56

UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE INGENIERIA

ANALISIS DE METODOLOGIAS PARA REALIZAR PRUEBAS DE MPLS

T E S I S

QUE PARA OBTENER EL TITULO DE:
INGENIERO MECANICO ELECTRICISTA
(AREA ELECTRICA-ELECTRONICA)
P R E S E N T A
PABLO DE LA O CRUZ

DIRECTOR DE TESIS: M. EN C. SERGIO CASTRO RESINES

MEXICO, D.F.

OCTUBRE 2002

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA
DIVISIÓN DE INGENIERÍA ELÉCTRICA

COORDINACIÓN DE SEMINARIOS Y SERVICIO SOCIAL

NOTIFICACIÓN DE JURADO PARA EXAMEN PROFESIONAL

JURADO		BORRADOR DE TESIS RECIBIDO FIRMA	BORRADOR DE TESIS RECIBIDO FECHA	BORRADOR DE TESIS AUTORIZADO FIRMA	BORRADOR DE TESIS AUTORIZADO FECHA
PRESIDENTE:	ING. ROBERTO MANDUJANO WILD		4 sep 02		9 sep 02
VOCAL:	M.C. SERGIO CASTRO RESINES		2 Sep '02		2 Sep '02
SECRETARIO:	M.I. MARIA JAQUELINA LÓPEZ BARRIENTOS		Sept.05.02		Sept.12.02
1° SUPLENTE	ING. JOSÉ LUIS RODRIGUEZ PÉREZ		2 Sep 02		4 Sep 02
2° SUPLENTE	ING. NORMA E. CHAVEZ RODRÍGUEZ		4 sep 02		M sep 02

TESIS: 021/123 ANÁLISIS DE METODOLOGÍAS PARA REALIZAR PRUEBAS DE MPLS

FECHA Y HORA DEL EXAMEN 1° *DE OCTUBRE DE 2002 A LAS 10:00 HORAS*

RESPETABLE PROFESOR (A)

Por este conducto, me es grato notificarle que ha sido designado(a) miembro del jurado para el examen profesional del Alumno *(a) PABLO DE LA O CRUZ* con número de cuenta *8410360-4* de la carrera *INGENIERO MECÁNICO ELECTRICISTA, ELÉCTRICA-ELECTRÓNICA*, para lo cual le solicito de la manera más atenta revise el trabajo de tesis con el fin de que usted haga saber por escrito a esta Coordinación en formato adjunto si considera necesario que el alumno (a) realice modificaciones al mismo en un plazo de 5 (cinco) días hábiles contados a partir de la fecha en que Ud. Reciba esta notificación. De no haber observación alguna de su parte, le agradeceré firmar el presente autorizando el trabajo con lo cual el alumno podrá imprimir definitivamente su tesis.

Atentamente
"POR MI RAZA HABLARÁ EL ESPIRITU"
Cd. Universitaria, D.F., a 2 de Septiembre de 2002
EL COORDINDAOR DE SEMINARIOS Y SERVICIO SOCIAL

ING. JOSÉ ARTURO ORIGEL COUTIÑO

JAOC/bal

Dedicatorias

A ti especialmente, gracias Padre mío..

Papi y Mami, cada vez que me tienen en su pensamiento, saben que así mismo los tengo yo en el mío, y cada vez que los tengo entre mis brazos, doy gracias a Dios y los bendigo al saberme tan afortunado de que sean mis padres, los amo mucho.

Blanca, cada palabra, cada beso, cada caricia..
jamás imagine que sería tan feliz a tu lado y lo soy,
te amo Bebe.

Mis amores, Blanca Paola y Daniela Lizet, cambiaron mi vida;
una mirada, una palabra, un abrazo, una sonrisa suya
y me parece que vuelvo a nacer a cada instante.
Gracias Dios mío.

Adriana, Sandra, Lizbeth, Oscar y Omar, a cada uno de ustedes lo amo como si fuera mi único hermano.

César Olvera Morales, amigo, gracias por todo.

Sergio Castro Resines, estimado, muchas gracias por tu apoyo.

Dios nos bendiga a todos.

Índice

Introducción

Internet ha revolucionado el cómputo y las comunicaciones de una manera sin precedentes. La invención del telégrafo, teléfono, radio, televisión y computadora en su momento marcaron una etapa definitiva en la integración de capacidades de comunicación. *Internet* como tal posee una capacidad de comunicación de alcance mundial, un mecanismo de diseminación de la información, y un medio de colaboración e interacción entre los individuos y sus computadoras sin consideración de fronteras geográficas. De manera que el *Internet* de hoy se ha convertido en una infraestructura global de información.

El surgimiento de las redes de computadoras se remonta a 1962, bajo la idea de *J.C.R. Licklider* y con el propósito de tener el acceso a datos y programas en computadoras interconectadas en cualquier lugar. Al cabo de varios años se desarrollaron las primeras redes de circuitos telefónicos conmutados que en breve evolucionaron hacia redes más eficientes que permitieron la conmutación de paquetes a través del uso de protocolos de comunicaciones con la funcionalidad para controlar la comunicación entre dos dispositivos y la capacidad para direccionar redes y máquinas más allá de un destino específico. Actualmente el protocolo de comunicaciones que cumple con esta funcionalidad y que se conoce como el protocolo *de facto* en *Internet* se denomina *"Transmission-Control Protocol/Internet Protocol"* (*TCP/IP*, Protocolo de Control de Transmisión / Protocolo de *Internet*)

Con el desarrollo de la tecnología *Ethernet* y de las *PCs (Personal Computer)*, se incrementa el número de redes en *Internet* y la necesidad de interconectarlas, administrarlas e identificarlas a través de nuevos mecanismos de direccionamiento, resolución jerárquica de nombres y el desarrollo de dispositivos de comunicaciones cada vez más rápidos y eficientes. Las redes se desarrollaron alrededor de los *routers* y éstos comenzaron entonces a utilizar un modelo jerárquico de *routing*, *IGP* (*Interior Gateway Protocol*) para el *routing* interno de cada región de *Internet*, y *EGP* (*Exterior Gateway Protocol*) para intercomunicar a las regiones. Este nuevo diseño permitió que regiones distintas utilizaran *IGP* distintos, logrando así que el costo, velocidad de configuración, estabilidad y escalabilidad pudieran ajustarse a cada situación particular.

Aunque actualmente los esquemas de direccionamiento en *Internet* se basan en la versión 4 del protocolo *IP* (*IPv4*), sobre el cual se han desarrollado diferentes técnicas y estrategias de adjudicación de direcciones *IP* bajo la técnica conocida como *VLSM (Variable-Length Subnet Mask)*, las necesidades de los servicios de red se han hecho más específicas y las soluciones más especializadas.

Es en la búsqueda de estas soluciones más especializadas que varios fabricantes de equipos de redes se ocuparon en desarrollar sus propias propuestas basadas en la mejora de las funcionalidades de *IP* y *ATM* (*Asynchronous Transfer Mode*), sin embargo ante tales desarrollos propietarios surgió inevitablemente la necesidad de una solución basada en estándares y útil a todos, a través de la conmutación por etiquetas. La Conmutación Multiprotocolo por Etiquetas o *Multiprotocol Label Switching* (*MPLS*) ha surgido como una solución versátil para resolver los problemas que se presentan hoy en día en las redes, tales como ancho de banda, administración de la calidad en el servicio e ingeniería de tráfico. *MPLS* mejora las condiciones relacionadas con la escalabilidad y el *routing* en redes de *backbone* y además puede convivir con redes *IP*, *ATM* y *Frame Relay*.

MPLS se concibió inicialmente como una forma de aumentar la velocidad a los *routers*, sin embargo con el desarrollo de los trabajos a este respecto, la *IETF (Internet Engineering Task Force)* ha venido presentando resultados que lo hacen ver como el estándar que ofrece nuevas capacidades para las redes *IP* de gran escala.

MPLS se puede ver como un sustituto de la conocida arquitectura *IP* sobre *ATM*, también como un protocolo que sustituye las técnicas habituales de "*tunneling*" y como una técnica para acelerar el *routing* de paquetes, ya que integra sin discontinuidades los niveles de capa de enlace y capa de red, estableciendo valores para posiciones significativas de la estructura de encapsulación de la capa de red por sí misma y combinando eficazmente las funciones de control del *routing* con la simplicidad y rapidez de la conmutación de paquetes a nivel de capa de enlace.

En la medida en que las necesidades de servicios a través de las redes de datos han evolucionado propiciando el desarrollo de nuevos equipos de comunicaciones, se han hecho presentes dos aspectos muy significativos:

- Garantizar el paso de la información para diferentes clases de tráfico a través de cualquier red.
- Las necesidades de robustecer la infraestructura *IP* multiusuario.

Parte de la tecnología *MPLS* como tal ya existe, y los proveedores de servicios de *Internet* (*ISP, Internet Service Provider*) están muy interesados en utilizarla, sin embargo para éstos es muy importante contar con una solución adecuada a sus necesidades e infraestructura particulares, de manera que les resulta mucho más factible contar previamente con los elementos de evaluación que les permita adaptar las soluciones que ofrece esta tecnología estrictamente a sus requerimientos. De ahí que esta tesis propone analizar algunos procedimientos y topologías que permitan llevar a cabo pruebas básicas con la tecnología *MPLS*, con el fin de tener una referencia que sirva a los interesados en instalar *MPLS* para la realización de sus pruebas y evaluaciones en los equipos requeridos para la instalación de una red con este tipo de requerimientos. La UNAM cuenta con un grupo de trabajo de investigación sobre *MPLS* (www.mpls.unam.mx) y los resultados de este trabajo de tesis facilitarán y extenderán los trabajos desarrollados para la RedUNAM.

Capítulo 1: Conceptos básicos de redes

1.1. Redes de datos

Las redes de computadoras se utilizan para hacer más eficientes el trabajo y las comunicaciones, ya que representa una gran ventaja el hecho de poder compartir recursos tanto de *hardware* como de *software* reduciendo así los costos de proveer estos servicios a cada persona en una compañía.

Una red de computadoras se construyó en torno a la idea y necesidad de que hay emisores y receptores remotos. El emisor o fuente es una computadora que envía información hacia otra computadora llamada receptor o destino. Además de las computadoras otros equipos tales como impresoras, *scanner*, arreglos de discos y otros equipos tienen la capacidad de comunicarse en una red por lo cual generalmente son referidos como elementos o nodos de una red.

Cuando estos elementos participan en una comunicación dentro de una red requieren de algún medio para intercambiar información, por lo que requieren estar interconectados. Este medio comúnmente está hecho de alambre de cobre pero también puede ser a través de fibras ópticas, aunque recientemente han tomado fuerza las comunicaciones a través del aire usando transmisiones de radiofrecuencia y microondas.

1.2. Tipos y topología de redes

1.2.1. Tipos de redes

En cuanto a la extensión geográfica existen tres tipos principales de redes:

Red de área local (*LAN-Local Area Network*-). Las redes locales pueden tener varias configuraciones, varian en velocidad de transmisión, las distancias que se pueden manejar, sus características de operación, las capacidades y los servicios que puedan ofrecer de acuerdo al software y al hardware. La *LAN* es un segmento de red con estaciones de trabajo y servidores enlazados, o un conjunto de segmentos de red interconectados, por lo general dentro de una misma área, como por ejemplo un edificio. Las caracteristicas de una red de área local son las siguientes:

- Se utilizan en áreas geográficamente pequeñas.
- Ofrecen alta velocidad de comunicación, tipicamente a velocidades que van de 10 *Mbps* hasta 1 *Gbps* (según *Kevin Boyne* de UUNet, para 2005 se alcanzarán velocidades de 10 *Petabit*/s ---> 1 *Petabit*/s = 1000 *Terabits*/s).
- Flexibilidad de instalación.
- Flexibilidad de expansión.
- Costos relativamente bajos.
- Proveen accesos para muchos dispositivos.

Red de área metropolitana (*MAN-Metropolitan Area Network*-). Es una red cuyo alcance se extiende sobre ciudades o municipios. La *MAN* se construye comúnmente sobre una arquitectura de *bus* dual, lo que significa que dos cables de fibra proporcionan transmisiones en direcciones opuestas al mismo tiempo. Un nodo perteneciente al *bus* dual puede enviar datos en ambas direcciones. El *bus* dual se incorpora en una topologia de anillo. Las principales caracteristicas son las siguientes:

- La *MAN* puede proporcionar un servicio no orientado a la conexión con tamaños de trama de hasta 9,188 *bytes*, que se transportan sobre tramas de 53 *bytes*.
- La *MAN* puede proporcionar servicios orientados a la conexión que transporta tramas de 53 *bytes* entre los nodos del *bus*, sobre conexiones de circuitos virtuales.
- La *MAN* utiliza dispositivos tales como *routers*, teléfonos, conmutadores *ATM* y antenas de microondas.

Red de área extensa (*WAN-Wide Area Network*-). Una *WAN* es una red que enlaza computadoras situadas fuera de las propiedades de una organización (edificios o *campus*) y se extiende a través de áreas públicas que están reguladas por autoridades locales, nacionales e internacionales y constituye un sistema de comunicación que interconecta sistemas de cómputo geográficamente remotos. Generalmente el enlace entre lugares remotos se realiza a través de una red digital pública, pero existen otras opciones para conformar una red *WAN* propia mediante enlaces de microondas y/o satelitales. Las principales características de las redes de área extensa son las siguientes:

- Pueden extenderse en áreas geográficas muy extensas y de difícil acceso.
- Comúnmente la velocidad de comunicación es baja en comparación con redes de tipo *LAN*.
- Utilizan dispositivos tales como *routers*, módems y conmutadores *WAN*.

1.2.2. Topología de redes

La topología de una red esta definida por los elementos y dispositivos que se conectan a ella, y se implanta de acuerdo a la influencia de algunos factores dentro de la red de trabajo.

Cuando hacemos referencia a la topología de redes, nos referimos a las topologías físicas o lógicas. Una topología física es la forma en que se conectan físicamente los dispositivos que forman parte de la red. Entre las diferentes topologías físicas se incluyen: *bus*, anillo, estrella, malla e híbrida.

Topología de *bus*. Una red con topología de *bus* consta de un sólo cable al cual se conectan todas las estaciones de trabajo, es decir, todos los nodos comparten el mismo medio pero el acceso al mismo es sólo para uno a la vez (Ver fig 1.1). Este tipo de topología facilita la agregación o desagregación de nodos, sin embargo una falla en el cable de conexión inhabilita por completo toda la red. *Ethernet* es un ejemplo de la

red más común con topología lógica de *bus*. El cable coaxial fue su medio principal de transmisión, y actualmente en la mayoría de las instalaciones se utiliza el cable de par trenzado.

Las características de la topología en *bus* son las siguientes:

- Se pueden agregar/desagregar fácilmente estaciones de trabajo.
- Sólo se puede incluir un número limitado de dispositivos.
- Cuando existe un problema es difícil que este pueda ser aislado.
- Precisa de terminadores.
- Cuando se tienen varios accesos al mismo tiempo la comunicación se vuelve lenta.
- Si se suscita una falla en cualquier punto del medio, se inhabilita a la red en su totalidad.

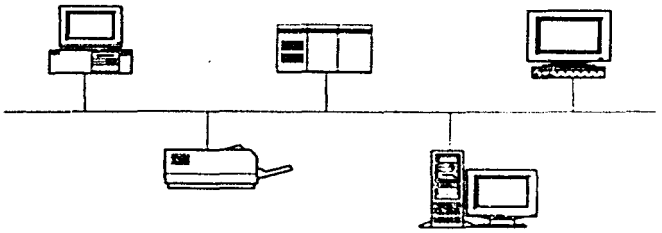

Figura 1.1 Topología tipo bus

Topología en anillo. Es una topología en ciclos cerrados que no precisa terminadores. Esta topología se conforma por un anillo lógico pero la topología física corresponde a una estrella, conectando la primera estación de trabajo a la siguiente y la última a la primera, cerrado el anillo (Ver fig. 1.2).

Las características de la topología en anillo son las siguientes:

- Puede transportar paquetes de datos con alta velocidad.
- No hay colisiones.
- Es fácil localizar problemas con dispositivos y en el cableado.
- No requiere de terminadores.
- Utiliza más cable que una red tipo bus.
- Si se presenta un corto en un cable provocará que algunos elementos de la red estén abajo.
- Si se requiere adicionar un nuevo dispositivo al anillo, todos los dispositivos de la red estarán suspendidos.

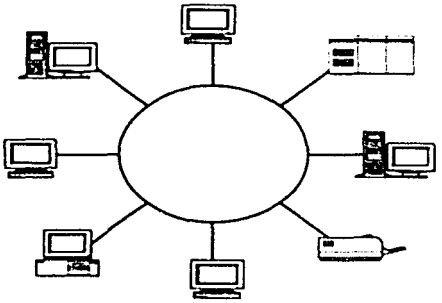

Figura 1.2 Topología de anillo

Topología en estrella. Las redes con topología en estrella pueden ser activas o pasivas (Ver fig. 1.3). En la topología de estrella pasiva, la estrella se configura con una caja que sencillamente sirve para la organización del cable, como un registro telefónico. En la topología de estrella activa, un concentrador (*hub*) es un dispositivo que regenera y repite las señales. El concentrador activo puede incluir características básicas para el diagnóstico de fallas o defectos en la transmisión de las estaciones de trabajo.

Actualmente el equipo activo ha evolucionado hasta convertirse en un conmutador. La falla de un nodo o en un cable no incapacita el resto de la red.

A continuación se muestran algunas características de este tipo de red.

- Es fácil agregar /desagregar nodos a/de la red.
- Si existe una falla en un cable, la red no se colapsa.
- Si existe una falla considerable en el equipo activo puede provocar que la red entera quede inactiva.
- Los costos de instalación y equipamiento son mas altos que para las redes tipo *bus*.

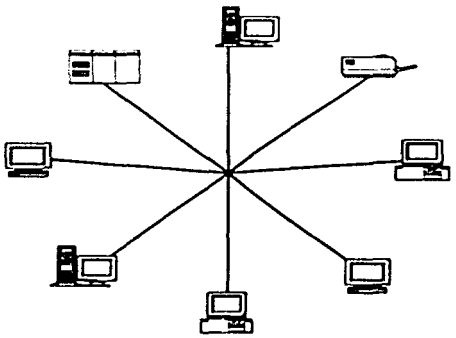

Figura 1.3 Topología de estrella

Topología de malla. Este tipo de topología frecuentemente se utiliza en redes *MAN* o *WAN* para conectar oficinas remotas mediante enlaces de telecomunicaciones. Se interconectan a través de *routers*, los cuales eligen la mejor y más eficiente trayectoria para comunicar los nodos fuente y destino a través de la malla. Los enlaces que fallan son suplantados por otros enlaces de la malla (Ver fig. 1.4).

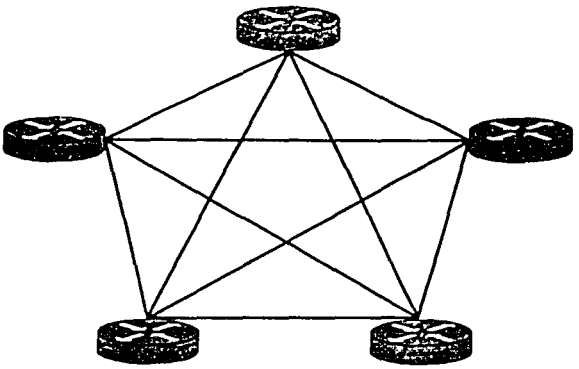

Figura 1.4 Topología de malla

Topología híbrida. Una topología híbrida combina dos o más diferentes topologías físicas en una red simple. Las redes híbridas más comunes que se encuentran actualmente son las topologías estrella-*bus* y estrella-anillo (Ver fig. 1.5).

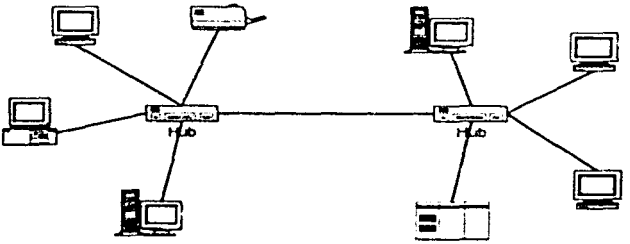

Figura 1.5 Topología híbrida

Una topología lógica define los métodos que utilizarán los dispositivos para acceder al medio de transmisión para comunicarse y transmitir datos, estos métodos son:

Métodos de Acceso al Medio

CSMA / CD (Carrier Sense Multiple Access / Collision Detection). Es el protocolo *IEEE* 802.3 para redes tipo *Ethernet*, donde cualquier dispositivo puede intentar enviar un *frame* en cualquier momento. Cada dispositivo conectado a la red, detecta cuando el medio o línea está ocupada o disponible para ser utilizada; si lo ve disponible empieza a transmitir, y si otro dispositivo intenta enviar sus *frames* en el mismo momento, se produce una colisión, provocando que los *frames* sean desechados y entonces ambos dispositivos esperan un tiempo aleatorio para volver a intentar su transmisión.

Token Ring. Es el protocolo de la *IEEE* 802.5 para redes de anillo, con el cual los dispositivos toman turno para transmitir datos haciendo uso de un *token*, en el cual colocan los *frames* que quieren transmitir y lo pasan al siguiente dispositivo conectado a la red de anillo junto con el identificador del destinatario, de manera que el *token* viaja por toda la red hasta encontrar a su destinatario. Cuando esto sucede, el destinatario copia el *frame* del *token* y lo envía de regreso al emisor, el cual al recibirlo, lo descarga y lo deja disponible para que otro dispositivo lo utilice.

Token Passing. Es el protocolo de la *IEEE* 802.4, que proporciona todos los beneficios de *Token Ring*, pero sin los requerimientos físicos de una red de anillo. En *Token Passing* el *token* lleva el control del acceso al medio de manera secuencial entre cada dispositivo conectado al mismo segmento de red. Cuando uno de los dispositivos recibe el *token*, puede enviar su información hasta que ya no tiene nada que enviar y entonces libera el *token* para que otro dispositivo lo utilice.

Modos de Transmisión

Broadcast. Implica que cada dispositivo envía *frames* al resto de los dispositivos conectados al mismo segmento de red y al mismo tiempo.

Unicast. Es la transmisión de sólo un dispositivo origen a sólo un dispositivo receptor de la red.

Multicast. Se utiliza para la transmisión de sólo un dispositivo de la red a múltiples dispositivos en específico conectados a la red.

Anycast. Permite la transmisión de sólo un dispositivo hacia uno (el más cercano) de varios dispositivos conectados a la red. Este modo de transmisión se utiliza en *IPv6* para tareas de autoconfiguración.

1.3. Protocolos de comunicaciones

Los protocolos de comunicaciones se definen dentro del contexto de arquitectura de red en capas. Cada capa especifica un protocolo o protocolos para manejar subsistemas o funciones del proceso de comunicación.

Es imposible hablar de interconexión de computadoras sin hacer referencia a los protocolos o procedimientos de comunicaciones que permiten el intercambio de paquetes de datos, y para ello generalmente se maneja el modelo de referencia de interconexión de sistemas abiertos (*OSI –Open System Interconnection–*) (Ver fig 1.6). El conjunto de protocolos de *OSI* define como los fabricantes pueden crear productos que puedan interactuar dentro de un proceso de comunicación, ya que cada fabricante empleaba sus propios protocolos de intercambio de señales haciendo imposible el intercambio de información.

Un protocolo es un modo definido de comunicación entre sistemas con reglas y procedimientos bien establecidos, que especifican la sincronización de las señales y la estructura de los datos comunicados.

El hecho de considerar a *OSI* como un modelo implica que éste no provee una solución tecnológica real, sino que aporta procedimientos y normas para el intercambio de información. El modelo *OSI* especifica siete capas en el conjunto de protocolos y cada una trabaja en diferentes niveles de *hardware* y *software*. A continuación se describen brevemente las siete capas:

Capa 1 - Física. Proporciona los medios mecánicos, eléctricos, funcionales y de procedimientos para activar, mantener y desactivar los enlaces físicos para la transmisión de *bits* entre los nodos interconectados a los sistemas de redes de comunicaciones, es decir; establece todo lo referente a los conectores y especifica los niveles de voltajes, potencia de la señal y distancias máximas de transmisión.

Capa 2 - Enlace de datos. Permite la transmisión segura de datos a través de los medios físicos de una red y de acuerdo al conjunto de protocolo que se trate, define diferentes características de red y protocolos, incluyendo direccionamiento físico, topología de red, notificación de errores, secuencia de *frames* y control de flujo.

El *IEEE* divide la capa de enlace de datos en dos subcapas: *Logical Link Control* (*LLC*) –*IEEE 802.2*- y *Media Access Control* (*MAC*). La subcapa *LLC* maneja las comunicaciones entre los dispositivos conectados a un mismo enlace de la red, soporta los servicios: orientado a conexión (*connection-oriented*) y no orientado a conexión (*connectionless)* que utilizan los protocolos de capas superiores, además de habilitar campos en los *frames* de la capa de enlace de datos para permitir que protocolos de capas superiores compartan un mismo enlace físico. La subcapa de *MAC* maneja el protocolo de acceso al medio y define las direcciones físicas *MAC* que permiten identificar a los dispositivos como únicos.

Capa 3 - Red. Provee funciones de conmutación y *routing* de los paquetes de datos a través de múltiples enlaces de datos, lo cual se logra por el manejo del direccionamiento lógico de los dispositivos interconectados a través de diferentes redes, soporta los servicios de *connectionless* y *connection-oriented* utilizados por los protocolos de capas superiores, además de reconocer la topología de red con el objeto de determinar la ruta más adecuada para el envío de los paquetes de datos.

Capa 4 – Transporte. Asegura la transferencia de datos punto a punto y de manera segura entre redes interconectadas. Entre sus funciones típicas están el control de flujo, multiplexaje, manejo de circuitos virtuales y mecanismos de chequeo y recuperación de errores, corrigiendo posibles fallas en la transmisión de los segmentos de datos.

Capa 5 – Sesión. Permite establecer, administrar y terminar sesiones de comunicación entre entidades de la capa de presentación, es decir maneja las solicitudes y respuestas entre aplicaciones localizadas en diferentes dispositivos de red que se comunican.

Capa 6 – Presentación. En esta capa se manejan las funciones de codificación y conversión para ser aplicadas a los datos en la capa de aplicación. Estas funciones aseguran que la información enviada de la capa de aplicación de un sistema será entendida por la capa de aplicación de otro sistema.

Capa 7 – Aplicación. Es la capa más cercana al usuario final, de manera que interactúan directamente con el *software* de aplicación. Entre sus funciones se encuentran la identificación de la identidad y disponibilidad de comunicación entre participantes para una aplicación que requiere transmitir datos, la identificación de la disponibilidad de recursos y la sincronización de la comunicación.

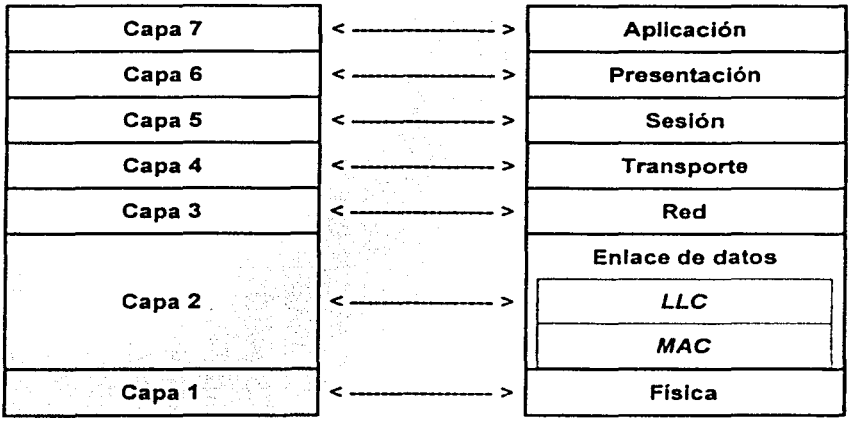

Figura 1.6 Modelo de referencia *OSI*

Un equipo que trabaje bajo un conjunto de protocolos no se puede conectar e interoperar con otro equipo que utilice otro conjunto de protocolos, sin embargo el uso de diversas técnicas de encapsulación y conversión de protocolos hace posible esta comunicación y que puedan convivir simult neamente.

