


Figura 5.4 Intercambio de rutas, paso 4 y 5

5.2.3 Truncated Broadcast Tree

DVMRP es un protocolo de modo denso que utiliza la distribución de árboles fuente. Estos árboles son construidos por los enrutadores asignados por el *truncated broadcast*

tree. que utiliza las métricas de las rutas para formarse, lo cual permite básicamente que un enrutador por debajo de otro le pida al enrutador por encima de él, que lo coloque en el *truncated broadcast tree* para la fuente multicast deseada de la red.

A continuación en la figura 5.5 se presenta el reenvío de las tablas de enrutamiento y la designación de enrutadores abajo o arriba de otros para la formación de *truncated broadcast tree*

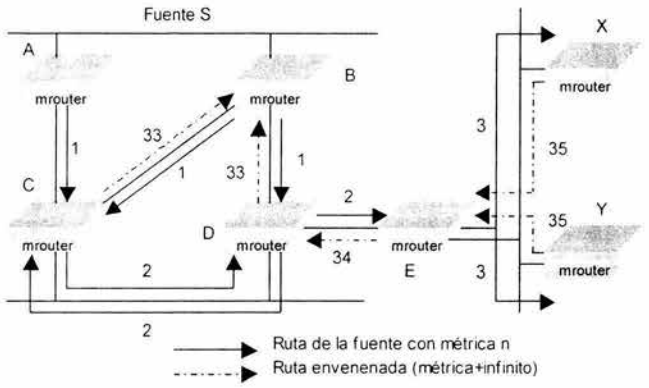


Figura 5.5 Truncated Broadcast Tree

En la figura 5.5 podemos observar como el enrutador D se encuentra por debajo del enrutador B con respecto a la red S por lo cual, envenena con un valor de infinito el anuncio de la ruta hacia la red S hacia el enrutador B.

En el caso del enrutador C, el cual se encuentra por debajo del enrutador A y B, habrá que designar por cual enrutador se recibirá el tráfico, como los dos tienen una métrica de 1, entonces se designa por medio de la dirección IP más pequeña.

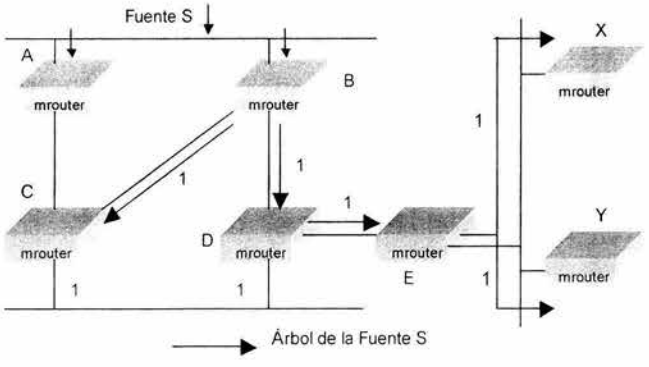


Figura 5.6 Resultado de Truncated Broadcast Tree para la Fuente S

En la figura 5.6 se describe la formación del árbol que será utilizada para enviar tráfico multicast de una fuente específica de la red a todos los enrutadores dentro de la red sin importar si hay miembros del grupo en la red. Cuando la fuente comienza a transmitir, los datos multicast fluyen hacia abajo de *truncated broadcast tree* a todos los puntos en la red. Los enrutadores DVMRP cortan el tráfico que no es necesario (este procedimiento se describirá más adelante).

Cada fuente de la red es asociada con un *truncated broadcast tree,* en la figura 5.7 se muestra como es construido el árbol para la fuente de la red S localizada en un lugar distinto al ejemplo anterior, cada árbol es distinto dependiendo donde se encuentre la fuente.

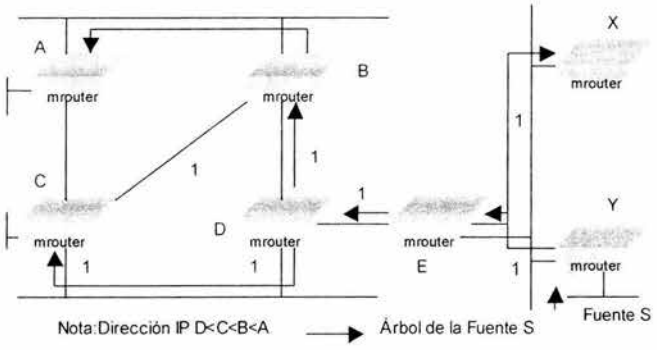


Figura 5.7 Truncated Broadcast Tree para la Fuente S

5.2.4 Reenvío de multicast DVMRP

Es importante determinar nuevamente la interfase por la cual se espera recibir un tráfico en especifico para evitar posibles *loops*, de tal forma que si un paquete llega por la interfaz incorrecta es descartado utilizando *Reverse Path Fowarding (RPF),* al proceso de chequeo de cada paquete se le llama *RPF-check.*

5.2.5 Abandono DVMRP

Como la mayoría de los protocolos de modo denso, DVMRP utiliza también mecanismos *Flood-and-Prune* para iniciar la entrega de tráfico multicast a los enrutadores, en este caso como ya hemos visto el tráfico fluye sin importar si existen miembros o no. Por lo que el trafico es delimitado por los enrutadores DVMRP, y esto lo logra enviando mensajes de abandono DVMRP al *truncated broadcast tree* para detener el flujo de tráfico multicast no necesario.

Desafortunadamente el abandono tiene un tiempo de expiración de 2 minutos, después el flujo volverá a inundar la red.

En las figuras siguientes se muestra el proceso de abandono:

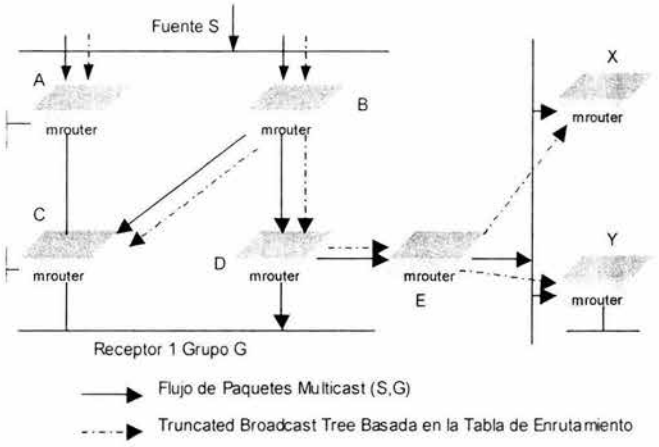


Figura 5.8 Condiciones Iniciales

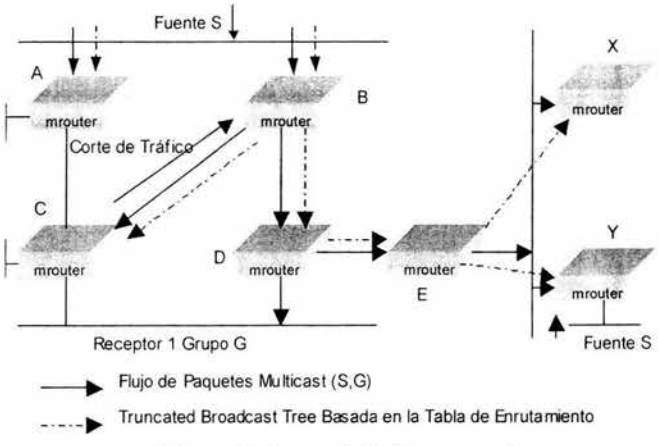


Figura 5.9 Cortes de Tráfico, paso 1

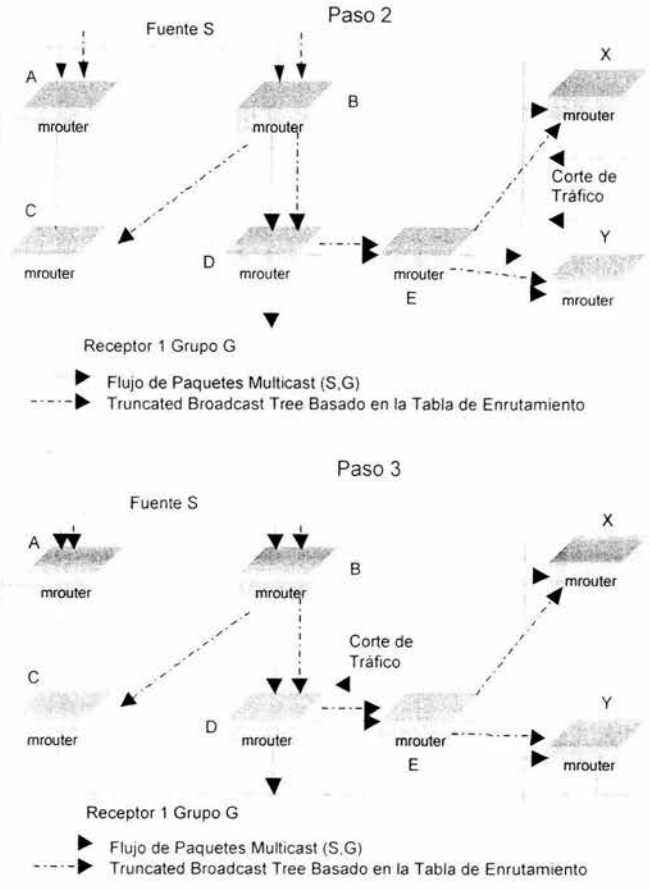


Figura 5.10 Cortes de tráfico, pasos 2 y 3

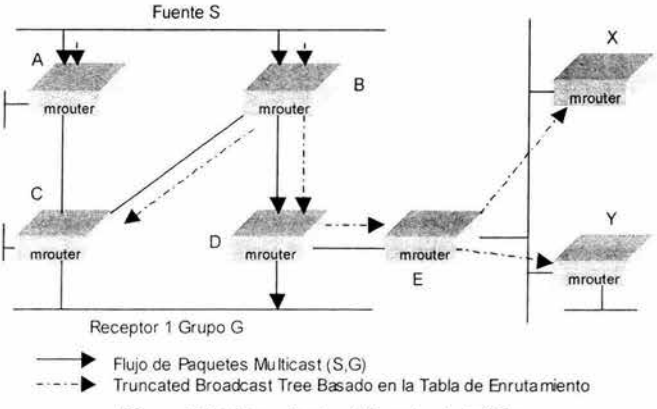


Figura 5.11 Resultado del corte de tráfico

5.2.6 Inserción DVMRP

DVMRP soporta un mecanismo muy confiable de inserción después de haber podado la rama donde se encuentra. Sin este mecanismo la latencia de enlace para nuevos *host* dentro del grupo pude ser afectada ya que el abandono de grupo en el enrutador superior tendría que esperar un tiempo antes de que el flujo comenzara de nuevo a fluir. Dependiendo del número de enrutadores a lo largo de la rama podada y los valores de espera en uso, puede tomar algunos minutos antes de que el *host* reciba tráfico multicast. Sin embargo utilizando el mecanismo de inserción, DVMRP reduce esta latencia de enlace a unos cuantos milisegundos.

A diferencia del mecanismo de abandono, la cual no es muy fiable, la inserción es muy segura utilizando mensajes *Graft-Ack*. Estos mensajes son devueltos por el enrutador superior en respuestas a los mensajes de inserción recibidos. En este paso se previene la perdida de mensajes a causa de la congestión, la cual es la principal causa de que los procesos de inserción fallen.

En la figura 5.12 se muestra la red inmediatamente después de que el receptor 2 en el enrutador Y desea enlazarse al grupo multicast, de tal forma que envía un mensaje de inserción al enrutador superior o que se encuentra arriba de él, en este caso el enrutador E.

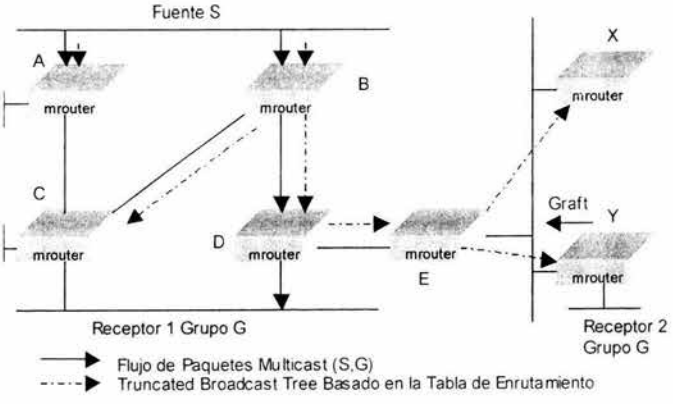


Figura 5.12 Condiciones iniciales de petición de tráfico

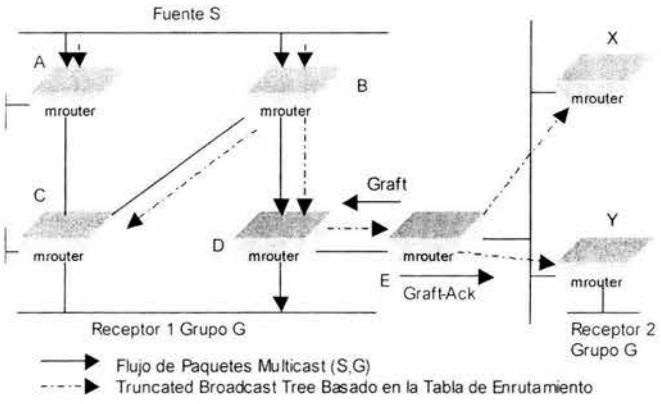


Figura 5.13 Petición de tráfico, paso 1

El enrutador E contesta a este mensaje enviando un mensaje *Graft-Ack*, como se muestra en la figura 5.13, de esta forma sucesiva se sigue hasta llegar al enrutador que este recibiendo tráfico multicast, como se muestra en la figura 5.14.

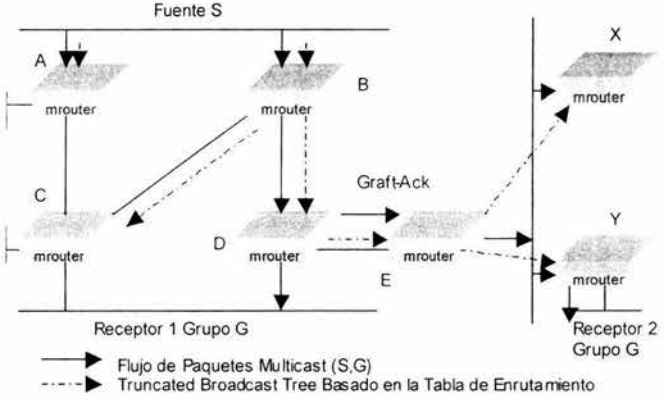


Figura 5.14 Resultados de la petición de tráfico

5.2.7 Resumen

En resumen podemos decir que DVRMP tiene muchas limitaciones derivado a que se basa en un algoritmo de enrutamiento tipo vector distancia. Si la ip multicast debe ser ubicada a través de Internet, es demasiado lento en cuanto a la convergencia y a los periodos de actualización.

5.3 PIM Dense Mode

El Protocolo Independiente Multicast utiliza la información de los protocolos de enrutamiento unicast, para el desempeño del reenvío multicast. Esto se logra al tomar en cuenta las tablas de enrutamiento unicast, incluso las rutas estáticas, para el control y manejo de *Reverse Path Forwarding* (RPF) de esta forma se mantiene una verificación en vez de hacer y actualizar por separado la tabla de enrutamiento multicast. PIM es incapaz de mantener por sí mismo una tabla de enrutamiento ya que no puede enviar ni recibir actualizaciones de tablas como otros protocolos.

PIM se puede configurar para que opere de dos formas: esparcido o denso, en este punto se estudiará el modo denso

Algunas características de PIM Modo Denso (PIM-DM):

- Es un protocolo Independiente (utiliza tablas de enrutamiento unicast para RPF check).
- No separa los protocolos de enrutamiento multicast, esto es, no envía ni recibe actualizaciones de rutas multicast como en el protocolo DVMRP, PIM-DM utiliza las tablas de enrutamiento unicast para entregar la información.
- Tiene un comportamiento de *Flood-and-Prune* como se explico en la parte de abandono DVMRP en ciclos de 3 minutos.
- *Classfull* o *Classless* (dependiendo del tipo de enrutamiento unicast que se utilice)

5.3.1 Selección del enrutador PIM-DM designado en redes de múltiple-acceso

Los mensajes de PIM Hello son utilizados para el establecimiento de la adyacencia entre los vecinos y también para el establecimiento del Enrutador Designado (DR) para la red-múltiple-acceso. Los enrutadores PIM hacen notar (por medio de mensajes PIM Hello) que el enrutador sobre la red con la dirección IP lo más grande posible se convertirá en el DR de la red.

Cuando uno o más enrutadores existen sobre un mismo segmento LAN, el ingeniero de la red tiene que designar un DR, sin embargo en algunas ocasiones es imposible renombrar las direcciones IP de los enrutadores de tal forma que este problema se resolvió adhiriéndole una nueva opción *DR-Priority* a los mensajes PIMv2 Hello , esta opción permite al ingeniero de red especificar una Prioridad DR a cada enrutador sobre el segmento LAN (por omisión la prioridad es 1), de tal forma que la elección se realiza de acuerdo a que enrutador tiene la prioridad más alta, si nuevamente todos los enrutadores tienen la misma prioridad se utilizará el concepto anterior: la dirección IP más alta.

5.3.2 PIM-DM árboles de distribución fuente

Ya que PIM-DM es un protocolo de modo denso, los árboles de distribución de fuentes serán construidos al mismo tiempo en que se inunda la red y se reciben los avisos de corte de tráfico de los receptores que no desean tráfico multicast.

PIM-DM utiliza la información de los vecinos para construir una árbol de distribución fuente, se basa en la tabla de enrutamiento unicast para poder determinar el envió hacia todos los vecinos PIM-DM. Esta inicial forma de SPT es referida como *Broadcast Tree* ya que el enrutador fuente envía el tráfico multicast a todos los vecinos.

5.3.3 Reenvío multicast PIM-DM

Cuando el enrutador recibe inicialmente un paquete multicast, el paquete experimenta un chequeo RPF para asegurar de que llegue por la interfaz correcta en la dirección de la fuente, usando para ello la información de la tabla de enrutamiento unicast. El enrutador PIM-DM busca en la tabla de enrutamiento unicast para comparar la dirección IP de la fuente dentro del paquete y utiliza esta información para determinar la interfaz entrante para el tráfico multicast de esa fuente. Si existen múltiples entradas para la fuente de la red en la tabla de enrutamiento unicast (esto ocurre cuando hay rutas de igual costo en la red). El enrutador escoge solo una interfaz, la que tenga la dirección IP más grande, la cual será checada por RPF y será tomada como la interfaz de entrada.

5.3.4 Corte de tráfico para el podado del árbol en PIM-DM

PIM-DM envía mensajes de corte de tráfico, bajo las siguientes condiciones

- EL tráfico llega por una interfaz punto a punto donde se no se cumple la regla RPF.
- Un enrutador en la punta del árbol sin receptores conectados directamente a él.
- Un enrutador intermedio en una rama del árbol que ha recibido un mensaje de corte de su vecino (en enlaces punto a punto).
- Cuando un enrutador intermedio en una rama del árbol y sobre un segmento LAN (sin receptores activos en dicho segmento) recibe un mensaje de corte de un vecino en el mismo segmento LAN y ningún otro vecino del segmento cancela el mensaje.

El siguiente ejemplo muestra más claramente el proceso de abandono.
Tome en cuenta que la métrica a la fuente es mejor por la vía del Enrutador A y C, como se muestra en la figura 5.15.

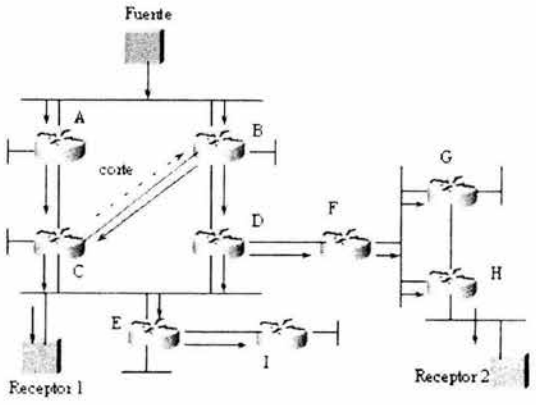


Figura 5.15 Corte de tráfico por una interfaz no RPF

Al recibir el mensaje, el enrutador B ya no envía tráfico al enrutador C, al mismo tiempo el enrutador I se da cuenta de que no tiene ningún receptor conectado y envía un mensaje de abandono al Enrutador E, figura 5.16.

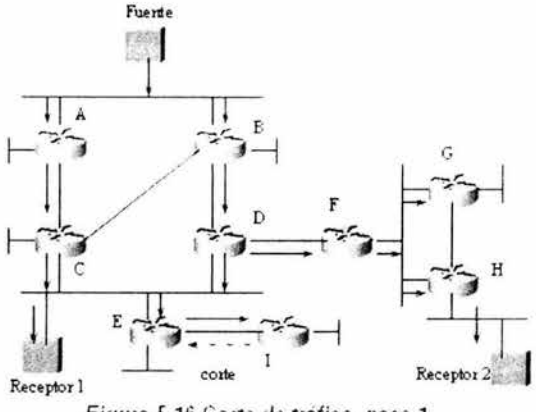


Figura 5.16 Corte de tráfico, paso 1

El enrutador E ahora responde, no enviando tráfico al enrutador I. Y ya que el enrutador E no tiene conectado directamente ningún receptor envía un mensaje de abandono a los enrutadores C y D, figura 5.17

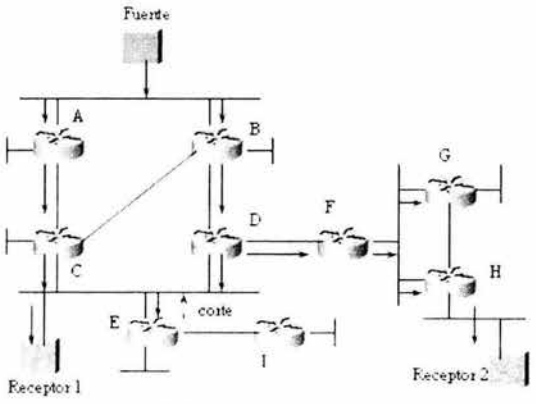


Figura 5.17 Corte de tráfico, paso 2

Sin embargo, el enrutador C y D ignoran la petición ya que tienen directamente conectado en la misma interfaz al Receptor 1, el problema es que cada 3 minutos enviara la petición de abandono consumiendo ancho de banda.

A esta característica se le conoce como invalidación de abandono, si después de 3 minutos no escucha a nadie, abandona el segmento y no envía tráfico.

Es importante mencionar que en algunas ocasiones la Latencia de Abandono es un punto a considerar ya que si el abandono se realiza en una rama muy grande existe un retraso considerable, como se ilustra en la figura 5.18

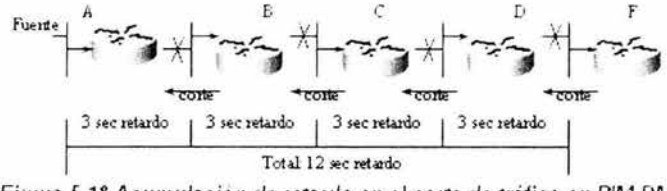


Figura 5.18 Acumulación de retardo en el corte de tráfico en PIM DM

En la figura 5.19 se puede ver que tanto el enrutador C como el D envían tráfico al mismo segmento duplicando la información para el receptor 1, para lo cual PIM utiliza un mecanismo llamado de "Aseguramiento" para elegir al enrutador que enviara el tráfico multicast.

Este mecanismo funciona de la siguiente manera; los enrutadores envían un mensaje de aseguramiento en el cual indican su métrica hacia la fuente, de esta forma los dos enrutadores comparan sus métricas con las de los vecinos, y la mejor será el enrutador que enviara el tráfico al segmento de red, todos los demás enrutadores colocaran en forma de corte de tráfico sus interfaces, si ambos enrutadores tienen la misma métrica, entonces se elegirá al que tenga la dirección IP más alta.

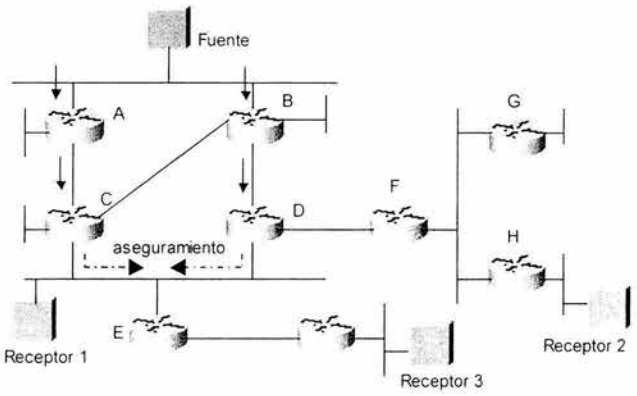


Figura 5.19 Aseguramiento

5.3.5 Petición de tráfico en PIM-DM

Para explicar de forma sencilla el proceso de inserción utilizaremos la siguiente figura 5.20, En esta se muestra como el enrutador envía un mensaje de petición de tráfico (*Graft*) al enrutador E como paso 1, el cual contesta con otro mensaje de respuesta a su petición (*Graft-Ack*) (paso 2), ahora el enrutador E envía un mensaje de petición de tráfico al enrutador C (paso 3), el cual a su vez contesta con el mensaje de respuesta a su petición (paso 4), de esta forma el flujo se restablece llegando al receptor 3.

