TÉCNICA MPLS PARA OPTIMIZACIÓN DE TRÁFICO EN REDES IP
ANEXO B

CAROLINA ANDREA CARRASCAL REYES
CLAUDIA XIMENA MOSQUERA LEYTON

UNIVERSIDAD DEL CAUCA
FACULTAD DE INGENIERÍA ELECTRÓNICA Y TELECOMUNICACIONES
GRUPO DE I+D EN NUEVAS TECNOLOGÍAS EN TELECOMUNICACIONES
POPAYÁN
2001

TÉCNICA MPLS PARA OPTIMIZACIÓN DE TRÁFICO EN REDES IP
ANEXO B

CAROLINA ANDREA CARRASCAL REYES
CLAUDIA XIMENA MOSQUERA LEYTON

Monografía para optar al título de
Ingeniero en Electrónica y Telecomunicaciones

Director
Ing. Esp. OSCAR J. CALDERÓN CORTÉS

UNIVERSIDAD DEL CAUCA
FACULTAD DE INGENIERIA ELECTRÓNICA Y TELECOMUNICACIONES
GRUPO DE I+D EN NUEVAS TECNOLOGÍAS EN TELECOMUNICACIONES
POPAYÁN
2001

TABLA DE CONTENIDO
ANEXO B – CALIDAD DE SERVICIO

INDICE DE FIGURAS
ANEXO B – CALIDAD DE SERVICIO

ANEXO B – CALIDAD DE SERVICIO

PARTE 1. TEORÍA DE COLAS

1. CONCEPTOS BÁSICOS

La formación de colas es un las redes de telecomunicaciones surgen cuando los paquetes llegan por un punto de entrada a un nodo intermedio en la trayectoria al destino. Los paquetes en este punto se almacenan temporalmente y se procesan para determinar la interfaz de salida apropiada.

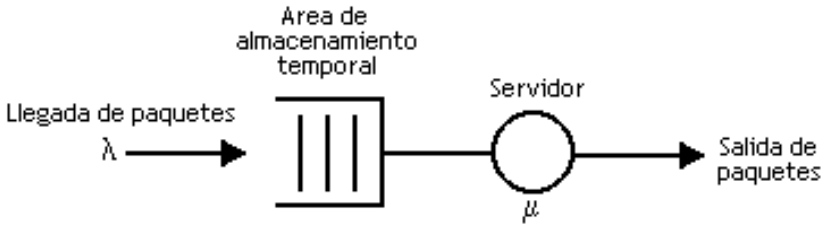


Figura 1.1. Modelo de Cola de Servidor Único.

Considérese el modelo de cola mas sencillo, como el mostrado en la Figura 1.1. Los paquetes llegan en forma aleatoria, a una velocidad promedio de λ paquetes/unidad de tiempo; forman cola de servicio en el área de almacenamiento temporal mostrada y luego, con alguna política de servicio especificada, son atendidos a una razón promedio de μ paquetes/unidad de tiempo. En el ejemplo de la Figura 1.1 se muestra un servidor único. En una situación más general, puede haber disponibilidad para muchos servidores, en cuyo caso es posible que haya en cualquier momento mas de un paquete de servicio.

En el contexto de las redes de datos, el servidor es el medio de transmisión: enlace, línea o troncal de salida que transmite datos a una velocidad prescrita C, dada en datos/unidad de tiempo. Lo mas común es que los datos se den en términos de bits/s ó caracteres/s.

La cola comienza a formarse conforme a la tasa de llegada de paquetes λ se aproxima a la capacidad de transmisión de paquetes μ. Para un área de almacenamiento temporal finita (situación real), la cola llegaría a un estado de saturación conforme λ exceda a μ. Cuando el área de almacenamiento temporal se satura, se bloquea la llegada de todos los paquetes

siguientes. Si se supone un área de almacenamiento temporal infinita (suposición que se hace a menudo para simplificar los análisis), la cola se vuele inestable a medida que $\lambda \rightarrow \mu$. Para este caso de cola con servidor único, $\lambda < \mu$ asegura la estabilidad.