Entre los conjuntos de protocolos se encuentran los siguientes:

- *OSI*
- *Xerox XNS*
- *TCP/IP*
- *Novell Netware*
- *IBM SNA / Microsoft Networking*
- *DEC DECnet*
- *Banyan Systems*
- *Apple Computer*

La relación entre los diferentes conjuntos de protocolos y el modelo *OSI* queda ilustrada a través de la figura 1.7.

Una clasificación adicional que se maneja para ubicar la funcionalidad de los conjuntos de protocolos, es la siguiente:

Protocolos de aplicación. Los protocolos de aplicación abarcan las capas de aplicación, presentación y sesión, que proporcionan fundamentalmente la funcionalidad de interacción entre aplicaciones-usuario e intercambio de datos.

Protocolos de transporte. Los protocolos de transporte proporcionan servicios de distribución de datos orientados a la conexión a través de redes. Fundamentalmente proporcionan los mecanismos de control para mantener sesiones o conexiones entre sistemas que realizan el intercambio de datos extremo a extremo.

Protocolos de red. Los protocolos de la capa de red proporcionan servicios de enlace entre los sistemas de comunicaciones, es decir, proporcionan los mecanismos de acceso a la red, de comprobación de errores y para peticiones de retransmisión de paquetes, además de encargarse del direccionamiento y el *routing* de los mismos.

OSI Capa	Apple Computer	Banyan Systems	DEC DECnet	IBM SNA Microsoft Networking		Novell NetWare
Aplicación Capa 7	Protocolos y Programas de Aplicación Para transferencias de archivos, correo electrónico, et					
Presentación Capa 6	AppleTalk Filing Protocol (AFP)	Remote Procedural Calls (Net RPC)	Network Management Network Application	Transaction Services Presentation Services	Server Message Block (SMB)	NetWare Core Protocols (NCP
Sesión Capa 5	AppleTalk Session Protocol (ASP)		Session	Data Flow Control	Network Basic Input/Output System (NetBIOS)	Network Basic Input/Output System (NetBIOS)
Transporte Capa 4	AppleTalk Transaction Protocol (ATP)	VINES InterProcess Communications (VIPC)	End Communications	Transmission Control	Network Basic Extended User Interface (NetBEUI)	Sequenced Packet Exchange (SPX)
Red Capa 3	Datagram Delivery Protocol (DDP)	VINES *Internet* Protocol (VIP)	*Routing*	Path Control		*Internet* Packet Exchange (IPX)
Enlace de Datos Capa 2	Tarjetas de Interfaz de red: Ethernet, Token-Ring, ARCNET, Star*LAN*, Loc Manejadores de NIC: Open Datalink Interface (ODI), Network Independent Inte					
Física Capa 1	Medio de Transmisión: Par trenzado, Coaxial, Fibra Optica, Inalámbrico, etc.					

Capítulo 2: *MPLS* (*Multiprotocol Label Switching*)

En este capítulo se hace una revisión de *MPLS*, centrándose en antecedentes, elementos componentes, estructura y operación de una red con *MPLS*, beneficios, aplicaciones, *MPLS* y el modelo *OSI*, finalizando con una pequeña exposición sobre los trabajos de *MPLS* en la UNAM.

2.1. Introducción

La Conmutación Multiprotocolo por Etiquetas o *Multiprotocol Label Switching* (*MPLS*) se concibió inicialmente como una forma de aumentar la velocidad a los *routers*, sin embargo con el desarrollo de los trabajos a este respecto, la *IETF (Internet Engineering Task Force)* ha venido presentando resultados que lo hacen ver como el estándar que ofrece nuevas capacidades para las redes *IP* de gran escala.

En la medida en que las necesidades de servicios a través de las redes de datos han evolucionado propiciando el desarrollo de los equipos de comunicaciones se han hecho presentes dos aspectos muy significativos:

- Garantizar el paso de la información para diferentes clases de tráfico a través de cualquier red.
- Las necesidades de robustecer la infraestructura *IP* multiusuario.

MPLS tiene la posibilidad de manejar de manera efectiva los requerimientos de las redes actuales proporcionando una solución basada en estándares y acorde a las perspectivas de los usuarios, con mayor y mejor control del que las redes *IP* convencionales pueden ofrecer. Las redes actuales bajo la perspectiva de *MPLS* cubren esas necesidades de la siguiente forma (información tomada de [APP 1998]):

a) **Funcionalidad**.- La conmutación de etiquetas proporciona funciones, que no están disponibles en una red *IP* convencional o eran ineficientes en el *routing* convencional, como por ejemplo utilizar *routing* explícito para seleccionar una ruta en base a ciertos atributos específicos, además de la dirección de destino, donde se requiere satisfacer cierta *QoS (Quality of Service)*.

b) **Escalabilidad**.- En la medida que crece el tamaño de una red la información de *routing* que debe manejarse aumenta considerablemente y eventualmente puede provocar una saturación o sobrecarga en los *routers* que la conforman. Comúnmente en la actualidad se superponen a las redes *IP* sobre redes *ATM* o bien sobre circuitos virtuales *Frame Relay*, pero ello no soluciona el problema, a través de *MPLS* se utilizan dispositivos de capa 2 (conmutadores *ATM*) que son capaces de manejar el plano de control de *IP* y con ello disminuyen el problema, que aminora a medida que se aplican soluciones de ingeniería de tráfico.

c) **Evolución**.- Tener la capacidad de cambio y crecimiento ante la necesidad de transportar múltiples tipos de tráfico sobre *IP*, sin que ello implique grandes alteraciones en la red, y lo que esto puede implicar en la industria.

d) **Integración**.- Convergencia de aplicaciones e integración de redes, a través del diseño de soluciones integrales para todos los niveles y todos los servicios.

Uno de los factores de éxito del *Internet* actual está en la aceptación de los protocolos *TCP/IP* como estándar *de facto* para todo tipo de servicios y aplicaciones. Si bien es cierto que *Internet* puede considerarse como una red pública de datos a gran escala, también es cierto que ya no satisface todos los requerimientos de los usuarios, principalmente en entornos corporativos que necesitan la red para el soporte de aplicaciones críticas, pues no permite seleccionar diferentes niveles de servicio para los distintos tipos de aplicaciones, por lo que la red *Internet* se valora más por el servicio de acceso y distribución de contenidos que por el servicio de transporte de datos, conocido como de "mejor esfuerzo".

La responsabilidad de la *IETF* en el desarrollo del estándar *MPLS* es la de consensuar diferentes soluciones de conmutación multinivel propuestas por distintos fabricantes, pues las implicaciones que supone su implantación real son enormemente

complejas. *MPLS* se puede presentar como: un sustituto de la arquitectura *IP* sobre *ATM*, un protocolo que sustituye las técnicas habituales de "*tunneling*" y una técnica para acelerar el *routing* de paquetes, integrando sin discontinuidades los niveles de capa de transporte y capa de red, a través de las funciones de control del *routing* y conmutación de paquetes a nivel de capa de enlace de datos.

Lo anterior es posible porque *MPLS* define una encapsulación que reside entre la encapsulación de capa de enlace de datos y capa de red, estableciendo valores para posiciones significativas de la estructura de encapsulación de la capa de red por si misma, como son los casos de *ATM* y *Frame Relay*.

MPLS es muchas cosas para mucha gente, pero es el primero de todos los métodos para lograr las características deseables de reenvio de las tecnologías de conmutación en tanto se mantenga la flexibilidad y escalabilidad del *routing* en capa de red.

Actualmente los dos usos más importantes de *MPLS* son la Ingeniería de Tráfico y las *VPNs (Virtual Private Networks)*. Aunque ambas se pueden realizar con otros protocolos ya existentes, *MPLS* lo hace más simple porque ofrece la ventaja de separar el *routing* y reenvio de paquetes para reducir o eliminar algunas limitaciones de *routing*.

En la Ingeniería de Tráfico por ejemplo, es posible especificar rutas explícitas durante el proceso del establecimiento de una ruta por donde los datos podrían ser reenviados a través de los puntos de red más concurridos o puntos de congestión. Los puntos de congestión, desarrollan como resultado de esta circunstancia que las rutas propuestas converjan en la selección de la ruta de menor costo para cada destino posible. Esto facilita la utilización de una ruta explícita para dirigir porciones significativas de este tráfico a partes de la red que no fueron seleccionadas por el proceso de *routing* permitiendo desviar los paquetes de los puntos de red más congestionados e ignorando las rutas que convergen en éstos.

Es relativamente fácil establecer túneles que permitan el transporte de paquetes que de otra manera podrían no enviarse (*routing*) correctamente a través del *backbone* de la red, estos túneles comúnmente se manejan a través del uso de las *VPN*s. *MPLS* permite utilizar túneles para enviar los paquetes a través de su *backbone* entre sitios *VPN* efectuando la traducción de direcciones, lo que resulta menos costoso que el uso del *tunneling*, y haciendo a este casi innecesario.

En resumen, *MPLS* provee una forma eficiente para designar, *routing*, reenviar y conmutar flujos de tráfico de datos, voz y video a través de una red, de la siguiente manera [IEC 2002]:

- Estableciendo mecanismos para manejar flujos de tráfico de diferentes granularidades, como flujos entre distintos componentes de *hardware* de comunicaciones y también entre diferentes aplicaciones de red.
- Manteniéndose independiente de los protocolos de capa 2 y capa 3 del modelo de referencia *OSI*.
- Aplicando un método de mapeo de direcciones *IP* a etiquetas de formato sencillo y de longitud fija, utilizadas por tecnologías de reenvío y conmutación de paquetes.
- Aprovechando protocolos tales como *Resource Reservation Protocol* (*RSVP*) y *Open Shortest Path First* (*OSPF*).
- Soportando los protocolos: *IP*, *ATM*, y *Frame-Relay*.

2.2. Antecedentes

Es importante comentar el contexto por el cual varios fabricantes se dieron a la tarea de realizar sus propios desarrollos para la conmutación de etiquetas, ya que si bien es cierto que *MPLS* cubre un espectro más amplio de soluciones, una de las razones más significativas tiene que ver directamente con la necesidad de integrar *IP* y *ATM*. [DAV 2000] y [APP 1998]

El estándar de *IP* sobre *ATM* se describe en el *RFC 1483* y trata de la forma de encapsular los paquetes sobre enlaces *ATM*, el cual aparentemente es un problema muy simple. El *RFC 1577* ya nos habla de "*IP* clásico sobre *ATM*". En este se dice que los *routers IP* y sus clientes se pueden comunicar siempre y cuando pertenezcan a la misma subred, es decir si la dirección de red pertenece a la misma subred. Si están en diferentes subredes, entonces uno o varios *routers* tendrían que cumplir con la función de reenviar los paquetes de la subred fuente hacia la subred destino.

En *IP* clásico sobre *ATM* se reconoce que un arreglo de equipos y *routers IP* pueden estar conectados a la misma red *ATM* aún perteneciendo a diferentes subredes, siempre y cuando compartieran una dirección *IP* de red y subred que fuera común, este concepto es conocido como subred *IP* lógica (*LIS –Logical IP Subnet*). Entonces la forma de llevar un paquete de una *LIS* a otra se haría a través de un *router* conectado a las dos *LIS* y ello implica que no podrían utilizar el mismo *VC (Virtual Channel)* a través de la red *ATM*, lo que al parecer se solucionaría si el conmutador *ATM* perteneciera a la misma *LIS*, pero esto resultaría un gran problema cuando más de una organización pretendiera administrar sus propias direcciones *IP*.

El mecanismo desarrollado para facilitar la comunicación entre dos dispositivos pertenecientes a la misma *LIS* se conoce como *ATM ARP*, el cual permite que dos dispositivos *IP* conozcan la dirección *ATM* del otro. A diferencia de *ARP* convencional, en *ATM* se requiere de utilizar un servidor *ARP*, es decir, un solo nodo en una *LIS* que efectúa la resolución de dirección *IP* a dirección *ATM*. Y con la dirección *ATM* puede establecer un *VC* para esa dirección y entonces enviar datos.

En cuanto a las tecnologías desarrolladas por diferentes fabricantes en la búsqueda de opciones para la conmutación de etiquetas, stas son: *Cell Switching Router, IP Switching, TAG Switching* y *Aggregate Route-based IP Switching (ARIS).*

2.2.1. *Cell Switching Router (CSR)*

La arquitectura de *CSR* es una tecnología desarrollada por *Toshiba* la cual introduce la idea de que un conmutador *ATM* pueda ser controlado por protocolos *IP* (como sucede en el caso de *routing IP* y *RSVP*) más que por protocolos de señalización *ATM*. *CSR* está diseñado para funcionar como un *router* para conectar *LIS* en una red basada en elementos *IP* clásico sobre *ATM*.

La decisión de enviar un paquete en un conmutador *ATM* puede desecharse si de alguna manera se asocia a la interfaz de entrada y el *VPI/VCI* de la celda *ATM* recibida con interfaz de salida *VPI/VCI* para ser usada en el reenvío de estos. Típicamente esto no puede ser realizado en ningún *router*/conmutador *ATM* porque los valores *VPI/VCI* entrando y saliendo de las interfaces corresponde a los clientes conectados a esta interfaz más que a la fuente o al destino de los paquetes *IP* contenidos en las celdas.

La gente de *Toshiba* reconoce que –si el protocolo de señalización se usa para establecer nuevos valores de *VPI/VCI* para flujos específicos de paquetes *IP* llegando a una interfaz de entrada- entonces esos valores especiales podrían estar ligados a sus correspondientes valores de *VPI/VCI* a los valores de una interfaz de salida. De esta manera una celda que llega con un valor *VPI/VCI* podría ser conmutada de la capa *ATM* hacia la interfaz de salida apropiada y podría asignarle el *VPI/VCI* correcto para reenviarlo al *router* del siguiente salto o al cliente directamente.

La idea básica fue que la mayoría del flujo de paquetes podrían ser procesados utilizando funciones de *routing* pero estos flujos específicos podrían ser reenviados a la capa de *ATM* basándose en el uso de un protocolo de señalización adicional. Los flujos implican un manejo especial y consumen gran cantidad de *VC*s en dos categorías: por omisión o dedicados.

El protocolo de notificación de atributos del flujo (*FANP –Flow Attribute Notification Protocol*) se encarga de identificar los *VCs* entre los *CSRs* y establece las asociaciones entre flujos individuales y *VCs* individualmente dedicados. El objetivo del *CSR* es permitir cortar a través del reenvío de flujos para conmutar el flujo de celdas *ATM* que constituyen el paquete más que reensamblarlo y tomar una decisión de reenvío a nivel *IP*.

2.2.2. *IP Switching*

Desarrollado por *Ipsilon IP Switching* habilita un dispositivo conmutador *ATM* para fungir como *router*, superando las limitaciones del *throughput* de los *routers* tradicionales. Las características principales de *IP Switching* son:

- Integrar conmutadores *ATM* y *routing IP* de una manera simple y eficiente.
- Utiliza la presencia de tráfico de datos para manejar el establecimiento de etiquetas.

El control de la unión de etiquetas se realiza por el protocolo *Ipsilon* de administración de flujo (*IFMP –Ipsilon Flow Management Protocol-*) y el control de los conmutadores *ATM* y de los circuitos virtuales se realiza a través del protocolo de administración general de conmutación (*GSMP –General Switch Management Protocol-*)

IFMP se utiliza para el establecimiento de valores *VPI/VCI* los cuales son usados por los conmutadores *ATM* vecinos para flujos específicos *IP*. La suposición en este método es que los conmutadores *IP* podrían reenviar los paquetes *IP* entre clientes y *routers* utilizando la encapsulación por omisión hasta que un flujo sea detectado y se envíe un mensaje de redireccionamiento. Una vez que un conmutador *IP* reenvía el mensaje de redireccionamiento -incluyendo el nuevo valor de encapsulación (*VPI/VCI*

en *ATM*)- el cliente vecino, *router* o conmutador *IP* podría reenviar paquetes pertenecientes al flujo utilizando nuevamente la encapsulación definida. El uso de la encapsulación de la nueva capa de enlace de datos -la cual podría ser localmente única para un flujo específico- podría permitir a un *router* vecino reenviar los paquetes *IP* asociados con este flujo la capa de enlace de datos.

IP Switching tiene la ventaja –similar al propósito de *CSR*- de ser potencialmente capaz de reducir el reenvío de la carga por omisión por un gran porcentaje de todos los paquetes *IP* reenviados a cualquier *router IP* en particular. Esta podría resultar en una sobrecarga significativa en el procesamiento de paquetes *IP* en el modo de reenvío por omisión y requerir la atención para la actividad de paquetes *IP* siempre y cuando haya flujo redireccionado.

2.2.3. *TAG Switching*

Es el método de conmutación desarrollado por *Cisco Systems* y en contraste a *CSR* e *IP Switching* es una técnica *control driving* que no depende del flujo de datos para estimular el levantamiento de tablas de reenvío de etiquetas en el *router*. Una red en *TAG Switching* consiste de *TAG Edge Routers y TAG Switching Routers* donde la responsabilidad del etiquetado de los paquetes corresponde al *Edge Router*. Se utilizan protocolos de *routing IP* para determinar el siguiente salto por tráfico. Las etiquetas se relacionan con las rutas en la tabla de *routing* y distribuidas en pares vía *TAG Distribution Protocol*.

2.2.4. *Aggregate Route-based IP Switching (ARIS)*

Este protocolo desarrollado por *IBM*, es muy similar a la arquitectura *Tag Switching*, ya que relaciona etiquetas para agregar rutas (grupos de prefijos de direcciones) en lugar de flujos (como e *CSR* o *IP Switching*). Las relaciones de

etiquetas y las rutas de conmutación de etiquetas se habilitan en respuesta al control del tráfico más que al flujo de datos, con el *router* de egreso como el iniciador. En *ARIS* los *routers* se llaman *Integrated Switch Routers o ISR*. *ARIS* se diseño enfocándose a *ATM* como una opción de capa de enlace de datos (provee mecanismos de prevención de *loops* no disponibles en *ATM*). El protocolo *ARIS* es un protocolo par a par que corre entre *ISRs* directamente sobre *IP* y provee un medio para establecer vecinos y para intercambiar arreglos de etiquetas. La distribución de etiquetas comienza en el *router* de egreso y se propaga ordenadamente hacia el *router* de ingreso.

2.3. Elementos componentes

A continuación se mencionan los principales términos y conceptos que maneja *MPLS*. La mayoría de la información de este tema se tomo de [IEC 2002]. Una propuesta de interacción entre los diferentes elementos de *MPLS* se muestra en la figura 2.1.

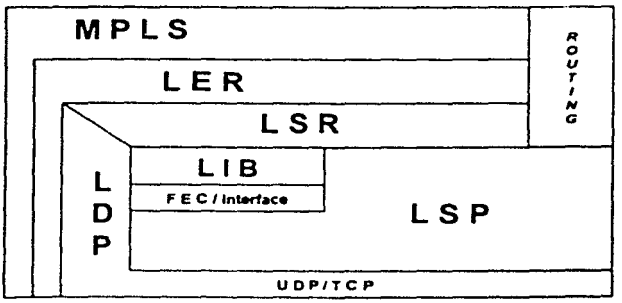

Figura 2.1 Interacción de elementos *MPLS*

La propuesta de la figura 2.1 muestra en términos muy generales la interacción de los diferentes elementos componetes de *MPLS*. En una red de *backbone*, el *routing* se

puede efectuar con algún protocolo de la industria como *OSPF*, *BGP*, etc. Sobre esta red es posible operar *MPLS* considerando la interacción de todos sus elementos, tal que los *LER* representarán la frontera entre las redes *IP* y la red MPLS, al interior del *backbone* los *LSRs* se encaragarán de llevar el intercambio de *frames* identificando los elementos vecinos y señalando su presencia en la red y que en conjunto con los *LERs* conformaran un dominio de *MPLS*, los *LSRs* operan el *LDP*, el cual utiliza *TCP* para la transmisión confiable de los datos de control y *UDP* durante la fase de descubrimiento de las *LSP* que se establecen para el envío de los *frames*, a su vez mantiene las *LIB* que residen en las interfaces de los *LSRs* relacionandolas con *FECs* específicas de acuerdo al tipo de tráfico que se requieran transmitir.

2.3.1. *LSR* (*Label Switching Router*)

Un *LSR* es un *router* de alta velocidad que trabaja al interior de una red *MPLS* y participa en el establecimiento de las *LSPs* utilizando los protocolos de señalización de etiquetas adecuados, además participa en la conmutación del tráfico de datos utilizando las *LSPs* establecidas (Ver fig. 2.2).

2.3.2. *LER* (*Label Edge Router*)

Un *LER* es un dispositivo que opera en la frontera entre la red de acceso y la red *MPLS* (Ver fig. 2.2). Los *LERs* tienen múltiples puertos que pueden estar conectados a redes distintas tales como *Frame Relay*, *ATM* ó *IEEE 802.3*. Un *LER* de ingreso recibe el tráfico de estas redes y lo reenvía hacia la red *MPLS* a través de las *LSPs* utilizando algún protocolo de señalización de etiquetas, asimismo un *LER* de egreso distribuye el tráfico de regreso hacia las redes de acceso conectadas a la red *MPLS*. Los *LER* son los responsables de la asignación por primera vez y finalmente el retiro de etiquetas, conforme el tráfico entra o sale de la red *MPLS*.

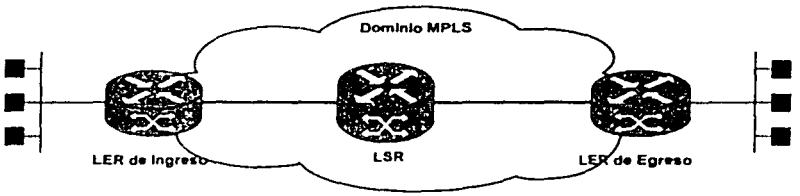

Figura 2.2 Ubicación de los equipos *LSR* y *LER*

2.3.3. *FEC* (*Forward Equivalence Class*)

Una *FEC* es una representación de un grupo de paquetes que comparten las mismas características y/o requerimientos para su transportación por la red. Contrariamente al reenvío convencional en *IP*, en *MPLS*, la inclusión de un cierto paquete dentro de una *FEC* se hace sólo una vez cuando el paquete entra en la red, de manera que todos los paquetes dentro de un grupo reciben el mismo trato durante su viaje hasta llegar a su destino. Las *FEC*s pueden seleccionarse con base a los requerimientos de servicio de los paquetes, o simplemente en prefijos de direcciones. Cada *LSR* construye una tabla que especifica como los paquetes deben reenviarse, esta tabla llamada *Label Information Base* (*LIB*) contiene las relaciones entre las etiquetas y las *FEC*s.

2.3.4. Etiquetas y Relaciones de las Etiquetas

MPLS define formatos de etiquetas especificas para *ATM* y *Frame Relay*, así como un formato propio de esta tecnología deseable para ser utilizada en la mayoría de otros medios. Las etiquetas *ATM*, corresponden a prefijos *VPI/VCI* y pueden ser cuando mucho de 24 *bits* de longitud. Las etiquetas de *Frame Relay* corresponden a prefijos *DLCI* y las hay de 10 o 23 *bits* de longitud. La etiqueta genérica de MPLS tiene una longitud de 20 *bits*. Una etiqueta se utiliza localmente para representar el reenvío de paquetes.

Una etiqueta en su forma simple, identifica la ruta por la cual el paquete debe viajar. Las etiquetas van encapsuladas con un *header* de capa de enlace de datos del paquete. El *router* receptor examina el paquete para conocer el contenido de su etiqueta y determinar cual debe ser el siguiente *router* al que será enviado el paquete. Una vez que el paquete ha sido etiquetado, el resto de su viaje a través de la red *MPLS* se basa en la conmutación por etiquetas. El valor de las etiquetas sólo tiene significado local, es decir, estas aplican sólo entre las interfaces de los *LSR*s conectados en los extremos del enlace.

Una vez que el paquete ha sido clasificado dentro de una *FEC*, se le asigna una etiqueta. Los valores de las etiquetas se derivan de algún protocolo de capa de enlace de datos; en redes de datos como *Frame Relay* o *ATM* pueden utilizarse como etiquetas los: *Data Link Connection Identifier* (*DLCI*) en el caso de redes de *Frame Relay* o los *Virtual Path Identifier/Virtual Channel Identifier* (*VPI/VCI*), en el caso de redes *ATM*.

Las etiquetas se ligan con una *FEC* como resultado de algún evento o política que indica la necesidad de dicha relación. Estos eventos pueden ser relaciones del tipo *Data-Driven* o *Control-Driven*. Siendo estas últimas preferibles por sus propiedades avanzadas de escalabilidad que pueden utilizarse en *MPLS*.

Las decisiones para la asignación de etiquetas, pueden basarse en criterios de reenvio, tales como:

- *Unicast Routing* (destinos específicos).
- Ingeniería de tráfico.
- *Multicast*.
- *Virtual Private Network* (*VPN*).
- *Quality of Service* (*QoS*).

El formato de una etiqueta *MPLS* genérica se ilustra en al figura 2.3. La etiqueta puede estar empotrada en el *header* de la capa de enlace de datos (*ATM –VPI/VCI-* en la figura 2.4 y *Frame Relay –DLCI-* en al figura 2.5 o al calce entre los *headers* de la capa de enlace de datos y capa de red como se muestra en la figura 2.6.

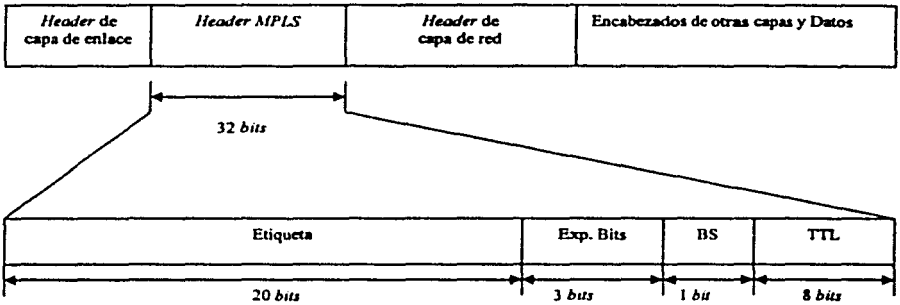

Figura 2.3 Formato de etiqueta *MPLS* genérica

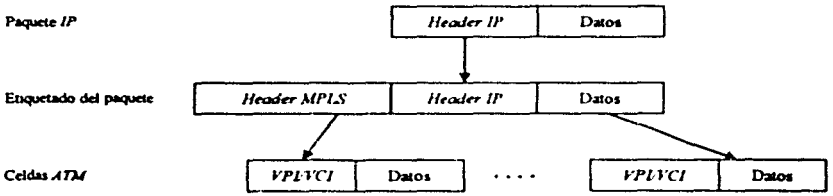

Figura 2.4 Formato de etiqueta para *ATM*

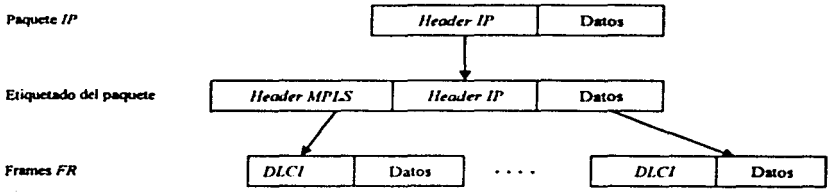

Figura 2.5 Formato de etiqueta para Frame Relay

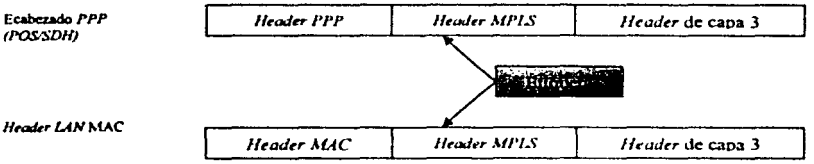

Figura 2.6 Formato de etiquetas para protocolos *Point to Point (PPP / Ethernet)*

2.3.5. Creación de Etiquetas

Existen varios métodos para la creación de etiquetas:

- ***topology-based***— (Método basado en Topologías) utiliza los procesos normales de los protocolos de *routing*, tales como: *OSPF* y *BGP*.
- ***request-based***— (Método basado en Peticiones) utiliza los procesos de control de tráfico basados en peticiones, tales como: *RSVP*.
- ***traffic-based***— (Método basado en Tráfico) utiliza la recepción de cierto paquete para iniciar la asignación y distribución de una etiqueta.