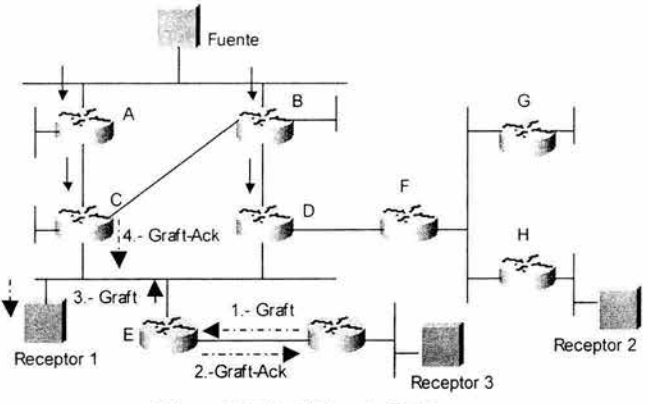


Figura 5.20 Petición de Tráfico

Como se puede observar existe nuevamente un tiempo de latencia.

5.3.6 Resumen

PIM-DM es potencialmente más escalable que DVMRP, ya que utiliza las tablas de enrutamiento unicast para el desempeño de RPF y a diferencia de DVMRP, no envía actualizaciones multicast separadas, Sin embargo tiene el mismo comportamiento básico *flood-and-prune*.

PIM-DM es utilizado en redes de alta velocidad, sin embargo en enlaces WAN no es recomendable. En redes donde existe un número grande de fuentes activas y grupos se tendrá un valor grande de latencia cuando se agreguen o se corten receptores, en cuanto a las aplicaciones en tiempo real sufrirán en gran medida de retardos.

5.4 PIM Sparse Mode

Al igual que el protocolo PIM de modo denso, utiliza las tablas de enrutamiento de los protocolos unicast para la aplicación de RPF y el mantenimiento de la tabla de enrutamiento multicast, por lo cual es un protocolo independiente.

Algunas caracteristicas de PIM *Sparse Mode* (PIM-SM)

- Es un protocolo independiente (utiliza las tablas de enrutamiento unicast para el desempeño de RPF).
- No separa los protocolos de enrutamiento multicast, esto es, no envía ni recibe actualizaciones de rutas multicast utiliza las tablas de enrutamiento unicast para entregar la información.
- Tiene un comportamiento explícito de petición de tráfico.
- Classfull o classless, (dependiendo del tipo de enrutamiento unicast que se utilice).

La más importante característica del modo esparcido es la capacidad de entregar el tráfico multicast únicamente al receptor que lo solicita y esto se logra enviando el tráfico de la fuente a la raíz del árbol compartido, la cual es un enrutador llamado punto de reunión (RP) y después al receptor. Una vez establecida la comunicación, si la trayectoria de la fuente al receptor es más corta, el algoritmo deja de utilizar el punto de reunión y envía el tráfico directamente de la fuente al receptor.

5.4.1 Registro de las fuentes

Para que las fuentes puedan enviar tráfico al punto de reunión es necesario que primero las fuentes se registren en él, el objetivo de registrarse es notificar al RP que la fuente S1 esta activa y preparada para enviar tráfico al grupo G, esto es entregar el paquete o paquetes al RP y después este entregará el tráfico hacia abajo del árbol.

El proceso comienza cuando la fuente envía tráfico al enrutador llamado DR, este crea un estado de entrada en su tabla de enrutamiento multicast (S,G) y encapsula la información enviándola en un *mensaje de registro PIM* por medio de unicast al RP, el cual tiene como objetivo notificar al RP que la fuente S esta enviando tráfico al grupo G, y le entrega el primer paquete multicast inicial enviado por la fuente S para que el RP lo entregue a los receptores.

Al recibir el mensaje de registro el RP verifica que el grupo G este activo, esto es, que haya recibido una petición de enlace al grupo G de un receptor.

Si esto ocurre el RP se enlaza por medio de SPT al la fuente S para poder recibir el tráfico multicast de una forma nativa en lugar de encapsular la información en mensajes de registro PIM.

En este momento el RP esta en condiciones de enviar un *mensaje de paro de registro PIM*, por otra parte si el grupo no esta activo enviara igualmente el mismo mensaje de paro de registro.

En la figura 5.22 se muestra este procedimiento.

5.4.2 Enlace al árbol compartido

Ahora bien para poder enviar el tráfico multicast al árbol compartido por medio del RP es necesario que previamente un receptor exprese su intención de recibir dicho tráfico, a continuación se presentan los pasos que sigue un receptor para lograr esto.

1. El receptor envía un Reporte de membresía IGMP al enrutador directamente conectado para poder recibir tráfico del grupo G por ejemplo.
2. El enrutador C como se muestra en la figura 5.21, crea la entrada (*,G) en su tabla de enrutamiento multicast, colocando la interfaz Ethernet en su lista de interfaces de salida para dicho grupo.
3. El enrutador C ahora trata de enlazarse por medio de mensajes PIM de enlace al RP utilizando su tabla de enrutamiento unicast para determinar por que interfaz lo puede alcanzar.
4. El RP en repuesta al mensaje crea un estado de entrada (*,G) en su tabla de enrutamiento multicast y agrega de igual forma la interfaz por la cual fue alcanzado a su lista de interfaces de salida.

Hasta este punto el árbol compartido para el grupo multicast G ha sido construido para el RP, el enrutador C y el receptor 1

Si por ejemplo otro receptor desea recibir tráfico del grupo G y se encuentra en el enrutador E, se repetirán los pasos 1 y dos tomando en cuenta que ahora el enrutador C será el E, ahora bien el enrutador E envía un mensaje PIM de enlace al enrutador C, se da cuenta que existe el estado de entrada (*, G) y solamente agrega a su lista de interfaces de salida, la interfaz por la cual fue alcanzado por el enrutador E. Hasta aquí llega el proceso, como se muestra en la figura 5.21

En la figura 5.22 se muestra el registro de la fuente

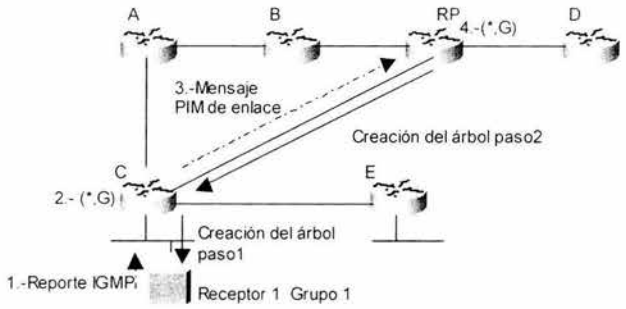


Figura 5.21 Enlace del receptor a un grupo

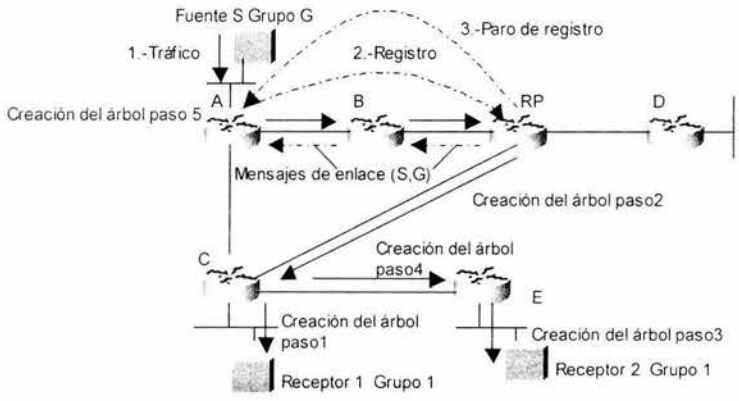


Figura 5.22 Registro de la fuente

5.4.3 Shortest Path Trees en PIM-SM

La gran ventaja de PIM-SM es la utilización de la entrega de tráfico solo a quien lo solicite, sin embargo al utilizar el concepto de RP la trayectoria entre este y los receptores tal ves no es la óptima, peor aún el receptor puede estar muy cerca de la fuente y muy lejos del RP, teniendo como resultado el aumento de congestión y latencia.

Sin embargo PIM-SM soluciona este problema y reduce la congestión que pueda tener el enrutador RP al implementar *Shortest Path Trees*, la única desventaja que presenta es la creación y el mantenimiento de los estados de entrada (S,G) en todos los enrutadores a lo largo de la ruta más corta, consumiendo los recursos en todos estos enrutadores, además de que esta condicionado a un limite de ancho de banda.

En esta parte se explica como se construye un árbol utilizando SPT

En la figura 5.23 se muestra como se realizan las peticiones de enlace a la fuente S utilizando SPT la cual utiliza las tablas de enrutamiento unicast para poder determinar la interfaz por la cual se puede alcanzar la fuente, a su vez en cada enrutador se actualizan las tablas de entrada y salida multicast, y se realiza la petición de corte hacia el RP si este esta fuera de la trayectoria del SPT.

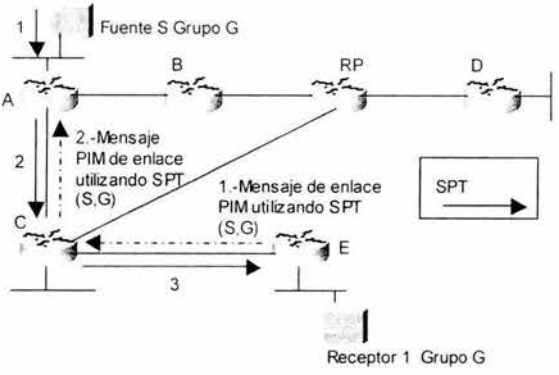


Figura 5.23 Shortest Path Tree en PIM-SM

5.4.4 Intercambio a la ruta más corta con SPT

En el subcapítulo anterior definimos de que manera SPT logra enlazarse directamente a la fuente considerando que la utilización de SPT se aplica tomando en cuenta un umbral de ancho de banda de cero sobre el árbol compartido, en este punto trataremos como se realiza el cambio de ruta a una más eficiente tomando en cuenta un umbral* referido al ancho de banda distinto de cero. Si este ancho de banda es excedido, el enrutador que es el último salto hacia el receptor utilizará SPT para llevar el tráfico a los receptores en lugar del árbol compartido.

En la figura 5.24 se muestra como el enrutador C realiza este cambio.

* En los enrutadores Cisco este umbral tiene un valor de cero por omisión

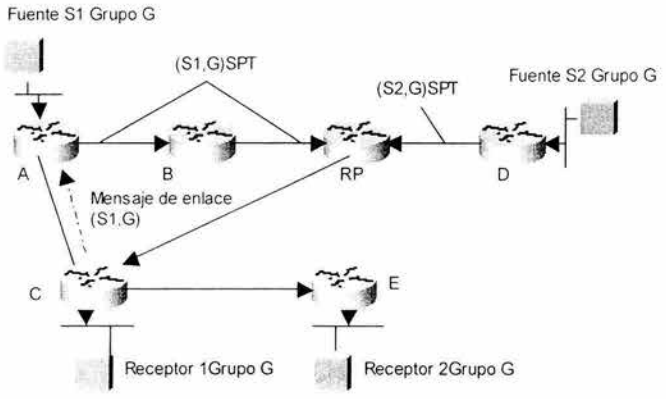


Figura 5.24 Intercambio a la ruta más corta

En la figura 5.24 se puede observar como el enrutador C obtiene el tráfico del RP sin embargo esta ruta no es la más óptima. Para que el enrutador C cambie la forma de obtener el tráfico multicast es indispensable que el umbral establecido sea superado, de esta forma empezará a enviar mensajes de enlace directamente al enrutador DR, en este caso el enrutador A, como se muestra en la figura.

Inmediatamente después el enrutador DR responde enviando tráfico al enrutador C de tal forma que ahora se tienen dos flujos hacia el enrutador C provocando congestión. Ahora es necesario decirle al RP que cese de enviar tráfico de la fuente S1, para esto el enrutador C utiliza un mensaje especial de corte de tráfico (S1,G)-*bit prune* que tiene la posibilidad de viajar hacia arriba del árbol al RP, figura 5.25.

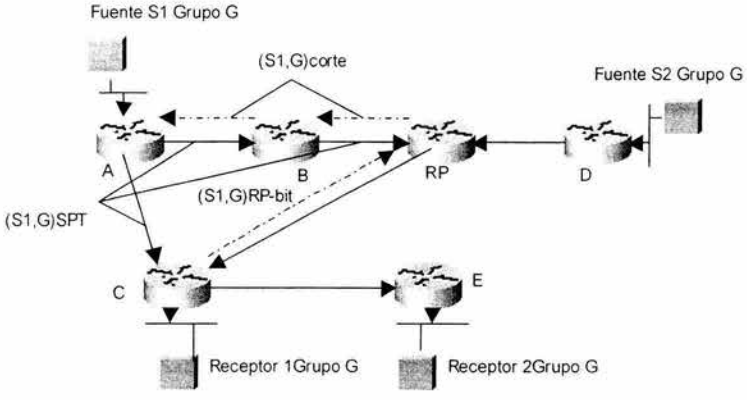


Figura 5.25 Mensajes PIM RP-bit

Al enviar el mensaje (S1,G)-bit al RP, éste de forma automática envía un mensaje de corte de tráfico al enrutador DR, el cual cesa el tráfico hacia el RP quedando de la siguiente forma.

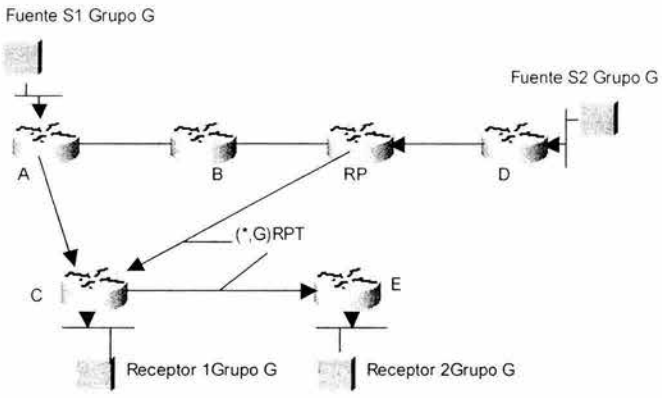


Figura 5.26 Resultado de la aplicación de un valor de umbral

El enrutador E seguirá recibiendo tráfico tanto de la fuente S1 como de la fuente S2 si así los solicita, figura 5.26.

Ahora bien hasta este momento nos hemos estado refiriendo a dos tipos de mensajes: uno para petición de enlace y otro para cortar el tráfico, en realidad se trata de un solo mensaje PIM corte /enlace a diferencia de PIM-DM que utiliza un mensaje corte y otro para enlace, PIM-SM envía uno solo para varios grupos y para sus respectivas fuentes como se muestra en el formato del mensaje, figura 5.27.

PIMver	Tipo	Reservado	Checksum
Codificado-Unicast-Dirección del vecino de arriba			
Reservado	Núm de grupos	Tiempo de retención	
Codificado-Dirección de Grupo Multicast-1			
Número de Fuente enlazada		Número de Fuente cortada	
Codificado –Dirección de fuentes a enlazar-1			

Codificado –Dirección de fuentes a enlazar-n			
Codificado-Dirección de la fuente a cortar -1			

Codificado-Dirección de la fuente a cortar-n			

Codificado-Dirección de Grupo Multicast-n			
Número de fuente enlazadas		Número de fuentes cortadas	

Figura 5.27 Formato del mensaje PIM corte/enlace

En los campos reservados puede tener las siguientes opciones:

- S Representado por 1 bit el cual indica que se utilizará la compatibilidad con los mensajes PIM v1.

- W El bit WC, toma el valor de 1 se aplica a las entradas (*,G) o (*,*,RP) si es corte o enlace, y 0 cuando se aplica a la entrada (S,G), donde S es la dirección de la fuente y se envía al RP ya sea de corte o de enlace.

- R El bit RPT, si el valor es 1, la información a cerca de la entrada (S,G) es enviada al RP, si es 0 la información debe enviarse a la fuente directamente.

En cuanto a la actualización de los estados se tiene como tiempo limite 3 minutos esto es, por ejemplo, si un enrutador pierde contacto con otro enrutador debajo de él por congestión por ejemplo, este mantendrá los estados (*,G) y (S,G) por tres minutos después serán borrados si no se establece la comunicación , por lo cual los enrutadores deben refrescar sus estados de forma periódica, esto se logra enviando mensajes PIM corte /enlace para actualizar los estados de reenvió multicast.

5.4.5 Enrutador designado

¿Qué pasa cuando existen dos enrutadores o mas conectados a una mismo segmento?, en esta parte se tratara la designación de uno de ellos como el enrutador DR. Antes de explicar como PIM-SM soluciona el problema, es necesario entender de manera clara el propósito que tiene el enrutador DR. Este tiene como principal tarea enviar al RP mensajes de registro de la fuente, en el caso de que el *host* directamente conectado a él fuera una fuente, y tiene la tarea de enviar mensajes de enlace al RP si el *host* es un receptor. Existe la posibilidad de que el *host* sea una fuente y un receptor al mismo tiempo, en este caso el DR tendrá que realizar las dos tareas antes mencionadas.

Al tener dos enrutadores en una misma LAN se designa como DR al que tenga la dirección IP más alta, como se muestra en la figura 5.28:

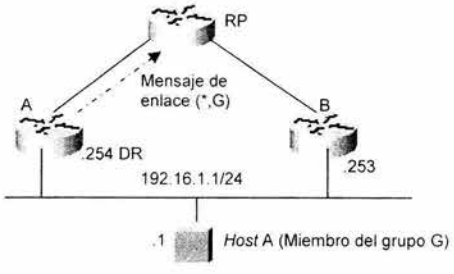


Figura 5.28 Designación de un DR

No solo eso, si el enrutador A llegara a fallar el enrutador B tomaría su lugar al no percibir los mensajes de adyacencia de su vecino.

5.4.6 Descubrimiento del RP

Para que PIM-SM trabaje apropiadamente, es necesario que todos los enrutadores en un dominio conozcan la dirección del RP. Para una red pequeña donde solo se utiliza un RP para todos los grupos multicast, es fácil configurar de manera manual esta dirección en todos los demás enrutadores, sin embargo es difícil en una red grande donde el RP debe cambiar de forma constante. Para solucionar este problema se utilizan distintos RP dentro del mismo dominio para optimizar el árbol compartido.

PIMv2 define un mecanismo "*Bootstrap*" que permite a todos los enrutadores dentro del dominio conocer de forma dinámica todos los grupos en un RP. Cisco por otra parte ha desarrollado un mecanismo "Auto-RP" el cual realiza la misma función que la realizada por PIMv2.

5.4.7 Resumen

PIM-SM tiene una gran ventaja con respecto a los demás protocolos antes vistos implementando un modelo de petición explícita de tráfico, así como la utilización de SPT las cuales reduce de manera considerable la latencia que se presenta en los árboles compartidos. De tal forma que PIM-SM es la mejor opción para un red multicast de propósito general de intra-dominio, sin embargo, cuando la aplicación que corre en la red es muy específica y requiere un completo control por parte de los administradores de la red tal vez no sea la mejor opción.

5.5 Core-Based-Trees

Este protocolo de enrutamiento multicast se originó con la versión 1 CBTv1 sin embargo ha evolucionado llegando a la versión 2 las cuales no son compatibles, de cualquier forma CBTv1 nunca se implemento de forma total, al igual que la primera versión, la versión dos, hasta el momento no se ha desplegado de una forma completa, lo cual tal vez no sea tan malo ya que se vislumbra las especificaciones de la tercera versión la cual incluirá la compatibilidad de la versión dos.

En esta sección se discutirá los conceptos y mecanismos utilizados por la segunda versión del protocolo CBT.

CBT es un protocolo de modo esparcido que tiene como meta principal la reducción de los grupos activos en los enrutadores dentro de la red del orden G, para completar esta meta sólo utiliza la forma bidireccional de los árboles compartidos para llevar el tráfico multicast a una porción de la red con un grupo específico, este árbol es arraigado en un enrutador núcleo, de ahí su nombre, permitiendo que el tráfico fluya en ambas direcciones: arriba y abajo del árbol. Esto permite que no se tomen medidas especiales para llevar el tráfico de la fuente al núcleo. En lugar de que el primer salto sea al núcleo, este simplemente lo envía hacia arriba del árbol, cada enrutador envía el tráfico por todas sus interfaces.

En la figura 5.29 se muestra el flujo en un CBT donde el M1 al M7 son miembros enlazados al árbol, el miembro 3 es también una fuente de tráfico multicast y se le llama *miembro fuente*:

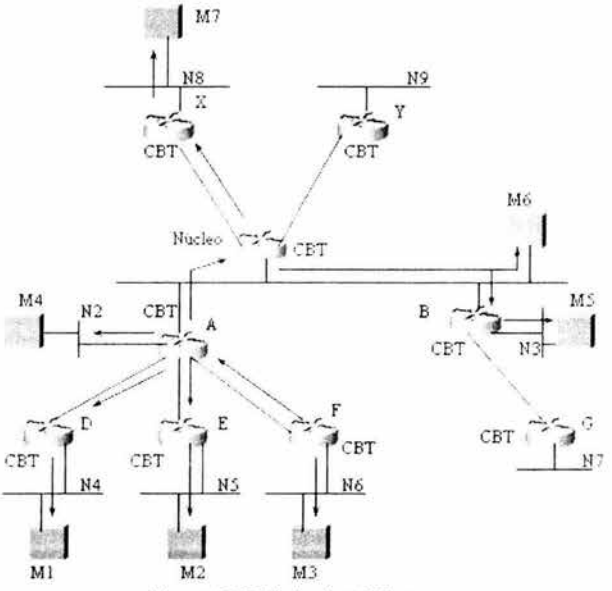


Figura 5.29 Flujo de tráfico

La importancia radica en que no se necesitan estados de reenvió en los enrutadores para el miembro fuente o miembros fuentes.

Desafortunadamente las fuentes que no son miembros, es decir, aquellos que sólo envían tráfico multicast pero que no son receptores y por lo tanto no se encuentran sobre el árbol compartido, deben enviar el tráfico vía túnel IP hacia el núcleo para que su tráfico pueda fluir hacia abajo a los receptores.

Finalmente, ya que CBT tiene noción del árbol compartido solo en estados minimizados en los enrutadores, éste no soporta SPT's. Como resultado CBT no es capaz de acortar rutas por medio de SPT's, sin embargo como ya se mencionó sólo se mantendrá un estado entrante (*,G) dentro de las tablas de enrutamiento multicast, como consecuencia es probable que se incremente la latencia por la elección de una ruta no óptima como en la figura 5.30.

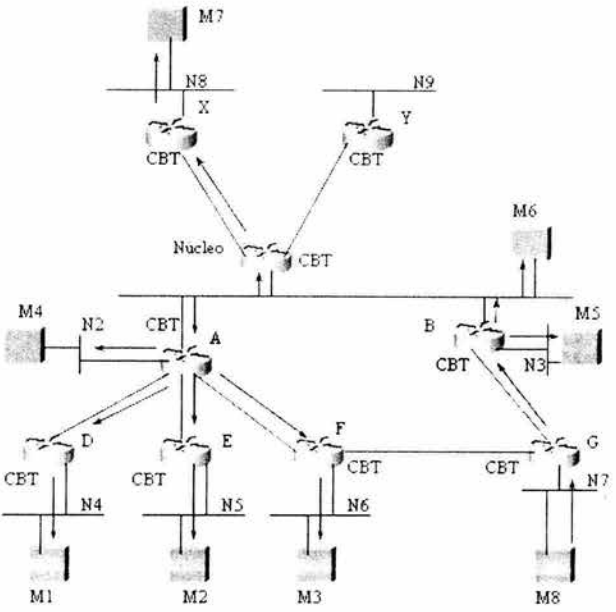


Figura 5.30 Flujo de tráfico no óptimo

En este ejemplo el miembro 8 causa que el enrutador G se enlace a través del enrutador B, adicionalmente, M8 es una fuente sobre la red N7 la cual comienza a reenviar el tráfico hacia el núcleo. El M3 requiere dicho tráfico por lo cual lo conseguirá siguiendo la ruta G-B-A-F a pesar de que el enrutador F y el G estén directamente conectados, esto es el resultado de que CBT no utiliza SPT.

5.5.1 Enlace con el árbol compartido

Cuando un enrutador recibe un reporte IGMP de un *host* conectado directamente, el enrutador inicia el proceso de enlace enviando un mensaje *CBT Join-Request* al siguiente *host* hacia el núcleo, utilizando las tablas unicast de tal forma que CBT es considerado un protocolo independiente.

Al llegar al núcleo envía de regreso un mensaje *Join*-Ack en respuesta confirmando el enlace, solo cuando este mensaje llega exitosamente a la rama del árbol donde se encuentra el receptor se permite el flujo del tráfico.