Los sistemas de colas de espera pueden definirse mediante cinco componentes. Conviene notar explícitamente que sólo se están considerando sistemas con un numero infinito de clientes (es decir, la existencia de una larga cola no reduce la población de clientes a tal grado que se reduzca materialmente la velocidad de entradas). Los características a tener en cuenta son:

- La función de densidad de probabilidad del tiempo entre llegadas. Describe el intervalo de tiempo entre llegadas consecutivas. Se podría suponer que se contrata a alguna persona para observar la llegada de los clientes. A cada llegada, el observador registraría el tiempo transcurrido desde que ocurrió la llegada previa. Después de que hubiese transcurrido un tiempo suficientemente largo de estar registrando las muestras, la lista de números podría clasificarse y agruparse, es decir, tantos tiempos entre llegadas de 0.1 segundos, tantos de 0.2 segundos , etc. Esta densidad de probabilidad caracteriza el proceso de llegadas.

- La función de densidad de probabilidad del tiempo de servicio. Cada cliente requiere de cierta cantidad de tiempo proporcionado por el servidor. El tiempo de servicio requerido varia entre un cliente y otro (por ejemplo, un cliente puede presentar un carro lleno de artículos que abarrote la caja, y el siguiente puede traer únicamente una caja de galletas). Para analizar un sistema de colas de espera, deben conocerse tanto la función de densidad de probabilidad del tiempo de servicio, como la función de densidad del tiempo entre llegadas.

- El número de servidores. La cantidad de servidores no necesita explicarse. Muchos bancos, por ejemplo, tienen una sola cola larga para todos sus clientes y, cada vez que un cajero se libera, el cliente que se encuentra al frente de la cola se dirige a dicha caja. A este sistema se le denomina sistema de cola multiservidor. En otros bancos , cada cajero tiene su propia cola particular. En este caso tendremos un conjunto de colas independientes de un solo servidor, y no un sistema multiservidor.

- El tamaño máximo de las colas. No todos los sistemas de colas de espera poseen una capacidad infinita de recepción de clientes. Cuando demasiados clientes quieren hacer cola, pero sólo existe un número finito de lugares en cola de espera, algunos de estos clientes se pierden o son rechazados.

- La disciplina de servicio en las colas. La disciplina de servicio de una cola describe el orden según el cual los clientes van siendo tomados de la cola de espera. Los supermercados utilizan el método del primero en llegar es el primero en ser atendido. En las salas de urgencia de los hospitales se utiliza, más a menudo, el criterio de atender primero al que esté más grave, no al primero en llegar. En un entorno amistoso de oficina, ante la fotocopiadora, se despacha primero al que tenga menor trabajo.

  Existen varias disciplinas de atención en colas, algunas de ellas son:

- ✓ Primero en Llegar Primero en Salir (FIFO – First In First Out). En la cual los clientes son atendidos en el orden de llegada.
- ✓ Ultimo en Llegar Primero en Salir (LIFO - Last In, First Out) en la cual los clientes se atienden en el orden inverso al de llegada, es decir, el ultimo que llega es el primero que se atiende.
- ✓ Round-Robin (RR). Se ofrece una cantidad de servicio fija y pequeña a cada cliente, de forma circular.
- ✓ Procesador Compartido (PS - Processor Sharing). Es el límite de la RR cuando se dedican tiempos de servicio infinitesimalmente pequeños a cada cliente.
- ✓ Servicio Aleatorio (SIRO - Service In Random Order). Los clientes que esperan en cola se atienden en orden aleatorio.

Los dispositivos de encaminamiento utilizan tradicionalmente la disciplina de atención en cola FIFO en cada uno de sus puertos de salida. En cada puerto de salida se mantiene una cola simple. Cuando llega un paquete y se encamina a un puerto de salida, éste se sitúa al final de la cola. Mientras la cola no esté vacía, el dispositivo de encaminamiento transmite paquetes de la cola, siendo el siguiente el mas viejo.