Los primeros dos métodos son ejemplos de relaciones tipo *Control-Driven*, mientras que el tercero es un ejemplo de relaciones tipo *Data-Driven*.

2.3.6. Distribución de Etiquetas

La arquitectura de *MPLS* no define un método único de señalización para la distribución de etiquetas. Los protocolos de *routing* existentes, tales como *BGP, IS-IS* y *OSPF* tienen la capacidad de transportar la información de las etiquetas dentro del contenido del protocolo. Así mismo el protocolo *RSVP*, soporta el intercambio de etiquetas.

La *IETF* ha definido un nuevo protocolo conocido como *Label Distribution Protocol* (*LDP*) para la señalización explicita y el manejo de etiquetas. Además se han definido extensiones al protocolo *LDP* básico, para soportar *routing* basado en *QoS* y *CoS*, estas extensiones están dentro de la definición del protocolo *Constraint-Based Routing* (*CR-LDP*).

Un resumen de los diferentes esquemas de intercambio de etiquetas, se presenta a continuación:

- *LDP*— mapea direcciones de destino *IP Unicast* en etiquetas.
- *RSVP*, CR–*LDP*— utilizado para ingeniería de tráfico y reservación de recursos.
- *Protocol-Independent Multicast* (*PIM*) — utilizado para enunciar el mapeo entre direcciones *Multicast* y las etiquetas.
- *BGP*— etiquetas externas (*VPN*)

2.3.7. *LSP* (*Label-Switched Path*)

Un conjunto de dispositivos habilitados para *MPLS* forman un dominio *MPLS*, dentro de ese dominio se crea una ruta para que un paquete viaje basándose en una *FEC*. La *LSP* se crea antes de la transmisión de los datos, y *MPLS* provee las siguientes dos opciones para la creación de una *LSP*:

- ***hop-by-hop routing***— (*routing* salto por salto) Cada *LSR* selecciona independientemente el siguiente salto para una cierta *FEC*. Esta metodología es la utilizada actualmente en las redes *IP*. El *LSR* utiliza cualquier protocolo disponible como: *OSPF*, *ATM private network-to-network interface* (*PNNI*.), etc.
- ***explicit routing***— (*routing* explícito) El *LER* de ingreso (es decir el *LSR* donde el flujo de datos entra a la red *MPLS* por primera vez) especifica la lista de los nodos que forman la *LSP*. La ruta especificada podría no ser óptima del todo, por lo que junto con la ruta pueden reservarse recursos para asegurar *QoS* al tráfico de datos. Esto facilita la ingeniería de tráfico a través de la red y pueden suministrarse servicios diferenciados utilizando flujos basados en políticas y métodos de administración de redes.

El establecimiento de una *LSP* para cierta *FEC* es unidireccional, el tráfico de regreso debe establecer otra *LSP*.

2.3.8. Clasificación de Etiquetas

Las etiquetas utilizadas por un *LSR* en las relaciones *FEC*-etiqueta se pueden categorizar como sigue:

- ***per platform***— (por plataforma) Los valores de las etiquetas son únicos en cada *LSR*. Dos etiquetas distribuidas en interfaces diferentes jamás tendrán el mismo valor. Las etiquetas se asignan de un grupo de etiquetas común para todo el *LSR*.
- ***per interface***— (por interfaz) Los intervalos de etiquetas están asociados a las interfaces. Se definen múltiples grupos de etiquetas para las interfaces, y las etiquetas asignadas a las interfaces provienen de diferentes grupos. Los valores de las etiquetas asignadas a distintas interfaces podrían ser los mismos.

2.3.9. Combinación de Etiquetas

Los flujos de tráfico de entrada de las diferentes interfaces pueden combinarse y conmutarse en forma conjunta haciendo uso de una etiqueta común si estos viajaran hacia el mismo destino final. Esto es conocido como encadenamiento de flujos o agregación de flujos. Si la red de transporte usada es una red *ATM*, el *LSR* podría emplear la combinación de ruta virtual (*VP-merge*) o bien canal virtual (*VC-merge*).

2.3.10. Memoria de Etiquetas

MPLS define dos modos para el tratamiento de las relaciones de etiquetas recibidas de los *LSR*s que no son el siguiente salto para un determinado *FEC*, estos son:

- ***conservative***— (conservativo) en este modo, las relaciones entre una etiqueta y una *FEC* recibida de los *LSR*s que no son el siguiente salto para una determinada *FEC*, son descartadas. Este modo requiere que un *LSR* mantenga pocas etiquetas. Es el modo recomendado para utilizar los *ATM-LSR*s.
- ***liberal***— (liberal) en este modo las relaciones entre una etiqueta y una *FEC* recibida de un *LSR* que no es el siguiente salto para una determinada *FEC*, son retenidos. Este modo permite una rápida adaptación a los cambios de topología y permite la conmutación del tráfico a través de otras *LSP*s en caso de cambios.

2.3.11. Control de Etiquetas

MPLS define dos modos para controlar la distribución de etiquetas hacia la red de *LSR*s:

- ***independent***— (independiente) En este modo, el *LSR* reconoce una *FEC* particular y la relaciona a una etiqueta, independientemente de distribuir las relaciones a sus puntos de conexión. Las nuevas *FEC*s se reconocen sin

importar que nuevas rutas se hagan visibles al *router*, pues estas convergen casi de manera inmediata durante el proceso de *routing*.

- ***ordered***— (ordenado) en este modo, un *LSR* relaciona una etiqueta con una *FEC* en particular, si y sólo si ésta, es del *router* de egreso o si ha recibido una relación de etiquetas para esa *FEC* desde el *LSR* de su próximo salto, por lo que el establecimiento de la *LSP* queda determinado de salida a entrada. Este modo es recomendado para *ATM–LSR*s.

Aún cuando cada uno de los métodos ofrece sus propias ventajas y desventajas, ambos métodos afectan la forma en que las *FEC*s son seleccionadas para relacionarlas a las etiquetas.

Con el método ordenado la selección de *FEC*s queda determinada por el *LSR* que inicia la *LSP* y con ello todos los *LSR*s que se encuentren en la misma *LSP* utilizarán dicha *FEC*, por lo que los *LSR*s deberán estar habilitados para determinar correctamente el próximo salto para la *FEC* en cuestión. Este método facilita la migración de una red convencional hacia una red *MPLS*, pues el administrador cuenta con el control del reenvío de los paquetes configurando listas de acceso en los *LER*s que iniciaran el levantamiento de las *LSP*s, sin embargo la complicación comienza con el tiempo de levantamiento de las *LSP*s, pues la propagación de la ruta podría tardar lo suficiente como para que los paquetes sean desechados o que estos tuvieran que ser procesados varias veces, lo que es una situación indeseable

El método independiente permite que cada *LSR* defina el número de particiones que hará de sus paquetes para relacionarlos a las *FEC*s, pero ello podría arrojar que cada *LSR* tomara decisiones distintas de otro de sus vecinos y no sería posible establecer ninguna *LSP* para algunas *FEC*s. Aún cuando esta no es la situación más común, ya que todos los LRS son previamente configurados, bien podría suscitarse cuando un *LSR* agregara nuevas rutas hacia los límites de una red *MPLS*. Bajo este método la migración hacia una red *MPLS* requiere de entrada que todos los equipos sean configurados al igual que con el método ordenado, pero este ofrece la ventaja de que los tiempos de convergencia no serían problema pues los tiempos de propagación

coinciden prácticamente con el establecimiento de las *LSP*s, aún cuando estas sean nuevas. En términos generales se puede percibir que una combinación de ambos métodos de control podrían favorecer la administración de una red *MPLS*.

2.3.12. Mecanismos de Señalización

- ***label request***— (petición de etiqueta) Utilizando este mecanismo, un *LSR* solicita una etiqueta de su vecino en cascada decendente que se encuentre sobre la misma *LSP*, y que esta relacionada a una *FEC* en especifico. Este mecanismo puede emplearse en el encadenamiento de *LSR*s, desde el *LER* de ingreso hasta el *LER* de egreso de la red *MPLS*.
- ***label mapping***— (mapeo de etiqueta) En respuesta a una petición de etiqueta, un *LSR* en cascada (ascendente o descendente) enviará una etiqueta al *LSR* iniciador usando el mecanismo de mapeo de etiqueta.

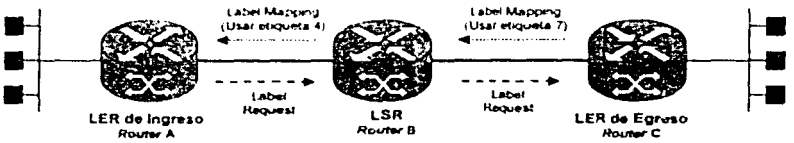

Figura 2.7 Mecanismos de señalización

2.3.13. *LDP* (*Label Distribution Protocol*)

El *LDP* es un nuevo protocolo para la distribución de la información de la relación de etiquetas hacia los *LSR*s en una red *MPLS*. Este protocolo se usa para mapear las *FEC*s a etiquetas, las cuales en su momento sirven para crear las *LSP*s. Las sesiones de *LDP* se establecen entre equipos no necesariamente adyacentes dentro de una red *MPLS*. Entre de las características de este protocolo se encuentran:

1. Provee un mecanismo de descubrimiento de *LSR*s ayudando a que pares de éstos se encuentren el uno al otro y establezcan comunicación.
2. Permite el intercambio de 4 tipos de mensajes *LDP*:
   - ***discovery messages***— (mensajes de descubrimiento) anuncia y mantiene la presencia de un LSR dentro de la red MPLS.
   - ***session messages***— (mensajes de sesión) establece, mantiene y termina las sesiones de LDP entre los equipos que la realizan.
   - ***advertisement messages***— (mensajes de aviso) crea, cambia y borra los mapeos de etiqueta a FEC.
   - ***notification messages***— (mensajes de notificación) provee información de notificación e información de señal de error.
3. Corre sobre *TCP* para proveer seguridad en la entrega de los mensajes (excepto los mensajes *Discovery*).
4. Esta diseñado para extenderse fácilmente a través de mensajes especificados como colecciones de objetos codificados *TLV* (*type, length, value*).

El campo *type* permite especificar que clase de objeto es este, el campo *length* especifica la longitud del objeto y el campo *value*, su significado depende del campo *type*.

2.3.13.1. Descubrimiento de vecinos *LSR*

El protocolo de descubrimiento *LDP* corre sobre *UDP* de la siguiente manera: Los *LSR*s envian periódicamente mensajes *HELLO* en *Multicast* hacia un puerto *UDP* común en todos los *routers* que componen la subred del grupo *Multicast*, de manera que en algún punto un *LSR* aprende de todos los otros *LSR*s con los que tiene conexiones directas utilizando *TCP*, siendo el momento para establecer una sesión *LDP* bidireccional, donde ambos *LSR*s anuncian y solicitan la relaciones de etiquetas-*FEC*

Así también pueden descubrir a otros *LSR*s que no se encuentran dentro de la misma subred, a través de mensajes *Unicast* que envian periódicamente hacia puertos *UDP* especificos de una *IP* especifica que pudo haber aprendido de otra manera (pre-configuración). Un ejemplo muy claro de esto es cuando un *LSR* de una subred A

quiere enviar paquetes etiquetados a través de una *LSP* que llega a un *LSR* perteneciente a una subred B. En estos casos el *LSR* de A requiere de aprender sobre las etiquetas que deberá aplicar a los paquetes que enviará hacia el *LSR* en B.

2.3.13.2. Transportación confiable (*TCP*)

La decisión de correr *LDP* sobre *TCP* obedece a las ventajas que ofrece este protocolo por demás probado y a la notoria dificultad de diseñar un protocolo de control. Aún cuando la distribución de etiquetas no requiere de un control tan estricto como lo hace *TCP*, visiblemente se entiende que cada mensaje de requerimiento de etiquetas para establecer las relaciones con los paquetes enviados por una red *MPLS*, deben ser confirmados y todos aquellos mensajes sin confirmación forzosamente deberían contar con un tiempo de vida, por lo que *LDP* delega esta responsabilidad a *TCP* el cual maneja sólo un reloj para toda una sesión y evita la sobrecarga de administrar grandes cantidades de cronómetros para cada mensaje a confirmar. *LDP* tiene entonces la posibilidad de utilizar una serie de funcionalidades con las que *TCP* ya cuenta tales como: manejo de los mensajes de capas superiores en paquetes *IP*, enviar confirmaciones en paquetes de datos y control de flujo.

2.3.13.3. Mensajes *LDP*

En *LDP* existen 7 tipos de mensajes básicos, que son:

- *INITIALIZATION*
- *KEEPALIVE*
- *LABEL MAPPING*
- *LABEL WITHDRAWAL*
- *LABEL RELEASE*
- *LABEL REQUEST*
- *LABEL REQUEST ABORT*

Los mensajes de *INITIALIZATION* son enviados al inicio de una sesión de *LDP* para que dos *LSR*s acuerden sobre los parámetros y opciones que manejarán durante la sesión, como modo de asignación de etiquetas, valor de los cronómetros y el rango de las etiquetas a utilizar por ambos *LSR*s. Ambos envían sus mensajes de inicialización y responden con un mensaje de *KEEPALIVE* si los parámetros son aceptables, de no ser el caso la sesión se dá por terminada.

Los mensajes *KEEPALIVE* son enviados periódicamente a fin de asegurar que cada par de *LDP*s están funcionando correctamente. En caso de que se suspenda el envío de estos mensajes o alguno otro de *LDP*, el *LSR* asume que su contraparte o su conexión a este, está caída o ha terminado su sesión.

Los mensajes de *LABEL MAPPING* permiten anunciar las relaciones entre las etiquetas y las *FEC*s, en tanto los mensajes *LABEL WITHDRAWAL* son el proceso inverso, con la finalidad de actualizar las tablas de *routing* del *LSR* implicado a fin de desligar la relación entre las etiquetas y las *FEC*s antes establecidas.

Un mensaje de *LABEL RELEASE* es utilizado por los *LSR*s que han recibido mensajes de *LABEL MAPPING* y no requieren más de estos mapeos. Típicamente sucede cuando los *LSR*s liberados encuentran que el próximo salto para la *FEC* no corresponde al *LSR* anunciado. Entonces los *LSR*s liberan de esta manera las relaciones entre las etiquetas y las *FEC*s cuando están operando en modo conservativo de retención de etiquetas.

Volver a llamar a estos *LSR*s podría generar la operación de asignación de etiquetas en modo *unsolicitated-downstream* o en modo *downstream-on-demand*. Operando bajo del segundo modo los *LSR*s requieren mapeo de etiquetas de sus vecinos en cascada utilizando mensajes de *LABEL REQUEST*. Si uno de estos mensajes necesita ser revocado antes de recibir respuesta, el *LSR* que recibe el requerimiento lo desecha y envía un mensaje de *LABEL REQUEST ABORT*.

2.3.13.4. Modos de distribución de etiquetas

Durante el proceso de inicialización de una sesión *LDP* se llevan al cabo todas las negociaciones necesarias entre los *LSR*s; la distribución y asignación de etiquetas puede llevarse a cabo de diferentes modos, por ejemplo, la asignación de etiquetas puede realizarse en *downstream–on-demand* o bien en modo *unsolicitated-downstream*, el control de las *LSP*s y el mecanismo de retención de etiquetas (liberal o conservativo).

En modo conservativo sólo se retienen los mapeos de etiqueta a *FEC* durante el tiempo que le son necesarios y en modo liberal los *LSR*s retienen todos los mapeos que le han sido anunciados, aún cuando estos no le sean de utilidad directamente, por ejemplo: si un *LSR* A anuncia a sus vecinos la relación entre alguna *FEC* y una etiqueta, y el vecino *LSR* B no lo ve como su próximo salto, no utilizará el mapeo que *LSR* A le anunció al menos por ese momento, pero sin embargo, lo almacenará y si en un momento *LSR* B actualiza su tabla de reenvío de etiquetas que hacen a *LSR* A su siguiente salto, entonces utilizará el mapeo que este le anunció para reenviar los paquetes etiquetados a través de la nueva ruta. La ventaja principal de operar bajo este modo es la velocidad de respuesta a los cambios en el *routing*, aunque la mayor desventaja es el gran desperdicio de etiquetas. Los equipos *ATM LSR*s comúnmente utilizan este modo de retención.

2.3.13.5. Elementos de *ATM*

Los principales elementos a considerar son:

- Encapsulación paquetes etiquetados en enlaces *ATM* (Ver fig. 2.8)
- *Looping* y ajustes al cronómetro *TTL*
- Intervalo de celdas y *VC-merge*

La encapsulación de paquetes etiquetados en enlaces *ATM* se basa en cargar la etiqueta en el campo *VPI/VCI*. El *header* del arreglo de etiquetas esta ligado al paquete de la capa de red antes de que dichos paquetes sean segmentados dentro de celdas.

Se representa un *AAL5 PDU*, constituido por el paquete de capa de red ligado a su arreglo de etiquetas al frente y seguido al final por un *trailer AAL5*. Posteriormente este *AAL5 PDU* será segmentado en celdas *ATM*, las cuales cargarán su etiqueta en el campo *VPI/VCI* de su *header*.

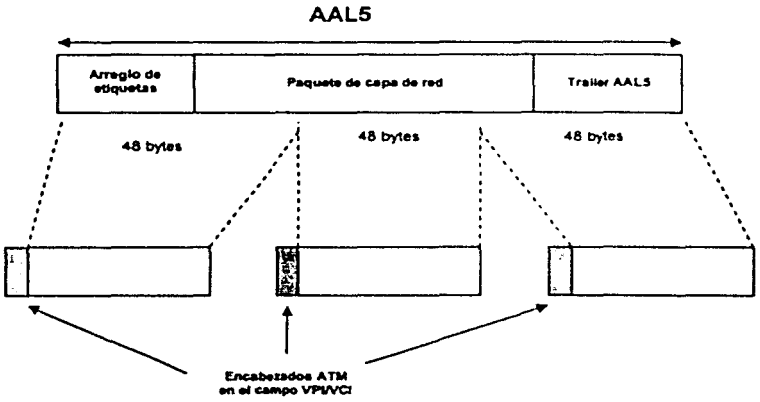

Figura 2.8 Encapsulación de un paquete etiquetado sobre un enlace *ATM*

La razón por la cual la etiqueta es cargada en el *header* de la celda y también en el *AAL5 PDU*, es para permitir arreglos de etiquetas de longitud arbitraria. La etiqueta más significativa del arreglo esta cargada en el campo *VPI/VCI* del *header* de *ATM* de manera que los *ATM-LSRs* pueden leerla y utilizarla para definir su reenvio.

Lo anterior representa un gran problema porque los paquetes con sólo una etiqueta, podrían no tener un *header* de arreglo de etiquetas, mientras los arreglos con

más de una etiqueta podrían tener un *header* de arreglo de etiquetas, y podría no haber forma de decir si el arreglo de etiquetas estuvo presente o no.

Otra función del arreglo de etiquetas es que este contiene el valor de los *bits* en los campos *TTL* y *EXP*. Estos campos podrían ser requeridos si el paquete fuera a ser conmutado por un *ATM-LSR* después de ser reensamblado.

La especificación *MPLS ATM* introduce una sutil distinción entre la interfaz "*Label Switching Controlled ATM* (*LC-ATM*)" y la interfaz convencional *ATM*. Una interfaz *LC-ATM* es aquella donde las etiquetas asignadas por un procedimiento de control de *MPLS* son cargadas al campo *VPI/VCI* de las celdas *ATM*. En una interfaz *ATM* convencional los campos de *VPI/VCI* podrían contener los valores asignados por los procedimientos de control *ATM*. Cuando un conmutador *ATM* opera como *LSR* todas sus interfaces son *LC-ATM* y en un conmutador *ATM* que no opera como *LSR* sólo se establecen circuitos convencionales de *ATM* en interfaces convencionales de *ATM*.

En referencia al tema de *loops*, lo más relevante es que maneja el problema de levantamiento de enlaces en ausencia de un *TTL* en *ATM*, sin embargo hay otros elementos de *TTL* a considerar. Lo primero es que el *TTL* de un paquete debiera ser ajustado correctamente es decir ser decrementado por el número de *ATM-LSR*s por los cuales este pase, sin embargo como esto no es posible esta tarea la realizan los no-*ATM-LSR*s que transmiten los paquetes sobre enlaces *LC-ATM*. Estos *LSR*s aprenden la forma apropiada para ajustar el *TTL* usando el campo de conteo en el mapeo de etiquetas. Si el ajuste lleva al *TTL* al valor cero o negativo, el paquete no es enviado como un paquete etiquetado a través de la región de los *ATM-LSR*s. De una u otra forma este es eliminado y el emisor recibe un mensaje *ICMP* o este es enviado salto a salto como un paquete no etiquetado y el *TTL* es decrementado en uno en cada salto, esto hará que el valor *TTL* alcance el valor de cero en el *LSR* apropiado lo que es importante para el diagnóstico de herramientas como *traceroute*.

El último elemento es el intervalo de celdas, el cual se basa en la asignación de etiquetas bajo el modo *downstream-on-demand* y utilizando múltiples etiquetas por

FEC o *VC-merge*. Una solución adicional propuesta para *MPLS* es el método *VP-merge*, en el cual la etiqueta se carga en el campo *VPI* de las celdas *ATM* y el campo *VCI* es usado como un identificador para distinguir los *frames* que vienen de diferentes fuentes pero son enviados a través del mismo enlace con la misma etiqueta en el campo *VPI*. Los *ATM-LSR*s reenvían basándose en el *VPI* y rescriben el valor de éste como lo hacen los conmutadores convencionales *ATM*.

Este método tiene sus desventajas, la primera es que limita el número de etiquetas al tamaño del espacio del *VPI* el cual a lo más es de doce *bits*, la segunda desventaja es que requiere de un identificador único para asignarlo al ingreso de cada *LSR*, el cual requiere de un grado extra de administración y configuración. La tercera desventaja es que la mayoría del hardware de conmutación no puede manejar *Early o Partial Packet Discard* (*EPD o PPD*) cuando se conmuta sólo con el *VPI*, pues estas características apuntan a descartar todos los paquetes de datos cuando se llega a eliminar sólo una celda del paquete de datos.

2.4. Estructura y operación de una red *MPLS*

La descripción de las componentes funcionales de una red *MPLS* refleja las diversas funciones con que cumple cada elemento que la conforma, sin embargo es importante resaltar que la estructura típica de una red *MPLS* sobresalen 4 elementos básicos:

- Los dispositivos *LER* que se encuentran ubicados en la frontera de la red *MPLS* y que son los responsables de la aplicación y retiro de etiquetas a los paquetes que circulan por la red *MPLS*.
- Los dispositivos *LSR*, mismos que se encargan de conmutar los paquetes etiquetados o las celdas basadas en etiquetas y que también tienen la capacidad de soportar *routing* (capa 3) o conmutación (capa 2) de paquetes, además de la conmutación de etiquetas.

- El protocolo *LDP* se utiliza en conjunto con los estándares de protocolos de *routing* de capa de red, para distribuir la información de etiquetas entre los dispositivos de la red *MPLS*.
- Topología del *backbone* en malla.

Una red *MPLS* se compone de equipos *LER* que se encuentran alrededor de la malla de equipos *LSR*, la cual constituye el corazón de la red. Las redes de usuarios o redes de acceso las cuales manejan el tráfico ordinario de *IP*, se conectan directamente a los *LER*s para enviar y recibir su información a través de la red *MPLS*. La figura 2.9 muestra la conexión de los elementos que estructuran una red *MPLS*.

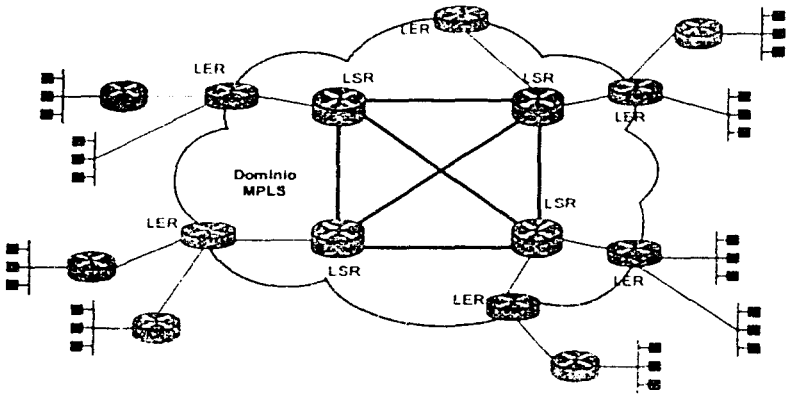

Figura 2.9 Estructura típica de una red *MPLS*

En *MPLS* la transmisión de datos ocurre a través de las *LSP*s, las cuales son una secuencia de etiquetas en todos y cada uno de los *LSR*s a través de una ruta desde un nodo fuente hasta un nodo destino, y que se establecen ya sea antes de la transmisión de datos (*control-driven*), o después de la detección de un cierto flujo de datos (*data-*

driven). La arquitectura de transporte hace parecer que cada par de *LSR*s están a sólo un salto, como si todos estuvieran unidos por una topología mallada donde la unión son precisamente las *LSP*s. Las etiquetas son marcadas por un identificador de protocolo específico y distribuidas por el *LDP*, protocolos de reservación de recursos como *RSVP* y protocolos de *routing* como: *BGP* y *OSPF*. Cada paquete de datos encapsula y transporta las etiquetas durante su recorrido desde la fuente hasta el destino, logrando la conmutación de datos en alta velocidad, pues las etiquetas de tamaño fijo fueron insertadas en el inicio del paquete o celda y son utilizadas por el *hardware* para conmutar los paquetes rápidamente entre los enlaces de la red. En un dominio *MPLS*, no toda la fuente de tráfico es enviada a través de la misma *LSP*. Dependiendo de las características del tráfico, se podrían crear diferentes *LSP*s para el tráfico de paquetes de diferentes características de *CoS*.

Deben tomarse en cuenta las siguientes acciones para hacer que un paquete de datos viaje a través de un dominio *MPLS*.

Acciones *MPLS*	**Descripción**
Creación de etiquetas y distribución de etiquetas.	• Antes de que comience a circular cualquier tráfico, los *LSR*s toman la decisión de pegar una etiqueta a una *FEC* en específico y construir sus tablas. • Respecto al protocolo *LDP*, los *LSR*s inician la distribución de etiquetas en cascada y las relaciones etiqueta - *FEC*. • Adicionalmente, las características del tráfico y las capacidades de *MPLS* se negocian usando el protocolo *LDP*. • Se debe utilizar un protocolo de transporte confiable y orientado a conexión, para el protocolo de señalización. *LDP* se utiliza *TCP*.