El enrutador CBT donde se originó el mensaje *Join-Request* es el responsable de la retransmisión del mismo si no recibe el mensaje *Join-Ack* en el intervalo de retransmisión que es de 5 segundos. En el caso de que esta retransmisión falle el otro enrutador enviará el mensaje *Join-Ack* con un intervalo de 7.2 segundos el cual es conocido como *Join-Timeout* y el proceso es abortado con la llegada del siguiente mensaje IGMP. Este

intercambio de mensajes se realiza por las mismas interfaces de tal forma que CBT no utiliza RPF.

5.5.2 Reenvío de los no miembros

CBT utiliza túneles IP para poder enviar el tráfico al núcleo de fuentes no miembros. En la figura 5.31 se muestra como el no-miembro (S1) envía al grupo, ya que el enrutador D no esta sobre el árbol, por lo cual se debe encapsular el tráfico y enviarlo al núcleo por medio de un túnel IP dentro de IP, este lo desencapsula y lo reenvía a todos los receptores debajo de la red:

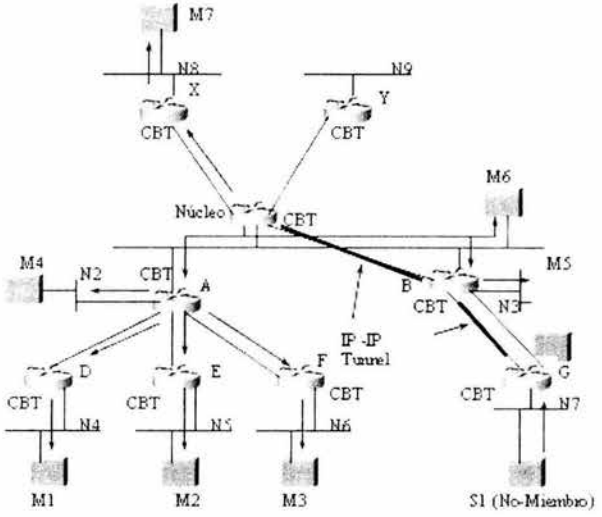


Figura 5.31 Envío a los no miembros

5.5.3 Mantenimiento del estado en CBT

Ya que CBT utiliza un modelo explícito de enlace, las ramas del árbol por medio de CBT necesitan periódicamente ser refrescadas y esta acción se basa en el envió de mensajes *keepalive* sobre los enlaces hacia el enrutador que está arriba (padre) y escuchando su correspondiente respuesta, este proceso permite al enrutador (hijo) monitorear la alcanzabilidad con el padre y toma acciones para reparar la conectividad perdida si es el caso. Para realizar esta tarea se basa en el envío de mensajes *echo-request* y *echo-response*.

5.5.3.1 Echo-request

Estos mensajes se reenvían por medio de los enrutadores CBT, con un periodo de 60 segundos al cuál se le llama ECHO-INTERVAL a los enrutadores (padres), este mensaje no contiene información de los grupos, los padres responden por medio de los echo-response en los cuales se agregan todos los grupos activos sobre la red.

5.5.3.2 Echo-reponse

Cuando los enrutadores reciben los mensajes *echo-request* sobre una de las interfaces de los "hijos", este contesta enviando un mensaje echo-response que contiene, como mencionamos, la lista de todos los grupos multicast conocidos por cada interfaz. Cuando esta respuesta es recibida, cada enrutador "hijo" busca esta información y actualiza la información con las entradas de reenvío, las entradas que no son actualizadas tienen un tiempo de vida y después del cual serán borradas si no son actualizadas, a este tiempo se le llama GROUP-EXPIRE-TIME y es aproximadamente de 90 segundos, al borrarlas el enrutador envía un mensaje *Quit-Notification* para el grupo debajo de él y un mensaje *Flush-Tree* para el grupo por arriba de él para que se realice la baja de esa rama del árbol.

5.5.4 Podado del árbol compartido

El abandono o podado del árbol es básicamente igual que el de PIM-SM, esto es, cuando un miembro directamente conectado al grupo lo abandona es necesario detener el tráfico que fluye por esa rama. Una de las formas de detenerlo es el envío de mensajes *echo-request* para actualizar los estados de los enrutadores "padres", este paso toma un periodo de tiempo finito en el cual el flujo de multicast sigue inundando la rama, por lo tanto el enrutador envía una forma de mensaje de podado hacia arriba del árbol para inmediatamente detener el tráfico, el podado dentro de los CBT es implementado utilizando los mensajes de *Quit-Notifiaction*.

Cuando estos mensajes se reciben simplemente se borra la interfaz de entrada para este grupo, sin embargo para ser borradas tiene que expirar un tiempo de aproximadamente 3 segundos, si dentro de ese periodo se recibe un mensaje *Join-Request* el tiempo es cancelado y la interfaz no es borrada.

5.5.5 Designación del enrutador en CBT

Para elegir un DR los enrutadores CBT periódicamente reenvían mensajes *Hello* (cada 60 segundos aproximadamente) a todos los enrutadores CBT del grupo con un TTL de 1. Estos mensajes contienen un valor de prioridad entre el 1 y el 255, (por default es 255) donde el 1 es considerado como el más elegible para ser el enrutador CBT DR, y este

valor puede ser configurado por el administrador de la red. Si dos o más enrutadores recibieran el mensaje con el valor de 1, se elegirá al enrutador con la IP más baja.

Una vez que se halla elegido un enrutador DR él advertirá a los demás que ha sido elegido incluyendo en los mensajes *Hello* el valor cero en el campo de preferencia por lo que existirá una supresión de mensajes *Hello*.

5.5.6 DR mediador de enlace

Cuando un CBT DR recibe un mensaje *join-request* de un enrutador por debajo de él, tal vez la mejor ruta para alcanzar al núcleo sea a través de uno u otro enrutador sobre la misma subred, ya que no necesariamente todos los enrutadores comparten el mismo punto de vista de enrutamiento. El CBT DR tomará la decisión de que ruta tomar (decisión de *join-routing*) lo cual evita que se formen *loops* en el árbol.

5.5.7 Descubrimiento del Enrutador Núcleo

Cuando se tiene una red pequeña, un enrutador núcleo simple se puede determinar en la configuración de cada enrutador. Sin embargo en una red grande en la cual el núcleo cambie frecuentemente ya no es tan sencillo, este problema puede ser resuelto por el hecho de que diferentes grupos multicast utilicen diferentes núcleos en otra locación dentro del dominio para optimizar el árbol. CBT utiliza el mismo mecanismo de *bootstrap* utilizado en PIMv2 que permite que todos los enrutadores CBT dentro de un dominio pueda leer dinámicamente todos los mapeos *<core, group>* y evite la configuración manual del núcleo.

5.5.8 CBT Versión 3

CBT v3 le concierne principalmente extensiones que permiten mejorar la interconexión con otros dominios multicast utilizando enrutadores de frontera (BRs), comprado con la versión anterior, esta requiere de mucha más información de los estados dentro de los enrutadores que trabajaran en la frontera para que puedan ser implementados eficientemente.

Por ejemplo el BR tiene que implementar nuevos estados (*,Core) y (S,G) dentro del orden para soportar podados de tráfico multicast que fluyen dentro, fuera y a través de este enrutador, por tal motivo el ,resultado de estos substanciales cambios no permite la compatibilidad de algunos paquetes con la versión 2.

5.5.9 Conveniencias y escalabilidad de CBT

Comparado con los protocolos que soportan SPT's, CBT's tiene grandes ventajas, es más eficiente en términos de cantidad de estados multicast creados dentro de los

enrutadores. Dentro de las redes con gran número de fuentes y grupos, el protocolo de enrutamiento CBT tiene la facilidad de minimizar los estados multicast, a demás La mayoría de los protocolos multicast dependen del protocolo unicast sobre el que se ejecuten. Pero esta dependencia del protocolo unicast puede ser un problema, ya que los diferentes *hosts* destinos de los paquetes multicast pueden pertenecer a redes con diferentes protocolos. El protocolo CBT construye su árbol multicast con independencia del protocolo de enrutamiento unicast, lo cual presenta grandes ventajas con respecto a los demás protocolos.

CAPÍTULO 6. Implementación de multicast en una red de datos IP

En este capítulo ejemplificaremos la implementación de multicast sobre una red de datos IP. Consideraremos el caso de una empresa de capacitación con presencia a nivel nacional que desea poder ofrecer cursos con aplicaciones de educación a distancia.

Para poder realizar las pruebas de implementación, elaboramos un prototipo en laboratorio (figura 6.1), el cual es un fragmento de la red de producción debido a la disponibilidad de equipos con la que se cuenta en el propio laboratorio. Sin embargo, el trabajo realizado dentro del presente tema de tesis puede servir como base para la implementación de multicast en la red de producción.

Cabe mencionar que todos los equipos de la red, enrutadores y *switches* LAN, son de la marca Cisco.

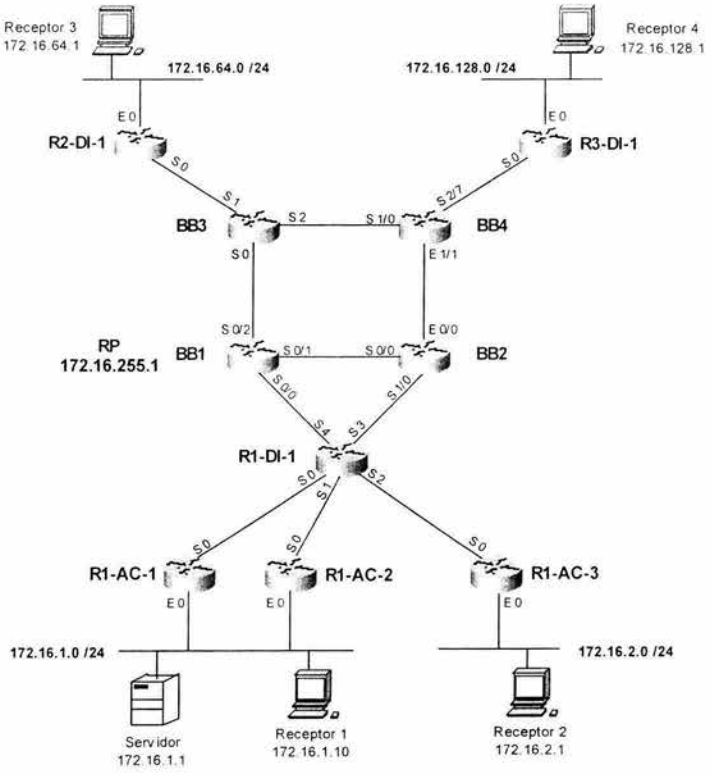


Figura 6.1 Prototipo de la red implementada

6.1 Necesidades de la empresa

Originalmente la empresa contaba con centros de capacitación únicamente en la Ciudad de México. Para algunos clientes foráneos esto representaba un problema debido a los gastos que representa el transporte y la estancia fuera de sus lugares de residencia.

Con la finalidad de ofrecer mejores opciones para sus clientes foráneos, la empresa ha establecido diversos centros de capacitación alrededor del país en los que se podrán impartir los cursos a distancia. Es decir, que el instructor podrá estar físicamente en un sitio y los participantes podrán estar en sitios remotos.

6.2 Opciones de solución

Para poder impartir los cursos en línea a múltiples participantes al mismo tiempo, ubicados en sitios distantes, se implementará una aplicación de educación a distancia a través de la cual los participantes podrán acceder a las sesiones de los cursos a los que deseen asistir.

Esta aplicación de educación a distancia puede operar tanto con IP Unicast como con IP multicast.

Unicast vs Multicast

- Con la opción IP Unicast, cada participante o asistente, establece una comunicación uno a uno (unicast) con el servidor. Con esta opción cada nuevo participante que se incorpore a la sesión incrementará la carga del servidor y el uso de ancho de banda, llegando en determinado momento a degradar el desempeño de la aplicación cuando los recursos sean insuficientes.

  Una ventaja de utilizar Unicast es que no hay que hacer modificaciones en los enrutadores de la red, ya que este es el tipo de tráfico que cursa normalmente en la red.

- Con la opción IP multicast, cada participante se agrega a un grupo multicast definido y a través de los mecanismos que ya fueron expuestos en capítulos previos, recibirá la información que se genera desde el servidor. En este caso la carga del servidor y el uso de ancho de banda se mantendrán siempre constantes sin importar el número de participantes en una sesión. De esta manera se logra mantener la eficiencia de la red y los servidores a un menor costo.

  La principal desventaja de multicast respecto a unicast, radica en que los enrutadores de la red, no encaminan los paquetes multicast de manera predeterminada. Para ello será necesario prepararlos, haciendo ciertas modificaciones, de tal manera que aprendan a encaminar este tipo de tráfico.

Analizando estas ventajas y desventajas, la mejor opción es IP multicast, a pesar de los ajustes que tengan que efectuarse, el esfuerzo realizado para la implementación será recompensado por el ahorro que se tendrá principalmente en el ancho de banda. Ya que éste es uno de los recursos de más valor dentro de la red.

6.3 Análisis de la Red

Previo a la implementación, haremos un estudio general de la red para determinar el estado actual y las facilidades o posibilidades para la implementación de IP multicast.

La red en cuestión es una red privada de paquetes IP, es decir, que solamente el protocolo IP es enrutable dentro de la red. Como protocolo de enrutamiento unicast se tiene el protocolo EIGRP [13].

Para el direccionamiento se utiliza la red clase B 172.16.0.0/16, que corresponde a uno de los rangos de direcciones reservados para redes privadas. La red no tiene interconexión con otras redes.

- Equipos que conforman la Red

  Como ya habíamos mencionado, todos los equipos que conforman la red son de la marca Cisco, en la tabla 6.1 se presentan los modelos y versiones de sistema operativo (IOS) de cada uno de ellos, además en la última columna incluimos también los protocolos multicast soportados por los sistemas operativos correspondientes.

Equipo	Tipo/Modelo	Versión Sistema	Protocolos Multicast
R1_AC_1	Enrutador 2500	12.1	PIMv2, IGMPv2
R1_AC_2	Enrutador 2500	12.1	PIMv2, IGMPv2
R1_AC_3	Enrutador 2500	11.2	PIMv2, IGMPv2
R2_DI_1	Enrutador 2500	10.3	PIMv2, IGMPv2
R3_DI_1	Enrutador 2500	10.3	PIMv2, IGMPv2
R1_DI_1	Enrutador 4500	12.1	PIMv2, IGMPv2
BB1	Enrutador 7505	12.1	PIMv2, IGMPv2
BB2	Enrutador 3640	12.0	PIMv2, IGMPv2
BB3	Enrutador 4500	10.3	**PIMv1**, IGMPv2
BB4	Enrutador 3640	12.0	PIMv2, IGMPv2
4 Switches LAN	Switch Catalyst 1900		CGMP, IGMPv2

Tabla 6.1 Equipos que conforman la Red

- Topología de la red

  La red es de tipo jerárquica, tiene un dorsal conformado por 4 enrutadores (BB1, BB2, BB3 y BB4) conectados entre sí por enlaces E1 (2Mbps).

  Todos los centros regionales (R1_AC3, R2_AC1 y R3_AC1) se encuentran conectados hacia el dorsal por enlaces E1.

  En el caso de la oficina central se tienen dos enrutadores de acceso (R1_AC1 y R1_AC2) configurados en modo de alta disponibilidad con el protocolo HSRP (Hot Standby Routing Protocol). Estos enrutadores se encuentran conectados a un nodo de tipo distribuidor (R1_DI1) por enlaces de tipo E1 también.

  En todos los enlaces se usa HDLC como protocolo de capa 2.

- Otras aplicaciones. Además de las aplicaciones de educación a distancia, el personal que laborará en los centros regionales continuará teniendo acceso a otras aplicaciones corporativas usando IP unicast como lo son:

  - Correo Electrónico.
  - Acceso a bases de datos centralizadas.
  - FTP's, acceso a archivos compartidos, impresoras en red.
  - Acceso a Intranet.

  Ver Anexo E

  El ancho de banda estimado para estas aplicaciones es de 500Kbps, mientras que para la aplicación de educación a distancia será de 480 Kbps. De esta información se deduce que la capacidad de los enlaces de la red (E1's, 2048 Kbps) soportará sin problemas el tráfico agregado por aplicaciones sobre multicast., ver Anexo C y Anexo F.

- Sistemas Operativos utilizados a nivel de usuario:

  Se cuenta actualmente con servidores y PC´s con sistemas operativos Windows principalmente.

  Win98, Win2000, WIN2000 Server, NT Server 4.0 y XP.

  Todos estos sistemas operativos soportan IGMPv2.

6.4 Implementación de multicast en la red prototipo

Ya que los enrutadores no soportan el enrutamiento de paquetes multicast de manera predeterminada, será necesario realizar algunos ajustes en la red para que esto sea posible. En primer lugar deberemos elegir un protocolo de enrutamiento multicast a través del cual los enrutadores decidirán como encaminar este tipo de paquetes.

Además para poder realizar pruebas en la red, emplearemos dos aplicaciones que simularán la aplicación de educación a distancia. La primera de ellas consiste en un servidor que estará enviando secuencias de video desde la oficina central (segmento 172.16.1.0) de manera constante. La otra es una aplicación para envío de mensajes en

línea (Chat) a través de la cual todos los participantes estarán en contacto de manera permanente. Ambas aplicaciones soportan el envío de datos a través de multicast.

6.4.1 Elección del protocolo de enrutamiento multicast

Para lograr lo anterior, primero elegiremos un protocolo de enrutamiento multicast. En nuestro caso, ya que no existe interconexión con redes externas con las cuales exista la necesidad de intercambiar tráfico multicast y además debido a que el número de fuentes dentro de la red será reducido, el protocolo de enrutamiento que elegiremos para la implementación es PIM-SM (PIM Sparse Mode) debido a sus principales características:

- Facilidad en la implementación, además que es soportado por todos los equipos de la red sin necesidad de realizar cambios ya sea en software o hardware.
- Protocolo independiente. Se basa en la tabla de enrutamiento unicast para tomar las decisiones de enrutamiento multicast.
- Integración explícita. Para que algún punto o nodo de la red reciba tráfico multicast debe existir al menos un receptor activo.
- Puede usar tanto árboles compartidos (Shared Trees) como árboles de ruta más corta (SPT) dependiendo de lo que resulte más conveniente.

En el Anexo D se muestra una opción alterna.

6.4.2 Implementación a nivel WAN

Una vez definido el protocolo de enrutamiento multicast a utilizar procederemos a definir los pasos a seguir para la implementación del mismo en la red.

- En primer lugar definiremos la ubicación del enrutador que funcionará como punto de reunión (RP). En este caso se eligió al enrutador BB1 por estar ubicado en el *backbone* y además ser uno de los más robustos dentro de la red.
- Posteriormente, para habilitar el enrutamiento de paquetes multicast en la red es necesario aplicar el siguiente comando en todos los enrutadores de la red:

  ip multicast routing

- Posteriormente se especifica el protocolo de enrutamiento multicast que vamos a utilizar, en este caso PIM-SM. Esto se realiza en cada una de las interfaces de cada enrutador de la red usando el comando:

  ip pim sparse-mode

- Por último, a cada enrutador de la red se le debe especificar cual será el punto de reunión (RP). En este caso se especifica una dirección IP de alguna de las interfaces del enrutador a través del comando:

  ip pim rp-address 172.16.255.1

6.4.3 Implementación a nivel LAN

La implementación a nivel LAN aplica solamente para *switches* de capa dos y aquellos enrutadores con conexión a redes locales. En este caso los pasos a seguir para la implementación son:

- Habilitar CGMP en *switches*.

  cgmp enable

- Habilitar CGMP en interfaces LAN de enrutadores con conexión a redes locales.

  ip cgmp

- Habilitar IGMP en interfaces LAN (puertos Ethernet) de los enrutadores con conexión a redes locales. Este protocolo está habilitado de manera predeterminada por lo que sólo es necesario verificarlo para las interfaces LAN:

  show igmp interfaces

En el Anexo A, se muestran las configuraciones finales de todos los equipos.

6.5 Pruebas realizadas

Una vez efectuadas las configuraciones correspondientes a la implementación multicast realizamos algunas pruebas con las aplicaciones escogidas para tal fin.

Estas pruebas consistieron principalmente en verificar el funcionamiento de dichas aplicaciones, así como observar el comportamiento de los enrutadores de la red (verificar la creación y el estado de tablas de enrutamiento multicast, etc.) Ver Anexo B.

Además también realizamos una comparación del uso de ancho de banda, utilizando ambos métodos de comunicación: multicast y unicast. Para esta prueba colocamos 1,2 y 3 receptores al mismo tiempo en un segmento de red determinado y con la ayuda de un analizador de red medimos el tráfico generado con ambos métodos.

6.5.1 Pruebas a nivel WAN

Como ya habíamos mencionado anteriormente, para poder realizar las pruebas en la red utilizamos dos aplicaciones que soportan el envío de datos a través de multicast. La primera de ellas consiste en un servidor que envía secuencias de audio y video de manera constante, ver Anexo C. La dirección IP de este servidor es 172.16.1.1 y esta será nuestra principal fuente de datos multicast. Los usuarios que recibirán la información de esta fuente estarán ubicados en las redes locales de los enrutadores: R1_AC1, R1_AC3, R2_AC1 y R3_AC1. Las direcciones IP de cada uno de los clientes son: 172.16.1.10, 172.16.2.1, 172.16.64.1 y 172.16.128.1 respectivamente.

La otra aplicación empleada en las pruebas es un mensajero instantáneo (Chat), a través de la cual se creará un cuarto de discusión con todos los usuarios participantes, ver Anexo C. Esta aplicación estará instalada en cada una de las computadoras de los usuarios, de tal manera que cada uno de ellos se desempeñarán también como fuentes multicast para esta otra aplicación.

La figura 6.2 muestra la ubicación en la red de la fuente y los receptores, así como el flujo de audio y video.

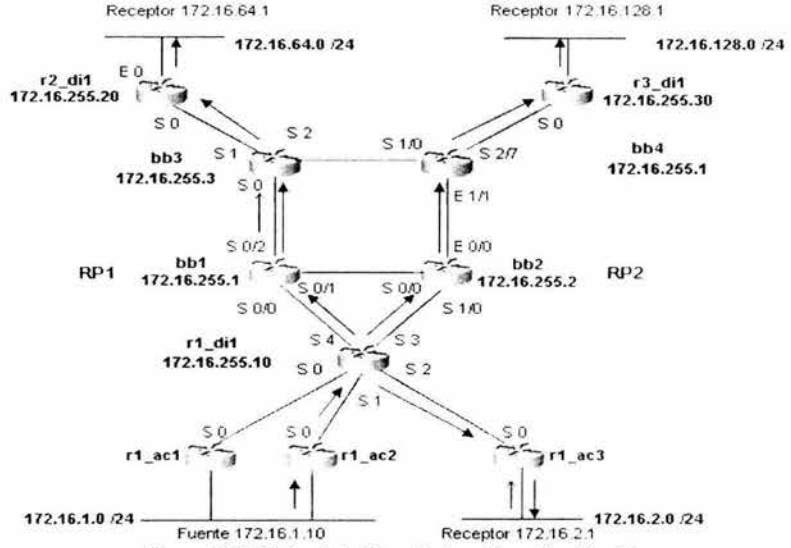


Figura 6.2 Flujo de tráfico de la aplicación de video

En la figura 6.3 se muestra el flujo correspondiente a la aplicación de mensajes instantáneos.

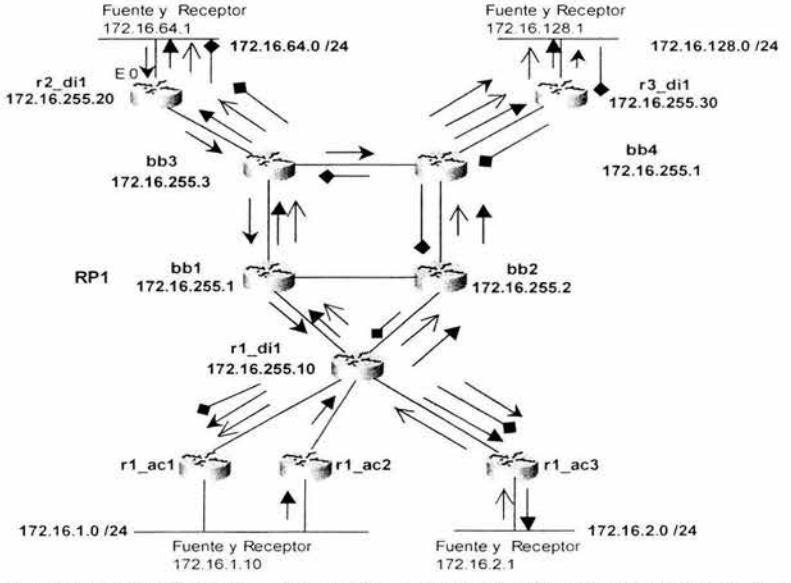


Figura 6.2 Representación Gráfica de las tablas multicast utilizando la aplicación Prochat

Observar en el Anexo B las tablas de enrutamiento multicast correspondientes.