La disciplina FIFO tiene varios inconvenientes:

- ✓ No se da ningún tratamiento especial a los paquetes de flujos que tienen una prioridad mas alta o que son mas sensibles al retardo. Si hay cierto número de paquetes de diferentes flujos para reenviar, todos ellos se tratan estrictamente en el orden FIFO.
- ✓ Si hay un número de paquetes pequeños que se sitúan en cola después de un paquete grande, entonces la disciplina FIFO da lugar a un retardo medio por paquete mas grande que si los paquetes pequeños se transmiten antes del grande. En general, los flujos con paquetes grandes obtienen un servicio mejor.
- ✓ Una conexión TCP "codiciosa" puede excluir a otras conexiones "altruistas". Si ocurre congestión y una conexión falla al retirarse, las otras conexiones en el camino de este segmento deben retirarse más de lo que deberían hacer en otras circunstancias.

Para resolver los inconvenientes de la disciplina FIFO se utiliza un tipo de esquema de atención equitativo de la cola, en el que un dispositivo de encaminamiento mantiene múltiples colas para cada puerto de salida (Figura 1.2). Con una atención equitativa de la cola, cada paquete de entrada se sitúa en una cola (flujo) para su atención. Las colas se atienden de una manera cíclica, tomando sucesivamente un paquete de cada cola no vacía. Las colas vacías no se atienden. Este sistema es equitativo en el sentido que cada flujo consigue enviar exactamente un paquete por ciclo. Además, es también una forma de balancear la carga entre varios flujos. No hay ninguna ventaja para las conexiones "codiciosas". Un flujo codicioso encuentra que su cola se va haciendo mas grande, incrementando el retardo, mientras que los otros flujos no se ven afectados por este comportamiento.

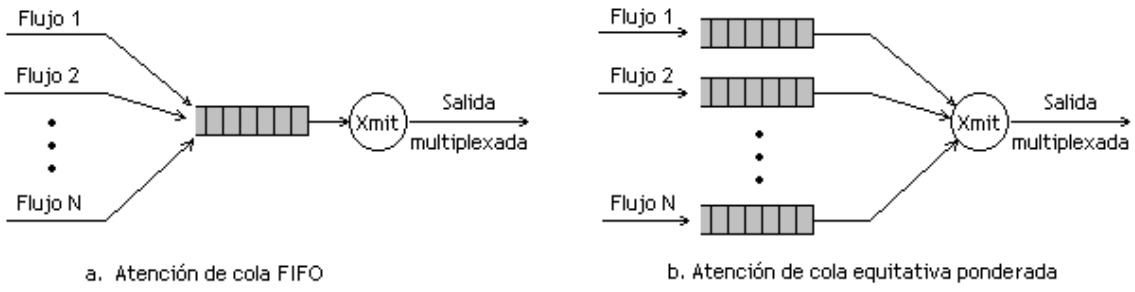


Figura 1.2. Atención Equitativa en cola FIFO.

Algunos fabricantes han implementado una mejora de la atención equitativa de colas conocido como Atención de Colas Equitativa Ponderada (WFQ – Weighted Fair Queuing). De forma resumida, WFQ tiene en cuenta la cantidad de tráfico en cada cola y asigna más capacidad a las colas mas ocupadas sin dejar de atender a las colas menos ocupadas. Además WFQ puede tener en cuenta la cantidad de servicios solicitados por cada flujo de tráfico y ajustar la disciplina de atención en cola adecuadamente.

Para es estudio de las colas en las redes de telecomunicaciones, el análisis se centra exclusivamente en sistemas de capacidad infinita con un solo servidor y una disciplina FIFO. Para estos sistemas se utiliza la notación A/B/m, en donde A es la densidad de probabilidad de tiempo entre llegadas, B es la densidad de probabilidad de tiempo de servicio y m es el número de servidores. Las densidades de probabilidad A y B son escogidas a partir del conjunto:

M - densidad de probabilidad exponencial (**M** significa Markov)
D - todos los clientes tienen el mismo valor (**D** significa determinístico)
G - general (es decir, densidad de probabilidad arbitraria).

El estado del arte actual cubre desde el sistema M/M/1, del cual se conoce absolutamente todo, hasta el sistema G/G/m, para el no se conoce, hasta la fecha, ninguna solución analítica exacta.