Acciones *MPLS*	Descripción
Creación de tablas	• Con la relación de etiquetas, cada *LSR* crea accesos en la *LIB*. • El contenido de la tabla especificará el mapeo entre una etiqueta y una *FEC*. i Realizando el mapeo entre el puerto de entrada y la tabla de etiquetas de entrada, hacia el puerto de salida y la tabla de etiquetas de salida. ii Los accesos son actualizados siempre y cuando ocurra la renegociación de la relación de etiquetas.
Creación de *LSPs*	• Las *LSPs* son creadas en dirección opuesta a la creación de los accesos en las *LIBs*.
Inserción de etiquetas / tabla de búsqueda.	• El *LER* de ingreso utiliza la tabla *LIB* o encuentra el siguiente salto y solicita una etiqueta para una *FEC* especifica. • Los *LSRs* subsecuentes sólo utilizan la etiqueta para encontrar el siguiente salto. • Una vez que el paquete alcanza el *LER* de egreso, la etiqueta es removida y el paquete entregado a su destino.
Reenvío de paquetes	• El paso de los paquetes por la red *MPLS* se da desde el *LER* de ingreso hasta el *LER* de egreso. • Si el *LER* de ingreso no cuenta con una etiqueta asignada a un paquete, la solicitara al *LSR* de primer salto que encuentre y este a su vez propagara la petición en cascada descendente a través de toda la red *MPLS*, hasta llegar el *LER* de egreso. • El *LER* de egreso regresara una etiqueta al *LSR* que lo antecede y propagará la señal hasta el *LER* de ingreso en modo de cascada ascendente. Si se requiere de ingeniería de tráfico, se utilizará *CD-LDP* para determinar la ruta a utilizar y que permita manejar los requerimientos de *QoS/CoS*. • El *LER* de ingreso inserta la etiqueta y reenvia el paquete hacia el *LSR* de primer salto.

Acciones *MPLS*	Descripción
Reenvío de paquetes Continúa..	• Los subsecuentes *LSR*s recibirán el paquete y remplazarán la etiqueta de entrada por otra de salida para proceder a reenviarlo. • El *LER* de egreso recibe el paquete y retira la etiqueta para proceder a liberarlo hacia su destino fuera del dominio *MPLS*.

2.5. Beneficios

La conmutación de etiquetas tiene la enorme ventaja de manejar por separado las funciones de control y las de reenvío. Cada una de ellas puede trabajar por separado sin impactar a la otra, permitiendo un desarrollo sencillo, fácil, menos costoso y menos propenso a errores.

El reenvío se basa en el intercambio de etiquetas. Cuando un conmutador de etiquetas recibe un paquete con una etiqueta, la etiqueta se utiliza como un índice en la base de datos de etiquetas (*LIB*). Cada campo en esta base de datos consiste en: etiqueta de entrada, puerto de entrada, etiqueta de salida, puerto de salida. El conmutador reemplaza la etiqueta de entrada en el paquete con la etiqueta de salida y envía el paquete por el puerto de salida.

La función de control se lleva al cabo por protocolos de *routing* en conjunto con procedimientos *MPLS* para la asignación y distribución de etiquetas, que permiten la conformación de las *LSP* y se encarga de mantener el sistema actualizado en caso de cambios en la topología de la red. Los protocolos de distribución de etiquetas son una parte importante de la función de control, pues son los responsables de establecer sesiones entre los *LSR*s e intercambian la información de etiquetas que necesita la función de reenvío.

Una de las mayores ventajas que trae consigo *MPLS*, es que es una tecnología de conmutación de etiquetas basada en estándares, lo que permitirá la interoperabilidad entre los diferentes fabricantes de soluciones. Inicialmente los beneficios más palpables en la actualidad son:

- Manejo de rutas explícitas –Las *LSP*s son mucho más eficientes que el *routing* convencional de *IP*, pues permiten transportar cualquier tipo de tráfico a través de los túneles que soportan los *LSR*s intermedios, los cuales sólo ven etiquetas de *MPLS* contenidas en los paquetes que se envían por estos.
- Soporte multiprotocolo y multienlace —La componente de reenvío de conmutación de etiquetas no es especifica para un protocolo de capa de red, es decir, lo mismo se trasporta tráfico *IP* que *IPX*, y lo mismo sucede con protocolos a nivel de capa de enlace de datos , aunque con especial énfasis sobre *ATM*.
- *Routing* Interdominio –La conmutación de etiquetas permite una clara separación entre el *routing* Interdominio e Intradominio, pues mejora los procesos de *routing* que permiten que gran cantidad de tráfico no requerido al interior de la red, sea claramente separado, evitando los problemas de sobrecarga asociados con el engranaje de las redes del tipo *IP-ATM*, por ejemplo
- Soporta todo tipo de tráfico – *Unicast*, *Unicast* con *ToS* y *Multicast*.
- Integra *IP* y *ATM* en la red –provee un puente entre el acceso *IP* y *ATM*, reutilizando el *hardware* de *routing*/conmutación y acoplando eficazmente dos redes dispares a través de la eliminación de complejos procedimientos y protocolos que manejan elementos como la resolución de direcciones.
- Soporta *QoS* y *CoS* para diferenciación de servicios –utiliza la configuración de rutas a través de ingeniería de tráfico y ayuda a conseguir las garantías de nivel de servicio.

Otro de los beneficios notorios que podemos enmarcar en la tecnología *MPLS* es el *routing* basado en restricciones (*Constraint-based routing* – *CR*-). Este protocolo toma en cuenta parámetros tales como características del enlace (ancho de banda, retardo, etc.), número de saltos y *QoS*. Una *LSP* puede ser *CR-LSP*, cuando las restricciones

sean un número de saltos explícito o cuando haya requerimientos de *QoS*. El número explícito de saltos determina la ruta que se tomará, y los requerimientos de *QoS* determinan qué enlaces y qué mecanismos de encolamiento o calendarización se emplearán para tráfico.

Cuando se usa *CR*, es enteramente posible que se seleccione una ruta más larga (en términos de costo), pero con menos carga. Sin embargo, es necesario que el camino seleccionado satisfaga los requerimientos de *QoS* de la *LSP*, ya que *CR* incrementa la utilización de la red, debido a que adiciona mayor complejidad a los cálculos de *routing*. *CR* puede utilizarse junto con *MPLS* para establecer *LSP*s.

2.6. Aplicaciones

Brevemente se comentarán las características de estas aplicaciones y las ventajas que *MPLS* supone para ello frente a otras soluciones tradicionales (parte de la información se tomo de [BAR 2001] y [DAV 2000]).

Las principales aplicaciones que hoy en día tiene *MPLS* son:

- Ingeniería de tráfico
- Diferenciación de niveles de servicio mediante clases (*CoS*)
- Servicio de redes privadas virtuales (*VPN*)
- *Multicast*

2.6.1. Ingeniería de Tráfico

La ingeniería de tráfico es un proceso para mejorar la utilización de la red al crear una distribución uniforme o diferenciada del tráfico a través de la red. Un resultado importante de este proceso es la prevención de congestión en cualquier ruta. Es importante mencionar que la ingeniería de tráfico no necesariamente selecciona la ruta

más corta entre dos dispositivos, ya que, permite utilizar los segmentos de red menos usados y proporcionar servicios diferenciados.

En *MPLS* la ingeniería de tráfico se alcanza usando rutas explícitas. Las *LSP*s se crean independientemente, especificando diferentes rutas basadas en políticas definidas por el usuario. Sin embargo, llevar esto a la práctica, puede requerir una amplia intervención manual del operador de la red. Los protocolos *RVSP* y *CR-LDP* son dos herramientas para proveer ingeniería de tráfico dinámica, además de *QoS* en *MPLS*.

2.6.2. Clases de Servicio (*CoS*)

MPLS está diseñado para manejar servicios diferenciados, según el Modelo *DiffServ* del *IETF*. Este modelo define una variedad de mecanismos para clasificar el tráfico en un reducido número de clases de servicio, con diferentes prioridades. Según los requisitos de los usuarios, *DiffServ* permite diferenciar servicios tradicionales tales como el Web, correo electrónico o la transferencia de archivos (para los que el retardo no es crítico), de otras aplicaciones mucho más dependientes del retardo y de la variación del mismo, como son las de video y voz interactivo. Para ello se emplea el campo *ToS* (*Type of Service*), rebautizado en *DiffServ*. Esta es la técnica *QoS* de marcar los paquetes que se envían a la red.

MPLS se adapta perfectamente a ese modelo, ya que las etiquetas de *MPLS* tienen el campo *EXP* para poder propagar la clase de servicio *CoS* en el correspondiente *LSP*. De este modo una red *MPLS* puede transportar distintas clases de tráfico, ya que:

- el tráfico que fluye a través de un determinado *LSP*, se puede asignar a diferentes colas de salida en los diferentes saltos *LSR*, de acuerdo con la información contenida en los *bits* del campo *EXP*

- entre cada par de *LSR*s se pueden habilitar múltiples *LSP*s, cada una de ellas con distintas prestaciones y con diferentes garantías de ancho de banda.

2.6.3. Redes Privadas Virtuales (*VPN*)

Una red privada virtual o *VPN* (*Virtual Private Network*) se construye a base de conexiones realizadas sobre una infraestructura compartida, con funcionalidades de red y de seguridad equivalentes a las que se obtienen con una red privada. El objetivo de las *VPN*s es el soporte de aplicaciones intranet/extranet, integrando aplicaciones multimedia de voz, datos y video sobre infraestructuras de comunicaciones eficaces y rentables. Las *IP VPN*s son soluciones de comunicación *VPN* basada en el protocolo de red *IP* de la *Internet*.

Las *VPN*s tradicionales se han venido construyendo sobre infraestructuras de transmisión compartidas con características implícitas de seguridad y respuesta predeterminada. Tal es el caso de las redes de datos *Frame Relay*, que permiten establecer *Private Virtual Circuits* (*PVC*) entre los diversos nodos que conforman la *VPN*. La seguridad y las garantías las proporcionan la separación de tráficos por *PVC* y el flujo asegurado (*CIR*). Algo similar se puede hacer con *ATM*, con diversas clases de garantías. Los inconvenientes de este tipo de solución es que la configuración de las rutas se basa en procedimientos más bien artesanales, al tener que establecer cada *PVC* entre nodos, con la complejidad que esto implica al administrador de la red.(por los costos asociados). Si se quiere tener conectados en una topología lógica de malla, añadir un nuevo nodo implicaría la necesidad de reconfigurar todos los nodos y restablecer todos los *PVC*s, algo similar a la solución *IP* sobre *ATM*.

La forma de utilizar las infraestructuras *IP* para servicio de *VPN* (*IP VPN*), se hace a través de la construcción de túneles *IP* de diversos modos.

El objetivo de un túnel sobre *IP* es crear una asociación permanente entre dos extremos, de modo que funcionalmente aparezcan conectados en una especie de

tuberías privadas por las que no puede entrar nadie que no sea miembro de esa *IP VPN*.

Realmente, el problema que plantean estas *IP VPN*s es que están basadas en un modelo topológico superpuesto sobre la topología física existente, a base de túneles punto a punto (o circuitos virtuales) entre cada par de *routers* cliente con cada *VPN*. De ahí las desventajas en cuanto a la poca flexibilidad en la configuración y administración del servicio, así como en el crecimiento cuando se quieren añadir nuevos nodos. En una arquitectura *MPLS* se obvian estos inconvenientes ya que el modelo topológico no se superpone sino que se acopla a la red del proveedor.

En el modelo acoplado *MPLS*, en lugar de conexiones punto a punto entre los distintos clientes de una *VPN*, se efectúan conexiones *IP* a una "nube común" en las que solamente pueden entrar los miembros de la misma *VPN*. Las "nubes" que representan las distintas *VPN*s se implementan mediante los caminos *LSP*s creados por el mecanismo de intercambio de etiquetas *MPLS*. Los *LSP*s son similares a los túneles en cuanto a que la red transporta los paquetes del usuario (incluyendo las cabeceras) sin examinar el contenido, a base de encapsularlos sobre otro protocolo. Aquí está la diferencia: en los túneles se utiliza el encaminamiento convencional *IP* para transportar la información del usuario, mientras que en *MPLS* esta información se transporta sobre el mecanismo de intercambio de etiquetas, que no ve para nada el proceso de *routing IP*. Sin embargo, sí se mantiene en todo momento la visibilidad *IP* hacia el usuario, que no sabe nada sobre rutas *MPLS* sino que ve una *Internet* privada (Intranet) entre los miembros de su *VPN*. De este modo, se pueden aplicar técnicas de *QoS* basadas en el examen del *header IP*, y que *MPLS* puede propagar hasta el destino reservando ancho de banda, priorizando tráfico de acuerdo a la aplicación, estableciendo *CoS* y optimizando los recursos de la red mediante técnicas de ingeniería de tráfico.

2.6.4. Multicast

La primera gran ventaja que ofrece *MPLS* para *Multicast* podría observarse en la conmutación de etiquetas y como segundo gran beneficio podría ser la habilidad de

manejar *IP Multicast* en equipos *ATM-LSR*s removiendo la necesidad de mapeos complejos de *IP* a *Multicast ATM*.

Los paquetes *Multicast* utilizan la misma encapsulación que los paquetes *Unicast* con una excepción, cuando utilizan la codificación de un arreglo de etiquetas el protocolo de capa de enlace utiliza el identificador de protocolo de capa de red para indicar que los paquetes etiquetados *MPLS* son llevados dentro de los paquetes de datos en capa 2. Diferentes identificadores de protocolos de capa de red son utilizados tanto para *Unicast* como *Multicast*, lo que representa algunas ventajas. Primero que es fácil reconocer los paquetes *Multicast* sin necesidad de analizar primero la etiqueta. Segundo, esto permite que los paquetes de *Unicast* y *Multicast* utilicen diferentes categorías de etiquetas, lo que habilita la *Multicast LFIB* para ser definida en un punto basado en interfaz aún si el *Unicast LFIB* es definido en un punto basado en *LSR*.

Los *loops* son un elemento particularmente importante para *Multicast* porque los paquetes que están inmersos en *loops* podrían ser replicados por el reenvío de *Multicast* propiciando grandes cantidades de tráfico indeseable hasta que el *loop* es eliminado. El campo *TTL* en la encapsulación de *MPLS* maneja este problema al igual que lo hace en *IP*, por tanto, la principal área de preocupación son los segmentos no *TTL*, por ejemplo, los enlaces *LC-ATM*, desafortunadamente no han sido desarrollados los métodos para prevención de *loops*.

El soporte para *MPLS Multicast* es muy complicado pues existen muchos y muy diferentes protocolos de *routing* de *Multicast*, mientras cualquier protocolo de *Unicast* genera una tabla de *routing* que puede utilizarse como un camino consistente para el manejo de la distribución de etiquetas, diferentes protocolos de *Multicast* generan diferentes ordenamientos en el estado de reenvío que necesitan ser manejados de manera diferente en el establecimiento de *LSP*s.

Algunos protocolos de *routing* de *Multicast* como *PIM* en modo disperso (*PIM-SM*) crea dos tipos de estado de reenvío conocidos como árboles compartidos (*shared trees*) y árboles de fuente específica (*source-specific trees*). Un árbol compartido libera

paquetes desde cualquier emisor hacia los receptores, mientras que un árbol de fuente específica libera paquetes desde un emisor en especial a los receptores. Los emisores deben estar en el árbol compartido para recibir paquetes previamente no escuchados desde los emisores y unir árboles de fuente específica para crear una ruta más óptima desde emisores específicos. Esto puede crear problemas en un ambiente *MPLS* cuando se establece un *LSP Multicast* para ambos estados de reenvío. Los paquetes de un emisor necesitan ser enviados en ambos estados de reenvío para asegurarse que serán alcanzados por todos los receptores. Como resultado un receptor en estado de árbol de fuente específica recibirá copias duplicadas de todos los paquetes enviados por el emisor. Una posible solución para este problema es utilizar *MPLS* sólo en el estado de árbol de fuente específica el cual es típicamente creado en respuesta de grandes flujos de tráfico; y para el reenvío convencional de *IP*, bajo el estado árbol compartido.

2.7. *MPLS* y el modelo *OSI*

En los *drafts* emitidos por la *IETF*, presentan a *MPLS* como una solución basada en la idea de que cualquier protocolo de capa de red debería poder transportarse sobre cualquier arquitectura de capa de enlace de datos. Sin embargo a medida que el desarrollo de los trabajos avanza, es fácil percatarse que se habla claramente de una compatibilidad evidente con protocolos como *ATM* o *Frame Relay* y muy probablemente la tendencia en el desarrollo de más *drafts*, reflejará mayores consideraciones para infraestructura óptica de capa de enlace de datos (*SONET*). Parte de la información de este tema se tomó de [NET 1998]

2.7.1. Funcionalidad de la Capa de Enlace de Datos y la Capa de Red

Observando que *MPLS* se fusiona entorno a las capas de enlace de datos y capa de red, es conveniente hablar un poco más al respecto, a fin de entender mejor su función.

2.7.1.1. Capa de Enlace de Datos

La capa de enlace de datos es la responsable de encapsular la información proveniente de la capa de red añadiéndole un header y un *trailer* a los datos, que son la información del protocolo utilizado en el proceso de enlace de datos y enviarlos como *frames* hacia la capa física la cual finalmente los transmite a través de un medio determinado como flujos de *bits*.

La capa de enlace de datos provee los siguientes servicios a la capa de red:

1) Para transmitir la información que le envía:
    i) Acepta la dirección de un nodo adyacente con el cual le transmitirá los datos.
    ii) Acepta los paquetes de datos de longitud arbitraria que le envían de capa de red.
    iii) Se asegura de haber recibido correctamente los paquetes completos.
2) Para recibir la información que le enviará, inicia el proceso de negociación para transmitirle los paquetes de datos.

A fin de proporcionarle estos servicios a la capa de red, la capa de enlace de datos debe:

1) Proporcionar una secuencia a los *frames*, con al finalidad de asegurar que estos serán recibidos con la secuencia correcta y de presentarse errores, tener la posibilidad de reconformar la secuencia a través del mecanismo de recuperación de errores.
2) Agrega códigos de detección y corrección de errores a los *frames* de datos, con el fin de detectar cuando un error ha ocurrido.
3) Agrega información del proceso de comunicación a los *frames* de datos, con la finalidad de corregir problemas suscitados durante el proceso de transmisión, como podría ser la perdida *frames*.

4) Se asegura de enviar las cuotas adecuadas de *frames* para que estos puedan ser manejados durante el proceso de transmisión.
5) Verifica los *frames* no tengan errores, aplicando las medidas correctivas apropiadas cuando un error es detectado, por ejemplo, cuando se recibe una petición de retransmisión; sin embargo, no tiene toda la responsabilidad de la detección de errores, pues esta labor es desempeñada por las capas superiores.
6) Ordena los *frames* en la secuencia correcta para reconstruir el paquete.

2.7.1.2. Capa de Red

La capa de red maneja todos los problemas relacionados con llevar un paquete de información de un nodo a otro a través de una red cuando los mensajes deben pasar por un nodo intermedio porque la fuente y el destino no están directamente conectados. La capa de red a diferencia de la capa de enlace de datos, establece procesos de comunicación a través de todos los enlaces ligados al nodo y envía paquetes (*datagrams*) en lugar de *frames*.

La capa de red toma la información de la capa de transporte y la encapsula colocándole un *header* a los datos, este *header* contiene la información utilizada por esta capa durante el proceso de comunicación con las capas adyacentes y posteriormente pasa la información a la capa de enlace de datos.

Si se trata de un nodo intermedio, la capa de red se encarga de reenviar los paquetes hacia el siguiente nodo y así hasta su destino, aún cuando los paquetes provengan de diferentes nodos, protocolos de comunicación y esquemas de direccionamiento.

La capa de red se sitúa en los límites entre las subredes y las redes que las contienen, por lo que provee los siguientes servicios:

1) Un esquema de direccionamiento unificado, cada nodo tiene una dirección única y el esquema de direccionamiento es consistente para toda la red.

2) Establece y mantiene los circuitos virtuales en las redes de circuitos conmutados.
3) Maneja *routing* independiente para cada paquete a través de los nodos intermedios de una red de conmutación de paquetes.

Las redes de circuitos conmutados establecen circuitos entre pares de nodos y a través de estos se intercambia la información de manera bidireccional, es decir:

1) El *routing* se establece cuando se realiza una conexión.
2) Los paquetes no requieren de direcciones, porque se ha establecido el *routing* y los nodos intermedios saben de antemano el destino de los paquetes.
3) Si un nodo intermedio falla se rompe el circuito y se debe reestablecer a través de otra ruta antes de que la transferencia se reactive.

En las redes de conmutación de paquetes:

1) Cada paquete contiene la dirección de red del nodo destino.
2) Cada nodo intermedio decide el siguiente destino del paquete.
3) Cuando un nodo falla, la red redirecciona inmediatamente los mensajes a través de otra ruta.
4) Como los circuitos son compartidos, el servicio puede ser afectado, si la demanda es mayor a la capacidad del circuito, pero por eso mismo los enlaces son mejor utilizados.

2.7.2. Protocolos de Capa de Enlace de Datos y Capa de Red

Idealmente la premisa que *MPLS* debería cumplir es que todo protocolo de la capa de red debería correr en la capa de enlace de datos, y de resultar viable en algún momento, se podrá observar la utilización de los siguientes protocolos que a continuación se enuncian.

2.7.2.1. Protocolos de Capa de Enlace de Datos

A nivel de la Capa de Enlace de Datos existen protocolos para *LAN* y para *WAN*, como a continuación se enlistan:

2.7.2.1.1. Protocolos de capa 2 para *LAN*

a. *RFC* 1042 *Sub Network Access Protocol (SNAP)*, en conjunto con *LLC*.
b. *IEEE* 802.2 *Logical Link Control (LLC)*
c. *IEEE* 802.3 *Ethernet*, con el método de acceso *CSMA/CD*.
d. *IEEE* 802.5 *Token Ring*, con el método de acceso *Token Passing*.
e. *IEEE* 802.6 *MAN*, con el método de acceso *DQDB*.
f. *IEEE* 802.12 VG-Any *LAN*.
g. *ANSI Fiber Distributed Data Interface (FDDI)*, con método de acceso *Token Passing*.

2.7.2.1.2. Protocolos de capa 2 para *WAN*

a. *Asychronous Transfer Mode* (*ATM*).
   i *ATM* Adaption Layer (AAL).
      - AAL1 Constant Bit Rate (CBR).
      - AAL2 Variable Bit Rate (VBR).
      - AAL3/4.
      - AAL5 Data.
      - Virtual LAN (VLAN).
b. *Integrated Services Digital Network (ISDN)*.
c. *High-level Data Link Control (HDLC), para Frame Relay*.
d. *Point to Point (PPP)*.
e. *Switched Multi-megabit Data Services (SMDS)*.
f. *Synchronous Data Link Control (SDLC)*.

2.7.2.2. Protocolos para Capa de Red

De acuerdo al conjunto de protocolos se ubican los siguientes:

2.7.2.2.1. *TCP/IP*

a. *Internet Protocol (IP).*
   i *Serial Line IP (SLIP).*
   ii *Compressed Serial Line IP (CSLIP).*
   iii *Packet Level Protocol (X.25).*
   iv *Internet Control Message Protocol (ICMP).*
b. *Distance Vector Multicast Routing Protocol (DVRMP).*
c. *Internet Group Management Protocol (IGMP).*
d. *Resource Reservation Protocol (RSVP).*
e. *Routing Information Protocol (RIP).*
f. *Border Gateway Protocol (BGP).*
g. *Exterior Gateway Protocol (EGP).*
h. *Gateway to Gateway Protocol (GGP).*
i. *Interior Gateway Protocol (IGP).*
j. *Exterior gateway Protocol (EGP).*
k. *Open Shortest Path First (OSPF).*

2.7.2.2.2. *Novell Netware*

a. *Internetwork Packet Exchange (IPX).*
   i *Routing Information Protocol (RIP).*
   ii *NetWare Link Service Protocol (NSLP)*

2.7.2.2.3. *IBM*

a. *Path Control.*
   i *Data Link Switching (DSLw).*

ii *Switch to Switch Protocol (SSP).*
iii *Qualified Logical Link Control (QLLC).*
iv *High Performance Routing (HPR).*
v *Advanced Peer-to-Peer Networking (APPN).*

2.7.2.2.4. *ISO*

a. *Network Protocol*
   i *Connection-oriented network Protocol (CONP).*
   ii *Connectionless Network Protocol (CLNP).*
b. *Packet Level Protocol (X-25).*
c. *Intermediate System to Intermediate System (IS-IS).*
d. *End System to Intermediate System (ES-IS).*

2.7.2.2.5. *DECnet Phase IV*

a. *DECnet Routing Protocol (DRP).*

2.7.2.2.6. *XNS Xerox Network System*

a. *Internetwork Datagram Protocol (IDP).*
   i *Routing Information Protocol (RIP).*
   ii *Error protocol.*
   iii *Echo Protocol.*

2.7.2.2.7. *Apple Talk*

a. *Datagram Delivery Protocol (DDP).*
   i *Apple Talk Update-based Routing Protocol (AURP).*
   ii *Routing Table Maintenance Protocol (RTMP).*

2.7.2.2.8. *Banyan VINES*

a. *VINES Internet Protocol (VIP).*
   i *VINES Fragmentation Protocol (VFP).*
   ii *VINES Routing Update Protocol (VRTP).*
   iii *VINES Sequenced Routing Update Protocol (VSRTP).*
   iv *VINES Address Resolution Protocol (VARP).*
   v *VINES Sequenced Address Resolution Protocol (VSARP).*
   vi *VINES Internet Control Protocol (VICP).*

2.8. *MPLS* en la UNAM

En la tendencia del desarrollo tecnológico, la Universidad Nacional Autónoma de México ha sido la institución precursora en México que ha marcado la pauta en lo referente al desarrollo de las telecomunicaciones a nivel académico y por mucho tiempo la única opción para la iniciativa privada. La UNAM ha desempeñado este papel protagónico desde los inicios de las redes académicas (*BITNET*) que evolucionaron hasta convertirse en redes comerciales y finalmente en *Internet*.

En su obligación social y cultural dentro de su ejercicio académico y de investigación, la UNAM ha continuado con sus trabajos en torno al crecimiento tecnológico del país, participando activamente en grupos de trabajo nacionales e internacionales responsables de promover el desarrollo, instalación y uso de las diferentes tecnologías de telecomunicaciones, tales como *QoS*, *Multicast*, *IPv6*, *VoIP*, *MPLS* e *Internet2*. De esta manera la UNAM colabora con instancias académicas, gubernamentales e iniciativa privada a través de sus conocimientos, experiencias y su ininterrumpida labor de investigación.

El Grupo de Investigación de *MPLS* de la UNAM (www.mpls.unam.mx), es el responsable de colaborar en los trabajos de *MPLS* que se vienen dando actualmente a

nivel de pruebas, instalación y desarrollo de estándares, ya que como tal *MPLS* aún no termina de estandarizarse. En vías de que esto suceda la UNAM se ha dado a la tarea en primera instancia de establecer mecanismos de evaluación, pruebas e instalación que permitan dar un mayor y mejor servicio a sus usuarios, y dentro del proyecto de *Internet* 2 dar impulso al desarrollo de infraestructura y aplicaciones que requieren de este tipo de tecnologías para promover igualmente la participación académica en todos sus niveles, comenzando por el campo de la investigación y la educación.

El Grupo de Investigación de *MPLS* de la UNAM tiene su presencia en las instalaciones de la Dirección General de Servicios de Cómputo Académico y sus integrantes incluyen personal de la Dirección de Telecomunicaciones, de otras áreas de DGSCA, así como personal de otras instituciones académicas y compañías nacionales y extranjeras. Desde enero del 2000 se viene trabajando con un plan que incluye temas de estudio, evaluación, instalación y difusión de *MPLS*. Los objetivos específicos del grupo son:

1. Investigar, probar e implantar *MPLS* en las redes de *Internet* e *Internet* 2 de la UNAM y de CUDI.
2. Participar en el desarrollo de proyectos nacionales e internacionales de *MPLS*.
3. Participar en la implantación y difusión de *MPLS* y sus aplicaciones en redes avanzadas de México y Latinoamérica.

Dentro de los proyectos que actualmente se siguen destacan los de [UNA 2001]:

- Requerimientos para pruebas de *MPLS*
- Pruebas *MPLS* UNAM (Fase 1)
- Pruebas *MPLS* UNAM (Fase 2)
- Pruebas *MPLS* UNAM (Fase 3)
- Propuesta de instalación de *MPLS* en la red CUDI

- Tutorial de *MPLS*.
- Estudio de *MPLS* asociado con otras tecnologías.