6.5.2 Prueba de uso de ancho de banda Unicast vs Multicast

Como ya hemos mencionado, una de las razones para utilizar multicast es el ahorro en el ancho de banda. Para comprobar lo anterior realizamos una prueba comparativa de unicast y multicast, la cual consistió en medir la utilización del ancho de banda con varios usuarios recibiendo tráfico desde el servidor de manera simultánea para ambos métodos.

Para esta prueba colocamos tres clientes en el segmento 172.128.1.0 (enrutador R3_DI1) y el servidor en el segmento 172.16.1.0 (enrutadores R1_AC1 y R1_AC2). La medición del tráfico generado se realizó en el enlace BB2-BB4 con la ayuda de un analizador de red.

La primera medición realizada fue utilizando multicast. Para esta prueba los clientes se fueron agregando uno por uno aproximadamente cada tres minutos. Como se observa en la figura 6.4, el patrón de utilización de ancho de banda se mantuvo constante sin importar el número de receptores que se fueron agregando.

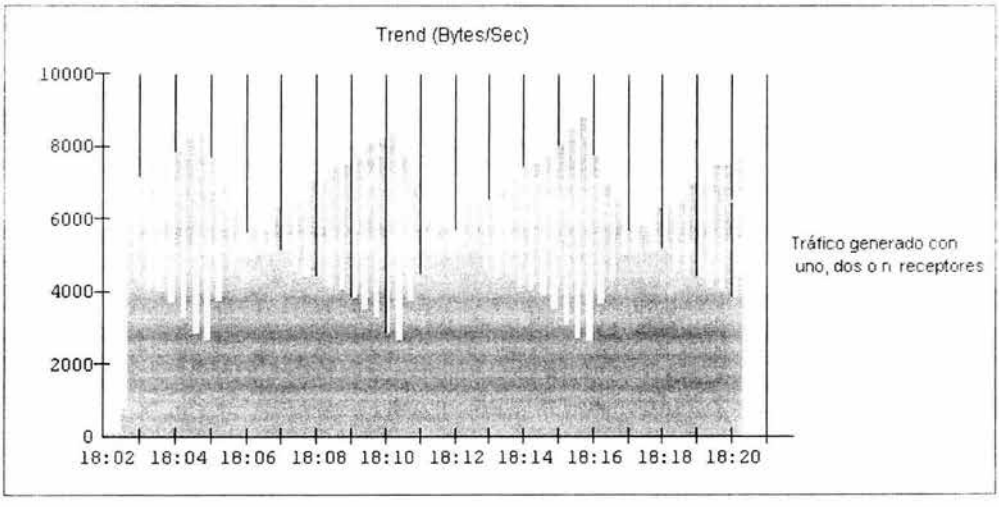


Figura 6.4 Ancho de banda utilizado para la transmisión de audio y video empleando multicast

La segunda medición realizada fue utilizando unicast. En esta prueba los clientes se fueron agregando uno por uno aproximadamente cada 4 minutos. En la figura 6.5 se observa como en este caso la utilización de ancho de banda se incrementa cada vez que se agrega un nuevo receptor.

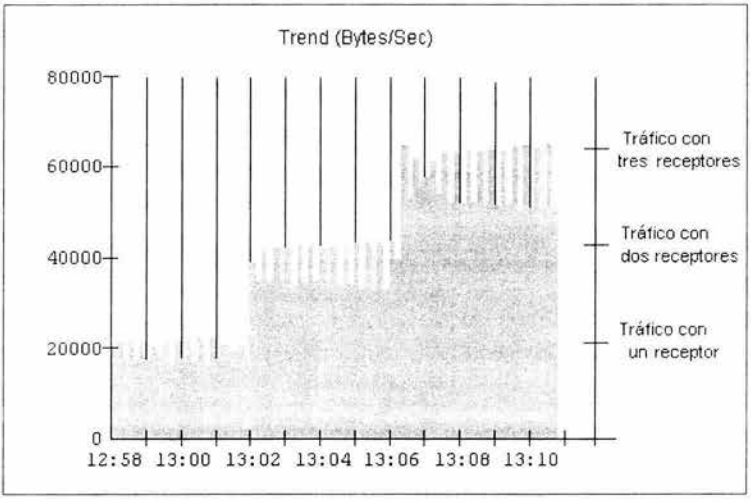


Figura 6.5 Utilización de ancho de banda en la transmisión de audio y video empleando Unicast

6.5.3 Pruebas a nivel LAN

En estas pruebas verificaremos la operación del *switch* LAN con y sin el protocolo CGMP activo. Estas pruebas se realizaron en el segmento 172.16.1.0, para este caso conectamos dos receptores multicast directamente en el *switch* correspondiente a dicho segmento. En la figura 6.6 se muestran las conexiones correspondientes.

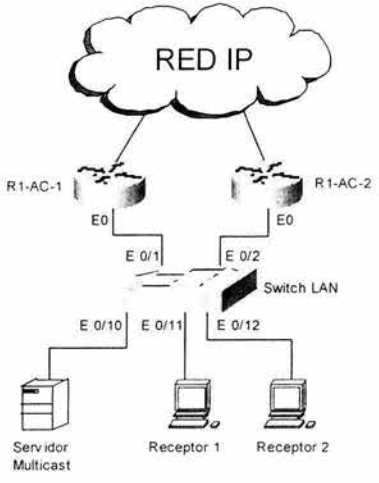


Figura 6.6 Diagrama de conexiones en el segmento LAN 172.16.1.0

- Con CGMP activo el tráfico *muticast* sólo debe fluir hacia los puertos donde hay receptores activos.
- Sin CGMP activo, el tráfico multicast deberá fluir por todos los puertos del *switch*.

Cuando CGMP está activo, con el comando *show cgmp* podemos verificar los puertos en los que se recibe tráfico multicast. En la tabla 6.3 podemos observar los datos que se obtienen con este comando.

```
R1-CM-Fastow cgmp
CGMP Status : Enabled
CGMP Fast Leave Status : Disabled
CGMP Holdtime (secs) : 600
Allow Reserved Address to Join : Enabled
VLAN     Address              Destination
------------------------------------------------------------------------------
1        0100.5E2D.9F4D       Et0/1, Et0/2, Et0/10, Et0/11, Et0/12

VLAN     Router Address       Expiration     Interface
------------------------------------------------------------------------------
1        0060.5C9F.A200       316 sec        Et0/1
1        00E0.B055.C98A       569 sec        Et0/2
```

Tabla 6.2 Salida al aplicar el comando show cgmp en el switch

Para este caso el grupo multicast utilizado fue 237.173.159.77, al cual corresponde la dirección MAC-Multicast 0100.5E2D.9F4D. Como se ve en la tabla esta dirección MAC tiene como destino los puertos 0/1, 0/2, 0/10, 0/11 y 0/12, que corresponden a los enrutadores del segmento, al servidor y a los dos receptores activos en el segmento para tal grupo multicast.

Además el *switch* también muestra los puertos donde se encuentran conectados los enrutadores y sus respectivas direcciones MAC.

Para verificar que efectivamente el tráfico multicast no se recibía en el resto de los puertos del switch, se conectó un analizador de red con el cual se comprobó que efectivamente no se recibía tráfico multicast en el resto de los puertos.

Con CGMP inactivo el tráfico multicast si se recibía en el resto de los puertos del switch.

CONCLUSIONES

Sin lugar a dudas Multicast es la mejor opción para lograr la comunicación de uno-a-muchos o muchos-a-muchos debido a la eficiencia en el uso de los recursos que esto representa. Sin embargo, y a pesar de que Multicast es un concepto antiguo, su desarrollo ha sido lento (comparado, por ejemplo, con servicios como el Web).

Aunque Multicast es un servicio que requiere una "inteligencia" adicional para lograr administrar y mantener los estados de los diferentes grupos activos dentro de la red, la dificultad está en el desarrollo y administración de estos servicios en una infraestructura, como Internet, que se caracteriza por ofrecer solamente un servicio unicast bajo la premisa del mejor esfuerzo.

Bajo esta perspectiva, la imagen de Multicast parece opacarse, sin embargo por todos los potenciales que multicast tiene, se ha convertido en un reto para los desarrolladores de Internet. Actualmente es una característica que muchos fabricantes desean incorporar en sus productos y finalmente está empezando a ser usado por un número de compañías que ofrecen servicios a gran escala.

Desde todos las perspectivas multicast se está convirtiendo en uno de los servicios con mayor interés dentro de la comunidad de Internet y más aún, en redes corporativas donde aplicaciones como capacitación a distancia ofrecen una muy atractiva reducción de costos.

Esta tesis se desarrollo en torno a la implementación de multicast sobre una red privada IP en operación, para la cual fueron necesarios los siguientes pasos:

- Análisis de la tecnología,
- Análisis de la arquitectura de la red en operación,
- Análisis de los requerimientos del cliente sobre la red en operación,
- Propuesta de arquitectura multicast sobre la red en operación,
- Pruebas de laboratorio en un prototipo de la red en operación.

Del trabajo realizado, se observó lo siguiente:

- La implementación de multicast sobre una red IP en operación, se puede lograr de forma transparente sin afectar el funcionamiento de la misma en su desempeño normal.
- Se recomienda el uso de PIM en modo Sparse-Dense para que la solución sea más robusta y escalable.

- Una vez implementado multicast, se puede crecer utilizando aplicaciones más completas dentro del mismo esquema con respecto al número de grupos ya que esto es independiente de la configuración general de la red.
- Sí se requiere diferenciar servicios (como voz sobre datos), será necesario implementar nuevas tecnologías que mejoren el desempeño de la red como MPLS, VPNS, QoS, etcétera, lo cual requerirá de nuevos análisis de la red.
- En cuanto al nivel de capa dos en particular de los Switches, es importante manejar de forma adecuada la asignación de las direcciones de grupo para evitar el traslape de grupos (por la dirección MAC) y tráfico no deseado en la red LAN.
- De forma general la implementación y el funcionamiento de la tecnología multicast se comporto de acuerdo a lo esperado y estudiado en la teoría, logrando de esta forma una reseña confiable para la implementación básica de Multicast en una red IP en operación. Por lo tanto, con base a estas pruebas podemos proceder a realizar una prueba piloto en una red real esperando contar con los mismos resultados.

Por último, podemos concluir lo siguiente con respecto a la tecnología Multicast:

- Es una tecnología que permite aprovechar adecuadamente los recursos de red (ancho de banda) y de servidores (procesamiento de CPU) logrando así una reducción de recursos, y por lo tanto de los costos.
- Es una tecnología con una amplia posibilidad de aplicaciones, entre ellas las de capacitación a distancia, las cuales son muy atractivas en términos de reducción de costos de transporte y hospedaje.
- Es fácil de implementar. Solo se requiere verificar la capacidad de los enrutadores para soportar multicast, si es así no se requerirán grandes cambios en la red para lograr su implementación.
- Existen distintos tipos de protocolos de enrutamiento multicast de acuerdo a la utilización y del número de grupos, siendo los más óptimos: PIM-Modo Esparcido y CBT.
- Si se requiere asegurar un cierto ancho de banda o dar prioridad para las aplicaciones multicast se pueden aplicar técnicas de calidad de servicio (QoS), para lo cual sería necesario un análisis más detallado de la red ya que para poder aplicarlas si se requieren cambios mayores (cambios de *hardware*, por ejemplo).
- Como una continuación al presente trabajo se pueden realizar investigaciones para la implementación de multicast en redes MPLS, así como también en redes privadas virtuales IP (IP-VPN) basadas en MPLS.

ANEXOS

Anexo A. Configuraciones

A continuación mostramos las configuraciones finales de todos los enrutadores de la red, después de la implementación de *multicast*:

Enrutador BB1

```
!
version 12.0
service timestamps debug datetime localtime
service timestamps log datetime localtime
!
hostname BB1
!
logging buffered 100000 debugging
enable password prov
!
clock timezone CST -6
clock summer-time CDT recurring
no ip domain-lookup
!
ip multicast-routing
!
interface Loopback0
 ip address 172.16.255.1 255.255.255.255
 no ip directed-broadcast
!
interface Serial0/0
 ip address 172.16.251.14 255.255.255.252
 no ip directed-broadcast
 ip pim sparse-mode
!
interface Serial0/1
 ip address 172.16.254.5 255.255.255.252
 no ip directed-broadcast
 ip pim sparse-mode
 clockrate 2000000
!
interface Serial0/2
 ip address 172.16.254.1 255.255.255.252
 no ip directed-broadcast
 ip pim sparse-mode
 clockrate 2000000
!
router eigrp 100
 network 172.16.0.0
!
ip pim rp-address 172.16.255.1
ip classless
!
ntp clock-period 17179634
ntp server 172.16.255.2
end
```

Enrutador BB2

```
!
version 12.0
service timestamps debug datetime localtime
service timestamps log datetime localtime
!
hostname BB2
!
logging buffered 1000000 debugging
enable password prov
!
clock timezone CST -6
clock summer-time CDT recurring
no ip domain-lookup
!
ip multicast-routing
!
interface Loopback0
 ip address 172.16.255.2 255.255.255.255
 no ip directed-broadcast
!
interface Ethernet0/0
 bandwidth 2048
 ip address 172.16.254.10 255.255.255.252
 no ip directed-broadcast
 ip pim sparse-mode
 delay 2000
 no fair-queue
!
interface Serial0/0
 ip address 172.16.254.6 255.255.255.252
 no ip directed-broadcast
 ip pim sparse-mode
!
interface Serial1/0
 ip address 172.16.251.18 255.255.255.252
 no ip directed-broadcast
 ip pim sparse-mode
!
router eigrp 100
 network 172.16.0.0
!
ip pim rp-address 172.16.255.1
ip classless
!
ntp master 2
end
```

Enrutador BB3

```
!
version 10.3
service timestamps debug datetime localtime
service timestamps log datetime localtime
!
hostname BB3
!
clock timezone CST -6
clock summer-time CDT recurring
enable password prov
!
no ip domain-lookup
ip pim rp-address 172.16.255.1
ip multicast-routing
!
interface Loopback0
 ip address 172.16.255.3 255.255.255.255
 no ip route-cache
!
interface Serial0
 ip address 172.16.254.2 255.255.255.252
 ip pim sparse-mode
 bandwidth 2048
!
interface Serial1
 ip address 172.16.252.2 255.255.255.252
 ip pim sparse-mode
 clockrate 2000000
!
interface Serial2
 ip address 172.16.254.13 255.255.255.252
 ip pim sparse-mode
 bandwidth 2048
 clockrate 2000000
!
router eigrp 100
 network 172.16.0.0
!
ip classless
ip pim rp-address 172.16.255.1

logging buffered 1000000
!
ntp clock-period 17179718
ntp server 172.16.255.2
end
```

Enrutador BB4

```
!
version 12.0
service timestamps debug datetime localtime
service timestamps log datetime localtime
!
hostname BB4
!
logging buffered 1000000 debugging
enable password prov
!
clock timezone CST -6
clock summer-time CDT recurring
no ip domain-lookup
!
ip multicast-routing
!
interface Loopback0
 ip address 172.16.255.4 255.255.255.255
 no ip directed-broadcast
!
interface Serial1/0
 ip address 172.16.254.14 255.255.255.252
 no ip directed-broadcast
 ip pim sparse-mode
 no fair-queue
!
interface Ethernet1/1
 bandwidth 2048
 ip address 172.16.254.9 255.255.255.252
 no ip directed-broadcast
 ip pim sparse-mode
 delay 2000
 no fair-queue
!
interface Serial2/7
 ip address 172.16.253.2 255.255.255.252
 no ip directed-broadcast
 ip pim sparse-mode
 no fair-queue
!
router eigrp 100
 network 172.16.0.0
!
ip classless
ip pim rp-address 172.16.255.1
!
ntp clock-period 17179804
ntp server 172.16.255.2
end
```

Enrutador R1-DI-1

```
!
version 12.1
service timestamps debug datetime localtime
service timestamps log datetime localtime
!
hostname R1-DI-1
!
logging buffered 1000000 debugging
enable password prov
!
```

Enrutador R2-DI-1

```
!
version 12.1
service timestamps debug datetime localtime
service timestamps log datetime localtime
!
hostname R2-DI-1
!
enable password prov
!
clock timezone CST -6
```

```
clock timezone CST -6
clock summer-time CDT recurring
no ip domain-lookup
!
ip multicast-routing
!
interface Loopback0
 ip address 172.16.255.10 255.255.255.255
!
interface Serial0
 ip address 172.16.251.2 255.255.255.252
 ip pim sparse-mode
 clockrate 2000000
!
interface Serial1
 ip address 172.16.251.6 255.255.255.252
 ip pim sparse-mode
 clockrate 4000000
!
interface Serial2
 ip address 172.16.251.10 255.255.255.252
 ip pim sparse-mode
 clockrate 4000000
!
interface Serial3
 ip address 172.16.251.17 255.255.255.252
 ip pim sparse-mode
 clockrate 2000000
!
interface Serial4
 ip address 172.16.251.13 255.255.255.252
 ip pim sparse-mode
 clockrate 2000000
!
router eigrp 100
 network 172.16.0.0
!
ip classless
ip pim rp-address 172.16.255.1
!
ntp clock-period 17179968
ntp server 172.16.255.2
end
```

```
clock summer-time CDT recurring
ip subnet-zero
no ip domain-lookup
!
ip multicast-routing
!
interface Loopback0
 ip address 172.16.255.20 255.255.255.255
!
interface Ethernet0
 ip address 172.16.64.254 255.255.255.0
 ip pim sparse-mode
!
interface Serial0
 ip address 172.16.252.1 255.255.255.252
 ip pim sparse-mode
 no fair-queue
!
router eigrp 100
 network 172.16.0.0
!
ip classless
no ip http server
ip pim rp-address 172.16.255.1
!
ntp clock-period 17179774
ntp server 172.16.255.2
end
```

Enrutador R3-DI-1

```
!
version 12.1
service timestamps debug datetime localtime
service timestamps log datetime localtime
!
hostname R3-DI-1
!
enable password prov
!
clock timezone CST -6
clock summer-time CDT recurring
ip subnet-zero
no ip domain-lookup
!
ip multicast-routing
!
interface Loopback0
 ip address 172.16.255.30 255.255.255.255
!
interface Ethernet0
 ip address 172.16.128.254 255.255.255.0
 ip pim sparse-mode
```

Enrutador R1-AC-1

```
!
version 12.1
service timestamps debug datetime localtime
service timestamps log datetime localtime
!
hostname R1-AC-1
!
logging buffered 1000000 debugging
enable password prov
!
clock timezone CST -6
clock summer-time CDT recurring
ip subnet-zero
no ip domain-lookup
!
ip multicast-routing
!
interface Loopback0
 ip address 172.16.255.11 255.255.255.255
!
interface Ethernet0
 ip address 172.16.1.253 255.255.255.0
```

```
!
interface Serial0
 ip address 172.16.253.1 255.255.255.252
 ip pim sparse-mode
 no fair-queue
 clockrate 2000000
!
router eigrp 100
 network 172.16.0.0
!
ip classless
ip pim rp-address 172.16.255.1
!
ntp clock-period 17179759
ntp server 172.16.255.2
end
```

```
 ip pim sparse-mode
 ip cgmp
 standby ip 172.16.1.254
 standby priority 105
 standby track Serial0 10
!
interface Serial0
 ip address 172.16.251.1 255.255.255.252
 ip pim sparse-mode
 no fair-queue
!
router eigrp 100
 network 172.16.0.0
!
ip classless
ip pim rp-address 172.16.255.1
!
ntp clock-period 17179944
ntp server 172.16.255.2 prefer
end
```

Enrutador R1-AC-2

```
!
version 12.1
service timestamps debug datetime localtime
service timestamps log datetime localtime
!
hostname R1-AC-2
!
logging buffered 1000000 debugging
enable password prov
!
clock timezone CST -6
clock summer-time CDT recurring
ip subnet-zero
no ip domain-lookup
!
ip multicast-routing
!
interface Loopback0
 ip address 172.16.255.12 255.255.255.255
!
interface Ethernet0
 ip address 172.16.1.252 255.255.255.0
 no ip redirects
 ip pim sparse-mode
 ip cgmp
 standby ip 172.16.1.254
 standby track Serial0 10
!
interface Serial0
 ip address 172.16.251.5 255.255.255.252
 ip pim sparse-mode
 no fair-queue
!
router eigrp 100
 network 172.16.0.0
!
ip classless
ip pim rp-address 172.16.255.1
!
ntp clock-period 17179795
ntp server 172.16.255.2
end
```

Enrutador R1-AC-3

```
!
version 11.2
service timestamps debug datetime localtime
service timestamps log datetime localtime
!
hostname R1-AC-3
!
enable password prov
!
no ip domain-lookup
ip multicast-routing
clock timezone CST -6
clock summer-time CDT recurring
!
interface Loopback0
 ip address 172.16.255.13 255.255.255.255
!
interface Ethernet0
 ip address 172.16.2.254 255.255.255.0
 ip pim sparse-mode
 ip cgmp
!
interface Serial0
 ip address 172.16.251.9 255.255.255.252
 ip pim sparse-mode
 no fair-queue
!
router eigrp 100
 network 172.16.0.0
!
ip classless
ip pim rp-address 172.16.255.1
!
logging buffered 1000000 debugging
!
ntp clock-period 17180090
ntp server 172.16.255.2 prefer
end
```

Y a continuación mostramos también el proceso para habilitar CGMP en los *switches*:

```
       Catalyst 1900 - Main Menu
     [C] Console Settings
     [S] System
     [N] Network Management
     [P] Port Configuration
     [A] Port Addressing
     [D] Port Statistics Detail
     [M] Monitoring
     [V] Virtual LAN
     [R] Multicast Registration
     [F] Firmware
     [I] RS-232 Interface
     [U] Usage Summaries
     [H] Help
     [K] Command Line

     [X] Exit Management Console
Enter Selection:  N

        Catalyst 1900 - Network Management
     [I] IP Configuration
     [S] SNMP Management
     [B] Bridge - Spanning Tree
     [C] Cisco Discovery Protocol
     [G] Cisco Group Management Protocol
     [H] HTTP Server Configuration
     [R] Cluster Management
     [X] Exit to Main Menu
Enter Selection:  G
   Catalyst 1900 - Cisco Group Management Protocol (CGMP) Configuration
    ---------------------- Settings ----------------------------------------
     [H] Router Hold Time (secs)                600
     [C] CGMP                                   Disabled
     [F] CGMP Fast Leave                        Disabled
    ---------------------- Actions -----------------------------------------
     [L] List IP multicast addresses
     [R] Remove IP multicast addresses
     [X] Exit to previous menu
Enter Selection:  C

This command enables or disables CGMP on the switch.

NOTE: Changing CGMP status from Disabled to Enabled will result
in deletion of all Multicast MAC Addresses of the form 0x01-00-5E-XX-XX-XX
that have been registered through the Multicast Registration Menu.

CGMP status may be [E]nabled or [D]isabled.
Current setting ===> Disabled
    New setting ===> Enabled

   Catalyst 1900 - Cisco Group Management Protocol (CGMP)
Configuration
    ---------------------- Settings ----------------------------------------
     [H] Router Hold Time (secs)                600
     [C] CGMP                                   Enabled
     [F] CGMP Fast Leave                        Disabled
    ---------------------- Actions -----------------------------------------
     [L] List IP multicast addresses
     [R] Remove IP multicast addresses
     [X] Exit to previous menu
Enter Selection:  X
```

Anexo B. Tablas de Enrutamiento

Enseguida mostraremos las tablas de enrutamiento multicast observadas cuando todos los receptores están recibiendo tráfico. Para poder observar dichas tablas en los enrutadores, se utiliza el comando "*show ip mroute*". Como resultado, este comando nos muestra en primer lugar el significado de las banderas* que se observan en cada una de las entradas de la tabla de enrutamiento, posteriormente nos muestra la tabla de enrutamiento propiamente.

* Las banderas nos indican la forma en como están operando los grupos multicast en la red, por ejemplo: si es modo denso (D) o esparcido (S), si el árbol debe crearse utilizando la ruta más corta SPT (T), etc.