La hipótesis de utilizar una probabilidad de tiempo entre llegadas exponencial es totalmente razonable para cualquier sistema que maneja una gran cantidad de clientes independientes. En semejantes condiciones, la probabilidad de que lleguen exactamente **n** clientes, durante un intervalo de longitud **t**, estará dada por la ley de Poisson:

$$Pn(t) = \frac{(\lambda t)^n}{n!} e^{-\lambda t} \qquad (1)$$

en la cual λ es la velocidad media de llegadas.

Ahora demostraremos que las llegadas de Poisson generan una densidad de probabilidad de tiempo entre llegadas de tipo exponencial. La probabilidad, $\mathbf{a(t)\delta t}$, de que un intervalo entre llegadas se encuentre entre **t** y $\mathbf{t+\delta t}$, es exactamente la probabilidad de que no existan llegadas durante un tiempo **t**, multiplicada por la probabilidad de que exista una sola llegada en el intervalo infinitesimal δ**t**:

$$a(t)\Delta t = P_0(t)P_1(\Delta t)$$

donde

$$P_0(t) = e^{-\lambda t}, P_1(\Delta t) = \lambda \Delta t e^{-\lambda \Delta t}$$

En el limite **δt→0** y el factor exponencial en P1 se acerca a la unidad, por lo tanto

$$a(t)dt = \lambda e^{-\lambda t} dt \quad (2)$$

Nótese que la integración de la ecuación (2), entre 0 y &inf; es igual a 1, como debe ser.

Aunque la hipótesis de una densidad de probabilidad de tiempo entre llegadas de tipo exponencial es normalmente razonable, en términos generales es más difícil defender la hipótesis de que los tiempos de servicios sean también de carácter exponencial. Sin embargo, para las situaciones en las cuales mientras más grande sea el tiempo de servio, menor será su probabilidad de ocurrir, el modelo M/M/1 puede ser una aproximación adecuada. A continuación se ve con más detenimiento el proceso de Poisson.

1.1. Proceso de Poisson

El proceso de llegada de Poisson es el más usado en el diseño de los modelos de colas. Este ha sido muy difundido para el trafico de redes telefónicas así como para la evaluación del desempeño de sistemas de conmutación en general. Además el proceso de Poisson se ha utilizado para modelar la generación de fotones y la estadística de fotodetectores, a fin de representar procesos de ruido por impacto y estudiar el fenómeno de generación de electrones-huecos en semiconductores, entre otras aplicaciones.

Se utilizan tres enunciados básicos para definir el proceso de llegada de Poisson.

Considérese un pequeño intervalo de tiempo **δt**(**δt**→0), separando los tiempos **t** y **t+δt**, como se muestra en la Figura 1.3. Entonces:

1. La probabilidad de una llegada en el intervalo δt se define como λδt<<1, siendo λ una constante de proporcionalidad especificada.
2. La probabilidad de cero llegadas en **δt** es **1-λδt+0(δt)**.
3. Las llegadas son procesos sin memoria: cada llegada (evento) en un intervalo de tiempo es independiente de eventos en intervalos previos o futuros.

Con esta última definición, el proceso de Poisson se ve como un caso especial de un proceso de Markov, en el cual la probabilidad de un evento en el tiempo **t+δt** depende de la probabilidad en el tiempo de sólo **τ**. Nótese que de acuerdo con los enunciados 1 y 2 , queda excluido el caso de más de una llegada u ocurrencia de un evento en el intervalo **δt**(**δt**→0), al menos a **0(δt)**.

Si ahora se toma un intervalo finito **T** mayor, se encuentra la probabilidad **p(k)** de **k** llegadas en **T** dada como :

$$P(k) = (\lambda \mathrm{T})^K e^{\frac{-\lambda \mathrm{T}}{K!}} \quad \text{k=0,1,2,...} \quad (3)$$

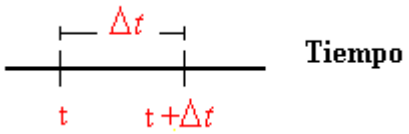


Figura 1.3. Intervalo de Tiempo Usado en la Definición del Proceso de Poisson.