Para mayor información en torno a estos trabajos referirse al sitio "http://www.*mpls*.unam.mx"

Este trabajo de tesis forma parte de los proyectos "Requerimientos para pruebas de *MPLS*" y "Pruebas *MPLS* UNAM (Fase 3)", cuyas pruebas y propuesta a desarrollar son:

Pruebas

- Etiquetado de tráfico *IP* en *Label Edge Routers* (*LER*).
- Reenvío de paquetes en *Label Switching Routers* (*LSR*).
- Análisis de desempeño en el reenvío y señalización en equipos.
- Establecimiento y desempeño de *Label Switched Paths* (*LSP*).
- Número total de *LSP*s que un equipo o red puede manejar.
- Transición de un *LSP* primario a un *LSP* secundario.
- Configuración de túneles e Ingeniería de Tráfico.
- Pruebas con *LDP* (*Unicast*), *CR-LDP* y *RSVP* (Ing. de Tráfico), *BGP* (*VPNs*) y *PIM* (*Multicast*).
- Pruebas en redes IP-*ATM*-*MPLS*.
- Reconfiguración dinámica de la red en cuanto a *QoS* y aplicaciones críticas como *VoIP* y video.
- Estudio y pruebas de *GMPLS*.

Propuesta

- Instalación de *MPLS* en la red *Internet2* de CUDI.

Capítulo 3: Pruebas de *MPLS*

3.1 ¿Por qué, qué y cómo probar?

En el vertiginoso desarrollo de la tecnología de telecomunicaciones se ha hecho patente un defasamiento entre las necesidades que cambian día con día y las soluciones disponibles en el mercado, ya que estas cambian constantemente. Los fabricantes suman más y nuevas funcionalidades a sus equipamientos, por lo que la selección de una solución no se hace una tarea tan evidente.

En este entendido, es cada vez más frecuente la búsqueda de soluciones a la medida, lo que ha implicado el desarrollo de escenarios de prueba que permitan simular las condiciones deseables para la implantación de una nueva red, de manera que se puedan prever distintas necesidades de producción. Tales pruebas permiten corroborar que la tecnología en prueba cumpla con los esquemas de conformidad, interoperabilidad y rendimiento que se esperan.

Sin embargo los escenarios de prueba deseables no siempre son viables por los costos que esto pueda implicar tanto para los proveedores de las soluciones como para los clientes y en tal caso se opta por obtener referencias similares a través de simuladores de redes, los cuales son comparativamente muy baratos y más versátiles en cuento a las opciones de configuraciones que ofrecen para generar pruebas, aunque no podrían reportar la información que provee un esquema de pruebas con equipos de diferentes marcas, pues se basan en estándares que se hacen interactuar idealmente.

A continuación se describen más detalladamente los elementos del por qué, qué y cómo realizar pruebas de tecnología de redes (gran parte de la información fue tomada de [ANG 2001] y [TEL 2000]).

3.1.1 ¿Por qué probar?

Desde hace algunos años los *routers* y conmutadores principales de una red han tenido que rediseñarse muy continuamente a fin de cumplir satisfactoriamente con los requerimientos del mercado. Actualmente estos requerimientos son:

- Soporte a interfaces ópticas de alta velocidad.
- Habilidad para manejar hasta 100,000 prefijos de direcciones a la vez que procesan un gran número de flujos de tráfico.
- Mecanismos para facilitar la ingeniería de tráfico.
- Proveer *QoS*.

La respuesta a estas necesidades se ha venido desarrollando gradualmente, tal que los nuevos *routers* cuentan con muchas de las funciones de *routing* y conmutación implícitas en el hardware. Muchos otros ya soportan *MPLS* además del *routing* tradicional de *IP* mediante *BGP*, *OSPF* e *IS-IS*.

Estos nuevos cambios en los equipos de *routing* han propiciado la necesidad de generar nuevos tipos de pruebas que prevean de alguna manera el comportamiento de los equipos en redes de producción.

3.1.2 ¿Qué probar?

3.1.2.1 Clases de pruebas

Se pueden realizar diferentes pruebas a los dispositivos de una red, y en términos generales se tipifican dentro de las siguientes: pruebas de conformidad, pruebas de interoperabilidad y pruebas de rendimiento.

Las pruebas de conformidad sirven para asegurarse que las implementaciones cumplen con los estándares establecidos. En el caso de las pruebas a los *routers* principales de una red, se implica la elasticidad de los mismos con los protocolos *BGP4*, *MPLS-BGP*, *RSVP-TE*, *LDP*, CR-*LDP*. Para cada protocolo el dispositivo bajo prueba (*DUT –Devise Under Test*) pasará un gran número de pruebas que servirán para verificar que los mensajes del protocolo tienen el formato correcto, que los *DUT*'s responden correctamente a los requerimientos y regresan los mensajes de error apropiados cuando reciben un requerimiento inválido.

Las pruebas de interoperabilidad sirven para verificar que diferentes implementaciones pueden trabajar en conjunto. A pesar de que hay una serie de pruebas distintas, la finalidad de ellas permite verificar la operación válida de funciones típicas como el establecimiento de conexiones, el intercambio y la actualización del *routing*, el establecimiento y la ruptura de *LSPs* y la recuperación de errores.

Las pruebas de rendimiento permiten medir la capacidad de los equipos bajo prueba tanto en condiciones normales como de sobrecarga. Con los nuevos *routers* centrales de red se han desarrollado nuevas técnicas para sustituir la metodología tradicional de pruebas para direccionar la nueva funcionalidad. Por ejemplo tradicionalmente las métricas de *routing* se enfocaban exclusivamente en la capacidad de los equipos bajo prueba para el manejo de los datos, métricas tales como *throughput*, *back to back*, latencia, y pérdida de paquetes. Sin embargo bajo una nueva perspectiva las pruebas de rendimiento para los *routers* centrales se enfocan particularmente a resolver las siguientes preguntas:

- ¿Cómo se afecta su rendimiento dependiendo del tamaño de la prueba de *routing*?
- ¿Qué pasa cuando maneja un gran número del flujo de tráfico?

- ¿Qué pasa con el tráfico de datos cuando están oscilando un gran número de *routers*?
- ¿Cuál es la diferencia en rendimiento entre *routing* en capa tres y conmutación de etiquetas?
- ¿Cómo afecta el rendimiento de una red el rol que cumple un *router* de *MPLS* (ingreso, tránsito o egreso)?
- ¿Cómo afecta el número de *LSP*s establecidos al costo de configuración de *LSP*?
- ¿Qué también se distribuye el tráfico cuando hay múltiples *LSP*s entre un par de *routers BGP*?
- ¿Qué tan rápido puede conmutarse el tráfico a una ruta secundaria si un enlace o nodo de la ruta primaria falla?

3.1.2.2. Tipos de pruebas para *MPLS*

Las pruebas de *MPLS* se pueden dividir en dos grupos principales de pruebas:

Pruebas de nodo único:

- Provee los resultados básicos de las pruebas de conformidad y desempeño de un equipo.
- Insuficiente para entender el comportamiento de *MPLS* de extremo a extremo de una red.

Pruebas de una red de nodos:

- Realizadas con una red de equipos
- Requerida para determinar a nivel de red la conformidad y le desempeño de la red *MPLS*.
- Alternativamente se pueden utilizar herramientas de simulación de una red *MPLS*.

Pruebas de nodo único: este tipo de pruebas se pueden dividir en dos subgrupos:

Pruebas de conformidad

Determinan la conformidad de las características obligatorias o deseadas de *MPLS*, implícitas en un determinado equipo.

La factibilidad de manejar los tres protocolos de distribución de etiquetas adecuadamente (*LDP*, CR-*LDP* y *RSVP-TE*).

Se enfocan en el análisis de:

- Formatos de mensaje.
- Clasificación de tráfico.
- Procedimientos de administración de etiquetas.
- Errores.

Pruebas de desempeño

Incluyen pruebas de rendimiento en el reenvío de paquetes:

- Determina el rendimiento de los *LSR*s.
- Compara el rendimiento de *MPLS* basándose en el rendimiento del reenvío *IP*.
- Se incluye la medición de métricas tales como *throughput*, latencia, rango de pérdida de paquetes, etc.

También incluyen pruebas de rendimiento de señalización:

- Determinación del rendimiento del plano de control.
- Relacionados a la capacidad de procesamiento de la entidad de señalización en el *LSR*.
- Medición y comparación de la eficiencia de los protocolos de distribución de etiquetas en términos de latencias y costos con respecto al control y administración de *LSP*s, por ejemplo; latencia de levantamiento de *LSP* como latencia de *re-routing* de *LSP*, máximo número de *LSP*s, etc

Pruebas de redes de nodos

El sistema bajo pruebas es una red con dos o más nodos.

Las pruebas *MPLS* para redes de nodos incluyen:

- Selección de ruta en base a *QoS*, *routing* explícito y *routing* basado en restricciones.
- *Re-routing* por fallas.
- Detección de *loops*.
- Clasificación de tráfico.
- Diferencias entre *LER*s y *LSR*s, por las características que soportan y su rendimiento.

Desafortunadamente este tipo de pruebas presentan una gran desventaja pues suelen ser costosas e ineficientes ya que se deben utilizar equipos altamente costosos, monitoreo de los enlaces entre los nodos para determinar la conformidad o la no conformidad, además de la gran cantidad de tiempo que consume el estar realizando la reconfiguración de diferentes topologías de red en forma manual.

3.1.3. ¿Cómo probar?

Por la complejidad que representa la conformación de una infraestructura de pruebas ajena a la infraestructura de operación de una red, resulta adecuado llevar a cabo este tipo de pruebas bajo las posibilidades que ofrezca la capacidad de cada instancia para realizar las mismas. Es por ello que deben seleccionarse las herramientas que permitirán satisfacer las expectativas sobre las pruebas *MPLS* que den respuesta a sus propios requerimientos. Si las pruebas se definen para utilizar dispositivos físicamente presentes se deben tomar en cuenta los siguientes requerimientos:

En el caso de pruebas de conformidad, las herramientas de prueba deberán:

- Generar paquetes *IP* con etiquetas, cumpliendo con las especificaciones de *MPLS*.

- Soportar múltiples tecnologías de capa dos (*ATM, FR, Ethernet, PPP*).
- Implementar protocolos de distribución de etiquetas (*LDP*, CR-*LDP*, *RSVP-TE*).
- Capacidad para el análisis de paquetes con etiquetas.
- Soportar una interfaz programable para implementar pruebas automatizadas.

En el caso de pruebas de rendimiento, adicionalmente se deberá:

- Tener la capacidad de generar tráfico independiente en múltiple flujos variando la longitud de paquete, costo, etc.
- Tener la capacidad de analizar múltiples flujos independientes.
- Contar con alta resolución para marcar los tiempos en las mediciones de retraso.
- Tener capacidad para generar un gran volumen de petición de etiquetas.

Otra forma de estimar los resultados esperados para la conformación de alguna configuración de red *MPLS* se puede llevar a cabo a través de herramientas simuladoras de redes. Estos simuladores deben incluir las siguientes características:

- La implantación de diferentes protocolos de etiquetas de distribución (*LDP*, CR-*LDP*, *RSVP-TE*).
- Capacidad de procesamiento para soportar las funciones del protocolo para cada nodo de la red.
- Rutas explícitas.
- Direccionamiento *IP* y protocolos de *routing* (*OSPF* o *RIP*).
- Encapsulación en capa dos.
- Configuración de topología a requerimiento del usuario.
- Facilidad de configuración por interfaz de usuario.

3.2. Esquema de pruebas

Entendiendo que el potencial de una red *MPLS* se basa en contar con una topología de red completamente mallada, se presenta en la figura 3.1 un esquema de pruebas.

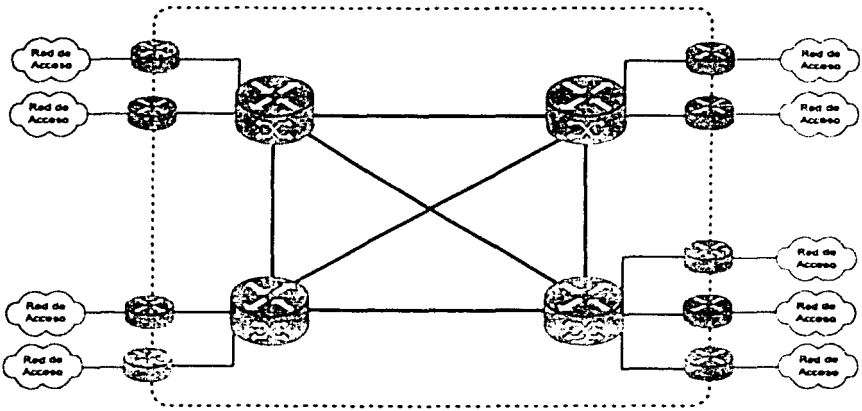

Figura 3.1 Esquema de pruebas

Este esquema ofrece un escenario de pruebas estructuralmente muy completo y básico para poder observar los beneficios que puede proporcionar una red *MPLS* visto desde el punto de vista práctico. Los elementos que lo componen son equipos cuyas características técnicas cumplen con los requerimientos de funcionalidad para una red de este tipo descritos en el apartado "2 5 Beneficios".

3.3. Requerimientos de equipo para esquema de pruebas

Para la realización de las pruebas es necesario contar con la infraestructura adecuada a tales fines. En términos generales deben considerarse los siguientes elementos:

- Equipos que cumplan con la funcionalidad de *LSR*.
- Un equipo generador-analizador de trafico *MPLS*.
- Equipos de cómputo para simulación de diferentes segmentos de redes.

Equipos que cumplan con las funcionalidad de *LSR*

Se sabe de la existencia de varios fabricantes en el mercado que actualmente cuentan con soluciones de *hardware* que cumplen con esta funcionalidad y entre ellos se encuentran *Cisco Systems, Foundry, Cabletron Systems, Juniper, Alcatel*, entre otros.

Estos equipos deben contar con interfaces de alta velocidad como son: *Fast Ethernet, GigaBit Ethernet, ATM y POS*.

Un equipo generador-analizador de trafico *MPLS*

Como su nombre lo indica son equipos que cuentan con la funcionalidad de generar y analizar el tráfico de paquetes con diferentes características como: *IP, MPLS, Multicast, Unicast, Broadcast*, etc. Asi también para el análisis de diferentes protocolos como: *TCP/IP, Novell Netware, Appletalk*, entre otros.

De estos equipos igualmente hay varios fabricantes que cuentan con todas estas características, entre estos los más relevantes son: *Agilent Technologies, Spirent Communications, Netwok Associates*, etc. esta ultima la compro Spirent recientemente.

Equipos de cómputo para simulación de diferentes segmentos de redes

Estos equipos pueden ser cualquier equipo de cómputo que permita el manejo de aplicaciones varias y se pueda conectar a una red *LAN*.

Capítulo 4: Metodología de pruebas

Existen metodologías para conocer de la factibilidad y el desempeño de diferentes tipos de *software* y *hardware*, sin embargo todas ellas buscan reportar los elementos más relevantes para cumplir con los requerimientos más especipificos de los usuarios y para este trabajo de tesis la metodología propuesta deja en claro todos los aspectos generales que se deben tomar en cuenta para llevar a cabo las propuestas de pruebas a desarrollar con la finalidad de obtener los resultados deseables a observar.

La metodología de pruebas propuesta incluye los siguientes aspectos:

- **Propósito**: Describe el objetivo general de la prueba.
- **Descripción**: Define en términos generales el desarrollo de la prueba.
- **Análisis**: Destacar la importancia de la realización de la prueba especifcada.
- **Escenario**: Define los supuestos de configuración deseable para llevar a cabo la prueba.
- **Configuración**: Especificar el esquema de pruebas a utilizar para la realización de la prueba.
- **Procesos**: Destacar los procesos más generales que tendrán a lugar para la realización de la prueba.
- **Procedimiento**: Describe los pasos a desarrollar durante la realización de la prueba, haciendo énfasis en los aspectos que se deben observar en cada caso tomando en cuenta el escenario y la configuración propuesta.
- **Monitoreo y resultados**: Define cuales deben ser los parámetros y la información a observar durante la realización de la prueba, para emitir los resultados obtenidos conforme a las variables propuestas.

Para la caracterización de los *routers* que se utilizan en las pruebas de funcionalidad de *MPLS*, aplicaciones de ingeniería de tráfico y configuración de *VPN*s, es necesario contar con un equipo con la capacidad de generar *LSP*s a través del

sistema bajo pruebas *SUT* (*System Under Test*), el cual debe contar con las siguientes componentes:

1. Permitir la utilización de protocolos de señalización de *MPLS* (*RSVP-TE* o *LDP*/CR-*LDP*), para el establecimiento dinámico de *LSP*s. Las extensiones de ingeniería de tráfico en esos protocolos son también importantes, ya que los *routers* rechazarían las peticiones del protocolo de señalización si no se cuenta con estas.
2. Permitir la configuración de *IGP* (extensiones de ingeniería de tráfico *OSPF* o *IS-IS*) permite anunciar la topología de una red simulada, pues los protocolos de señalización trabajan mejor en conjunto con este.
3. Tener capacidad para insertar etiquetas automáticamente y enviar paquetes etiquetados a través de rutas dinámicamente preestablecidas en una configuración de mallado completo desde cualquier interfaz hacia cualquier red anunciada en cualquier otra interfaz.

Aquí se enlistan las pruebas que se discutirán a continuación en base a la metodología propuesta (el desarrollo de las mismas hace referencia en algunas de sus partes a las propuestas por *Agilent Technologies* [AGI 2001-A], [AGI 2001-B] y [AGI 2002]):

4.1. Etiquetado de paquetes *IP* en *Label Edge Router* (*LER*).
4.2. Reenvío de paquetes en *Label Switching Router* (*LSR*).
4.3. Establecimiento y desempeño de *Label Switched Path* (*LSP*).
4.4. Establecimiento dinámico y número total de *LSP*s.
4.5. Transición de un *LSP* primario a un *LSP* secundario.
4.6. Configuración de *VPN*s en una red *MPLS*.
4.7. Pruebas en redes *IP-ATM-MPLS*.

4.1. Etiquetado de paquetes *IP* en *Label Edge Router* (*LER*)

Propósitos

Se probarán las funcionalidades de ingreso (Prueba A) y egreso (Prueba B) de un *LER*.

La prueba A permitirá verificar la capacidad de un *LER* de Ingreso (*DUT*) para insertar etiquetas en los paquetes que entran a una red *MPLS*, y enviarlos al *LSR* de siguiente salto que se encuentre en la misma *LSP* (Ver fig. 4.1).

La prueba B permitirá verificar la capacidad de un *LER* de egreso (*DUT*) para remover las etiquetas a los paquetes que circulan en una red *MPLS*, con la finalidad de enviarlos sin etiqueta a sus destinos correctos (Ver fig. 4.2).

Descripción

Utilizando *LDP*, crear una *LSP*:

- Para la prueba A entre el *LER* de Ingreso (*DUT*) y el *LER* de egreso. El tráfico sin etiquetas será enviado desde el *DUT* hasta el *LER* de egreso. El *DUT* deberá insertar las etiquetas con los valores de señalización a los paquetes y reenviar el tráfico hacia el *LSR* de siguiente salto incluido en el mismo *LSP*.
- Para la prueba B entre el *LER* de ingreso y el *LER* de egreso (*DUT*), este último extraerá las etiquetas y reenviará el tráfico *IP* hacia su destino final.

Análisis

La funcionalidad de los *LER* es una de las pruebas más básicas que se realizan a los equipos que operan *MPLS*, para conocer su capacidad de insertar y extraer etiquetas a los paquetes de datos que se intercambian entre dos o más dispositivos de cómputo pertenecientes a redes físicamente interconectadas a través de una red *MPLS*.

Escenario

Para la realización de estas pruebas suponemos la configuración de los siguientes elementos:

- La comunicación entre un cierto número de clientes que pertenecen a la red de usuarios A y un servidor de aplicaciones que esta conectado a una red B. Las redes A y B se basan en *IP*, se encuentran interconectadas a través de la red *MPLS* propuesta y son simuladas por un dispositivo generador analizador.
- Las *FEC*s estarán basadas en los prefijos de las direcciones *IP* de las redes A y B.
- La creación de las etiquetas se basará en el método *topology-based*.
- La distribución de etiquetas se hará a través del protocolo *LDP*.
- Las *LSP*s se crearán utilizando el método *hop-by-hop routing*.
- Las relaciones entre *FEC*-etiqueta (*LIB*), se construirán con base a la disposición de interfaces.
- El control de las etiquetas se hará a través del método *ordered*.
- La señalización en la red *MPLS* se hará a través del mecanismo *label request*.
- Variables a considerar:
  1. Longitud del paquete (*Bytes*).
  2. Carga aplicada (%).

Configuraciones

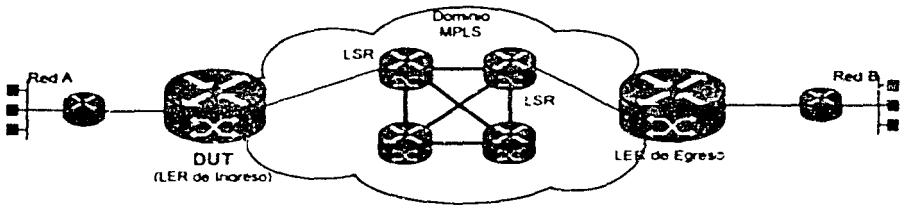

Figura 4.1 Configuración de la Prueba A

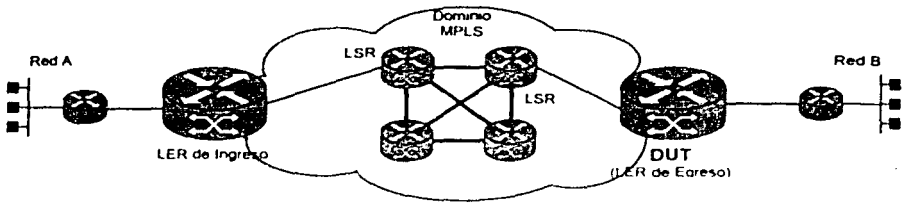

Figura 4.2 Configuración de la Prueba B

Procesos

Para la prueba A:

1. Utilizar el protocolo *OSPF* para la topología de red propuesta.
2. Utilizando *LDP*, crear una *LSP* entre el *DUT* y el *LER* de egreso.
3. Enviar los paquetes etiquetados desde el *DUT*, hasta el *LER* de egreso.
4. Verificar y reportar el número de paquetes transmitidos y recibidos por el *DUT* y el *LER* de egreso respectivamente.

Para la prueba B:

1. Utilizar el protocolo *OSPF*, para la topología de red propuesta.
2. Utilizando *LDP*, crear una *LSP* entre el *LER* de ingreso y el *DUT*.
3. Enviar los paquetes etiquetados con los valores previamente determinados, desde el *LER* de ingreso y hasta el *DUT*.
4. Verificar y reportar el número de paquetes transmitidos y recibidos por el *LER* de ingreso y el *DUT* respectivamente.

Procedimiento

1. Conforme al protocolo de *routing* especificado se construirán de manera automática las tablas de *routing* para la topología propuesta y se configurarán los elementos requeridos para cumplir con la funcionalidad de *MPLS*, conforme la plataforma de pruebas lo especifique.

2. De los paquetes IP
    - Para la prueba A los paquetes de datos generados por la red A serán insertados en el *LER* de ingreso (*DUT*) a través de la interfaz de ingreso, para que este los etiquete y los transfiera al *LSR* de siguiente salto.
    - Para la prueba B el *LER* de egreso (*DUT*) extraerá las etiquetas de los paquetes insertados a la red *MPLS* por el *LER* de ingreso y los enviará a su destino final.
3. Como se utilizará el método de *topology-based*, las etiquetas serán diferentes en cada interfaz que se encuentre a lo largo de la *LSP* creado en base al protocolo de *routing* utilizado y al método *hop-by-hop routing*, por lo que debe ser transparente y clara la opción de más de una *LSP* para transmitir la información, así mismo definida por el mecanismo de *label request*.
4. Como existirá una *LIB* por interfaz, deberá verificarse que correspondan a la *LSP* en la que este implícito el prefijo de la dirección *IP* destino y atendiendo al método de control de etiquetas *ordered*.
5. Es recomendable realizar varios ensayos con paquetes de longitudes variables y aplicando diferentes porcentajes de flujo conforme a la capacidad de los medios a través de los cuales se interconectaran las redes A y B, los *LER*s y los *LSR*s del dominio *MPLS* en la configuración propuesta.

Monitoreo y resultados

- Obtener las *LIB*s generadas en las interfaces de los *LER*s de ingreso y egreso.
- Verificar y reportar la cantidad de paquetes etiquetados correctamente vs. las *LIB*s generadas.
- Reportar la cantidad de paquetes etiquetados incorrectamente.
- Reportar el número de paquetes que perdieron su etiqueta y cuantos no.
- Verificar y reportar el número de paquetes enviados vs. los recibidos.
- Obtener el número de paquetes perdidos.
- Monitorear los diferentes resultados obtenidos vs. las variaciones en la longitud y el porcentaje de flujo de paquetes.

4.2. Reenvío de paquetes en *Label Switching Router* (*LSR*)

Propósito

Esta prueba permite verificar la habilidad del *LSR* de tránsito para intercambiar etiquetas de los paquetes en una red *MPLS* y reenviarlos correctamente reetiquetados.

Descripción

Utilizando *LDP*, se crea una *LSP* entre el *LER* de ingreso y el *LER* de egreso, se configura también el *LSR* de tránsito (*DUT*), y el tráfico que se genere del *LER* de ingreso se transmitirá a través de éste *router* hasta el *LER* de egreso. El *DUT* deberá intercambiar las etiquetas de los paquetes y reenviarlos al siguiente *LSR* dentro de la misma *LSP*.

Análisis

La funcionalidad de los *LSR*s es otra de las pruebas requisito que se realizan a los equipos que operan *MPLS*, con la finalidad de evaluar su capacidad para intercambiar las etiquetas a los paquetes de datos que circulan a través de una red *MPLS*.

Escenario

Para la realización de esta prueba suponemos la configuración de los siguientes elementos (Ver fig. 4.3):

- La comunicación entre un cierto número de clientes que pertenecen a la red de usuarios A y servidores de aplicaciones conectados a las redes B y C. Las redes A, B y C se basan en *IP*, se encuentran interconectadas a través de la red *MPLS* propuesta y son simuladas por un dispositivo generador analizador.
- Las *FEC*s estarán basadas en los prefijos de las direcciones *IP* de las redes A, B y C.
- La creación de las etiquetas se basará en el método *topology-based*.
- La distribución de etiquetas se hará a través del protocolo *LDP*.
- Las *LSP*s se crearán utilizando el método *hop-by-hop routing*.

- Las relaciones entre *FEC*-etiqueta (*LIB*), se construirán con base a la disposición de interfaces.
- El control de las etiquetas se hará a través del método *ordered*.
- La señalización en la red *MPLS* se hará a través del mecanismo *label request*.
- Variables a considerar:
  1. Longitud del paquete (*Bytes*).
  2. Carga aplicada (%).

Configuración

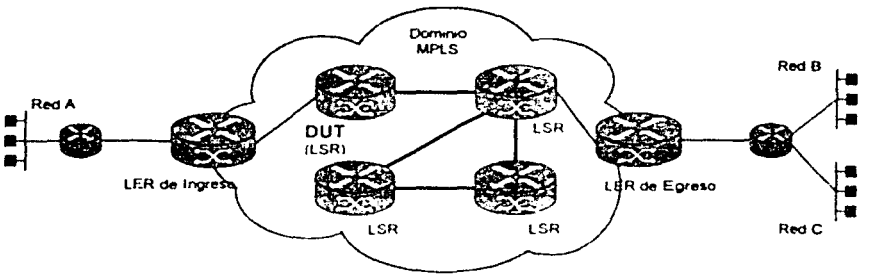

Figura 4.3 Configuración para funcionalidad *LSR*

Proceso

1. Utilizar el protocolo *OSPF*, para la topología de red propuesta.
2. Utilizando *LDP*, crear una *LSP* entre el *LER* de ingreso y el *LER* de egreso.
3. Enviar los paquetes etiquetados con los valores previamente determinados desde el *LER* de ingreso a través del *DUT*. Éste deberá intercambiar las etiquetas de los paquetes y reenviarlos al *LSR* de siguiente salto dentro de la misma *LSP*.
4. Medir el número de paquetes transmitidos y recibidos entre el *LER* de ingreso y el *LER* de egreso, verificando que los paquetes recibidos tengan las etiquetas correctas.