Para abreviar mostraremos la sección de la tabla de enrutamiento multicast que describe el significado de las banderas por separado, figura 1. Posteriormente mostraremos las tablas de cada uno de los enrutadores omitiendo dicha sección.

```
Router#show ip mroute
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, C - Connected, L - Local, P - Pruned
       R - RP-bit set, F - Register flag, T - SPT-bit set, J - Join SPT
       M - MSDP created entry, X - Proxy Join Timer Running
       A - Advertised via MSDP
Outgoing interface flags: H - Hardware switched
Timers: Uptime/Expires
```

Figura 1. Significado de las banderas (Flags) en las tablas de enrutamiento multicast

Tabla de enrutamiento multicast en enrutador BB1:

```
Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 237.173.159.77), 00:47:44/00:03:12, RP 172.16.255.1, flags: S
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Serial0/0, Forward/Sparse, 00:28:20/00:03:11
    Serial0/2, Forward/Sparse, 00:28:28/00:02:15

(172.16.1.1, 237.173.159.77), 00:47:44/00:02:59, flags: TA
  Incoming interface: Serial0/0, RPF nbr 172.16.251.13
  Outgoing interface list:
    Serial0/2, Forward/Sparse, 00:28:28/00:02:15

(*, 227.0.0.2), 00:32:38/00:03:25, RP 172.16.255.1, flags: S
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Serial0/2, Forward/Sparse, 00:25:02/00:02:46
    Serial0/0, Forward/Sparse, 00:25:06/00:03:25

(172.16.1.10, 227.0.0.2), 00:02:04/00:02:52, flags: TA
  Incoming interface: Serial0/0, RPF nbr 172.16.251.13
  Outgoing interface list:
    Serial0/2, Forward/Sparse, 00:02:04/00:02:46

(172.16.2.1, 227.0.0.2), 00:10:12/00:02:52, flags: TA
  Incoming interface: Serial0/0, RPF nbr 172.16.251.13
  Outgoing interface list:
    Serial0/2, Forward/Sparse, 00:10:13/00:02:46
```

```
(172.16.64.1, 227.0.0.2), 00:05:07/00:02:52, flags: TA
  Incoming interface: Serial0/2, RPF nbr 172.16.254.2
  Outgoing interface list:
    Serial0/0, Forward/Sparse, 00:05:07/00:02:54

(172.16.128.1, 227.0.0.2), 00:16:06/00:02:52, flags: TA
  Incoming interface: Serial0/1, RPF nbr 172.16.254.6
 Outgoing interface list:
    Serial0/2, Forward/Sparse, 00:00:14/00:02:45
    Serial0/0, Forward/Sparse, 00:16:07/00:02:54

(*, 224.0.1.40), 02:42:57/00:00:00, RP 0.0.0.0, flags: DJCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Serial0/0, Forward/Sparse, 02:42:14/00:03:06
    Serial0/1, Forward/Sparse, 02:42:39/00:02:38
    Serial0/2, Forward/Sparse, 02:42:57/00:02:27
```

Tabla de enrutamiento multicast en enrutador BB2:

```
(*, 237.173.159.77), 00:28:43/00:02:59, RP 172.16.255.1, flags: SP
  Incoming interface: Serial0/0, RPF nbr 172.16.254.5
  Outgoing interface list: Null

(172.16.1.1, 237.173.159.77), 00:28:43/00:03:29, flags: T
  Incoming interface: Serial1/0, RPF nbr 172.16.251.17
  Outgoing interface list:
    Ethernet0/0, Forward/Sparse, 00:28:43/00:02:34

(*, 227.0.0.2), 00:25:17/00:02:59, RP 172.16.255.1, flags: SP
  Incoming interface: Serial0/0, RPF nbr 172.16.254.5
  Outgoing interface list: Null

(172.16.1.10, 227.0.0.2), 00:02:19/00:03:28, flags: T
  Incoming interface: Serial1/0, RPF nbr 172.16.251.17
  Outgoing interface list:
    Ethernet0/0, Forward/Sparse, 00:02:19/00:02:59

(172.16.2.1, 227.0.0.2), 00:10:27/00:03:28, flags: T
  Incoming interface: Serial1/0, RPF nbr 172.16.251.17
  Outgoing interface list:
    Ethernet0/0, Forward/Sparse, 00:10:27/00:02:59

(172.16.128.1, 227.0.0.2), 00:16:20/00:03:28, flags: T
  Incoming interface: Ethernet0/0, RPF nbr 172.16.254.9
  Outgoing interface list:
    Serial0/0, Forward/Sparse, 00:16:20/00:03:10

(*, 224.0.1.40), 00:29:54/00:00:00, RP 172.16.255.1, flags: SJCL
  Incoming interface: Serial0/0, RPF nbr 172.16.254.5
  Outgoing interface list:
    Serial1/0, Forward/Sparse, 00:29:54/00:00:00
    Ethernet0/0, Forward/Sparse, 00:29:54/00:02:47
```

Tabla de enrutamiento multicast en enrutador BB3:

```
(*, 237.173.159.77), 00:28:55/0:02:54, RP 172.16.255.1, flags: S
  Incoming interface: Serial0, RPF neighbor 172.16.254.1
  Outgoing interface list:
    Serial1, Forward state, Sparse mode, uptime 00:28:55, expires 0:02:51
    Serial2, Forward state, Sparse mode, uptime 00:28:55, expires 0:02:50

(172.16.1.1/32, 237.173.159.77), uptime 00:28:55, expires 0:02:55, flags: T
  Incoming interface: Serial0, RPF neighbor 172.16.254.1
  Outgoing interface list:
    Serial1, Forward state, Sparse mode, uptime 00:28:55, expires 0:02:51
    Serial2, Prune state, Sparse mode, uptime 00:28:55, expires 0:02:51
```

```
(*, 227.0.0.2), 00:25:28/0:02:57, RP 172.16.255.1, flags: S
  Incoming interface: Serial0, RPF neighbor 172.16.254.1
  Outgoing interface list:
    Serial1, Forward state, Sparse mode, uptime 00:25:28, expires 0:02:33
    Serial2, Forward state, Sparse mode, uptime 00:25:28, expires 0:02:14

(172.16.1.10/32, 227.0.0.2), uptime 00:02:30, expires 0:02:47, flags: T
  Incoming interface: Serial0, RPF neighbor 172.16.254.1
  Outgoing interface list:
    Serial1, Forward state, Sparse mode, uptime 00:02:30, expires 0:02:37
    Serial2, Prune state, Sparse mode, uptime 00:02:30, expires 0:02:17

(172.16.2.1/32, 227.0.0.2), uptime 00:10:38, expires 0:02:45, flags: T
  Incoming interface: Serial0, RPF neighbor 172.16.254.1
  Outgoing interface list:
    Serial1, Forward state, Sparse mode, uptime 00:10:38, expires 0:02:34
    Serial2, Prune state, Sparse mode, uptime 00:10:38, expires 0:02:17

(172.16.64.1/32, 227.0.0.2), uptime 00:02:30, expires 0:02:46, flags: T
  Incoming interface: Serial1, RPF neighbor 172.16.252.1
  Outgoing interface list:
    Serial2, Forward state, Sparse mode, uptime 00:02:30, expires 0:02:16
    Serial0, Forward state, Sparse mode, uptime 00:02:30, expires 0:02:27

(172.16.128.1/32, 227.0.0.2), uptime 00:00:38, expires 0:02:21
  Incoming interface: Serial2, RPF neighbor 172.16.254.14
  Outgoing interface list:
    Serial1, Forward state, Sparse mode, uptime 00:00:38, expires 0:02:37

(*, 224.0.1.40), 00:29:23/0:02:50, RP 172.16.255.1, flags: S
  Incoming interface: Serial0, RPF neighbor 172.16.254.1
  Outgoing interface list:
    Serial2, Forward state, Sparse mode, uptime 00:29:23, expires 0:02:52
    Serial1, Forward state, Sparse mode, uptime 00:29:04, expires 0:02:41
```

Tabla de enrutamiento multicast en enrutador enrutador BB4:

```
(*, 237.173.159.77), 00:29:03/00:03:14, RP 172.16.255.1, flags: S
  Incoming interface: Serial1/0, RPF nbr 172.16.254.13
  Outgoing interface list:
    Serial2/7, Forward/Sparse, 00:29:03/00:03:14

(172.16.1.1, 237.173.159.77), 00:29:02/00:03:29, flags: T
  Incoming interface: Ethernet1/1, RPF nbr 172.16.254.10
  Outgoing interface list:
    Serial2/7, Forward/Sparse, 00:29:02/00:03:14

(*, 227.0.0.2), 00:25:36/00:02:59, RP 172.16.255.1, flags: S
  Incoming interface: Serial1/0, RPF nbr 172.16.254.13
  Outgoing interface list:
    Serial2/7, Forward/Sparse, 00:25:36/00:02:40

(172.16.1.10, 227.0.0.2), 00:02:38/00:03:09, flags: T
  Incoming interface: Ethernet1/1, RPF nbr 172.16.254.10
  Outgoing interface list:
    Serial2/7, Forward/Sparse, 00:02:38/00:02:40

(172.16.2.1, 227.0.0.2), 00:10:46/00:03:09, flags: T
  Incoming interface: Ethernet1/1, RPF nbr 172.16.254.10
  Outgoing interface list:
    Serial2/7, Forward/Sparse, 00:10:46/00:02:40

(172.16.64.1, 227.0.0.2), 00:05:41/00:03:09, flags: T
  Incoming interface: Serial1/0, RPF nbr 172.16.254.13
  Outgoing interface list:
    Serial2/7, Forward/Sparse, 00:05:41/00:02:40

(172.16.128.1, 227.0.0.2), 00:16:38/00:03:09, flags: T
  Incoming interface: Serial2/7, RPF nbr 172.16.253.1
```

```
    Outgoing interface list:
      Ethernet1/1, Forward/Sparse, 00:16:38/00:02:40
(*, 224.0.1.40), 00:29:27/00:02:56, RP 172.16.255.1, flags: SJ
  Incoming interface: Serial1/0, RPF nbr 172.16.254.13
  Outgoing interface list:
    Serial2/7, Forward/Sparse, 00:29:07/00:03:19
```

Tabla de enrutamiento multicast en enrutador R1-DI-1:

```
(*, 237.173.159.77), 00:29:49/00:02:59, RP 172.16.255.1, flags: S
  Incoming interface: Serial4, RPF nbr 172.16.251.14
  Outgoing interface list:
    Serial2, Forward/Sparse, 00:29:04/00:02:56

(172.16.1.1, 237.173.159.77), 00:29:11/00:03:29, flags: T
  Incoming interface: Serial1, RPF nbr 172.16.251.5
  Outgoing interface list:
    Serial2, Forward/Sparse, 00:29:04/00:02:56
    Serial3, Forward/Sparse, 00:29:10/00:03:05
    Serial4, Forward/Sparse, 00:29:11/00:03:02

(*, 227.0.0.2), 00:25:48/00:03:02, RP 172.16.255.1, flags: S
  Incoming interface: Serial4, RPF nbr 172.16.251.14
  Outgoing interface list:
    Serial0, Forward/Sparse, 00:02:46/00:02:42
    Serial2, Forward/Sparse, 00:25:48/00:02:02

(172.16.1.10, 227.0.0.2), 00:02:46/00:03:01, flags: T
  Incoming interface: Serial1, RPF nbr 172.16.251.5
  Outgoing interface list:
    Serial3, Forward/Sparse, 00:02:46/00:02:32
    Serial4, Forward/Sparse, 00:02:46/00:02:43
    Serial2, Forward/Sparse, 00:02:46/00:02:02

(172.16.2.1, 227.0.0.2), 00:10:54/00:03:01, flags: T
  Incoming interface: Serial2, RPF nbr 172.16.251.9
  Outgoing interface list:
    Serial0, Forward/Sparse, 00:02:46/00:02:12
    Serial3, Forward/Sparse, 00:10:54/00:02:32
    Serial4, Forward/Sparse, 00:10:54/00:02:43

(*, 224.0.1.40), 02:44:24/00:00:00, RP 172.16.255.1, flags: SJCL
  Incoming interface: Serial4, RPF nbr 172.16.251.14
  Outgoing interface list:
    Serial2, Forward/Sparse, 02:44:02/00:02:52
    Serial0, Forward/Sparse, 02:44:09/00:02:30
    Serial1, Forward/Sparse, 02:44:24/00:03:25
```

Tabla de enrutamiento multicast en enrutador R1-AC-1

```
(*, 237.173.159.77), 00:48:52/00:02:59, RP 172.16.255.1, flags: SPF
  Incoming interface: Serial0, RPF nbr 172.16.251.2
  Outgoing interface list: Null

(172.16.1.1, 237.173.159.77), 00:48:52/00:02:59, flags: PFT
  Incoming interface: Ethernet0, RPF nbr 0.0.0.0
  Outgoing interface list: Null

(*, 227.0.0.2), 00:03:11/00:02:59, RP 172.16.255.1, flags: SCF
  Incoming interface: Serial0, RPF nbr 172.16.251.2
 Outgoing interface list:
   Ethernet0, Forward/Sparse, 00:03:12/00:02:11

 (172.16.1.10, 227.0.0.2), 00:03:13/00:02:05, flags: PCFT
Incoming interface: Ethernet0, RPF nbr 0.0.0.0
Outgoing interface list: Null

(*, 224.0.1.40), 02:44:11/00:00:00, RP 172.16.255.1, flags: SJCL
  Incoming interface: Serial0, RPF nbr 172.16.251.2
  Outgoing interface list:
```

```
    Ethernet0, Forward/Dense, 02:44:11/00:00:00
```

Tabla de enrutamiento multicast en enrutador R1-AC-2

```
(*, 237.173.159.77), 00:49:02/00:02:58, RP 172.16.255.1, flags: SP
  Incoming interface: Serial0, RPF nbr 172.16.251.6
  Outgoing interface list: Null

(172.16.1.1, 237.173.159.77), 00:49:02/00:03:29, flags: T
  Incoming interface: Ethernet0, RPF nbr 0.0.0.0
  Outgoing interface list:
    Serial0, Forward/Sparse, 00:29:46/00:02:52

(*, 227.0.0.2), 00:03:21/00:02:59, RP 172.16.255.1, flags: SP
  Incoming interface: Serial0, RPF nbr 172.16.251.6
  Outgoing interface list: Null

(172.16.1.10, 227.0.0.2), 00:03:22/00:02:26, flags: T
  Incoming interface: Ethernet0, RPF nbr 0.0.0.0
  Outgoing interface list:
    Serial0, Forward/Sparse, 00:03:21/00:03:07

(*, 224.0.1.40), 02:44:11/00:00:00, RP 172.16.255.1, flags: SJCL
  Incoming interface: Serial0, RPF nbr 172.16.251.6
  Outgoing interface list:
    Ethernet0, Forward/Dense, 02:44:11/00:00:00
```

Tabla de enrutamiento multicast en enrutador R1-AC-3

```
(*, 237.173.159.77), 00:30:32/00:02:59, RP 172.16.255.1, flags: SJC
  Incoming interface: Serial0, RPF nbr 172.16.251.10
  Outgoing interface list:
    Ethernet0, Forward/Sparse, 00:29:46/00:02:02

(172.16.1.1/32, 237.173.159.77), 00:29:46/00:02:59, flags: CJT
  Incoming interface: Serial0, RPF nbr 172.16.251.10
  Outgoing interface list:
    Ethernet0, Forward/Sparse, 00:29:46/00:02:02

(*, 227.0.0.2), 00:26:31/00:02:59, RP 172.16.255.1, flags: SCF
  Incoming interface: Serial0, RPF nbr 172.16.251.10
  Outgoing interface list:
    Ethernet0, Forward/Sparse, 00:26:31/00:02:55

 (172.16.2.1/32, 227.0.0.2), 00:11:38/00:01:48, flags: CFT
  Incoming interface: Ethernet0, RPF nbr 0.0.0.0
  Outgoing interface list:
    Serial0, Forward/Sparse, 00:11:38/00:02:29

(*, 224.0.1.40), 04:41:37/00:00:00, RP 172.16.255.1, flags: SJPCL
  Incoming interface: Serial0, RPF nbr 172.16.251.10
  Outgoing interface list: Null
```

Tabla de enrutamiento multicast en enrutador R2-DI-1

```
(*, 237.173.159.77), 00:29:21/00:02:58, RP 172.16.255.1, flags: SJC
  Incoming interface: Serial0, RPF nbr 172.16.252.2
  Outgoing interface list:
    Ethernet0, Forward/Sparse, 00:27:29/00:02:11

(172.16.1.1, 237.173.159.77), 00:27:29/00:02:59, flags: CJT
  Incoming interface: Serial0, RPF nbr 172.16.252.2
  Outgoing interface list:
    Ethernet0, Forward/Sparse, 00:27:29/00:02:11

(*, 227.0.0.2), 00:25:54/00:02:59, RP 172.16.255.1, flags: SCF
  Incoming interface: Serial0, RPF nbr 172.16.252.2
```

```
  Outgoing interface list:
    Ethernet0, Forward/Sparse, 00:25:40/00:02:05

(172.16.64.1, 227.0.0.2), 00:05:58/00:02:22, flags: CFT
  Incoming interface: Ethernet0, RPF nbr 0.0.0.0
  Outgoing interface list:
    Serial0, Forward/Sparse, 00:05:04/00:02:53

(*, 224.0.1.40), 00:29:53/00:00:00, RP 172.16.255.1, flags: SJPCL
  Incoming interface: Serial0, RPF nbr 172.16.252.2
  Outgoing interface list: Null
```

Tabla de enrutamiento multicast en enrutador R3-DI-1

```
(*, 237.173.159.77), 00:29:29/00:02:59, RP 172.16.255.1, flags: SJC
  Incoming interface: Serial0, RPF nbr 172.16.253.2
  Outgoing interface list:
    Ethernet0, Forward/Sparse, 00:29:29/00:02:16

(172.16.1.1, 237.173.159.77), 00:29:28/00:02:59, flags: CJT
  Incoming interface: Serial0, RPF nbr 172.16.253.2
  Outgoing interface list:
    Ethernet0, Forward/Sparse, 00:29:28/00:02:16

(*, 227.0.0.2), 00:26:01/00:02:59, RP 172.16.255.1, flags: SJCF
  Incoming interface: Serial0, RPF nbr 172.16.253.2
  Outgoing interface list:
    Ethernet0, Forward/Sparse, 00:26:01/00:02:14

(172.16.1.10, 227.0.0.2), 00:03:03/00:02:15, flags: CJT
  Incoming interface: Serial0, RPF nbr 172.16.253.2
  Outgoing interface list:
    Ethernet0, Forward/Sparse, 00:03:03/00:02:14

(172.16.2.1, 227.0.0.2), 00:11:11/00:02:14, flags: CJT
  Incoming interface: Serial0, RPF nbr 172.16.253.2
  Outgoing interface list:
    Ethernet0, Forward/Sparse, 00:11:11/00:02:14

(172.16.64.1, 227.0.0.2), 00:06:06/00:02:14, flags: CJT
  Incoming interface: Serial0, RPF nbr 172.16.253.2
  Outgoing interface list:
    Ethernet0, Forward/Sparse, 00:06:06/00:02:14

(172.16.128.1, 227.0.0.2), 00:17:03/00:02:45, flags: CFT
  Incoming interface: Ethernet0, RPF nbr 0.0.0.0
  Outgoing interface list:
    Serial0, Forward/Sparse, 00:17:03/00:03:15

(*, 224.0.1.40), 00:29:32/00:00:00, RP 172.16.255.1, flags: SJPCL
  Incoming interface: Serial0, RPF nbr 172.16.253.2
  Outgoing interface list: Null
```

Nota: El grupo multicast 224.0.1.40 es un grupo bien conocido por todos los enrutadores multicast en la red y es creado por el RP para anunciar los grupos multicast activos en la red.

Como podemos observar en las tablas de enrutamiento multicast de los enrutadores, el tráfico recibido por cada uno de los clientes proviene de la ruta más corta hacia la fuente respectiva.

Anexo C. Configuración de las aplicaciones

En esta parte mostraremos por medio de pantallas la configuración realizada en la aplicación del Administrador de Windows Media. (figura 1 a la 15) posteriormente se mostrara la pantalla de propiedades de la emisora, figura 16 y el control de la emisión, figura 17.

Figura 1 Pantalla inicial (Administrador de Windows Media)

Figura 2 Configuración de una emisora (1)

Figura 3 Configuración de una emisora (2)

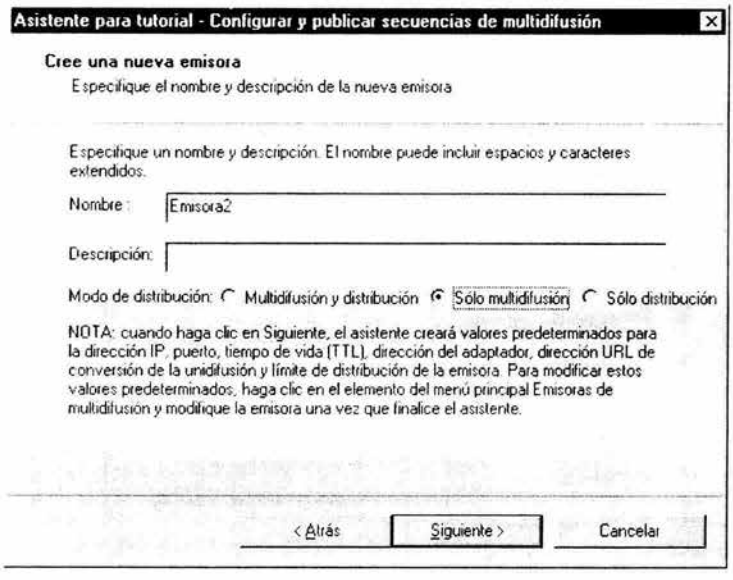


Figura 4 Configuración de una emisora (3)

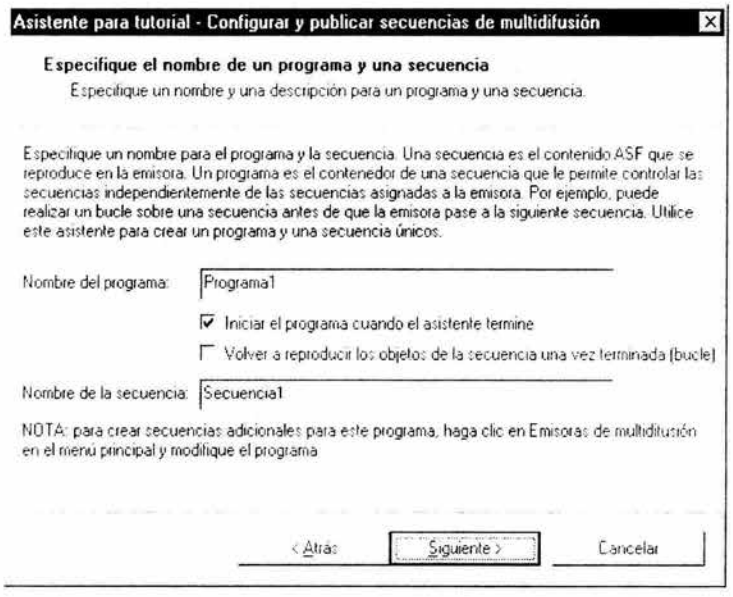


Figura 5 Configuración de una emisora (4)

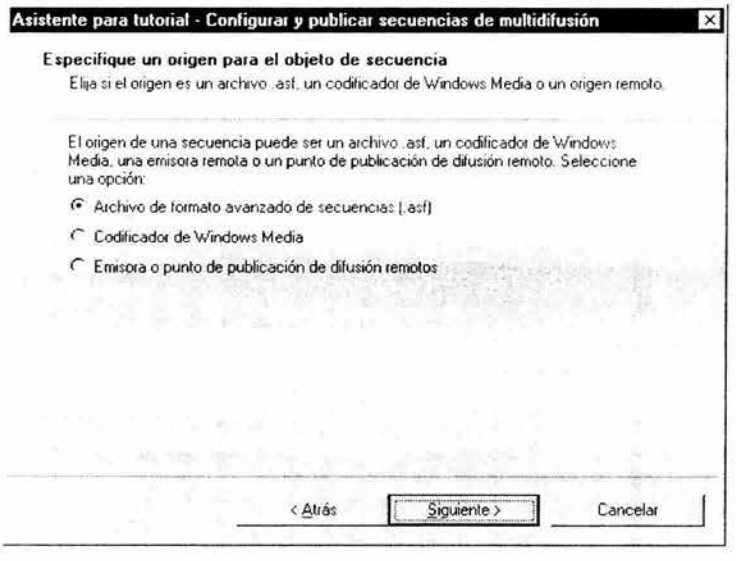


Figura 6 Configuración de una emisora (5)

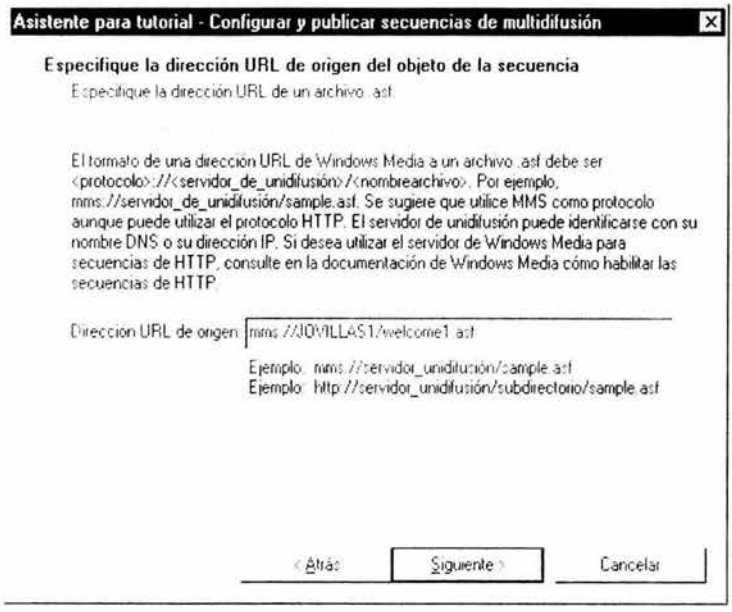


Figura 7Configuración de una emisora (6)