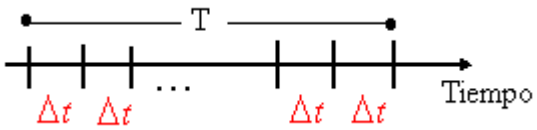


Figura 1.4. Derivación de la Distribución de Poisson.

Esta se conoce como la distribución de Poisson. Esta distribución está debidamente normalizada:

$$\sum_{k=0}^{\infty} p(k) = 1$$

y que el valor esperado está dado por

$$E(k) = \sum_{k=0}^{\infty} kp(k) = \lambda \mathrm{T} \quad (4)$$

La varianza:

$$\sigma_k^2 \equiv E\left[k - E(k)\right]^2 = E(k^2) - E^2(k) \quad \text{resulta ser}$$

$$\sigma_k^2 = E(k) = \lambda \mathrm{T} \qquad (5)$$

El parámetro **λ**, definido inicialmente como una constante de proporcionalidad, resulta ser un parámetro de velocidad:

$$\lambda = \frac{E(k)}{T}$$

de la ecuación (4) . Este representa entonces la tasa de promedio de llegadas de Poisson.

De las ecuaciones (4) y (5) se desprende que la desviación estándar σ_k de la distribución, normalizada al valor promedio E(K), tiende a cero conforme **λT** aumenta: $\frac{\sigma_k}{E(k)} = 1\sqrt{\lambda \mathrm{T}}$. Esto implica que para valores grandes de **λT**, la distribución se encuentra concentrada alrededor de valores muy cercanos al valor promedio **λT**. De esta forma , si se mide el número (aleatorio) de llegadas **n** en un intervalo **T** grande ("grande" implica **λT**>>1, o **T**>>1/**λ**), **n/T** sería la buena estimación de **λ**. Nótese también que $p(0) = e^{-\lambda \mathrm{T}}$. Conforme **λT** aumenta y la distribución

alcanza valores de alrededor de **E(k)= λT**, la probabilidad de no llegadas en el intervalo **T** se aproxima exponencialmente a cero con **T**.

La distribución de Poisson de la ecuación (3) se deriva sin dificultad usando los tres enunciados del proceso de Poisson. Con referencia a la Figura 1.4, considérese una secuencia de m pequeños intervalos, cada uno de longitud **δt**. Sea **p=λδt** la probabilidad de un evento (llegada) en cualquier intervalo **δt**, mientras que la probabilidad de 0 eventos es **q=1-λδt**. Usando el enunciado de independencia, parece entonces que la probabilidad de **k** eventos(llegadas) en cualquier intervalo **T=mδt** está dada por la distribución binomial:

$$p(k) = \binom{m}{k} p^k q^{m-k} \quad (6)$$

con

$$\binom{m}{k} \equiv \frac{m!}{(m-k)!k!}$$

Ahora, sea **δt→0**, pero con **T=mδt** fijo. Usando la ecuación que define la exponencial,

$$\lim_{t\to\infty} = (1+at)^{\frac{k}{t}} = e^{ak}$$

y calculando los términos factoriales mediante la aproximación de Stirling, se encuentra la ecuación (3).

Ahora considérese un intervalo grande de tiempo, y señálense los intervalos en los que ocurre un evento (llegada) Poisson. Se obtiene una secuencia aleatoria de puntos como la demostrada en la Figura 1.5. El tiempo entre las llegadas sucesivas se representa con el símbolo **τ**. Es evidente que τ es una variable aleatoria positiva con distribución continua. En la estadística de Poisson, **τ** es una variable aleatoria de distribución exponencial; es decir, su función de densidad de probabilidad $f_\tau(\tau)$ está dada por

$$f_\tau(\tau) = \lambda e^{-\lambda\tau} \qquad \tau \geq 0 \quad (7)$$

Esta distribución exponencial entre llegadas se esboza en la Figura 1.6. En procesos de llegada de Poisson, el tiempo entre las llegadas es más bien pequeño, y la probabilidad entre dos eventos (llegadas) sucesivos disminuye en forma exponencial con el tiempo **τ**.
Después de un calculo simple se ve que el valor medio **E(τ)** de esta distribución exponencial es