Procedimiento

1. Conforme al protocolo de *routing* especificado se construirán de manera automática las tablas de *routing* para la topología propuesta y se configurarán los elementos requeridos para cumplir con la funcionalidad de *MPLS*, conforme la plataforma de pruebas lo especifique.
2. Los paquetes de datos generados por la red A serán insertados en el *LER* de ingreso, el cual se encargará de etiquetarlos por primera vez y los transfiera al *DUT* (*LSR* de siguiente salto).
3. Como se utilizará el método de *topology-based*, las etiquetas que se puedan observar insertas a los paquetes de datos serán diferentes a la salida de cada interfaz que se encuentre a lo largo de la *LSP* creada en base al protocolo de *routing* utilizado y al método *hop-by-hop routing*, por lo que debe ser transparente y clara la opción de más de una *LSP* para transmitir la información, así mismo definida por el mecanismo de *label request*.
4. Se observará especialmente el intercambio de etiquetas que se a los paquetes de datos, entre la interfaz de salida del *LER* de ingreso y la interfaz de salida del *DUT*, de acuerdo a la red destino (B ó C).
5. Como existirá una *LIB* por interfaz, deberá verificarse que correspondan a la *LSP* en la que este implícito el prefijo de la dirección *IP* destino y atendiendo al método de control de etiquetas *ordered*.
6. Es recomendable realizar varios ensayos con paquetes de longitudes variables y aplicando diferentes porcentajes de flujo conforme a la capacidad de los medios a través de los cuales se interconectaran la red A, el *LER* de ingreso y el *DUT* del dominio *MPLS* en la topología propuesta.

Monitoreo y resultados

1. Obtener las *LIB*s generadas en las interfaces del *LER* de ingreso y el *DUT*.
2. Verificar y reportar la cantidad de paquetes etiquetados correctamente vs. las *LIB*s generadas.
3. Reportar la cantidad de paquetes etiquetados incorrectamente.
4. Verificar y reportar el número de paquetes enviados vs. los recibidos.
5. Obtener el número de paquetes perdidos

6. Monitorear los resultados reportados por el *DUT* vs. las variaciones en la longitud y el porcentaje de flujo de paquetes.

4.3. Establecimiento y desempeño del *Label Switched Path* (*LSP*)

Propósito

Esta prueba permitirá determinar el tiempo que le toma a un *SUT* levantar un cierto número de *LSP*s.

Descripción

Se crearán un número específico de *LSP*s en la red *MPLS* propuesta (*SUT*) y se medirán los tiempos mínimo, máximo y promedio que se requieren para levantarlas.

Análisis

Es importante conocer el desempeño de los *LSR*s en su funcionalidad generadora de las *LSP*s, a fin de establecer parámetros de calidad referentes al buen desempeño de una red *MPLS*, tal como es el *timestamp* para levantar una *LSP* con cada destino diferente.

Escenario

Para la realización de esta prueba suponemos la configuración de los siguientes elementos (Ver fig. 4.4):

- La comunicación entre un cierto número de clientes que pertenecen a la red de usuarios A y servidores de aplicaciones conectados a las redes B, C, D y E. Las redes A, B, C, D y E se basan en *IP*, se encuentran interconectadas a través de la red *MPLS* propuesta y son simuladas por un dispositivo generador analizador.
- Las *FEC*s estarán basadas en los prefijos de las direcciones *IP* de las redes implicadas en las pruebas.
- La creación de las etiquetas se basará en el método *topology-based*.
- La distribución de etiquetas se hará a través del protocolo *LDP*.

- Las *LSP*s se crearán utilizando el método *hop-by-hop routing*.
- Las relaciones entre *FEC*-etiqueta (*LIB*), se construirán con base a la disposición de interfaces.
- El control de las etiquetas se hará a través del método *ordered*.
- La señalización en la red *MPLS* se hará a través del mecanismo *label request*.
- Variables a considerar: El número de *LSP*s a establecer.

Configuración

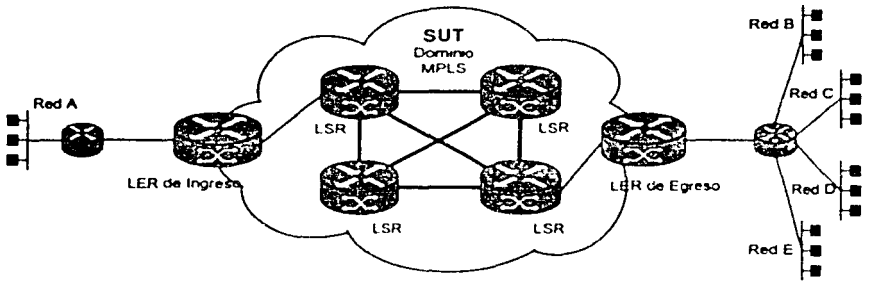

Figura 4.4 Configuración para prueba de *timestamp*

Proceso

1. Utilizar el protocolo *OSPF*, para la topología de red propuesta.
2. Utilizando *LDP*, crear un número especifico de *LSP*s entre los puntos terminales del *SUT*.
3. Registrar el valor de "*timestamp*" del primer mensaje de inicialización introducido al *SUT* y sustraer el valor del "*timestamp*" del mensaje de última inicialización recibido.

Procedimiento

1. Conforme al protocolo de *routing* especificado se construirán de manera automática las tablas de *routing* para la topología propuesta y se configurarán los

elementos requeridos para cumplir con la funcionalidad de *MPLS*, conforme la plataforma de pruebas lo especifique.

2. Los paquetes de datos generados por la red A serán insertados en el *LER* de ingreso, el cual a su vez se encargará de generar las peticiones de etiquetas al *LSR* de siguiente salto, cada vez que se detecte una petición para una *FEC* distinta y sobre la base descrita en el punto anterior.
3. La utilización del método *topology-based* implicará inserción de diferentes etiquetas a la salida de cada interfaz que se encuentre a lo largo de las *LSP*s creadas en base al protocolo de *routing* utilizado y al método *hop-by-hop routing*.
4. La generación y utilización de múltiples *LSP*s se hará a través del mecanismo de *label request*, mismo que se activará cada vez que se detecte una petición para una *FEC* distinta.
5. Se observará el desempeño del *LER* de ingreso, pues es este el dispositivo que mostrará por primera vez la presencia de una nueva *LSP* en el dominio *MPLS* (*SUT*), por lo que resulta conveniente contar con la simulación de varios destinos distintos a fin de tener más de una referencia de *timestamp* por ensayo.
6. En cada ocasión que se repita la prueba para el *SUT*, deberá verificarse que la *LIB* del *LER* de ingreso cuente con las *LSP*s en la que este implícito el prefijo de las direcciones *IP* destino y atendiendo al método de control de etiquetas *ordered*.
7. Es recomendable realizar varios ensayos aplicando el criterio de generar el mismo número de *LSP*s en cada ocasión a fin de obtener un número significativo de referencias de *timestamp* que permita observar un promedio más exacto de este valor en el *DUT* del dominio *MPLS* para la topología propuesta.

Monitoreo y resultados

- Obtener las *LIB*s generadas en el *DUT* para cada ensayo.
- Obtener los valores de *timestamp* con el requerimiento de cada *FEC* en los varios ensayos realizados.
- Obtener los valores mínimo, máximo y promedio de *timestamp* de todos los valores obtenidos, primero por cada destino y luego entonces un promedio de cada indicador.

- Monitorear la conformación de las *LSP*s en cada ensayo.

4.4. Establecimiento dinámico y número total de *LSP*s

Propósito

Determinar el número máximo de *LSP*s que puede establecer un *LSR* de forma dinámica y mantenerlas en operación bajo un *SUT* propuesto, utilizando un protocolo de señalización en específico.

Descripción

Se configurará el *SUT* para que trabaje con un *IGP* específico (*OSPF*) y se cambiará el protocolo de señalización de *RSVP* a *LDP* o viceversa, a fin de que se generen el número máximo de *LSP*s, a fin de verificar su operación enviando tráfico a través de estas.

Análisis

Determinar el máximo número de *LSP*s que un *SUT* puede generar dinámicamente y mantenerlas operando de forma adecuada bajo diferentes condiciones de tráfico; esta es una de las pruebas de mayor interés para evaluar la capacidad de diferentes equipos interconectados en un mismo dominio de *MPLS*, aún cuando estos puedan ser de diferentes fabricantes.

Escenario

Para la realización de esta prueba suponemos la configuración de los siguientes elementos (Ver fig. 4.5):

- La comunicación entre un cierto número de clientes que pertenecen a las redes de usuarios A, B, C y D, y los servidores de aplicaciones conectados a las redes E, F, G y H. Todas las redes se basan en *IP*, se encuentran interconectadas a través de la red *MPLS* propuesta y son simuladas por un dispositivo generador analizador.

- Las *FEC*s estarán basadas en los prefijos de las direcciones *IP* de las redes implicadas en las pruebas.
- La creación de las etiquetas se basará en el método *topology-based*.
- La distribución de etiquetas se hará a través de los protocolos *LDP* y RSVP alternadamente.
- Las *LSP*s se crearán utilizando el método *hop-by-hop routing*.
- Las relaciones entre *FEC*-etiqueta (*LIB*), se construirán con base a la disposición de interfaces.
- El control de las etiquetas se hará a través del método *ordered*.
- La señalización en la red *MPLS* se hará a través del mecanismo *label request*.
- Variables a considerar:
  1. Número inicial de *LSP*s.
  2. Número de *LSP*s a sumar en cada evento.
  3. Longitud del paquete (*Bytes*).
  4. Carga aplicada (%).

Configuración

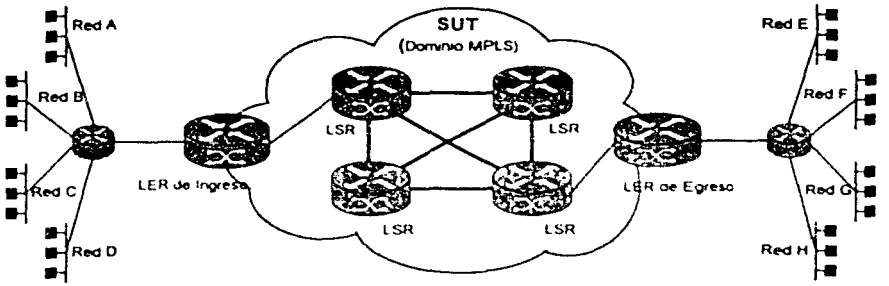

Figura 4.5 Configuración para establecimiento de *LSP*s

Proceso

1. Utilizar el protocolo *OSPF* para el *SUT* propuesto.

2. Utilizando *RSVP-TE y LDP* (primero uno y después otro).
3. Enviar los paquetes etiquetados con los valores previamente determinados, desde y hacia direcciones *IP* dentro del *SUT*.
4. Medir el número de paquetes transmitidos y recibidos, el número de paquetes etiquetados recibidos (por tipo de etiqueta), y el número de paquetes recibidos sin etiqueta.
5. Continuar sumando *LSP*s hasta el límite posible, o hasta que el *router* no pueda enviar los paquetes etiquetados a través de la *LSP* correcta.

Procedimiento

1. Conforme al protocolo de *routing* especificado se construirán de manera automática las tablas de *routing* para la topología propuesta y se configurarán los elementos requeridos para cumplir con la funcionalidad de *MPLS*, conforme la plataforma de pruebas lo especifique.
2. Los paquetes de datos generados por las redes A, B, C y D deberán ser peticiones para diferentes *FEC*s en todos los casos, de manera que con cada petición obligue al *SUT* a construir una nueva *LSP*.
3. Se hace clave el buen desempeño de los mecanismos de *label request* para la generación de múltiples *LSP*s y que se activará cada vez que se detecte una petición para una *FEC* distinta, así también, el protocolo de *routing* utilizado y al método *hop-by-hop routing*.
4. Se observarán especialmente el desempeño del *LER* de ingreso, por ser el dispositivo que mostrará por primera vez la presencia de una nueva *LSP* en el dominio *MPLS*, y los dispositivos cuyos enlaces sean el de mayor capacidad y el de menor capacidad a fin de tener tres referencias distintas por ensayo.
5. En cada ocasión que se repita la prueba para el *SUT*, deberá verificarse que las *LIB*s cuenten con las *LSP*s en la que este implícito el prefijo de las direcciones *IP* destino y atendiendo al método de control de etiquetas *ordered*.
6. Se realizarán varios ensayos con paquetes de longitudes variables y aplicando diferentes porcentajes de flujo conforme a la capacidad de los medios configurados en el *SUT*.

Monitoreo y resultados

- Obtener el número máximo de *LSP*s generadas por ensayo, de acuerdo al tamaño de los paquetes y porcentajes de flujo manejados.
- Obtener las *LIB*s generadas para cada ensayo en el *LER* de ingreso y en los dispositivos cuyos enlaces sean el de mayor capacidad y el de menor capacidad.
- Monitorear la conformación de las *LSP*s en cada ensayo.
- Verificar y reportar la cantidad de paquetes etiquetados correctamente vs. las *LIB*s generadas.
- Reportar la cantidad de paquetes etiquetados incorrectamente.
- Verificar y reportar el número de paquetes enviados vs. los recibidos.
- Obtener el número de paquetes perdidos.
- Monitorear los resultados reportados por el *SUT* en general vs. las variaciones en la longitud y el porcentaje de flujo de paquetes.

4.5. Transición de una *LSP* primaria a una *LSP* secundaria

Propósito

Medir el tiempo que le toma a un *SUT* converger y redireccionar el tráfico hacia una ruta secundaria o de respaldo, después de que la primera se ha caído por cambios en la topología de la red.

Descripción

Se crearan dos *LSP*s en la red *MPLS* propuesta, ambas a un salto del puerto destino, una de estas se definirá como primaria y otra como secundaria o respaldo de la primera. Cuando el *SUT* actualice sus tablas de *routing* y reenvío, se provocará una falla en la *LSP* primaria con la finalidad de que el *router* sea forzado a utilizar la *LSP* secundaria para reenviar el tráfico interrumpido. Se medirán el tiempo de convergencia y la perdida de paquetes.

Análisis

Determinar los tiempos de convergencia y redireccionamiento de tráfico, son parámetros de calidad básicos para medir la factibilidad de la configuración de una red de comunicaciones, es decir, la capacidad de respuesta de una red a cambios repentinos en el direccionamiento de los paquetes por fallas en la operación normal de las rutas y que obligan a los equipos involucrados en el envío de los mismos, a definir rutas alternativas para continuar con el proceso de comunicación y garantizar que los paquetes alcanzarán su destino final de manera casi transparente. Esta funcionalidad de control es una característica esencial en una red *MPLS*.

Escenario

Para la realización de esta prueba suponemos la configuración de los siguientes elementos(Ver fig. 4.6):

- La comunicación entre un cierto número de clientes que pertenecen a la red de usuarios A y un servidor de aplicaciones conectados a una red B. Las redes A y B se basan en *IP*, se encuentran interconectadas a través de la red *MPLS* propuesta y son simuladas por un dispositivo generador analizador.
- Las *FEC*s estarán basadas en los prefijos de las direcciones *IP* de las redes A y B.
- La creación de las etiquetas se basará en el método *topology-based*.
- La distribución de etiquetas se hará a través del protocolo *LDP*.
- Las *LSP*s se crearán utilizando el método *hop-by-hop routing*.
- Las relaciones entre *FEC*-etiqueta (*LIB*), se construirán con base a la disposición de interfaces.
- El control de las etiquetas se hará a través del método *ordered*.
- La señalización en la red *MPLS* se hará a través del mecanismo *label request*.
- Variables a considerar:
  1. Longitud del paquete (*Bytes*).
  2. Carga aplicada (%).

Configuración

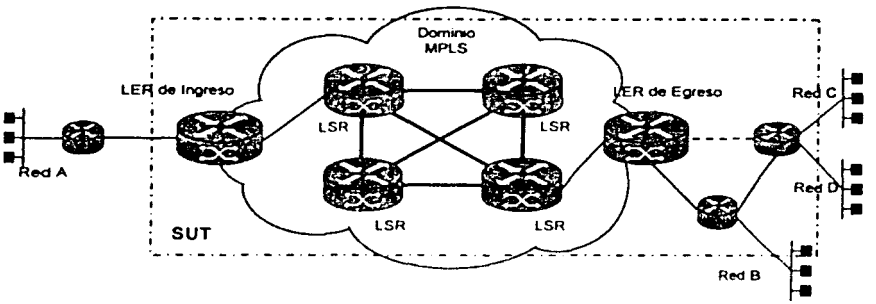

Figura 4.6 Configuración para transición a *LSP* secundaria

Proceso

1. Utilizar el protocolo *OSPF* para el *SUT* propuesto.
2. Utilizando *LDP*, y establecer la *LSP* primaria hacia el puerto destino para enviar *un flujo de tráfico continuo a través de este, en seguida establecer la LSP secundaria hacia el mismo puerto destino.*
3. *Desconectar el enlace establecido hacia el puerto destino a través de la LSP primaria.*
4. *Verificar al momento (punto 3) si el tráfico esta siendo reenviado por la LSP secundaria, a fin de medir el tiempo de convergencia y la pérdida de paquetes.*

Procedimiento

1. *Conforme al protocolo de routing especificado se construirán de manera automática las tablas de routing para la topología propuesta y se configurarán los elementos requeridos para cumplir con la funcionalidad de MPLS, conforme la plataforma de pruebas lo especifique.*
2. *Los paquetes de datos generados por la red A serán insertados en el LER de ingreso, el cual se encargará de etiquetarlos por primera vez y los transfiera al LSR de siguiente salto.*

3. *La utilización del método topology-based implicará inserción de diferentes etiquetas a la salida de cada interfaz que se encuentre a lo largo de las LSPs creadas en base al protocolo de routing utilizado y al método hop-by-hop routing.*
4. *La generación y utilización de múltiples LSPs se hará a través del mecanismo de label request, mismo que se activará cuando se detecte la petición para una FEC específica*
5. *Se observarán especialmente las LSPs creadas para alcanzar el destino específico de la red B y físicamente se identificará un enlace como primario, lo que implicará que la LSP generada a través de este, será la LSP primaria y el enlace identificado como secundario, implicará a la LSP secundaria. Lo anterior conforme a la red MPLS propuesta.*
6. *Se provocaran las fallas en la LSP primaria desconectando físicamente el enlace que la soporta a fin de forzar el levantamiento de la ruta alterna (LSP secundaria).*
7. *Como existirá una LIB por interfaz, deberán verificarse que correspondan a las LSP en la que este implícito el prefijo de la dirección IP destino y atendiendo al método de control de etiquetas ordered.*
8. *Es recomendable realizar varios ensayos con paquetes de longitudes variables y aplicando diferentes porcentajes de flujo conforme a la capacidad de los medios a través de los cuales se interconectaran la red A y B del dominio MPLS en la topología propuesta.*

Monitoreo y resultados

- *Obtener las LIBs generadas en las interfaces del LER de ingreso y LER de egreso*
- *Medir el tiempo de convergencia a través de la ruta secundaria en los varios ensayos realizados.*
- *Obtener los valores mínimo, máximo y promedio del tiempo de restablecimiento de la comunicación a través de la ruta secundaria.*
- *Verificar y reportar la cantidad de paquetes etiquetados correctamente vs. las LIBs generadas.*
- *Reportar la cantidad de paquetes etiquetados incorrectamente.*

- *Verificar y reportar el número de paquetes enviados vs. los recibidos.*
- *Obtener el número de paquetes perdidos.*
- *Monitorear los resultados reportados por el LER de egreso vs. las variaciones en la longitud y el porcentaje de flujo de paquetes.*

4.6. Configuración de *VPNs* en una red *MPLS*

Propósito

Estas pruebas permitirán medir la habilidad de los LERs para configurar VPNs usando MP-IBGP (Multi-protocol Internal Border Gateway Protocol) sobre LSPs preestablecidas, cumpliendo con su función de insertar y retirar etiquetas.

Descripción

Utilizando MP-IBGP primero se intercambiaran mensajes de entre el LER de ingreso y el LER de egreso para propagar la información de miembros de la VPN y su alcance. Luego entonces se establecerán las LSPs utilizando un protocolo de señalización LDP, a fin de conformar las VPNs que quedaran conformadas entre los routers R1 y R2, así como entre R1 y R3. Luego entonces se inyectará tráfico sin etiquetas al *LER* de ingreso desde el R1, y este a través de la red *MPLS* propuesta hasta el *LER* de egreso que se encargará de retirar las etiquetas insertadas por el *LER* de ingreso, para reenviar los paquetes *IP* hacia el destino correcto utilizando la *VPN* correspondiente.

Análisis

La posibilidad de configurar *VPN*s en una red *MPLS* agrega un beneficio sustantivo a una red de este tipo porque permite la comunicación en un nivel de seguridad y privacidad propios de una intranet aún y cuando compartan una infraestructura de *backbone* común a otras *VPN*s. Bajo la topología de red propuesta deberá observarse que la conformación de las *VPN*s queda supeditada a las *LSP*s generadas por mecanismos de *MPLS*, pero que a diferencia de una red *VPN* basada en

IP, operarán a través del intercambio de etiquetas, característica esencial en una red *MPLS*.

Escenario

Para la realización de esta prueba suponemos la configuración de los siguientes elementos (Ver fig. 4.7):

- La comunicación entre un cierto número de clientes que pertenecen a las redes A, B y C, y servidores de aplicaciones conectados a las redes D, E, F y G. Todas las redes se basan en *IP*, se encuentran interconectadas a través de la red *MPLS* propuesta y son simuladas por un dispositivo generador analizador.
- Las *FEC*s estarán basadas en los prefijos de las direcciones *IP* de las redes configuradas.
- La creación de las etiquetas se basará en el método *topology-based*.
- La distribución de etiquetas se hará a través del protocolo *LDP*.
- Las *LSP*s se crearán utilizando el método *hop-by-hop routing*.
- Las relaciones entre *FEC*-etiqueta (*LIB*), se construirán con base a la disposición de interfaces.
- El control de las etiquetas se hará a través del método *ordered*.
- La señalización en la red *MPLS* se hará a través del mecanismo *label request*.
- Las *VPN*s se conformaran como sigue: *VPN*1 entre los *router*s R1 y R2, *VPN2* entre los *router*s R1 y R3.
- Variables a considerar:
  1. Longitud del paquete (*Bytes*).
  2. Carga aplicada (%).

Configuración

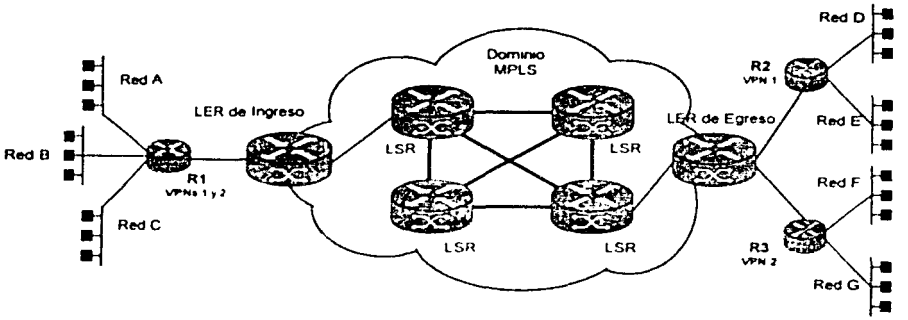

Figura 4.7 Configuración para generación de *VPNs*

Proceso

1. Utilizar el protocolo *OSPF* para la topología de red propuesta.
2. Configurar *MP-IBGP* en los *LER*s para actualizar las tablas de ingreso y egreso de las *VPN*s a las que se conectan.
3. Configurar los túneles para enviar el tráfico a través de las redes *VPN*1 y *VPN*2.
4. Realizar mediciones del número de paquetes transmitidos y recibidos, así como el número de paquetes etiquetados y sin etiqueta recibidos por los *router*s R2 y R3 a través del *LER* de egreso.

Procedimiento

1. Conforme al protocolo de *routing* especificado se construirán de manera automática las tablas de *routing* para la topología propuesta y se configurarán los elementos requeridos para cumplir con la funcionalidad de *MPLS*, conforme la plataforma de pruebas lo especifique.
2. Configurar *MP-IBGP* en los *LER*s para actualizar las tablas de ingreso y egreso de las *VPN*s a las que conectan.
3. Configurar los túneles para enviar el tráfico desde R1 hasta R2 por la red *VPN*1, y desde R1 hasta R2 por la red *VPN*2.

4. Los paquetes de datos generados por las redes A, B y C serán insertados al *LER* de ingreso, el cual se encargará de etiquetarlos y transferirlos por las *VPN*s configuradas hasta el *LER* de egreso a través de la red *MPLS* propuesta. Se podrá observar que el *LER* de ingreso inserta dos etiquetas, una para identificar a la *VPN* destino y otra identificar el próximo salto.
5. La utilización del método *topology-based* implicará inserción de diferentes etiquetas a la salida de cada interfaz que se encuentre a lo largo de las *LSP*s creadas en base al protocolo de *routing* utilizado y al método *hop-by-hop routing*.
6. La generación y utilización de múltiples *LSP*s se hará a través del mecanismo de *label request*, mismo que se activará cuando se detecte la petición para una *FEC* especifica.
7. Se identificarán las *LSP*s que conforman las *VPN*s definidas entre los *LER*s de ingreso y egreso.
8. Se realizarán ensayos separados para transferir información entre las redes que conforman las diferentes *VPN*s.
9. Se realizaran ensayos para intentar la comunicación entre redes que pertenezcan a diferentes *VPN*s.
10. Como existirá una *LIB* por interfaz, deberán verificarse que correspondan a las *LSP* en la que este implícito el prefijo de la dirección *IP* destino y atendiendo al método de control de etiquetas *ordered*.
11. Es recomendable realizar varios ensayos con paquetes de longitudes variables y aplicando diferentes porcentajes de flujo conforme a la capacidad de los medios a través de los cuales se interconectaran las *VPN*s del dominio *MPLS* en la topología propuesta.

Monitoreo y resultados

- Obtener las *LIB*s generadas para las *LSP*s de cada *VPN*.
- Verificar y reportar la cantidad de paquetes etiquetados correctamente por tipo de etiqueta vs. las *LIB*s generadas.
- Reportar la cantidad de paquetes etiquetados incorrectamente.
- Verificar y reportar el número de paquetes enviados vs. los recibidos.
- Obtener el número de paquetes perdidos.

- Monitorear los resultados reportados vs. las variaciones en la longitud y el porcentaje de flujo de paquetes.

4.7. Pruebas en redes *IP-ATM-MPLS*

Propósito

Esta prueba permite verificar la habilidad de los *LSR-ATM* para recibir paquetes *IP* y llevar a cabo el intercambio de los mismos a través de una red *MPLS* una vez que han sido correctamente etiquetados.

Descripción

Utilizando *LDP*, se crea una *LSP* entre el *LER* de ingreso y el *LER* de egreso, se configura también el *LSR-ATM* de tránsito, y el tráfico *IP* que se introduzca al *LER* de ingreso deberá ser etiquetado por este y retransmitido a través del *LSR-ATM* hasta el *LER* de egreso, el cual se encargará de retirar la etiquetas a los paquetes transmitidos por el mismo *VC*.

Análisis

Verificar la funcionalidad de *MPLS* montado sobre una red *ATM*. Bajo la topología de red propuesta deberá observarse que la conformación de las *LSP*s generadas por mecanismos de *MPLS*, así como la presencia de las etiquetas dentro de los campos *VPI/VCI* de cada paquete en el trayecto de los *LSR-ATM* implicados en las *LSP*s generadas.