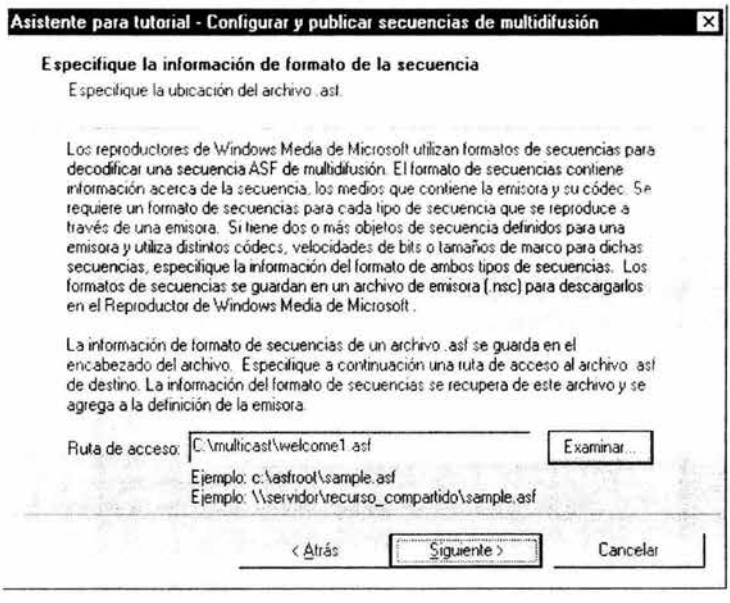


Figura 8 Configuración de una emisora (7)

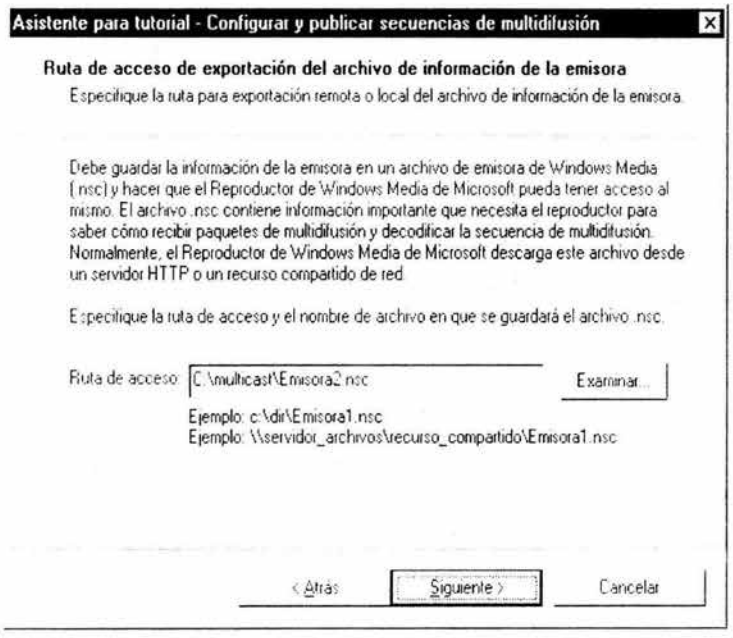


Figura 9 Configuración de una emisora (8)

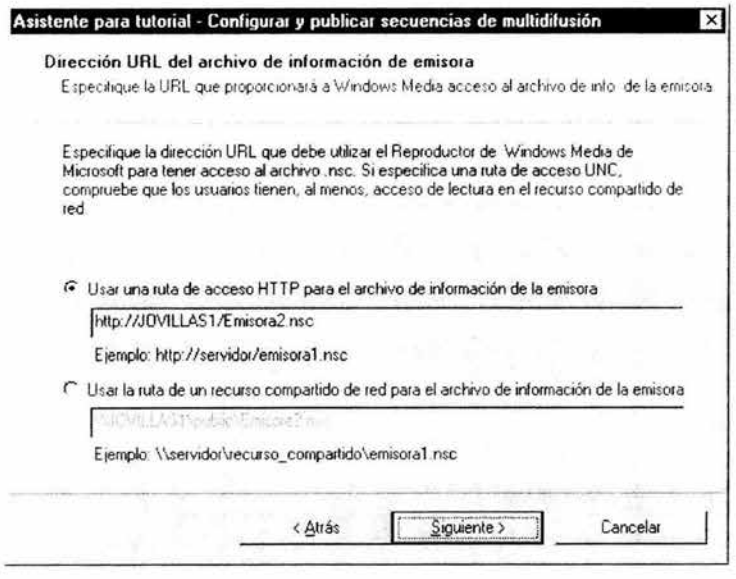


Figura 10 Configuración de una emisora (9)

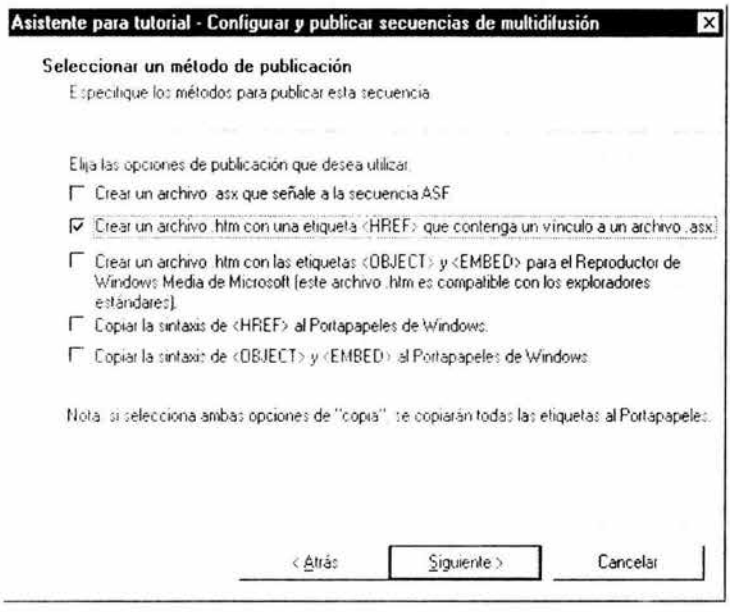


Figura 11 Configuración de una emisora (10)

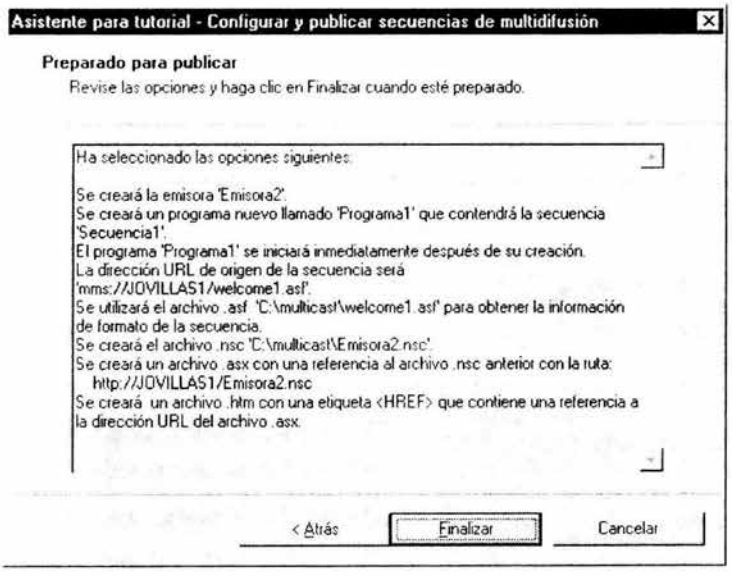


Figura 12 Configuración de una emisora (11)

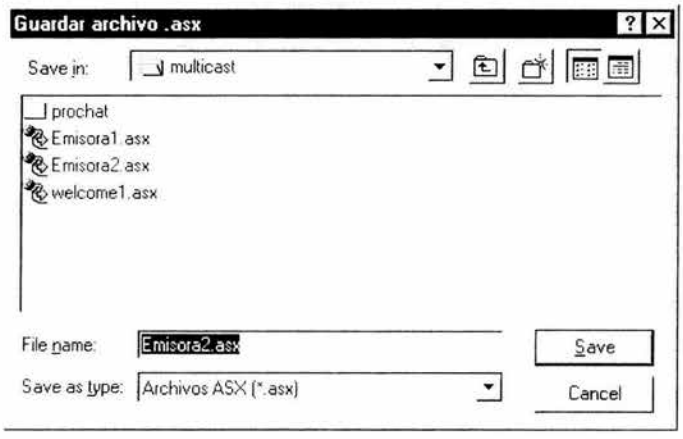


Figura 13 Configuración de una emisora (12)

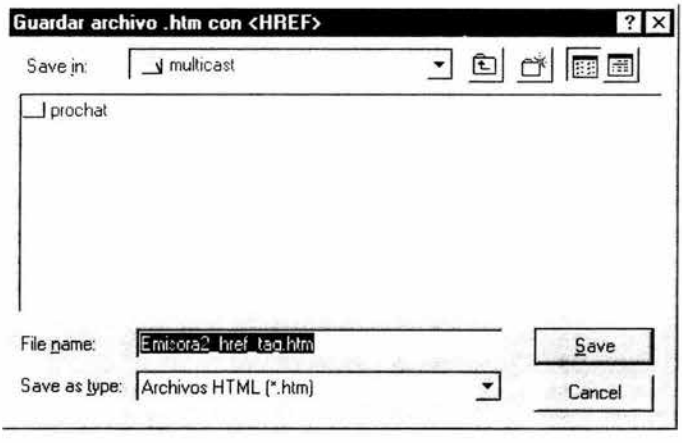


Figura 14 Configuración de una emisora (13)

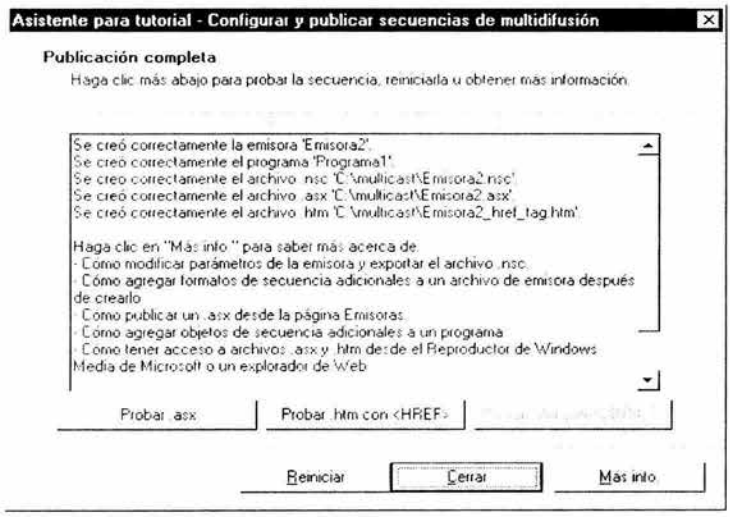


Figura 15 Configuración de una emisora (14)

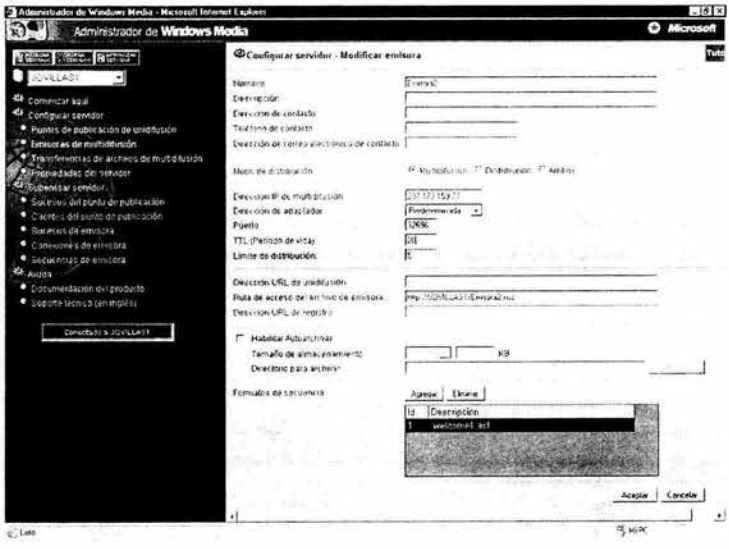


Figura 16 Propiedades de la emisora creada

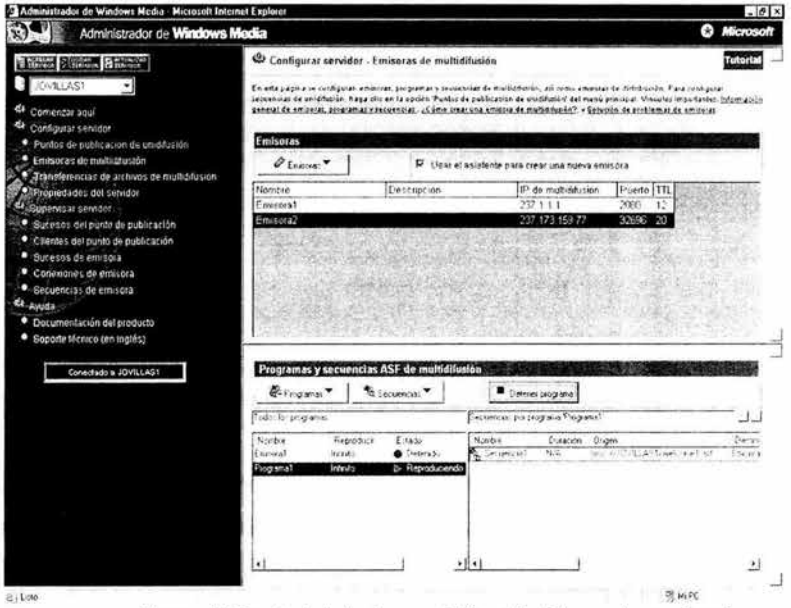


Figura 17 Control de la transmisión del video welcome1.asf

La configuración de la aplicación ProChat se muestra en las figuras 18 y 19

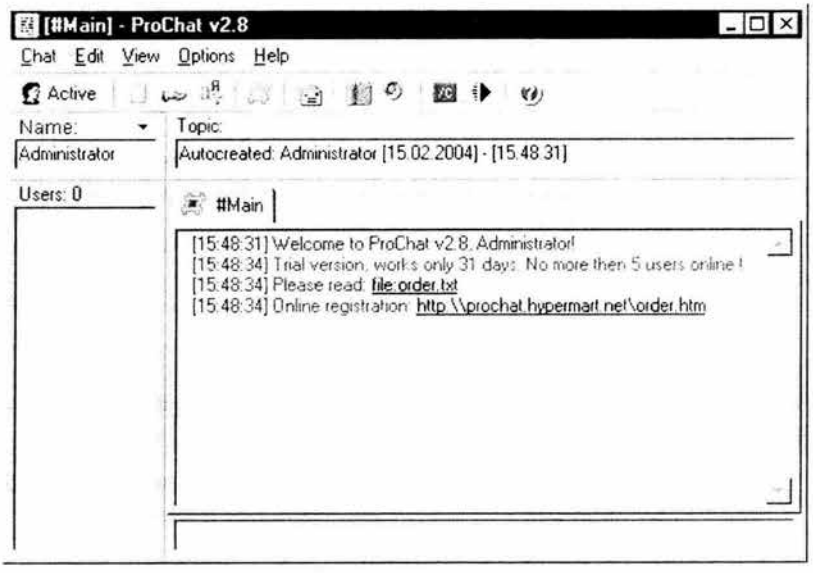


Figura 18 Pantalla inicial (ProChat)

En la figura 18 se muestra la pantalla de inicio, en esta pantalla se selecciona **Options** y después **connection** como se muestra en la figura 19

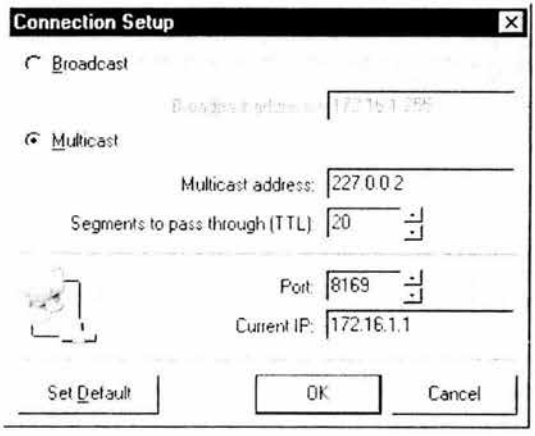


Figura 19 Configuración de multicast en la aplicación

Anexo D. PIM Sparse-Dense Mode

Las versiones IOS más recientes de CISCO soportan un tipo de operación especial del protocolo PIM, que consiste en la posibilidad de utilizar tanto modo esparcido como modo denso (PIM Sparse-Dense Mode), esto dependerá de la presencia o la falta de un punto de reunión (RP), es decir, cuando exista un punto de reunión disponible funcionará el modo esparcido, cuando no en modo denso.

Otra característica importante de este modo de operación es el auto-descubrimiento del punto de reunión. Con esta característica no es necesario configurar el punto de reunión en cada uno de los enrutadores de la red, con lo que la administración se vuelve más sencilla haciendo la solución más escalable.

Esta característica es soportada a partir de la versión IOS 11.1. En la red de la empresa en cuestión existen algunos enrutadores con versiones anteriores a la mencionada lo cual no permitía utilizar esta característica especial. Para remediar esta situación serían necesarios algunos cambios en la red. En algunos enrutadores, por ejemplo, solo sería necesaria una actualización de software, pero en otros casos se requeriría también cambios o actualizaciones de hardware. Debido a esta situación no se propuso originalmente la implementación de *PIM-Sparse-Dense Mode*.

En la red prototipo implementada para la pruebas de Multicast, el enrutador BB3 tenía una versión **10.3.18**, la cual no soporta *PIM Sparse-Dense Mode*.

Suponiendo que todos los enrutadores en la red soportaran la operación de PIM en modo *Sparse-Dense*, a continuación mostraremos las configuraciones necesarias para implementarlo.

Configuración del enrutador BB1, punto de reunión (RP), este equipo anunciará los grupos multicast disponibles en la red a través de un grupo multicast bien conocido 224.0.1.40, con el comando:

ip pim send-rp-announce loopback0 scope 20

En este modo de operación existen otros enrutadores cuya función es la de descubrir que enrutadores son RP, esto lo logra uniéndose al grupo 224.0.1.40 y seleccionar uno de ellos de acuerdo a la dirección *loopback* más alta, utilizando la dirección de grupo 224.0.1.39 para anunciar al RP seleccionado. A estos enrutadores se les denominan *Mapping Agent*, esto se configura con el comando:

ip pim send-rp-discovery scope 20

El Mapping Agent puede ser el mismo RP, como se muestra en la configuración siguiente.

Enrutador BB1

```
!
version 12.0
service timestamps debug datetime localtime
service timestamps log datetime localtime
!
hostname BB1
!
logging buffered 100000 debugging
enable password prov
!
clock timezone CST -6
clock summer-time CDT recurring
ip subnet-zero
no ip domain-lookup
!
ip multicast-routing
!
interface Loopback0
 ip address 172.16.255.1 255.255.255.255
 no ip directed-broadcast
!
interface Serial0/0
 ip address 172.16.251.14 255.255.255.252
 no ip directed-broadcast
 ip pim sparse-dense-mode
!
interface Serial0/1
 ip address 172.16.254.5 255.255.255.252
 no ip directed-broadcast
 ip pim sparse-dense-mode
 clockrate 2000000
!
interface Serial0/2
 ip address 172.16.254.1 255.255.255.252
 no ip directed-broadcast
 ip pim sparse-dense-mode
 clockrate 2000000
!
router eigrp 100
 network 172.16.0.0
!
ip pim send-rp-announce loopback0 scope 20
ip pim send-rp-discovery scope 20
ip classless
!
ntp clock-period 17179634
ntp server 172.16.255.2
end
```

En los demás enrutadores de la red ya no será necesario especificar la dirección del punto de reunión, debido a que lo aprenderán de manera dinámica a través de un grupo multicast bien conocido: 224.0.1.39 generado por el *Mapping Agent*.

Para el resto de los enrutadores solo será necesario configurar las interfaces en el modo sparse-dense como se muestra en el siguiente ejemplo de configuración:

Enrutador no RP

```
!
version 12.0
service timestamps debug datetime localtime
service timestamps log datetime localtime
!
hostname ROUTER
!
logging buffered 1000000 debugging
enable password prov
!
clock timezone CST -6
clock summer-time CDT recurring
ip subnet-zero
no ip domain-lookup
!
ip multicast-routing
!
interface Loopback0
 ip address 172.16.255.2 255.255.255.255
 no ip directed-broadcast
!
interface Ethernet0/0
 bandwidth 2048
 ip address 172.16.254.10 255.255.255.252
 no ip directed-broadcast
 ip pim sparse-dense-mode
 delay 2000
 no fair-queue
!
interface Serial0/0
 ip address 172.16.254.6 255.255.255.252
 no ip directed-broadcast
 ip pim sparse-dense-mode
!
interface Serial1/0
 ip address 172.16.251.18 255.255.255.252
 no ip directed-broadcast
 ip pim sparse-dense-mode
!
router eigrp 100
 network 172.16.0.0
!
ip classless
!
ntp clock-period 17179634
ntp server 172.16.255.2
end
```

Anexo E. Calidad de Servicio en Redes de Datos IP

La Calidad de Servicio (QoS, por sus siglas en inglés) se refiere a la capacidad de la red para ofrecer un mejor servicio a cierto tipo de tráfico, con esto nos referimos al poder ofrecer prioridad incluyendo ancho de banda dedicado, controlando tanto la variación del retardo (*jitter*) y el retardo (parámetros críticos para aplicaciones en tiempo-real, por ejemplo: transmisiones de voz y video), así como reducir la probabilidad de pérdida de paquetes en la red. También es importante asegurar que al ofrecer prioridad a ciertos flujos de tráfico, no se elimine el flujo de otros.

Beneficios de QoS:

- Control y uso eficiente de los recursos: Por ejemplo, se puede controlar el ancho de banda utilizado en un enlace por aplicaciones como el web, o la transferencia de archivos FTP, así como para las aplicaciones críticas. Logrando así asegurar la operación adecuada del negocio.
- Es la base para lograr establecer una red integrada de servicios.

Básicamente existen tres etapas para lograr la implementación de QoS en una red. Para cada una de estas etapas existen diferentes técnicas o estándares definidos.

- Clasificación. Marcado de paquetes con una prioridad específica, indicando el requerimiento del servicio especial de la red. Algunas técnicas de clasificación son por ejemplo:
  - Clase de Servicio (CoS- Class of Service) (802.1D) para Ethernet.
  - Precedencia IP (IP Precedence) o Servicios diferenciados (DSCP) dentro del campo de Tipo de Servicio (ToS) del encabezado IP.
- Encolamiento: Asignación de los paquetes a las colas de despacho (basadas en la clasificación) las cuales tienen un trato especial a través de la red. Algunas técnicas de encolamiento son:
  - PQ – Priority Queuing
  - WFQ – Weighted Fair Queuing
  - CBWFQ – Class Based Weighted Fair Queuing
  - WRED – Weighted Random Early Detection
- Aprovisionamiento. Asignación adecuada de ancho de banda para cada una de las clasificaciones realizadas.

Ubicación del campo **Tipo de Servicio** en el encabezado IP, figura 1.

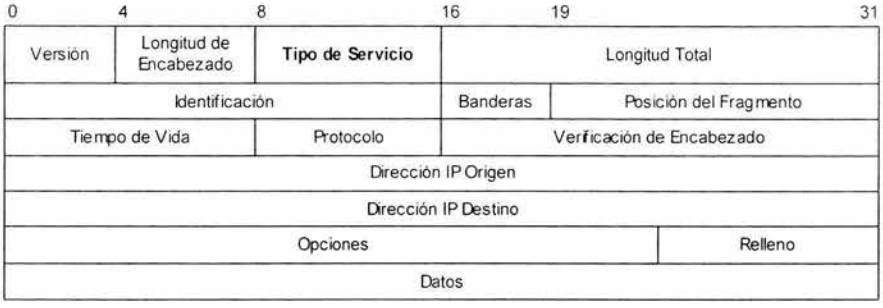


Figura 1. Contenido del encabezado IP

Opciones de *Tipo de Servicio*

- Precedencia IP (RFC 1122) [44]: Utiliza los tres bits más significativos del campo **Tipo de Servicio** (ToS) del encabezado IP. El tráfico puede ser clasificado hasta en 6 clases de servicio (las opciones 6 y 7 son reservadas para uso interno en la red) como se muestra en la figura 2.
  Las técnicas de encolamiento en la red pueden entonces usar este campo para tratar el tráfico de la manera apropiada.