Escenario

Para la realización de esta prueba suponemos la configuración de los siguientes elementos (Ver fig. 4.8):

- La comunicación entre un cierto número de clientes que pertenecen a la red de usuarios A y servidores de aplicaciones conectados a las redes B. Las redes A y

B se basan en *IP*, se encuentran interconectadas a través de la red *MPLS* propuesta y son simuladas por un dispositivo generador analizador.

- Las *FEC*s estarán basadas en los prefijos de las direcciones *IP* de las redes configuradas.
- La creación de las etiquetas se basará en el método *topology-based*.
- La distribución de etiquetas se hará a través del protocolo *LDP*.
- Las *LSP*s se crearán utilizando el método *hop-by-hop routing*.
- Las relaciones entre *FEC*-etiqueta (*LIB*), se construirán con base a la disposición de interfaces.
- El control de las etiquetas se hará a través del método *ordered*.
- La señalización en la red *MPLS* se hará a través del mecanismo *label request*.
- Variables a considerar, las especificadas.
  1. Longitud del paquete (*Bytes*).
  2. Carga aplicada (%).

Configuración

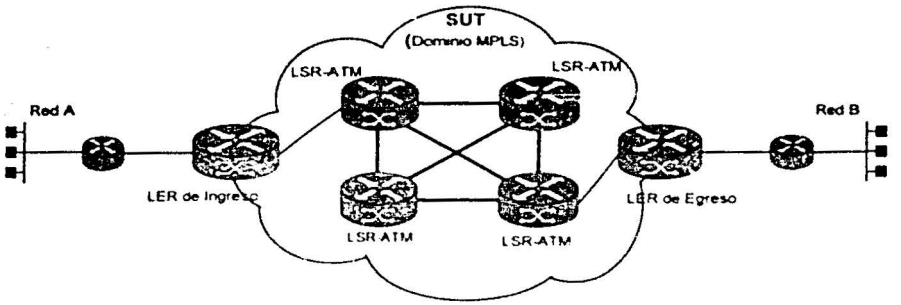

Figura 4.8 Configuración de dominio *MPLS* con *LSR-ATM*s

Proceso

1. Utilizar *ATM* para configurar la topología de red propuesta y comprobar su funcionamiento.

2. Configurar la funcionalidad de *MPLS* sobre la red *ATM* funcionando.
3. Utilizando *LDP*, generar las *LSP*s entre el *LER* de ingreso y el *LER* de egreso sobre los *VCs* preestablecidos por *ATM*.
4. Introducir tráfico *IP* -atendiendo al valor de las variables- al *LER* de ingreso, para que este lo retransmita debidamente etiquetado al *LSR-ATM* y este al *LER* de egreso, a través de la *LSP* creada.
5. Medir el número de paquetes transmitidos y recibidos entre el *LER* de ingreso y el *LER* de egreso, verificando que los paquetes recibidos tengan las etiquetas correctas.

Procedimiento

1. Conforme a la topología de red propuesta, se configurarán los elementos requeridos para habilitarla inicialmente como un *backbone ATM*, asegurándonos que los paquetes circulan a través ésta, cumplen con todas la características propias de la tecnología.
2. Configurar la funcionalidad de *MPLS* a los equipos de la red *ATM* en operación y habilitar el protocolo *LDP* para construir las *LSP*s entre el *LER* de ingreso y el *LER* de egreso sobre los *VCs* preestablecidos por *ATM*, las *LIB*s se construirán de manera automática al quedar identificadas las *LSP*s entre las redes A y B.
3. Los paquetes *IP* generados por la red A serán insertados en el *LER* de ingreso, mismo que se encargará de etiquetarlos en el campo *VPI /VCI* y transferirlos al *LSR-ATM* de siguiente salto correspondientemente, el *LER* de egreso y la red destino.
4. La utilización del método *topology-based* implicará inserción de diferentes etiquetas a la salida de cada interfaz que se encuentre a lo largo de las *LSP*s creadas al habilitar *LDP*.
5. La generación y utilización de múltiples *LSP*s se hará a través del mecanismo de *label request*, mismo que se activará cuando se detecte la petición para una *FEC* especifica.
6. Como existirá una *LIB* por interfaz, deberán verificarse que correspondan a la *LSP* en la que este implícito el prefijo de la dirección *IP* destino y atendiendo al método de control de etiquetas *ordered*.

7. Es recomendable realizar varios ensayos con paquetes de longitudes variables y aplicando diferentes porcentajes de flujo conforme a la capacidad de los medios a través de los cuales se interconectaran las redes A y B en la topología propuesta.

Monitoreo y resultados

- Obtener las *LIB*s generadas para las *LSP*s.
- Verificar y reportar la cantidad de paquetes etiquetados correctamente vs. las *LIB*s generadas.
- Reportar la cantidad de paquetes etiquetados incorrectamente.
- Verificar y reportar el número de paquetes enviados vs. los recibidos.
- Obtener el número de paquetes perdidos.
- Monitorear los resultados reportados vs. las variaciones en la longitud y el porcentaje de flujo de paquetes.

Conclusiones

Las conclusiones que se desprenden de este trabajo de tesis son las siguientes:

El desarrollo de la tecnología *MPLS* hasta el momento ofrece mejores posibilidades para la transmisión de grandes volúmenes de información para diferentes tipos de tráfico y amigabilidad en la configuración de los equipos involucrados, facilitando las funciones de control y monitoreo de grandes redes de *backbone*.

MPLS convive con cualquier tipo de red y cualquier protocolo estándar a nivel de *backbone*, por lo cual podría referirse como una solución ideal para redes grandes y topología en malla.

Una de las características más sobresalientes de la tecnología *MPLS* es la capacidad de integrar los niveles de capa de red y capa de transporte sin discontinuidades a través de las funciones de control de *routing* y conmutación de paquetes a nivel de capa de red.

La simplicidad de la encapsulación en *MPLS* se basa en el establecimiento de valores específicos para posiciones significativas de las estructuras de encapsulación de la capa de red, y como tal entre ésta y la capa de enlace de datos.

Actualmente los dos usos más importantes de *MPLS* son la Ingeniería de Tráfico y las *VPNs* (*Virtual Private Networks*).

Bajo el uso de la Ingeniería de Tráfico *MPLS* elimina algunas limitaciones de *routing*, pues permite especificar rutas explícitas durante el proceso de establecimiento de rutas, dirigiendo éstas hacia puntos de congestión y como resultado de ello los puntos congestión seleccionan las rutas de menor costo para cada destino posible, permitiendo que los paquetes sean desviados hacia rutas ignoradas por el proceso de *routing* e ignorando las rutas que convergen hacia estos.

MPLS permite establecer túneles entre sitios *VPN* para enviar los paquetes a través de su *backbone* efectuando la traducción de direcciones, lo que resulta ser un proceso menos costoso que el uso del *tunneling*.

Las pruebas propuestas aún no han sido realizadas, sin embargo este material conjunta una serie de procedimientos para la realización de trabajos específicos en las siguientes fases del proyecto de *MPLS* de la UNAM, y que nos permitirán obtener los valores de referencia más indicativos en términos generales, para el buen conocimiento y utilización de las plataformas en evaluación.

Por otro lado, considero que en la medida que la tecnología *MPLS* reporte mayores avances en su estandarización y en consecuencia el abatimiento en los costos que representa su implementación, sería una solución idónea para la RedUNAM - cuyo *backbone* trabaja en *Gigabit Ethernet*- y su interacción con otras redes de alta velocidad como lo es la red de *Internet* 2 en México proyecto de la Coorporación Universitaria para el Desarrollo de Internet (CUDI) de la cual es un nodo del *backbone*, y a la que a su vez se conectan muchas otras instituciones involucradas con el proyecto, de tal manera que favorezca la cooperación para el desarrollo de los diferentes trabajos científicos con requerimientos de grandes anchos de banda manejando diferentes servicios, que podrían ir desde transmisión de videoconferencia en tiempo real, como utilización de laboratorios remotos, robots electro-mecánicos, ejecución de aplicaciones de cómputo distribuido, simulaciones complejas, etc.

Bibliografía

[AGI 2001-A] *Agilent Technologies, INSIGHT, Edition 1*, (2001).

[AGI 2001-B] *Agilent Technologies, The Journal of Internet test Methodologies, Edition 1.0*, (2001).

[AGI 2002] *Agilent Technologies, Testing MPLS (Multi Protocol Label Switching) Enabled Networks.* http://advanced.comms.agilent.com/routertester/member/appnotes/mplstest.htm, (2002).

[ANG 2001] *Angus, M., MPLS Test & Analysis, Seminar for Developers and Service Providers. AHM Technology Corporation,* (2001).

[APP 1998] *Applied Technologies Group, Inc., Multiprotocol Label Switching (MPLS), The Technology Guide Series*, (1998).

[BAR 2001] Barberá, J., *MPLS*: Una Arquitectura de Backbone para la *Internet* del siglo XXI. Unisource Iberia, S.A., (2001).

[CIC 1999] *Ciccarelli, P., Faulkner, C., CCNA JumpStart, Networking and Internetworking Basics. Network Press, ISBN* 0-7821-2592-1, (1999).

[DAV 2000] *Davie, B., Rekhter, Y., MPLS: Technology and Applications. Morgan Kaufmann Publishers, Inc. ISBN* 1-55860-656-4, (2000).

[DOW 1998] *Downes, K., Ford, M., Lew, H.K., Spanier, S., Stevenson, T., Internetworking Technologies Handbook. Second Edition, Cisco Press, Cisco Systems, Inc. ISBN* 1-57870-102-3, (1998).

[DRA 2001] *Dragos, S., Dragos, R. Bandwith Management in MPLS Networks. School of Electronic Engineering – DCU, Broadband Switching and Systems Laboratory*, (2001).

[GRA 2001] *Gray, E. W., MPLS: Implementing the Technology, Addison Wesley, ISBN* 0-201-65762-7, (2001).

[HAL 2001] *Halaba, S., McPherson, D.* Arquitecturas de Enrutamiento en *Internet.* Segunda Edición, *Cisco Press, Cisco Systems Inc.*, (2001).

[HOB 2002] *Hobbes, R., Hobbes' Internet Timeline. v5.6,* http://www.zakon.org/robert/Internet/timeline/, (2002).

[IEC 2002] *International Engineering Consortium, Multiprotocol Label Switching.* http://www.iec.org/online/tutorials/mpls/topic02.html?Next.x=41&Next.y=16, (2002).

[JUN 2000] *Juniper Networks, The MPLS Advantage,Public and Private IP Services integration. Juniper Networks Inc.,* www.juniper.net, (2000).

[LEI 2000] *Leiner,B. M., Cerf, V. C., Clark, D. D., Kahn, R. E., Kleinrock, L., Lynch, D. C., Postel, J., Roberts, L. G., Wolff, S., A Brief History of the Internet.* V3.31, http://www.isoc.org/Internet/history/brief.shtml , *Internet Society,* (2000).

[MAT 2001] *Matyasovszki, I., Flanagan, C., Scalable Routers, Programmable Network Processors, MPLS and Others, are this real solutions?. Department of Electronic & Computer Engineering, University of Limerick, Ireland.* (2001).

[NET 1998] *Network Associates, Network Associates Guide to Communications Protocols,* (1998).

[SHE 1998] *Sheldon, T., Encyclopedia of Networking, Electronic Edition. McGraw-Hill Companies, ISBN* 0-07-882333-1, (1998).

[TEL 2000] *Telcordia Technologies, MPLS and IP DiffServ Testing Requirements,* (2000).

[UNA 2001] UNAM, *MPLS Projects,* http://www.mpls.unam.mx/projects.html, (2001).

Apéndice A: Acrónimos

AAL5	*Asynchronous Transfer Mode Advanced Layer 5*
AAL5-PDU	*Asynchronous Transfer Mode Advanced Layer 5-Protocol Data Unit*
ATM	*Asynchronous Transfer Mode*
ATM-LSR	*Asynchronous Transfer Mode-Label Switching Router*
ATM-VP	*Asynchronous Transfer Mode-Virtual Path*
BGP	*Border Gateway Protocol*
BITNET	*Because It's Time NETwork*
CIR	*Committed Information Rate*
CoS	*Class of Service*
CR-LSP	*Constraint-based Routing-Label Switched Path*
CR-LDP	*Constraint-based Routing-Label Distribution Protocol*
DiffServ	*Differentiated (or Differential) Services definitions.*
DLCI	*Data Link Connection Identifier*
DUT	*Devise Under Test*
EDP	*Early- Packet Discard*
EGP	*Exterior Gateway Protocol*
FEC	*Forwarding Equivalence Class*
FR	*Frame Relay*
GMPLS	*General Multi-Protocol Label Switching*
GSMP	*General (or Generic) Switch Management Protocol*
ICMP	*Internet Control Message Protocol*
IEEE	*The Institute of Electrical and Electronics Engineers, Inc.*
IETF	*Internet Engineering Task Force*
IFMP	*Ipsilon's Flow Management Protocol*
IGP	*Interior Gateway Protocol*
IP	*Internet Protocol*

IPv4	*Internet Protocol version 4*
IPv6	*Internet Protocol version 6*
IS-IS	*Intermediate Service to Intermediate Service*
ISP	*Internet Service Provider*
LAN	*Local Area Network*
LC-ATM	*Label Switching Controlled-ATM*
LDP	*Label Distribution Protocol*
LER	*Label Edge Router*
LFIB	*Label Forwarding Information Base*
LIB	*Label Information Base*
LIS	*Logical IP Subnet*
LLC	*Logical-Link Control*
LSP	*Label Switched Path*
LSR	*Label Switch (Switched or Switching) Router*
MAC	*Media Access Control*
MAN	*Metropolitan Area Network*
MPLS	*Multi-Protocol Label Switching*
OSI	*Open System Interconnection*
OSPF	*Open Shortest Path First*
PIM	*Protocol-Independent Multicast*
PIM-SM	*Protocol-Independent Multicast-*
POS	*Packet Over SONET*
PPD	*Partial-Packet Discard*
PPP	*Point to Point Protocol*
PVC	*Permanent Virtual Circuit*
QoS	*Quality of Service*
RFC	*Request For Comments*

RIP	*Routing Information Protocol*
RSVP	*ReSerVation Protocol*
RSVP-TE	*ReSerVation Protocol - Traffic Extention*
SONET	*Synchronous Optical NETworking*
SUT	*System Under Test*
TCP	*Transmission Control Protocol*
TCP/IP	*Transmission Control Protocol/Internet Protocol*
TLV	*Type Length Value*
ToS	*Type of Service*
TTL	*Time To Live*
UDP	*User Datagram Protocol*
VCI	*Virtual Channel Identifier*
VC-merge	*Virtual Circuit-merge*
VLSM	*Variable-Length Subnet Mask*
VoIP	*Voice over IP*
VPI	*Virtual Path Identifier*
VP-merge	*Virtual Path-merge*
VPN	*Virtual Private Network*
WAN	*Wide Area Network*

Apéndice B: Glosario

ATM Adaptation Layer 5 (AAL5)	Soporta *Variable Bit Rate (VBR)*, tolerancia al retardo, esta orientado a conexiones punto a punto y punto multipunto, requiere mínimo secuenciamiento para tráfico de datos o soporte para detección de errores.
Asynchronous Transfer Mode Advanced Layer 5-Protocol Data Unit (AAL5-PDU)	Contiene información especifica del control y administración de servicios como: cronometraje entre fuente y destino, velocidad de transmisión y tipo de conexión, definidos para la subcapa de Convergencia y la subcapa de Segmentación y Reensamble, para el transporte de datos en múltiplos de 48 *bytes*.
Asynchronous Transfer Mode (ATM)	Es un modo de transferencia en el cual la información es organizada en celdas. Es asíncrona en el sentido de que las celdas que contiene la información de un usuario en especial, no son necesariamente periódicas.
ATM-Virtual Path (ATM-VP)	Se la ruta establecida entre dos dispositivos *ATM* a través de un *VC*.
Backbone	Cualquier enlace de comunicaciones de alta velocidad que interconecta a los principales dispositivos de distribución de tráfico al interior de un sistema autónomo.
BITNET	*BITNET* es una abreviación de *Because It's Time NETwork*, fue una red de comunicaciones que funciono alrededor de 1981, para comunicar universidades y centros de investigación en el mundo.

Border Gateway Protocol (BGP)	El único protocolo *Exterior Gateway* para *routing* entre dominios independientes. Actualmente existe la versión 4.
Binary digit (Bit)	La unidad más pequeña de información que una computadora puede procesar, representando uno de dos estados usualmente indicados por "1" y "0".
Broadcast	Transmisión de información que se envia simultáneamente a todos los dispositivos pertenecientes a la misma red.
Committed Information Rate (CIR)	Es la velocidad de transferencia de información que una red con servicios de *Frame Relay* esta obligada a ofrecer en condiciones normales. La velocidad se promedia por encima de un incremento mínimo de tiempo.
Control-Driven	Es el tipo de relación utilizado para ligar las etiquetas con las *FEC*s y que en este caso permiten establecer las *LSP*s de una red *MPLS* antes de la transmisión de datos.
Class of Service (CoS)	*CoS* es una forma de administrar el tráfico de una red agrupando tipos de tráfico similares (ej. video, correo electrónico, voz, transferencia de archivos, etc.) y procesar cada tipo de tráfico como una clase con su propia prioridad de nivel de servicio, aunque no garantiza el servicio en términos de ancho de banda y tiempo de entrega. Las tres tecnologías de CoS son: • 802.1p *Layer 2 Tagging* • *Type of Service (ToS)* • *Differentiated Services (**DiffServ**)*

Constraint-based Routing-Label Distribution Protocol (CR-LDP)	Para mayor información referirse a: *draft-ietf-MPLS-LDP-06;* *draft-ietf-MPLS-cr-LDP-03;* *draft-fan-MPLS-lambada-signalling-00* Contiene las extensiones para que *LDP* amplíe sus capacidades. Permite ampliar la información utilizada para conformar rutas más allá de las disponibles para un protocolo de *routing*.
Data-Driven	Es el tipo de relación utilizado para ligar las etiquetas con las *FEC*s y que en este caso permiten establecer las *LSP*s de una red *MPLS* después de la detección de un cierto flujo de datos.
Datagram	Una unidad de mensaje que contiene la dirección fuente, la dirección destino y datos, un datagrama es routed a través de una red de paquetes conmutados.
Differentiated (or Differential) Services definitions (DiffServ)	Esencialmente es una técnica de *QoS* para proveer diferentes clases de servicio basado en algún conjunto de suposiciones comunes con requerimientos especificos de tráfico. La base para un tratamiento especifico esta explícitamente implicada en los paquetes y no que estos sean clasificados por un dispositivo, este método de *QoS* es reconocido como "*less state-full*".
Data Link Connection Identifier (DLCI)	Se utiliza en *Frame Relay* para identificar la conexión de un circuito entre dos conmutadores *Frame Relay* adyacentes.

Downstream-on-demand	Las etiquetas son asignadas y suministradas de manera ascendente sólo cuando son solicitadas. Este modo es comúnmente utilizado cuando los *LSR*s usan el modo conservativo para retención de etiquetas.
Devise Under Test (DUT)	Es el dispositivo de una red sobre el cual se llevan a cabo pruebas diversas para comprobar y medir su desempeño.
Early Packet Discard (EDP)	Cuando se utiliza un conmutador *ATM* como conmutador de backone *LAN*, *EPD* y *PPD* pueden aumentar el rendimiento de una red, ya que en situaciones de congestión, las celdas son desechadas aleatoriamente y ello implica que tales paquetes deban ser retransmitidos por el servidor o la aplicación correspondiente creando más congestión. Los mecanismos inteligentes para descartar paquetes son: *EDP* y *PPD*, se encargan de desechar todas las celdas relacionadas al paquete y creando un espacio para que otras celdas puedan pasar por el conmutador libremente.
Exterior Gateway Protocol (EGP)	Es un protocolo para intercambio de información de *routing* entre dos *router*s con funcionalidad de gateway, que pertenecen a diferentes sistemas autónomos y que intercambian sus tablas de *routing* con información asociada a la lista de *router*s conocidos, las direcciones IP que pueden ver directamente en sus interfaces y el costo de las métricas asociadas con las rutas de cada *router* conocido.
Ethernet	Protocolo de *LAN* sinónimo del estándar *IEEE* 802.3. *Ethernet* es el estándar de red para comunicación de datos más utilizado, desarrollado por DEC, Intel y Xerox. Utiliza topología de bus y *CMSA/CD* como método de acceso.

Frame Relay	Una forma de conmutación de paquetes, pero usando paquetes pequeños con menos revisión de errores que las formas tradicionales de conmutación de paquetes (como en X.25), actualmente un nuevo estándar para redes de alta velocidad y transmisión de ráfagas de datos en rede *WAN*.
Frame	Un mensaje encapsulado que contiene un *header DLL*, carga útil – frecuentemente la última parte de un paquete de la capa de red- y el *trailer*.
General Multi-Protocol Label Switching (GMPLS)	*draft-ietf-MPLS-generalized-signaling-02.txt* *GMPLS* es la extensión del protocolo *MPLS* que direccionará las necesidades de tráfico bajo plano de control óptico que requerirá de un complejo sistema de administración llamado Operating Support System (OSS) y utilizará sistemas como *SONET/SDH, DWDM y OXC*, para incrementar la capacidad de la red
General (or Generic) Switch Management Protocol (GSMP)	*RFC1987* http://www.cis.ohio-state.edu/htbin/rfc/rfc1987.html El *GSMP* es un protocolo de propósito general para control de un conmutador *ATM*, permite controlar el establecimiento y liberación de conexiones a través de un conmutador, agrega y elimina conexiones punto a multipunto, administra los puertos del conmutador, solicita información de configuración y estadísticas. Los paquetes *GSMP* son de longitud variable y se encapsulan directamente en *AAL5* con un *header LLC/SNAP* 0x00-00-00-88-0C, para especificar que se trata de un mensaje *GSMP*.

Internet Control Message Protocol (ICMP)	*RFC792* http://www.cis.ohio-state.edu/htbin/rfc/rfc792.html *RFC950* http://www.cis.ohio-state.edu/htbin/rfc/rfc950.html Los mensajes *ICMP* generalmente contienen información referente a las dificultades para llevar a cabo el *routing* con datagramas *IP* o de cambios de timestamp y transacciones echo.
The Institute of Electrical and Electronics Engineers, Inc. (IEEE)	El *IEEE* se describe asimismo como la sociedad profesional técnica más grande del mundo. Promoviendo el desarrollo y aplicación de la electrotecnología y ciencias relacionadas, para beneficio de la humanidad y de la profesión. La *IEEE* patrocina el desarrollo de estándares que frecuentemente se aceptan como nacionales e internacionales.
Internet Engineering Task Force (IETF)	El *IETF* es el organismo que define los estándares de los protocolos para la operación de *Internet*, como *TCP/IP* y es supervisado por la sociedad de *Internet Architecture Board (IAB)*. Los estándares publicados por la *IEFT* se expresan en forma de *RFCs*.
Ipsilon's Flow Management Protocol (IFMP)	*RFC1953* http://www.cis.ohio-state.edu/htbin/rfc/rfc1953.html *IFMP* es un protocolo para instruir al nodo adyacente para que ligue una etiqueta de capa 2 con un flujo *IP* especifico para enviarlo a través de un conmutador *IP*. La etiqueta permite un acceso más eficiente para ocultar información de *routing* para que dicho flujo pueda ser conmutado. *IFMP* se compone de dos sub-protocolos: *Adjacency Protocol y Redirection Protocol*. Los mensajes de *IFMP* son encapsulados en paquetes de *IPv4* y son enviados a la dirección límite de Broadcast (255.255.255.255). El campo de protocolo en el *header IP* contiene el valor 101 para indicar que se trata de un mensaje *IFMP*.

Interior Gateway Protocol (IGP)	Es un protocolo para intercambio de información de *routing* entre dos *routers* con funcionalidad de gateway, que pertenecen al mismo sistema autónomo. Dos de los protocolos más comunes que utilizan *IGP* son: *Routing Information Protocol (RIP) y Open Shortest Path First (OSPF) protocol.*
Internet 2	*Internet* 2 es un consorcio formado por aprox. 190 universidades, trabajando conjuntamente con la industria y el gobierno, para desarrollar y poner a la vanguardia las aplicaciones y tecnología en redes, para la creación de la *Internet* del mañana. Los objetivos primarios de *Internet* 2 son: • Crear una red de telecomunicaciones con capacidades avanzadas para la comunidad nacional de investigación. • Habilitar aplicaciones revolucionarias para *Internet*. • Garantizar la rápida transferencia de servicios y aplicaciones de red a toda la comunidad de *Internet*. Este proyecto existe en México bajo el nombre de Corporación Universitaria para el Desarrollo de *Internet* (CUDI).
Internet Protocol (IP)	*RFC 791* http://www.cis.ohio-state.edu/htbin/rfc/rfc791.html El protocolo de *Internet* el servicio de datagrama de la capa de red de *TCP/IP*, todos los demás protocolos de *TCP/IP*, excepto *ARP y RARP*, lo utilizan para enviar los frames de un host a otro. El *header* de *IP* contiene información de *routing* y de control asociada con los datagramas entregados.
Internet Protocol version 4 (IPv4)	Consultar *IP*.

	RFC1883 http://www.cis.ohio-state.edu/htbin/rfc/rfc1883.html *RFC1827* http://www.cis.ohio-state.edu/htbin/rfc/rfc1827.html
Internet Protocol version 6 (IPv6)	La versión 6 de *IP* es una nueva versión de *IP* basada en *IPv4*. *IPv6* incrementa el tamaño de *IP* de 32 *bits* a 128 *bits*, para soportar más niveles de direccionamiento, un número de direcciones disponibles mucho mayor y autoconfiguración de direcciones. Se introduce el concepto de escalabilidad de direccionamiento multicast y adicionalmente el direccionamiento anycast, para enviar un paquete a cualquiera de un grupo de nodos.
Intermediate System to Intermediate System (IS-IS)	http://www.iso.ch/cate/d18673.html *IS-IS* es un protocolo de capa de red que permite que sistemas intermedios dentro de un dominio de *routing* intercambien su configuración e información de *routing* para facilitar la operación del *routing* y transmisión de funciones de la capa de red.
Internet Service Provider (ISP)	Proveedor de acceso a los servicios de *Internet*.
Label Forwarding Information Base (LFIB)	Son las *LIBs* utilizadas por los *LSRs* cuando están configurados para el manejo de tráfico *Unicast* y *Multicast*.

Logical-Link Control (LLC)	El protocolo *IEEE* 802.2 ó *LLC* provee un mecanismo de enlace hacia los protocolos de capas superiores. El *LLC* tipo I provee enlace de datos en modo no orientado a conexión y el tipo II provee servicio de conexión orientada en la capa de enlace de datos.
Loop	En una red de comunicaciones es la repetición de un grupo de instrucciones e información circulando en la red, hasta que este es descartado por que su tiempo de vida ha terminado sin alcanzar su destino.
Media Access Control (MAC)	Un sistema de reglas para mover datos de un medio físico a otro ***MAX*** – *Media Access Exchange*. El *MAX* es una unidad de acceso a sistema de capa de red que soporta múltiples aplicaciones como: el acceso a redes *LAN* y líneas dedicadas de respaldo, entre otras.
Multicast	En tecnología de comunicaciones, es la habilidad para distribuir señales desde una sola fuente y hacia varios destinos al mismo tiempo, como el correo de voz y el correo electrónico.
Open System Interconnection (OSI)	Es un modelo de 7 capas de estándares de redes de comunicaciones que habilita a diferentes sistemas para intercambiar información independientemente del fabricante o la plataforma.
Open Shortest Path First (OSPF)	*RFC1583* http://www.cis.ohio-state.edu/htbin/rfc/rfc1583.html *OSPF* es un protocolo *IGP* para llevar a cabo el *routing* dentro de un grupo de *routers*. Utiliza tecnología *link-state* por lo que los *routers* se envían información los unos a los otros acerca de las conexiones directas y de los enlaces que tiene otros *routers*.

Protocol-Independent Multicast (PIM)	Consultar *PIM-SM.*
Protocol Independent Multicast-Sparse Mode (PIM-SM)	*RFC2362 ftp://ftp.isi.edu./in-notes/rfc2362.txt* Es un protocolo para hacer más eficiente el *routing* entre grupos de *multicast* que podrían abarcar grandes áreas Interdominio en *Internet*. El protocolo no depende de ningún protocolo de *routing Unicast* y esta diseñado para soportar grupos esparcidos.
Packet Over SONET (POS)	La tecnología de *POS* permite hacer más eficiente el transporte de datos sobre *Synchronous Optical Network/Synchronous Digital Hierarchy (SONET/SDH)* y ha sido clave en el crecimiento explosivo de *Internet*, permitiendo altos rendimientos a un costo efectivo entre grandes proveedores de servicios de *Internet*, pues se basa en el uso de la fibra óptica en redes *IP* de gran escala y alta velocidad. Las interfaces de los *routers* de *POS*, frecuentemente se conectan a dispositivos *Add Drop Multiplexers (ADMs)*, finalizando enlaces *SONET/SDH* punto a punto y conexiones directa a la fibra negra o via *Dense Wave-Division Multiplexing* (*DWDM*).
Partial-Packet Discard (PPD)	Ver *EDP.*

Point to Point Protocol (PPP)	*RFC1548* http://www.cis.ohio-state.edu/htbin/rfc/rfc1548.html *RFC1661* http://www.cis.ohio-state.edu/htbin/rfc/rfc1661.html *RFC1662* http://www.cis.ohio-state.edu/htbin/rfc/rfc1662.html Esta diseñado para simples enlaces que transportan paquetes entre dos puntos. Los enlaces son *full-duplex* y permiten la comunicación bidireccional entregando los paquetes en el orden que se generan.
Permanent Virtual Circuit (PVC)	Es un enlace definido a través de una ruta estática configurada manualmente.
Quality of Service (QoS)	El manejo o tratamiento específico de paquetes en servicios end-to-end. Comúnmente se utilizan dos modelos para proveer *QoS* a redes *IP*, *Integrated Services (IntServ) and Differentiated Services* (*DiffServ*).
Request For Comments (RFC)	Son una serie de notas acerca de *Internet*, que discuten muchos aspectos de computación y comunicación de computadoras enfocándose en los protocolos de red, procedimientos, programas y conceptos, peor también incluyen notas de reuniones de trabajo, opiniones y humor en algunas ocasiones. Los documentos de especificación de los protocolos de *Internet* definidos por la *IETF* son publicados como *RFCs*.

	Este protocolo mantiene una base de datos de los nodos de una red e intercambia información sobre la topología de la misma (*XNS*).
Routing Information Protocol (RIP)	Cada *router* mantiene una lista de todas las redes conocidas a través de los procesos de *routing* que le permiten saber del número de saltos hacia cada una de estas. *XNS* distribuye la información de las tablas de *routing* en la red a través de mensajes de b*roadcast* emitidos por los *routers* cada 30 segundos. Este protocolo tablas de *routing* como resultado de cambios en el servicio o la topología de la red y en respuesta a los requerimientos de información de *routing* por otros *routers*. *XNS* utiliza el protocolo *Echo* para demostrar la existencia y accesibilidad de otros nodos en la red, mientras el protocolo Error para señalar errores de *routing*.
Router	Es un dispositivo utilizado para enviar paquetes a nivel de capa de red. Físicamente es un dispositivo que reenvía paquetes de basándose en la información de capa de red y permite la operación de uno o varios protocolos de *routing*.
Routing	Es un esquema para seleccionar una de muchas posibles trayectorias. El proceso de aplicar un algoritmo matemático a una base de datos de topología para computar las rutas. Hay muchos tipos para cálculo del *routing*, uno de ellos es el algoritmo de *Djikstra*.

Resource ReSerVation Protocol (RSVP)	*draft* http://info.Internet.isi.edu:80/in-drafts/files/draft-ietf-rsvp-spec-16.txt *draft* http://info.Internet.isi.edu:80/in-drafts/files/draft-ietf-rsvp-md5-06.txt *RFC2205* http://www.cis.ohio-state.edu/htbin/rfc/rfc2205.html *RFC2750* http://www.cis.ohio-state.edu/htbin/rfc/rfc2750.html *RSVP* es in protocolo diseñado para integrar servicios de *Internet* y utilizado por algún dispositivo en beneficio de un arreglo de datos de alguna aplicación con requerimientos específicos de *QoS* y para un flujo de datos en particular.
Resourse ReSerVation Protocol Traffic Extension (RSVP-TE)	*draft-ietf-rsvp-spec-13.txt;* *draft-ietf-rsvp-md5-02.txt;* *draft-ietf-MPLS-rsvp-LSP-tunnel-05.txt;* *draft-fan-MPLS-lambda-signaling-00.txt* El protocolo *RSVP-TE* es una adición al protocolo *RSVP* con extensiones especiales para permitirle configurar trayectorias ópticas de manera simple en una red óptica.
Shim	Es una codificación de las etiquetas de *MPLS* presente para todo el medio cuando un arreglo de etiquetas esta en uso, ya sea con números de protocolo o identificadores de conexión la encapsulación de capa 2.
Synchronous Optical NETworking (SONET)	Es un estándar de ANSI para transmisión de información en fibra óptica y es una variación del estándar internacional.

Transmission Control Protocol (TCP)	*RFC793* http://www.cis.ohio-state.edu/htbin/rfc/rfc793.html *RFC1146* http://www.cis.ohio-state.edu/htbin/rfc/rfc1146.html *RFC1072* http://www.cis.ohio-state.edu/htbin/rfc/rfc1072.html *RFC1693* http://www.cis.ohio-state.edu/htbin/rfc/rfc1693.html *TCP* provee la entrega confiable de información y servicio de conexión virtual para aplicaciones a través del uso de acuses secuenciados con retransmisión de paquetes cuando es necesario.
Transmission Control Protocol/Internet Protocol (TCP/IP)	*RFC793* http://www.cis.ohio-state.edu/htbin/rfc/rfc793.html *RFC1146* http://www.cis.ohio-state.edu/htbin/rfc/rfc1146.html *RFC1072* http://www.cis.ohio-state.edu/htbin/rfc/rfc1072.html *RFC1693* http://www.cis.ohio-state.edu/htbin/rfc/rfc1693.html La familia de estándares de protocolo formulada para permitir que diferentes tipos de computadoras se comunica en entre estas. Aplicaciones particulares de TCP/IP son: el correo electrónico, establecimiento de sesión remota y transferencia de archivos.
Tester	Es un dispositivo que tiene la funcionalidad y *hardware* requerido para poder generar tráfico de paquetes con diferentes características, según se requiera y medirlo de manera simultanea.
Throughput	La velocidad efectiva de una red.

Type Length Value (TLV)	Una descripción de objeto con significado altamente intuitivo, compuesto de tres campos: *type, length and value*. *Type* aporta el significado semántico de *value*, *Length* aporta el número de *bytes* en el campo *value* y *value* es la longitud (*bytes*) de los datos en un formato un consistente con *Type*. *TLV* se utiliza en *LDP*.
Token	En una red *token-ring*, la presencia de un *token* el cual es simplemente un *frame* en particular circulando continuamente sobre un flujo de transmisión y que permite a los dispositivos conectados cambiar su estado, es decir tomar el *token* para colocar un mensaje en su lugar y enviarlo a otro dispositivo, cuando el destinatario notifica al emisor de haber y tomado el mensaje, este libera el *token* y lo regresa al anillo para que otro dispositivo pueda tomarlo y transmitir su información.
Type of Service (ToS)	Es una tecnología de *CoS* (Ver *CoS*).
Traceroute	Es un programa que registra la ruta que siguen los paquetes para llegar de una computadora a otra a través de *Internet*. Además de calcular y desplegar el tiempo que transcurre entre salto y salto, lo que lo hace una herramienta importante para localizar donde se pueda localizar una falla en la red.
Time To Live (TTL)	Especifica el tiempo de vida asignado a cada paquete de datos que circula en una red.

Tunneling	Permite la transmisión de información perteneciente a una red corporativa que utiliza la red pública y que los nodos de esta ignoran que se trate de una red privada. El *tunneling* se lleva a cabo encapsulando los datos de una red privada y el protocolo de información con las unidades de transmisión de la red pública y de esta manera el protocolo de información de la red privada aparece como un dato normal dentro de la red pública., creando así una especie de *VPN*.
User Datagram Protocol (UDP)	*RFC768* http://www.cis.ohio-state.edu/htbin/rfc/rfc768.html El protocolo *UDP* provee un servicio de mensajes simples y poco confiables para servicio orientados a transacción. Cada *header* de *UDP* lleva consigo el identificador del puerto fuente y el identificador del puerto destino.
Unicast	Es la comunicación entre un solo emisor y un solo receptor sobre una red.
Unsolicitated-downstream	Las etiquetas asignadas y provistas en forma ascendente con el aviso de una nueva ruta, y más eficiente cuando el equipo vecino que se encuentra a un salto en forma ascendente utiliza el modo liberal de retención.
Virtual Channel (VC)	Un canal de comunicaciones provisto para el transporte unidireccional de celda *ATM*.
Virtual Channel Identifier (VCI)	Un único identificador numérico definido por un campo de 16 *bits* en el *header* de las celdas *ATM* que identifica el canal virtual a través del cual viajan las celdas.

Virtual Circuit merge (VC-merge)	Es el proceso en que la etiqueta de *MPLS* se agrega al campo *VCI* de *ATM*, lo que permite que múltiples *VCs* se asocien dentro de un sólo *VC*.
Variable-Length Subnet Mask (VLSM)	Es el método que permite aplicar más de una mascara red a un espacio de direcciones *IP*, permitiendo dividirlo en grupos de espacios de red más pequeños en términos de dispositivos y subredes.
Voice over IP (VoIP)	En *VoIP*, el procesador de señales digitales segmenta la señal de voz en frames y los almacena en paquetes. Esos paquetes de voz son transportados usando *IP* en combinación con H.323 –la especificación para transmisión multimedia- a través de la red. Como es una aplicación sensible al retardo, se necesita contar con buena ingeniería, en redes *end-to-end* para garantizar el uso de *VoIP*.
Virtual Path Identifier (VPI)	Un campo de 8 *bits* en el *header* de las celdas *ATM*, el cual indica la trayectoria virtual sobre la cual las celdas son enviadas.
Virtual Path merge (VP-merge)	Es el proceso en que la etiqueta de *MPLS* se agrega al campo *VPI* de *ATM*, lo que permite que múltiples *VPs* se asocien dentro de un sólo *VP*. En este caso dos celdas podrían tener el mismo valor de *VCI* sólo si se originan del mismo nodo, de otra forma tendrán diferente valor de *VCI*.
Virtual Private Network (VPN)	Una *VPN* es una red de datos privada que hace uso de la infraestructura de telecomunicaciones pública, manteniendo la privacidad a través del uso de túneles y procedimientos de seguridad, lo que es muy diferente al uso de líneas dedicadas y resultan más económicas

Apéndice C: Tabla de trabajos referentes a *MPLS* (*Drafts* y *RFCs*)

[GRA 2001].

Tema	Versión	Fechas	Autores	Título
[BGP-MPLS-VPN]	0 1 2547 0 1 2	11/1998 12/1998 3/1999 3/2000 5/2000 7/2000	*E. Rosen,* *Y. Rekhter,* *T. Bogovic,* *R. Vaidyanathan,* *S. Brannon,* *M. Morrow,* *M. Carugi,* *C. Chase,* *T. Wo Chung,* *J. De Clercq,* *E. Dean,* *P. Hitchin,* *M. LeeLANivas,* *D. Marshall* *L. Martini,* *V. Srinivasan*	*BGP/MPLS VPNs* *draft-rosen-vpn-MPLS-00* and *01*, *RFC2547* and *draft-rosen-rfc2547bis-00*, *01*, *02*
[CLIP]	1577 2225	1/1994 4/1998	*M. Laubach*	*Classical IP and ARP over ATM* *RFC1577* and *RFC2225*
[CR-LDP]	0 1 2 3 4	1/1999 2/1999 8/1999 9/1999 7/2000	*B. Jamoussi (Ed.)*	*Constraint-Based LSP Setup using LDP* *draft-ietf-cr-LDP-00*, *01*, *02*, *03*, *04*
[CR-LDP-0]	0	10/1998	*L. Andersson,* *A. Fredette,* *B. Jamoussi,* *R. Callon,* *R. Dantu,* *P. Doolan,* *N. Feldman,* *M. Girish, E. Gray,* *J. Halpern,* *J. Heinanen,* *T. Kilty, A. Malis,* *K. Sundell,* *P. Vaananen,* *T. Worster, L. Wu*	*Constraint-Based LSP Setup using LDP* *draft-jamoussi-mpls-cr-ldp-00*

Tema	Versión	Fechas	Autores	Titulo
[CR-LDP-APP]	0 0 1	8/1999 9/1999 7/2000	*G. Ash, M. Girish,* *E. Gray,* *B. Jamoussi,* *G. Wright*	*Applicability Statement for CR-LDP* *draft-jamoussi-mpls-crldp-applic-00* *draft-ietf-mpls-crldp-applic-00 and 01*
[ECN]	0	6/1999	*K. Ramakrishnan,* *S. Floyd, B. Davie*	*A Proposal to Incorporate ECN in MPLS* *draft-ietf-mpls-ecn-00*
[Encapsulation]	0 1 2 3 4 5 6 7 8 3032	11/1997 2/1998 7/1998 9/1998 4/1999 8/1999 9/1999 9/1999 7/2000 1/2001	*E. Rosen,* *Y. Rekhter,* *D. Tappan,* *D. Farinacci,* *G. Fedorkow,* *T. Li, A. Conta*	*MPLSMPLS Label Stack Encoding* *draft-ietf-mplsMPLS-label-encaps-00, 01, 02, 03, 04, 05, 06, 07, 08* *and RFC3032*
[Framework]	0 1 2 3 4 5	5/1997 7/1997 11/1997 6/1999 7/1999 9/1999	*R. Callon,* *N. Feldman,* *A. Fredette,* *G. Swallow,* *P. Doolan,* *A. ViswaNathan*	*A Framework for Multiprotocol Label Switching* *draft-ietf-mplsMPLS-framework-00, 01, 02, 03, 04, 05*
[IP-MCAST]	0 1 2 0 1	8/1998 2/1999 5/1999 6/1999 5/2000	*D. Ooms,* *W. Livens,* *B. Sales,* *M. Ramalho* *A. Acharya* *F. Griffoul,* *F. Ansari*	*Framework for IP Multicast in MPLS* *draft-ooms-mpls-multicast-00, 01, 02* *draft-ietf-mpls-multicast-00 and 01*
[IP-MCAST-PIM-1]	0	11/1998	*W. Livens,* *D. Ooms, B. Sales*	*MPLS for PIM-SM* *draft-ooms-mplsMPLS-pimsm-00*
[IP-MCAST-PIM-2]	0	6/1999	*D. Farinacci,* *Y. Rekhter,* *E. Rosen*	*Using PIM to Distribute MPLS Labels for Multicast Routes* *draft-farinacci-MPLS-multicast-00*

Tema	Versión	Fechas	Autores	Titulo
[LDP]	0 0 1 2 3 4 5 6 7 8 9 10 11 3036	11/1997 3/1998 8/1998 11/1998 1/1999 5/1999 6/1999 10/1999 6/2000 6/2000 8/2000 8/2000 8/2000 1/2001	*L. Andersson,* *P. Doolan,* *N. Feldman,* *A. Fredette,* *R. Thomas*	*LDP Specification* *draft-feldman-ldp-LDP-spec-00 and* *draft-ietf-mpls-MPLS-ldp-00, 01, 02, 03, 04, 05, 06, 07, 08, 09, 10, 11* *and RFC3036*
[LDP-APP]	0 0 1 2 3037	8/1999 10/1999 8/2000 8/2000 1/2001	*R. Thomas,* *E. Gray*	*LDP Applicability* *draft-thomas-mpls-ldp-applic-00 and* *draft-ietf-mpls-ldp-applic-00, 01, 02* *and RFC3037*
[LDP-MIB]	0 1 2 3 4 5 6	8/1998 6/1999 10/1999 10/1999 1/2000 3/2000 7/2000	*J. Cucchiara,* *H. Sjostrand* *J. Luciani*	*Definitions of Managed Objects for the Multiprotocol Label Switching, Label Distribution Protocol (LDP)* *draft-ietf-mpls-ldp-mib-00, 01, 02, 03, 04, 05, 06*
[LDP-State]	0 0 1 2 3	10/1998 2/1999 6/1999 10/1999 1/2000	*L. Wu, P. Cheval,* *C. Boscher* *E. Gray*	*LDP State Machine* *draft-wu-mpls-ldp-state-00 and* *draft-ietf-mpls-ldp-state-00, 01, 02, 03*
[Loop-Prevention]	0 1 2 0 1 2 3	3/1998 7/1998 11/1998 5/1999 5/1999 10/1999 4/2000	*Y. Ohba,* *Y. Katsube,* *E. Rosen,* *P. Doolan*	*MPLS Loop Prevention Mechanism Using LSP-Id and Hop Count* *draft-ohba-mpls-loop-prevention-00 and* *MPLS Loop Prevention Mechanism* *draft-ohba-mpls-loop-prevention-01, 02 and* *draft-ietf-mpls-loop-prevention-00, 01, 02, 03*

Tema	Versión	Fechas	Autores	Titulo
[LSR-MIB]	0 1 2 3 4 5 6	6/1999 2/2000 3/2000 4/2000 5/2000 7/2000 7/2000	*C. Srinivasan,* *T. Nadeau,* *A. VisWANathan*	*MPLS Label Switch Router Management Information Base Using SMIv2* *draft-ietf-mpls-lsr-mib-00, 01, 02, 03, 04, 05, 06*
[MPLS-ARCH]	0 1 2 3 4 5 6 7 3031	7/1997 3/1998 7/1998 2/1999 2/1999 4/1999 8/1999 7/2000 1/2001	*E. Rosen,* *A. VisWANathan,* *R. Callon*	*A Proposed Architecture for MPLS* *draft-rosen-mpls-arch-00 and* *Multiprotocol Label Switching Architecture* *draft-ietf-mpls-arch-01, 02, 03, 04, 05, 06, 07* *and RFC3031*
[MPLS-ATM]	0 1 0 1 2 3 4 3035	11/1997 7/1998 9/1998 11/1998 4/1999 5/2000 6/2000 1/2001		*Use of Label Switching with ATM* *draft-davie-mpls-atm-00 and 01 and* *draft-ietf-mpls-atm-00, and* *MPLS using LDP and ATM VC Switching* *draft-ietf-mpls-atm-01, 02, 03, 04* *and RFC3035*
[MPLS-ATM-SVC]	0	10/1997	*N. Demizu,* *K. Nagami,* *P. Doolan,* *H. Esaki*	*ATM SVC Support for ATM-LSRs* *draft-dymitzu-mpls-atm-svc-00*
[MPLS-BGP]	0 1 2 3 4	4/1998 8/1998 2/1999 7/1999 1/2000	*Y. Rekhter,* *E. Rosen*	*Carrying Label Information in BGP-4* *draft-ietf-mpls-bgp4-mpls-00, 01, 02, 03, 04*
[MPLS-CAP]	0 1	2/1999 10/1999	*L. Andersson,* *B. Jamoussi,* *M. Girish,* *T. Worster*	*MPLS Capability set* *draft-loa-mpls-cap-set-00 and 01*
[MPLS-CPE-VPN]	0	10/1998	*T. Li*	*CPE based VPNs using MPLS* *draft-li-mpls-vpn-00*

Tema	Versión	Fechas	Autores	Titulo
[MPLS-Diff]	0 1 2 3 4 5 6 7	3/1999 6/1999 10/1999 2/2000 3/2000 6/2000 7/2000 8/2000	*F. le Faucheur,* *L. Wu, B. Davie,* *S. Davari,* *P. Vaananen,* *R. Krishnan,* *P. Cheval,* *J. Heinanen*	*MPLS Support of Differentiated Services by ATM LSRs and Frame Relay LSRs* draft-ietf-mpls-diff-ext-00 and 01 *MPLS Support of Differentiated Services* draft-ietf-mpls-diff-ext-02, 03, 04, 05, 06, 07
[MPLS-Diff-H]	0	6/1999	*J. Heinanen*	*Differentiated Services in MPLS Networks* draft-heinanen-diffserv-mpls-00
[MPLS-Diff-PPP-D]	0	4/1999	*S. Davari,* *R. Krishnan,* *P. Vaananen*	*MPLS Support of Differentiated Services over PPP links* draft-davari-mpls-diff-ppp-00
[MPLS-Diff-PPP-L]	0	6/1999	*F. le Faucheur,* *S. Davari,* *R. Krishnan,* *P. Vaananen,* *B. Davie*	*MPLS Support of Differentiated Services over PPP links* draft-lefaucheur-mpls-diff-ppp-00
[MPLS-DiffServ]	0 1	11/1998 2/1999	*L. Wu, P. Cheval,* *P. Vaananen,* *F. le Faucheur,* *B. Davie*	*MPLS Extensions for Differential Services* draft-wu-mpls-diff-ext-00 and 01
[MPLS-FR]	0 1 0 1 2 3 4 5 6 3034	9/1997 11/1997 12/1997 8/1998 10/1998 11/1998 5/2000 6/2000 6/2000 1/2001	*A. Conta,* *P. Doolan,* *A. Malis*	*Use of Label Switching With Frame Relay Specification* draft-conta-mpls-fr-00 and *Use of Label Switching on Frame Relay Networks Specification* draft-conta-mpls-fr-01 and draft-ietf-mpls-fr-00, 01, 02, 03, 04, 05, 06 and RFC3034
[MPLS-GIT-UUS]	0 1 2 3 4 3033	6/1998 12/1998 3/1999 7/1999 1/2000 1/2001	*M. Suzuki*	*The Assignment of the Information Field and Protocol Identifier in the Q.2941 Generic Identifier and Q.2957 User-to-user Signaling for the Internet Protocol* draft-ietf-mpls-git-uus-00, 01, 02, 03, 04 and RFC3033

Tema	Versión	Fechas	Autores	Titulo
[MPLS-GVPN]	0	8/1998	*J. Heinanen, B. Gleeson*	*MPLS Mappings of Generic VPN Mechanisms draft-heinanen-generic-vpn-mpls-00*
[MPLS-ICMP-Ext]	0 1 0 1 2	2/1999 5/1999 7/1999 12/1999 8/2000	*R. Bonica, D. Tappan, D. Gan*	*ICMP Extensions for MultiProtocol Label Switching draft-bonica-icmp-mpls-00, 01 and draft-ietf-mpls-icmp-00, 01, 02*
[MPLS-IP-ATM-ARP]	0	7/1997	*H. Esaki, Y. Katsube, K. Nagami, P. Doolan, Y. Rekhter*	*IP Address Resolution and ATM Signaling for MPLS over ATM SVC services draft-katsube-mpls-over-svc-00*
[MPLS-IP-VPN]	0	11/1998	*L. Casey, I. Cunningham, R. Eros*	*IP VPN Realization using MPLS Tunnels draft-casey-vpn-mpls-00*
[MPLS-LAN-R]	0	11/1997	*E. Rosen, Y. Rekhter, D. Tappan, D. Farinacci, G. Fedorkow, T. Li, A. Conta*	*MPLS Label Stack Encoding on LAN Media draft-rosen-mpls-lan-encaps-00*
[MPLS-LAN-V]	0	8/1997	*D. Bussiere, H. Esaki, A. GhanWANi, S. Matsuzawa, J. Pace, V. Srinivasan*	*Labels for MPLS over LAN Media draft-srinivasan-mpls-lans-label-00*
[MPLS-OMP]	0 1	11/1998 2/1999	*C. Villamizar*	*MPLS Optimized Multipath (MPLS-OMP) draft-villamizar-mpls-omp-00 and 01*
[MPLS-RSVP]	0 1 0	5/1997 11/1997 3/1998	*B. Davie, Y. Rekhter, E. Rosen, A. ViswANathan, V. Srinivasan, S. Blake*	*Use of Label Switching With RSVP draft-davie-mpls-rsvp-00 and 01, and draft-ietf-mpls-rsvp-00*

Tema	Versión	Fechas	Autores	Titulo
[MPLS-RTG-DYN]	0	3/1998	*S. Ayandeh, Y. Fan*	*MPLS Routing Dynamics draft-ayandeh-mpls-dynamics-00*
[MPLS-SIN]	0	8/1998	*B. Jamoussi, N. Feldman, L. Andersson*	*MPLS Ships in the Night Operation with ATM draft-jamoussi-mpls-sin-00*
[MPLS-TM-Frmwrk]	0	3/1998	*P. Vaananen, R. Ravikanth*	*Framework for Traffic Management in MPLS Networks draft-vaananen-mpls-tm-framework-00*
[MPLS-VCID]	1 0 0 1 2 3 4 5 3038	10/1997 2/1998 3/1998 8/1998 12/1998 4/1999 7/1999 8/2000 1/2001	*K. Nagami, N. Demizu, H. Esaki, Y. Katsube, P. Doolan*	*VCID Notification over ATM link draft-demizu-mpls-vcid-01, draft-nagami-mpls-vcid-atm-00 and draft-ietf-mpls-vcid-atm-00, 01, 02, 03 VCID Notification over ATM link for LDP draft-ietf-mpls-vcid-atm-04 and 05 and RFC3038*
[MPLS-VPN]	0 1	12/1997 3/1998	*J. Heinanen, E. Rosen*	*VPN support with MPLS draft-heinanen-mpls-vpn-00 and 01*
[MPLS-VPN-ARCH]	0	8/1998	*D. Jamieson, B. Jamoussi, G. Wright, P. Beaubien*	*MPLS VPN Architecture draft-jamieson-mpls-vpn-00*
[RFC1483]	1483 2684	7/1993 9/1999	*J. Heinanen D. Grossman*	*Multiprotocol Encapsulation over ATM Adaptation Layer 5 RFC1483 and RFC2684*
[RFC2547]	2547	3/1999	*E. Rosen, Y. Rekhter*	*BGP/MPLS VPNs RFC2547*
[RFC2684]	2684	9/1999	*J. Heinanen, D. Grossman*	*Multiprotocol Encapsulation over ATM Adaptation Layer 5 RFC2684*
[RFC2702]	2702	7/1999	*D. Awduche, J. Malcolm, J. Agogbua, M. O'Dell, J. McManus*	*Requirements For Traffic Engineering Over MPLS*

Tema	Versión	Fechas	Autores	Titulo
[RSVP-ATM]	0	6/1999	*W. Wimer*	*MPLS Using RSVP and ATM Switching* draft-wimer-mpls-atm-rsvp-00
[TE-MIB]	0 1 0 1 3 4 5	11/1998 1/1999 2/1999 6/1999 3/2000 7/2000 11/2000	*C. Srinivasan,* *A. VisWANathan,* *T. Nadeau*	*MPLS Traffic Engineering Management Information Base* draft-srinivasan-mpls-te-mib-00 and 01 *MPLS Traffic Engineering Management Information Base Using SMIv2* draft-ietf-mpls-te-mib-00, 01, 02, 04, 05
[TER]	0 0 1 2702	4/1998 10/1998 6/1999 9/1999	*D. Awduche,* *J. Malcolm,* *J. Agogbua,* *M. O'Dell,* *J. McManus*	*Requirements For Traffic Engineering Over MPLS* draft-awduche-mpls-traffic-eng-00 draft-ietf-mpls-traffic-eng-00 and 01 and RFC2702
[VP-Switching-DT]	0	2/1999	*N. Feldman,* *B. Jamoussi,* *S. Komandur,* *A. VisWANathan,* *T. Worster*	*MPLS using ATM VP Switching* draft-feldman-mpls-atmvp-00