Figura 2. Tipos de Servicio utilizando Precedencia IP

- Servicios Diferenciados (DiffServ) (RFC 2474) [40]: Este es un nuevo modelo que se superpone y a la vez es compatible con **Precedencia IP**. DiffServ utiliza 6 bits, los cuales permiten una clasificación de hasta 64 valores, cada uno de los cuales es llamado Punto de Código de Servicios Diferenciados (DSCP), figura 3

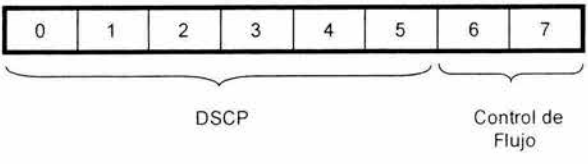


Figura 3 DiffServ

Anexo F. Aplicaciones Multimedia

H.320

El protocolo H.320 [23] define el estándar para videoconferencia sobre *RDSI* y otros medios de transmisión sobre banda estrecha definidos por la *ITU* (Internacional Telecommunications Union). Este protocolo define un "paraguas" que comprenden tres grupos de protocolos, cada uno de los cuales atiende a una necesidad dentro de la videoconferencia, a saber: *H.261* para vídeo, *G.711*, *G.722* y *G.728* para audio y *T.120* para datos.

Arquitectura H.320

H.261 es un formato de compresión de vídeo para ser usado en canales que vayan de 64 Kbits a 2 Mbits. También llamado *px64* donde *p* es un rango comprendido entre 1 y 30 (los múltiplos que puede tener un canal B. Este algoritmo utiliza la codificación tanto intratrama como intertrama. La primera utiliza *DCT* (Discrete Cosine Transform), similar a la utilizada por *JPEG*. La segunda por su parte utiliza un esquema de codificación basado en las diferencias entre bloques. *H.261* define a su vez dos tamaños de ventana *CIF* (Common Intermediate Format) con una resolución de 352 x 288 y *QCIF* (Quarter CIF) con una resolución de 176 x 144.

En cuanto a los protocolos de audio soportados por *H.320* (*G.711*, *G.722* y *G.728*) diseñados para distintas necesidades de audio. *G.711* utiliza la codificación PCM proporcionando calidad de audio a 64 Kbits (en el tramo de 3 KHz). *G.722* es idéntico al anterior pero a 7 Khz. El último utiliza 16 Kbits a 3Khz.

H.221 define la estructura de las tramas para comunicaciones sobre canales de 64 a 2 Mbits. Las tramas tienen un tamaño de 80 bytes de longitud. Cada byte contiene audio, vídeo y datos multiplexados, generados por otros protocolos de la norma *H.320*. Tal como construye la trama la norma *H.221* se puede decir que utiliza un cuarto del canal para audio, otro cuarto para datos y la mitad para la señal de vídeo.

H.231 define el estándar para multipunto y cifrado de datos. Cuando se utiliza la utilidad multipunto entra en juego la MCU (Multipoint Central Unit) . cada uno de los participantes utilizará entonces los protocolos H.242 y H.243 para intercambio de información con ésta. Durante esta comunicación, la MCU guarda los datos acerca de formatos de video (CIF, QCIF), tipo de codificación de audio soportado por cada uno de los clientes. Una vez conseguida esa información, establecerá conexiones con cada uno de ellos de acuerdo a los datos conseguidos.

H.323

El estándar H.323 [24] proporciona una base para las comunicaciones de audio, video y datos a través de una red IP como Internet. Los productos que cumplen con el estándar H.323 pueden ínteroperar con los productos de otros, permitiendo de esta manera que los usuarios puedan comunicarse sin preocuparse con problemas de compatibilidad.

H.323 es un estándar bajo el amparo de la ITU, es un conjunto de estándares para la comunicación multimedia sobre redes que no proporcionan calidad de servicio (QoS). Estas redes son las que predominan hoy en todos los lugares, como redes de paquetes conmutadas TCP/IP e IP sobre Ethernet, Fast Ethernet y Token Ring. Por esto, los estándares H.323 son bloques importantes de construcción para un amplio rango de aplicaciones basadas en redes de paquetes para la comunicación multimedia y el trabajo colaborativo.

El estándar tiene amplitud e incluye desde dispositivos específicos hasta tecnologías embebidas en ordenadores personales, además de servir para comunicación punto-punto o conferencias multi-punto. H.323 habla también sobre control de llamadas, gestión multimedia y gestión de ancho de banda, además de los interfaces entre redes de paquetes y otras redes (RTC p.e.)

H.323 forma parte de una gran serie de estándares que permiten la videoconferencia a través de redes. Conocidos como H.32X, esta serie incluye H.320 y H.324, que permiten las comunicaciones RDSI y RTC respectivamente.

Arquitectura H.323

La Recomendación H.323 cubre los requerimientos técnicos para los servicios de comunicaciones entre Redes Basadas en Paquetes (PBN) que pueden no proporcionar calidad de servicio (QoS). Estas redes de paquetes pueden incluir Redes de Área Local (LAN's), Redes de Área Extensa (WAN), Intra-Networks y Inter-Networks (incluyendo Internet). También incluye conexiones telefónicas o punto a punto sobre RTC o ISDN que usan debajo un transporte basado en paquetes como PPP. Esas redes pueden consistir de un segmento de red sencillo, o pueden tener topologías complejas que pueden incorporar muchos segmentos de red interconectados por otros enlaces de comunicación.

La recomendación describe los componentes de un sistema H.323, estos son: Terminales, Gateways, Gatekeepers, Controladores Multipunto (MC), Procesadores Multipunto (MP) y Unidades de Control Multipunto (MCU)

Terminales

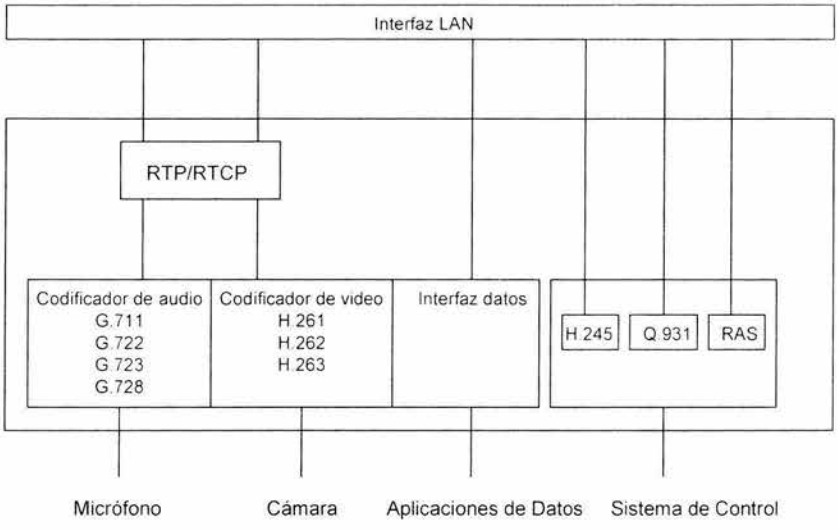


Figura 1. Diagrama de las terminales H323

Los terminales son puntos finales de la comunicación. Proporcionan comunicación en tiempo real bidireccional. Los componentes de un terminal se pueden ver en la figura 1.

Para permitir que cualesquiera terminales inter operen se define que todos tienen que tener un mínimo denominador que es, soportar voz y con un codec G.711. De esta manera el soporte para video y datos es opcional para un terminal H.323.

Todos los terminales deben soportar H.245, el cual es usado para negociar el uso del canal y las capacidades. Otros tres componentes requeridos son: Q.931 para señalización de llamada y configuración de llamada, un componente llamado RAS (Registrantion/Admisión/Status), este es un protocolo usado para comunicar con el Gatekeeper; y soporte para RTP/RTCP para secuenciar paquetes de audio y video.

Otros componentes opcionales de los terminales H.323 son: los codec de video, los protocolos T.120 para datos y las capacidades MCU.

Gateways

El Gateway (o Pasarela) es un elemento opcional de una conferencia H.323. Es necesario solo si necesitamos comunicar con un terminal que está en otra red (por ejemplo RTC), como se muestra en la figura 2. Los Gateways proporcionan muchos servicios, el más común es la traducción entre formatos de transmisión (por ejemplo H.225.0 a H.221) y entre procedimientos de comunicación (por ejemplo H.245 a H.242). Además el Gateway también traduce entre los codecs de video y audio usados en ambas

redes y procesa la configuración de la llamada y limpieza de ambos lados de la comunicación.

El Gateway es un tipo particular de terminal y es una entidad llamable (tiene una dirección).

En general, el propósito del Gateway es reflejar las características del terminal en la red basada en paquetes en la terminal en la Red de Circuitos Conmutados (SCN) y al contrario. Las principales aplicaciones de los Gateways son:

- Establece enlaces con terminales telefónicos analógicos conectados a la RTB (Red Telefónica Básica)
- Establecer enlaces con terminales remotos que cumple H.320 sobre redes RDSI basadas en circuitos conmutados (SCN)
- Establecer enlaces con terminales remotos que cumple H.324 sobre red telefónica básica (RTB)

Los Gateways no se necesitan si las conexiones son entre redes basadas en paquetes.

Muchas funciones del Gateway son dejadas al diseñador. Por ejemplo, el número de terminales H.323 que pueden comunicar a través del Gateway no es asunto de estandarización. De la misma manera el número e conexiones con la SCN, el número de conferencias individuales soportadas, las funciones de conversión de audio/video/datos, y la inclusión de funciones multipuntos son dejadas al diseñador. Debido a la incorporación de los Gateways a la especificación H.323, la ITU posicionó H.323 como el pegamento que junta todos los terminales para conferencias funcionando juntos.

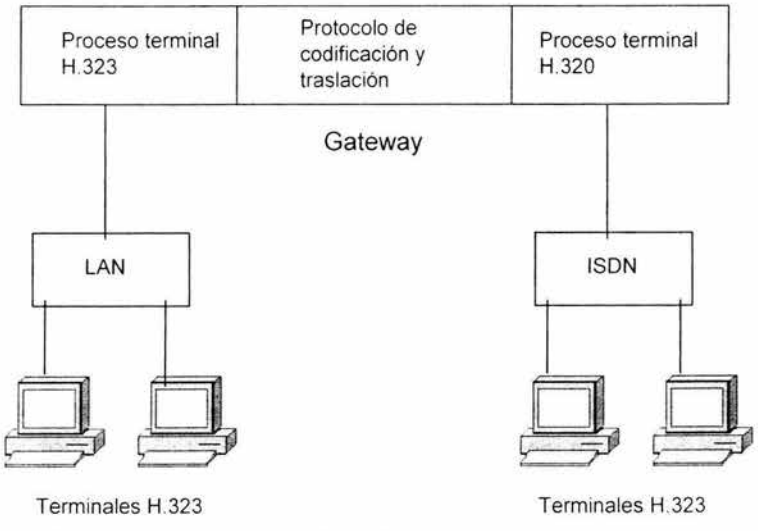


Figura 2. Diagrama del funcionamiento del Gateway para H.323

Gatekeepers

Son un elemento opcional en la comunicación entre terminales H.323. No obstante, son el elemento más importante de una red H.323. Actúan como punto central de todas las llamadas dentro de una zona y proporcionan servicios a los terminales registrados y control de las llamadas. De alguna forma, el gatekeeper H.323 actúa como un conmutador virtual.

Los Gatekeepers proporcionan dos importantes funciones de control de llamada:

- Traducción de direcciones desde alias de la red H.323 a direcciones IP o IPX, tal y como está especificado en RAS.
- Gestión de ancho de banda, también especificado en RAS. Por ejemplo, si un administrador de red ha especificado un umbral para el número de conferencias simultáneas, el Gatekeeper puede rechazar hacer más conexiones cuando se ha alcanzado dicho umbral. El efecto es limitar el ancho de banda total de las conferencias a alguna fracción del total existente para permitir que la capacidad remanente se use para e-mail, transferencias de archivos y otros protocolos.

A la colección de todos los Terminales, Gateways y MCU's gestionados por un gatekeeper se la conoce como Zona H.323. Ver figura.

Una característica opcional, pero valiosa de los gatekeepers es la habilidad para enrutar llamadas. Si se enruta la llamada por un gatekeeper, esta puede ser controlada más efectivamente. Los proveedores de servicio necesitan esta característica para facturar por las llamadas realizadas a través de su red. Este servicio también puede ser usado para re-enrutar una llamada a otro terminal en caso de estar no disponible el llamado. Además con esta característica un gatekeeper puede tomar decisiones que involucren el balanceo entre varios gateways. Por ejemplo, si una llamada es enrutada por un gatekeeper, ese gatekeeper puede re-enrutar la llamada a uno de varios gateways basándose en alguna lógica de enrutamiento propietaria.

Mientras que un Gatekeeper está lógicamente separado de los extremos de una conferencia H.323, los fabricantes pueden elegir incorporar la funcionalidad del Gatekeeper dentro de la implementación física de Gateways y MCU's.

A pesar de que el Gatekeeper no es un elemento obligatorio, si existe, los terminales deben usarlo. RAS define para estos la traducción de direcciones, control de admisión, control de ancho de banda y gestión de zonas.

Los Gatekeepers juegan también un rol en las conexiones multipunto. Para soportar conferencias multipunto, los usuarios podrían emplear un Gatekeeper para recibir los canales de control H.245 desde dos terminales en una conferencia punto-punto. Cuando la conferencia cambia a multipunto, el Gatekeeper puede redireccionar el Canal de Control H.245 a un controlador multipunto, el MC. El Gatekeeper no necesita procesar la señalización H.245, solo necesita pasarla entre los terminales o entre los terminales y el MC.

Las redes que posean un Gateway pueden también tener un Gatekeeper para traducir llamadas entrantes E.164 (número de teléfono convencionales) a direcciones de

transporte. Debido a que una Zona está definida por su Gatekeeper, las entidad H.323 que contengan un Gatekeeper interno necesitan de un mecanismo para desactivar su funcionamiento cuando hay varias entidades H.323 que contiene un Gatekeeper dentro de la red, las entidades pueden ser configuradas para estar en la misma Zona.

Existen dos formas para que un terminal se registre en un gatekeeper, sabiendo su ip y enviando entonces un mensaje de registro unicast a esta dirección o bien enviando un mensaje multicast de descubrimiento del gatekeeper (GRQ) que pregunta ¿quién es mi gatekeeper?

Funciones obligatorias Gatekeeper

- Traducción de Direcciones: Traducción de alias a direcciones de transporte, usando para ello una tabla que es modificada con mensajes de Registration. Se permiten otros métodos de modificar la tabla.
- Control de Admisión: El Gatekeeper debería autorizar el acceso a la red usando mensajes H.225.0 ARQ/ACF/ARJ. Esto puede basarse en autorización de llamada, ancho de banda, o algún otro criterio que es dejado al fabricante. También puede ser una función nula que admita todas las peticiones.
- Control de Ancho de Banda: El Gatekeeper debería soportar mensajes BRQ/BRJ/BCF. Esto puede usarse para gestión del ancho de banda. También se puede aceptar todas las peticiones de ancho de banda.
- Gestión de Zona: El Gatekeeper debería suministrar la funciones anteriores a: todos los terminales, MCU's y Gateways que se encuentren registrados en su Zona de control.

Funciones opcionales del Gatekeeper

- Señalización de control de llamada: El Gatekeeper puede elegir completar la señalización de llamada con los extremos y procesar la señalización de llamada el mismo. Alternativamente, puede elegir que los extremos conecten directamente sus señalizaciones de llamada. De esta manera el Gatekeeper puede evitar gestionar las señales de control H.225.0.
- Autorización de llamada: El Gatekeeper puede rechazar una llamada desde un terminal basándose en la especificación Q.931. (H.225.0) Las razones para rechazar la llamada pueden ser, pero no están limitadas a, acceso restringido desde o hacia un terminal particular o Gateway, y acceso restringido durante un periodo de tiempo. El criterio para determinar si se pasa la autorización o falla, está fuera del alcance de H.323.
- Gestión de llamada: El Gatekeeper puede mantener una lista de las llamadas en curso, esta información puede ser usada para indicar si un terminal está ocupado o para dar información a la función de gestión de ancho de banda.
- Otros como: estructura de datos de información para la gestión, reserva de ancho de banda y servicios de directorio.

Unidades Control Multipunto (MCU)

La MCU soporta conferencias entre tres o más extremos. En terminología H.323, el MCU se compone de: Controlador Multipunto (MC) que es obligatorio, y cero o más Procesadores Multipunto (MP). El MC gestiona las negociaciones H.245 entre todos los terminales para determinar las capacidades comunes para el procesado de audio y video. El MC también controla los recursos de la conferencia para determinar cuales de los flujos, si hay alguno, serán multicast. Las capacidades son enviadas por el MC a todos los extremos en la conferencia indicando los modos en los que pueden transmitir. El conjunto de capacidades puede variar como resultado de la incorporación o salida de terminales de la conferencia.

El MC no trata directamente con ningún flujo de datos, audio o video. Esto se lo deja a el MP, este mezcla, conmuta y procesa audio, video y/o bits de datos. Las capacidades del MC y MP pueden estar implementadas en un componente dedicado o ser parte de otros componentes H.323, en concreto puede ser parte de un Gatekeeper, un Gateway, un terminal o una MCU.

El MP recibe flujos de audio, video o datos desde los extremos, estos pueden estar involucrados en una conferencia centralizada, descentralizada o híbrida. El MP procesa esos flujos y los devuelve a los extremos.

La comunicación entre el MC y el MP no es asunto de estandarización.

Conferencias Multipunto

Existen una variedad de métodos de gestionar las conferencias multipunto. La Recomendación hace uso de los conceptos de conferencia centralizada y descentralizada.

La conferencias centralizadas requieren de una MCU. Todos los terminales envían audio, video, datos y flujos de control a la MCU en un comportamiento punto-punto. La MC gestiona de forma centralizada la conferencia usando las funciones de control H.245 que también definen las capacidades de cada terminal. El MP mezcla el audio, distribuye los datos y mezcla/conmuta el video y envía los resultados en flujos de vuelta a cada terminal participante.

En conferencia multipunto descentralizadas se puede hacer uso de tecnología multicast. Los terminales H.323 participantes envían audio y video a otros terminales participantes sin enviar los datos a una MCU. Sin embargo el control de los datos multipunto sigue siendo procesado de forma centralizada por la MCU, y la información del canal de control H.245 sigue siendo transmitida de modo unicast a un MC.

Son los terminales que reciben múltiples flujos de audio y video los responsables de procesarlos. Los terminales usan los canales de control H.245 para indicar a un MC cuantos flujos simultáneos de video y audio son capaces de decodificar. El número de capacidades simultáneas de un terminal no limita el número de flujos de audio y video que son enviados por multicast en una conferencia.

Las conferencias multipunto híbridas usan una combinación de características de las centralizadas y descentralizadas. Las señalizaciones y cualquier flujo de audio o video es, procesado a través de mensajes punto a punto enviados a la MCU. Las restantes señales (audio o video) son enviadas a los participantes a través de multicast.

Una ventaja de las conferencias centralizadas es que todos los terminales soportan comunicaciones punto a punto. La MCU puede sacar varios flujos unicast a los participantes y no se requiere ninguna capacidad de la red especial. También es posible que la MCU reciba varios flujos unicast, mezcle el audio, y conmute el video, y saque un flujo multicast, conservando de esta manera el ancho de banda de la red.

H.323 también soporta conferencias multipunto mixtas en las cuales algunos terminales están en una conferencia centralizada, mientras otros están en una descentralizada, y una MCU proporciona el puente entre los dos tipos. Al terminal le es transparente la naturaleza mixta de la conferencia, solo tiene en cuenta el modo en que envía o recibe.

Multicast hace más eficiente el uso del ancho de banda de la red, pero supone una más alta carga computacional en los terminales que tienen que mezclar y conmutar entre los flujos de audio y video que reciben. Además, el soporte multicast es necesario en elementos de la red como routers y switches.

Un MC puede estar localizado en un Gatekeeper, un Gateway, un terminal o una MCU. En la figura 3 se muestra el área determinada para H323

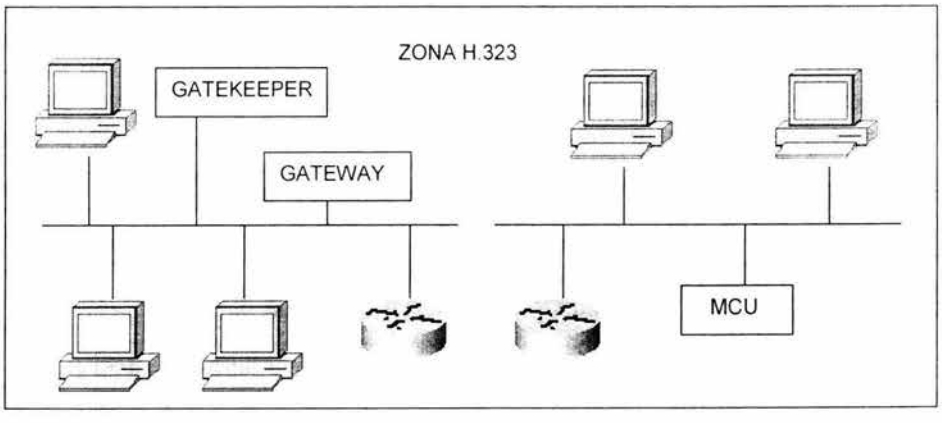


Figura 3. Zona que conforma H.323

Videoconferencia multicast

Se describe en este apartado las herramientas necesarias para la realización de videoconferencias (multiconferencias) utilizando la tecnología multicast. Es importante señalar que no sólo estamos hablando de utilidades de vídeo o audio, sino de un conjunto de herramientas que aportan entre otras cosas pizarra electrónica, WWW multicast, editores compartidos, a continuación se presentan algunas de estas:

- **vic**: Se utiliza para recibir/emitir vídeo.
- **vat**: Idem para audio.
- **rat**: Idem para audio.
- **nt**: Editor compartido.
- **wb**: Pizarra compartida.
- **sdr**: Directorio de sesiones.

Es interesante, además, instalar herramientas de control/calidad de la red multicast. Como herramienta para controlar la calidad de nuestra red multicast se recomienda utilizar ***beacon***. Es una utilidad cliente/servidor basada en JAVA que nos ofrece una serie de datos sobre distintas medidas (perdidas, retardo, *jitter*,...). Es necesario ejecutar un cliente y conectarse a un servidor instalado en el MBone.

Existen algunos paquetes que intentan facilitar el uso de estas herramientas. La idea es integrarlas bajo una única interfaz. Como ejemplo podemos mencionar ***ReLaTE*** y ***Deta***[1]

Streaming

Se entiende por *streaming* la capacidad de distribución de contenido multimedia, con la característica de poder visualizar estos contenidos mientras esa información está siendo trasmitida por la red. Este sistema tiene la ventaja frente al sistema existente anteriormente, (era necesario bajar completamente el vídeo para comenzar su visualización) pero necesita que tanto el servidor de vídeo como las redes de datos sean capaces de mantener un flujo constante de esa información. Básicamente cualquier sistema de *streaming* estará formado por: Compresión, transmisión y *buffering*.

Las configuraciones de este tipo de servicios, pueden variar desde un número pequeño de usuarios y contenidos, hasta un gran número de ellos y mucho volumen de información. Evidentemente, el planteamiento será distinto según la solución a tratar. Aquí la red de comunicaciones también es importante, tanto en su ancho de banda como en la posibilidad de manejar tráfico multicast.

A la hora de dimensionar una solución de *streaming*, es necesario tener en cuenta varios factores: número de usuarios simultáneos, número de horas de almacenamiento y cómo se va a utilizar el servicio. Cuando el almacenamiento crece, es necesario plantearse el

incorporar a la solución un sistema, tanto de catalogación, como de recuperación de esa información. Así como un sistema que permita manejar todo ese volumen de datos.

Además, es importante tener en cuenta otros factores como: formatos de vídeo soportados por la servidor de *streaming* (Real, MPEG-1, MPEG-2, MPEG-4, etc), protocolos utilizados para el transporte (RTP, RTSP, soluciones propietarias), y soporte de multicast.

En relación a los servidores de *streaming* se pueden dividir en dos tipos: *Intranet video servers* e *Internet video servers*

Los primeros, suelen utilizar formatos de vídeo de más calidad y más ancho de banda. Aquí estamos hablando de *MPEG-1* (1 a 3 Mbits) con una calidad similar al *VHS*, *MPEG-2* (3 a 10 Mbits) con calidad *DVD*. Además necesitaremos las tarjetas capturadoras (encoders) que nos generen esta salida. Aquí el precio variará dependiendo del formato que hayamos elegido. La elección de un formato u otro dependerá de varios factores: uso que se la dará a esos contenidos, disponibilidad de ancho de banda en la red, calidad y tipo del material (master), lugar de visualización de contenidos.

Como ejemplo de ancho de banda/calidad diremos que una ventana pequeña con calidad VHS se puede estar hablando de (80 Kbits para una solución propietaria y hasta 0.5 Mbits para MPEG-1). Una pantalla grande con calidad S-VHS necesitaremos de 1 a 2 Mbits para MPEG-1. Para una mayor calidad como Betacam-SP hablaremos ya de 3 Mbits en MPEG-1 a 5 Mbits en MPEG-2. Evidentemente con este volumen de información tenemos que plantearnos la elección de un sistema de almacenamiento acorde a nuestras necesidades.

El segundo tipo de servidores se utilizarán para dar servicio a Internet. Aquí los anchos de banda (a día de hoy) son considerablemente menores. Se está hablando de velocidad que va desde los 28-56 Kbits, a 128 o 384 kbits. Con formatos como *Realvideo* o *ASF* (Advanced Stream Format). Evidentemente, las necesidades de almacenamiento cambian radicalmente con el primer tipo de servidores, y los precios de la solución final (incluyendo estaciones de codificación) también.

En cuanto a la forma de trabajar y planificar el trabajo, es independiente del tipo de servidor que tengamos (internet o intranet). En las tablas 1 y 2 se pueden ver las distintas formas de planificación existentes.

Por último, como elementos a resaltar en una solución de *streaming* podemos citar:

- **Servidor de streaming**: estará formado por el software de *streaming* y sistema de catalogación en caso de ser necesario.
- **Estación de captura** (codificación): nos permite realizar la captura de la señales de audio y vídeo para trabajar de cualquiera de las formas vistas anteriormente.
- **Sistema de almacenamiento**: se dimensionará según las necesidades.
- **Herramientas de producción**: En algunos casos puede ser interesante disponer de software adicional que nos permita generar contenidos más elaborados.

BAJO DEMANDA	DIRECTO	DIFERIDO
Se puede acceder en cualquier momento.	Sólo se puede ver cuando se emite.	Igual que en directo
Ficheros almacenados en el servidor.	No están guardados en el servidor.	Similar a los de bajo demanda.
Vemos la emisión desde el principio.	Todo el mundo ve la misma parte de la emisión al mismo tiempo.	Igual que en directo.
Podemos hacer pausa, rebobinar,..	No podemos interactuar con la emisión.	Igual que en directo.

Tabla 1 Planificaciones (1)

	BAJO DEMANDA	DIRECTO	DIFERIDO
Emisión	Grabado previamente.	En directo.	En diferido
Acceso	Se puede acceder en cualquier momento. Rebobinar, pausa, avanzar.	Sólo cuando se emite.	Sólo cuando se emite.
Método	Streaming, Spliting (con algunas soluciones)	Unicast, Spliting (división de la carga entre varios servidores) multicast	Unicast, Spliting, multicast
Ámbito	- Número limitado de usuarios. - Necesita ancho de banda (dependiendo calidad y número).	En unicast número limitado de usuarios	- En multicast número ilimitado de usuarios. - La red debe soportar multicast.

Tabla 2 Planificación (2)

Real

Bajo este epígrafe se describe la solución que ofrece *RealNetworks* para la producción y distribución de contenido multimedia.

RealServer: Es el software de Real para la distribución de contenidos vía *streaming* tanto en directo como en diferido. El software es multiplataforma y puede ser instalado tanto en *Windows* como en sistemas *Unix* (*Linux* y *Solaris, HP-UX, IRIX,*). Se puede instalar una versión de demo que permite la conexión de hasta 25 usuarios simultáneos. Esta solución viene limitada en algunas de sus funcionalidades (*mutlicast* y *splitting*).

Como herramientas para producir podemos citar entre otras: ***RealProducer*** (codificación en formato RealVideo tanto en directo como en diferido), ***RealPresenter*** (para realizar

presentaciones de audio, vídeo con PowerPoint y WWW), ***RealSlideShow*** (presentaciones con gráficos y audio).

Como características de este conjunto de software se puede resaltar:

- Facilidad de instalación y configuración.
- Multiplataforma en Servidor, codificación y visualizadores.
- Administración y configuración vía WWW.
- Se apoya en estándares (SMIL, RTP/RTCP, RTSP,..).
- Soporte multicast (real y simulado).
- Soporta protocolos de sesión como SAP/SDP para su comunicación con las herramientas típicas de videoconferencia multicast (sdr).
- Herramientas de control y estadísticas de uso (JAVA).
- Control de acceso (por dirección IP).
- Soporta formatos como: AVI, MOV, QT, MP3, RealVideo8, MPEG-1, MPEG-2

Windows Media

Es la tecnología perteneciente a Microsoft engloba una serie de herramientas para la generación de elementos audiovisuales y su difusión por intranets o internet.

Soporta video bajo de manda (VoD), emisión en vivo y programada. La emisión puede ser entregada de múltiples formas al receptor:

- Multicast: si la red lo soporta y es un contenido en vivo o diferido.
- UDP: si no soporta multicast.
- TCP: si los puertos UDP están filtrados
- HTTP: Si las conexiones TCP están filtradas puede ser entregado por HTTP, a través de proxies. Esta forma de entrega es la menos eficiente pero en muchas empresas todo lo demás está cortado.

Windows Media es una solución propietaria que aporta sus propios protocolos. La conexión entre cliente y servidor se negocia usando el protocolo MMS (Multi Media Server) o también se puede hacer streaming sobre HTTP. MMS funciona encima de TCP, usa el puerto 1755 y con él se negocian las características de velocidad de la conexión así como el modo de entrega que en particular puede ser sobre HTTP. Los servidores que sirven contenidos de Windows Media usan urls del tipo MMS://, o bien MMSU:// (para forzar UDP) o bien MMST:// (para forzar TCP). La negociación de la forma de entrega es la siguiente: primero se intenta Multicast (sí el servidor está configurado para hacerlo), después UDP, TCP y HTTP. El cliente puede forzar un modo si quiere en Herramientas->Opciones->Red.

Los contenidos de Windows Media pueden ser difundidos desde un servidor web por HTTP. Comparado con un servidor Web, el Windows Media Service aporta varias ventajas:

- Uso más eficiente del ancho de banda. El Windows Media Service puede hacer uso como ya se indicó de varios transportes que hacen más eficiente la entrega.
- Mejor calidad para el usuario
- Envió de flujos multistream. Para que según el ancho de banda del cliente se envié con distintas calidades.
- Protección de contenidos con copyright
- Escalabilidad. Soporta más clientes.
- Control del ancho de banda en uso.

Por lo que se conoce, MMS es equivalente a RTP, RTCP y RTSP del IETF.

Los formatos de archivo propios de Windows Media son asf, wmv y wma. En realidad los archivos wmv y wma (video y audio respectivamente) pueden ser renombrados a asf ya que su estructura es idéntica. En ciertos sitios puede demandarse un archivo de tipo asf y nosotros tener uno codificado que sea wmv por ejemplo, en cuyo caso solo lo tenemos que renombrar a asf.

Una de las ventajas de esta solución de streaming frente a otros productos es la calidad de sus CODEC. Actualmente se soportan los siguientes CODEC para video: MPEG4 v3 (no cumple el estándar), MPEG4 ISO, Windows Media V7, Windows Media V8.

Otro punto fuerte a favor de esta solución es el precio, Microsoft lo distribuye desde su Web de forma gratuita.

Codificación

Para la generación de archivos en formatos asf, wmv y wma se pueden utilizar dos soluciones:

- Batch Encode Utility: para generar archivos asf, wmv y wma a partir de archivos en otros formatos (por ejemplo avi, mp3, wav, mpg, etc.
- Windows Media Encoder: Se puede usar para lo mismo que el anterior pero además permite tomar como fuente una captura audio y/o video que es codificada en tiempo real. En este se pueden especificar varias fuentes (por ejemplo un video de entrada, un anuncio, un gráfico, etc) e ir pasando de unas a otras. Se pueden generar eventos que son generados en el reproductor del cliente. El contenido que está siendo generado puede almacenarse en un archivo, difundirse a Media Players por HTTP o bien difundirse a un Windows Media Services también por HTTP y que sea este el que lo distribuya a los clientes. No se aconseja más de 30 clientes conectados directamente al Windows Media Encoder. El inconveniente principal de conectar de esta forma a los clientes es que sólo se pueden entregar los contenidos por HTTP.

Difusión

Para la difusión del contenido multimedia como ya se dijo además de un servidor web se pueden utilizar los Windows Media Services. Estos son administrados desde una interface web simple.

Una vez instalado el servicio, que puede funcionar tanto en NT como en Windows 2000, habrá que instalar los puntos de publicación de unidifusión a petición. Esto es equivalente a dar la raíz en un servidor WEB. Una vez hecho esto cualquier archivo archivo asf colocado en dicho directorio será servido bajo demanda usando la sintaxis mms://nombreservidor/archivo.asf. También se puede dar un alias al punto de montaje y entonces sería mms://nombreservidor/alias/archivo.asf. Esta url la entienden Explorer o Media Player, en el caso del Explorer lanza el Media Player para que la use.

Con lo explicado en el párrafo anterior tendríamos video bajo demanda. Otras posibilidades son: emisora difundiendo contenidos en vivo (desde un Encoder) o contenidos grabados, ambas distribuidas por unicast o multicast. Todas ellas se encuentran extensamente documentadas en la ayuda.

Puesto que una url de tipo mms:// solo es entendida por Internet Explorer, se usa un formato de archivo denominado asx que además sirve para definir el comportamiento del Media Player. Este archivo se baja por HTTP y está asociado al Media Player. Dentro de este archivo se encuentra la url mms:// o http:// del contenido a visualizar, también se pueden poner banners, y metainformación sobre el contenido en emisión, además de listas de reproducción y más características interesantes.

El video puede ser visualizado embebido dentro de una página web en cuyo caso la apariencia del video puede ser cambiada, por ejemplo con botones o sin ellos, etc.

Referencias

[1] A. Ballardie, "Core Based Trees (CBT) Multicast Routing Arquitecture", RFC 2201. Septiembre, 1997.

[2] American National Standards Institute "Digital Hierarchy - Electrical Interfaces", Article: ANSI T1.102-1993 (R1999)

[3] American National Standards Institute "Network and customer Installation Interfaces-DS1- Electrical Interface" , Article:ANSI T1.403-1999.

[4] American National Standards Institute, SONET (Synchronous Optical Network). 1988.

[5] American National Standards Institute, T1.413 "Asimetric Digital Subscriber Line (ADSL) metallic interface". 1995.

International Telecommunication Union, G.992.1 "Asimetric Digital Subscriber Line (ADSL) transceivers". 1999.

[6] C. Hedrick, "Routing Information Protocol", RFC 1058. Rutgers University. Junio, 1988.

[7] Cable Television Laboratories, Inc. "Data-Over-Cable Service Interface Specifications DOCSIS 1.1 Radio Frequency Interface Specification" 2003

[8] D. Estrin, D. Farinacci, A. Helmy, D. Thaler, S. Deering, M. Handley, V. Jacobson, C. Liu, P. Sharma y L. Wei, "Protocol Independent Multicast-Sparse Mode (PIM-SM): Protocol specification", RFC 2117. Junio, 1997.

[9] D. Estrin, D. Farinacci, V. Jacobson, C. Liu, L. Wei, P. Sharma y A. Helmy, "Protocol Independent Multicast-Dense Mode (PIM-DM): Protocol specification". 1996

[10] D. Farinacci y A. Tweedly, "Cisco Group Management Protocol", Cisco Systems. 1996

[11] D. Perkins, "Point-to-Point Protocol: A Proposal for Multi-Protocol Transmission of Datagrams Over Point-to-Point Links", RFC 1134, CMU. Noviembre, 1989.

[12] D. Waitzman, C. Partridge y S. Deering, "Distance Vector Multicast Routing Protocol", RFC 1075, BBN STC - Standford University. Noviembre, 1988.

[13] EIGRP desarrollado por Cisco Systems.

http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/en_igrp.htm

[14] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", RFC 1889. GMD Fokus, Precept Software, Inc. XPARC, Lawrence Berkeley National Laboratory. Enero, 1996.

[15] IEEE 802.16 Working Group on Broadband Wireless Access Standards. "Broadband Wireless Metropolitan Area Networks. 802.16", Diciembre 2001

[16] IEEE 802.3 "CSMA/CD Access Method and Physical Layer Specifications". 1980.

[17] International Standards Organization, ISO/IEC 13239 "Telecommunications and Information Exchange Between Systems - High-level Data Link Control (HDLC) procedures".

[18] International Standards Organization, ISO/IEC 7776 "Telecommunications and information exchange between systems - High-level Data Link Control procedures -- Description of the X.25 LAPB-compatible DTE data link procedures". 1995

[19] International Standards Organization, ISO/IEC 8473 "Protocol for Providing the Connectionless Network Service, Edition 2", 1994.

[20] International Telecommunication Union, CCITT I.120 "Recommendations of ISDN". 1984.

[21] International Telecommunication Union, CCITT I.363, "B-ISDN ATM Adaptation Layer specification", CCITT Study Group XVIII. Enero, 1993.

[22] International Telecommunication Union, CCITT Recommendation X.25, 1976.

[23] International Telecommunication Union, H.320 "Narrow-band visual telephone systems and terminal equipment". Mayo, 1999

[24] International Telecommunication Union, H.323 "Packet-based multimedia communications systems". Febrero, 1998.

[25] International Telecommunication Union, Q.920/921 LAPD "ISDN user-network interface data link layer protocol", 1993.

[26] International Telecommunication Union, Q.922 LAPF "ISDN data link layer specification for frame mode bearer services", 1991.

[27] International Telecommunication Union, Q.922A "Data link layer specification for frame mode bearer services". Febrero, 1992.

[28] International Telecommunication Union, recomendaciones para SDH (Synchronous Digital Hierarchy) : G.703, G.707, G.708, G.709, G.803, etc. 1989.

[29] International Telecommunication Union. Recommendation G.703 (10/98) "Physical/electrical characteristics of hierarchical digital interfaces", Article number S 15433. Octubre 1998.

[30] IPX desarrollado por Novell, Inc.

[31] J. Moy, "Multicast Extensions to OSPF", RFC 1584. Proteon, Inc. Marzo, 1994.

[32] J. Moy, "OSPF specification", RFC 1131. Proteon, Inc. Octubre, 1989.

[33] J. Postel (ed.), "Internet Protocol - DARPA Internet Program Protocol Specification," RFC 791, USC Information Sciences Institute. Septiembre, 1981.

[34] J. Postel (ed.), "Transmission Control Protocol - DARPA Internet Program Protocol Specification," RFC 793, USC Information Sciences Institute. Septiembre, 1981.

[35] J. Postel y J. K. Reynolds, "File Transfer Protocol", RFC 959, USC Information Sciences Institute. Octubre, 1985.

[36] J. Postel, "Domain Name System Implementation Schedule", RFC 897, USC Information Sciences Institute. Febrero, 1984.

[37] J. Postel, "Simple Mail Transfer Protocol", RFC 788, USC Information Sciences Institute. Noviembre, 1981.

[38] J. Postel, "User Datagram Protocol", RFC 768, USC Information Sciences Institute. Agosto, 1980.

[39] K. Lougheed, Y. Rekhter, "A Border Gateway Protocol (BGP)", RFC 1105. Cisco Systems - T.J. Watson Research Center, IBM Corp. Junio, 1989.

[40] K. Nichols, S. Blake, F. Baker, D. Black, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", RFC 2474. Cisco Systems, Torrent Networking Technologies, EMC Corporation. Diciembre, 1998.

[41] K. R. Sollins, "The TFTP Protocol (Revision 2)", RFC 783, MIT. Junio, 1981.

[42] M. Handley, V. Jacobson, "SAP: Session Announcement Protocol", Internet draft draft-ietf-mmusic-sap-01.txt. MMUSIC WG, IETF. Noviembre, 1996.

[43] NetBios/NetBEUI desarrollado por Microsoft, Inc.

[44] R. Braden (ed.), "Requirements for Internet Hosts - Communication Layers", RFC 1122, Internet Engineering Task Force. Octubre, 1989.

[45] Robert R. Metcalfe y David R. Boggs, "Ethernet: Distributed Packet Switching for Local Computer Networks", Xerox Palo Alto Research Center. Julio, 1976.

[46] S. Dering & D. cheriton. "Multicast Routing in a Datagram Network". Diciembre 1991.

[47] S. E. Deering y D. R. Cheriton, "Host groups: A multicast extension to the Internet Protocol" RFC 966, Stanford University. Diciembre, 1985.

[48] S. E. Deering, "Host extensions for IP multicasting", RFC 988, Standford University. Julio, 1986.

[49] S. E. Deering, D. Estrin, D. Farinacci, V. Jacobson, C. Liu, L. Wei, P. Sharma y A. Helmy, "Protocol Independent Multicast (PIM): Motivation and Architecture". 1996

[50] SPX desarrollado por Novell, Inc. 1980.

[51] Synchronous Data Link Control (SDLC), desarrollado por IBM. 1976.

[52] T. Berners-Lee, R. Fielding y H. Frystyk, "Hypertext Transfer Protocol - HTTP/1.0", RFC 1945. MIT/LCS, UC Irvine. Mayo, 1996.

[53] T. Li, B. Cole, P. Morton, D. Li, "Cisco Hot Standby Router Protocol (HSRP)", RFC 2281. Juniper Networks, Cisco Systems. Marzo 1998

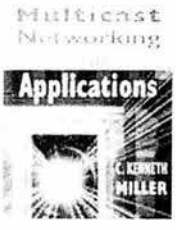
[54] V. Jacobson, "Serial Line Internet Protocol".

[55] VINES desarrollado por Banyan Systems

Bibliografía

Beau Williamson
Developing IP Multicast Networks Volume 1
Cisco Press
201 West 103rd Street Indianapolis, IN 46290 USA
2000

Ralph Wittman and Martina Zitterbart
Multicast Communication: Protocols and Applications
Ed. Morgan Kaufman
Mayo, 2000



C. Kenneth Miller
Multicast Networking and Aplications
Ed. Addison Wesley
Enero, 1999

Sitios en Internet:

- Bernardo Alarcos. La Red Mbone.
  Disponible en WWW: http://greco.dit.upm.es/~encarna/doctorado/mbone/asptec.htm

- Cisco Systems. Multicast Routing. 1999
  Disponible en WWW: http://www.cisco.com/warp/public/614/17.html

- IANA (Internet Assigned Numbers Authority). Internet Multicast Addresses. Última actualización: enero 2004.
  Disponible en WWW: http://www.iana.org/assigments/multicast-addresses

- INTERDIC Informática en Internet. Multidifusión de Audio/Video en Internet Multicasting – Mbone

Disponible en WWW: http://www.arrakis.es/~aikido/interdic/articul2.htm

- Kevin C. Almeroth. The Evolution of Multicast: From the Mbone to Interdomain Multicast to Internet2 Deployment. University of California.
  Disponible en WWW: http://multicast.internet2.edu/almeroth.pdf

- Kevin Savetz, Neil Randall, Yves Lepage. Mbone: Multicasting Tomorrow's Internet. 1998-1996.
  Disponible en WWW: http://www.savetz.com/mbone/

- Yee-Ting Lee. IP Extensions for Multicast. UCL, London. 2003.
  Disponible en WWW: http://www.hep.ucl.ac.uk/~ytl/multi-cast/iprouting_01.html

- Jaime Escoms Mendoza, Oscar Mora Climent, Jordi Pariagua Soriano, Francisco Sánchez Pardo. Aplicaciones DBS Futuras: Interactividad y Multimedia.
  Disponible en WWW: http://www.upv.es/satelite/trabajos/grupo8_99.00/index.html

- Red IRIS. Red Académica y de Investigación Nacional de España.
  Disponible en WWW: http://www.rediris.es

Acrónimos

ADSL	Asymetric Digital Subscriber Line Línea de Subscritor Digital Asimétrica
AS	Autonomous System Sistema Autónomo
ASICS	Application Specific Integrated Circuit Circuito Integrado de aplicación Específica
ATM	Asynchronous Transfer Mode Modo de transferencia Asincrona
BGP	Border Gateway Protocol Protocolo de encaminamiento en interdominios
BSR	Bootstrap Router. Nombre que se le da a un enrutador que desecha cierto tipo de paquetes
CAM	Content Addressable Memory Memoria de Conexión Accesible por una dirección
CBT	Cored-Based-Tree Árbol basado en un núcleo
CCITT	Consultative Committee on International Telegraphy and Telephony Comité Consultativo Internacional Telefónico y Telegráfico
CGMP	Cisco Group Management Protocol Protocolo de Gestión de Grupos de Cisco
CHAP	Challenge Handshake Authorization Protocol Protocolo de autenticación por intercambio de saludo
CRC	Cyclic Redundancy Check Comprobación por Redundancia Cíclica.
CSMA/CD	Carrier Sense Multiple Access/ Collision Detection. Acceso Múltiple con Detección de Portadora/Detección de Colisiones
DARPA	Defense Advanced Research Projects Agency Agencia de Proyectos de Desarrollo para Defensa
DAS	Dual Attachment Station FDDI Estación de Doble Conexión
DCE	Data Communications Equipment Equipo de Comunicación de Datos
DECNET	Digital Equipment Corporation Networking. Conjunto de Protocolos desarrollado y soportado por Digital Equipment Corporation

DNS	Domain Name Service
	Servicio de Nombramiento de Dominios
DR	Designated Router
	Enrutador Designado Cisco
DSL	Digital Subscriber Line
	Línea Digital de Suscriptor
DTE	Data Terminal Equipment
	Equipo Terminal de Datos
DVMRP	Distance Vector Multicast Routing Protocol
	Protocolo de Enrutamiento Multicast por Vector Distancia
EIA	Electronics Industry Association
	Asociación de Industrias Electrónicas
FCS	Frame Check Sequence
	Secuencia de Verificación de Trama
FDDI	Fiber Distributed Data Interface
	Interfase de Datos Distribuida por Fibra
GARP	IEEE´s Generic Attribute Resolution Protocol
	Protocolo creado por la IEEE para la Resolución de Atributos Genéricos
HDLC	High-level Data Link Control
	Control de Enlace de Datos de Alto Nivel
HSRP	Host Standby Router Protocol
	Protocolo de Enrutamiento de "hot standby"
IANA	Internet Assigned Numbers Authority
	Autoridad para la Asignación de Nombres en Internet
IEEE	Institute of Electrical and Electronics Engineers
	Instituto de Ingenieros en Electrónica y Electricidad
IETF	Internet Engineering Task Force
	Fuerza de Trabajo de Ingeniería en Internet
IGMP	Internet Group Management Protocol.
	Protocolo Administrador de Grupos en Internet
IP	Internet Protocol
	Protocolo de Internet
IPTV	Internet Protocol Television
	Televisión por IP

ISO	International Organization for Standardization
	Organización Internacional para la Estandarización
LAN	Local Area Network
	Red de Área Local
LAPB	Link Access Procedure Balanced
	Protocolo Balanceado de Acceso al Enlace
LAPD	Link Access Procedure for the D-Channel
	Procedimiento de Acceso al Enlace en el Canal D
MAC	Media Access Control
	Control de Acceso a Medios
MAN	Metropolitan Area Network
	Red de Área Metropolitana
MAU	Médium Access Unit
	Unidad de Conexión a Medios
MBONE	Multicast Backbone
	Columna vertebral de la red multicast en Internet
MOSPF	Multicast Open Shortest Path First
	Protocolo de Intra-dominio de Enrutamiento Multicast
MPEG	Moving Picture Expert Group
	Estándar para video digital y compresión de audio
NBMA	Non-Broadcast Multi-Access
	Termino que describe una Red Multiacceso que no soporta broadcast
NetBIOS	Network Basic Input/Output System
	Protocolo de red no ruteable, utilizado en redes LAN
NIC	Network Interface Card.
	Tarjeta de Interfase de Red
NTP	Network Time Protocol.
	Protocolo de Adaptación de Tiempo en la Red
OSPF	Open Shortest Path First.
	Algoritmo Abierto de Primero la Trayectoria más Corta "basado en el estado de enlaces"
PDU	Protocol Data Unit
	Unidad de Datos de Protocolo
PGM	Pragmatic General Multicast.

PHY	Physical Layer Capa Física
PIM	Protocol Independent Multicast Protocolo Independiente Multicast
PIM-DM	Protocol Independent Multicast-Dense Mode PIM de Modo Denso
PIM-SM	Protocol Independent Multicast-Sparse Mode PIM de Modo Esparcido
PPP	Point to Point Protocol Protocolo Punto a Punto
PVC	Permanent Virtual Circuit Circuito Virtual Permanente
RDSI	Integrated Services Digital Network Red Digital de Servicios Integrados
RIP	Routing Information Protocol Protocolo de Información de Enrutamiento basado en Vector Distancia
RP	Rendezvous Point Punto de Reunión
RPF	Reverse Path Forwarding Ruta de Reenvío de Regreso Multicast
RTP	Real Time Protocol Protocolo de Transporte en Tiempo Real
RTCP	Real Time Transport Control Protocol Protocolo de Control de Transporte en Tiempo Real
SDLC	Synchronous Data Link Control Control de Enlace de Datos Síncrono
SLIP	Serial Line Internet Protocol Protocolo de Conexión Serial de Internet
SMDS	Switched Multimegabit Data Service Servicio de Datos Conmutados Multimegabit
SSM	Source Specific Multicast Fuente Especifica Multicast

SPT	Shortest Path Tree
	Árbol de la Ruta más corta
SDP	Session Description Protocol
	Protocolo de Descripción de Sesiones
SVC	Switch Virtual Circuits
	Circuito Virtual Conmutado
TCP	Transmission Control Protocol
	Protocolo de Control de Transmisión
TTL	Time to Live.
	Tiempo de vida. Es un campo dentro del encabezado IP que indica el periodo dentro del cual se considera válido un paquete
UNI	User-Network Interface
	Interfaz de red de usuario ATM, FR
WAN	Wide Area Network
	Red de Comunicación Extendida