$$E(\tau) = \int_0^\infty \tau f_\tau(\tau) d\tau = 1/\lambda \quad (8)$$

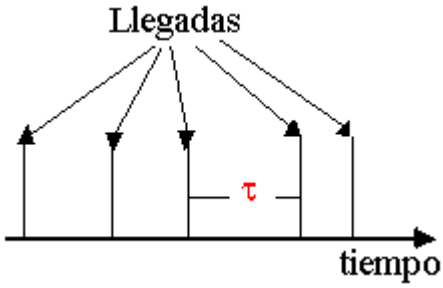


Figura 1.5. Llegadas de Poisson

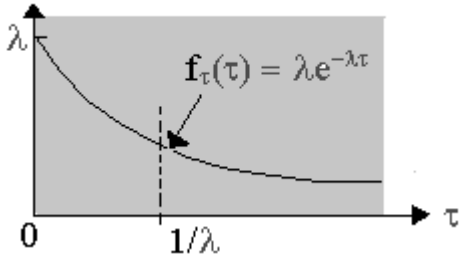


Figura 1.6. Distribución Exponencial entre Llegadas

mientras que la varianza está dada por

$$\sigma_\tau^{\,2} = 1/\lambda^2 \quad (9)$$

El tiempo promedio entre llegadas resulta el esperado, ya que si la tasa de llegadas es **λ**, el tiempo entre ellas debería ser **1/λ**.

El hecho de que la estadística del proceso de Poisson de lugar a una distribución exponencial entre llegadas se deduce con facilidad de la distribución de Poisson de la ecuación (3).

Considérese el diagrama de tiempo de la Figura 1.7. Como muestra, sea **τ** la variable aleatoria que representa el tiempo transcurrido desde un origen arbitrario hasta el tiempo de la primera llegada. Tómese cualquier valor **x**. No ocurren llegadas en el intervalo (0,x) si, y sólo si, **τ>x**. La probabilidad de que **τ>x** es exactamente la probabilidad de que no ocurran llegadas en (0,x); es decir,

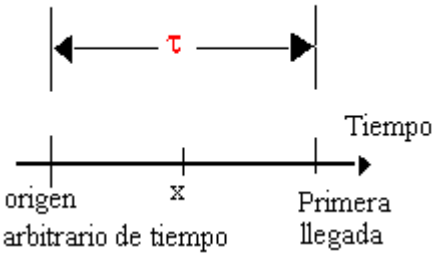


Figura 1.7. Derivación de la distribución exponencial.

P(τ>x)=prob. (número de llegadas en (0,x)=0)= $e^{-\lambda x}$ de la ecuación (3). Entonces la probabilidad de que **τ≤x** es

$$P(\tau \le x) = 1 - e^{-\lambda x}$$

que es justamente la distribución acumulativa de probabilidad $F_\tau(x)$ de la variable aleatoria **τ**. Por tanto, se tiene

$$F_\tau(x) = 1 - e^{-\lambda x} \quad (10)$$

de donde sigue la función de densidad de probabilidad $f_\tau(x) = dF_\tau(x)/dx = \lambda e^{-\lambda x}$. La cercana relación entre el proceso de llegada de Poisson y el tiempo entre llegadas con distribución exponencial puede aplicarse después de discutir las propiedades de la distribución exponencial del tiempo de servicio. Así, considérese una cola con usuarios (paquetes o llamadas) en espera de servicio. Céntrese la atención en la salida de la cola y señálese el tiempo en que un usuario completa el servicio. Esto se muestra de manera esquemática en la Figura 1.8. Sea **r**, como se muestra, la variable aleatoria que representa el tiempo entre cumplimiento de servicio. También puede ser el tiempo de servicio si el siguiente usuario es atendido tan pronto como el que está en servicio abandona el sistema. En particular, tómese el caso en que **r** tiene distribución exponencial, con un valor promedio **E(r)=1/μ**. Entonces

$$f_r(r) = \mu e^{-\mu r} \quad r \ge 0 \quad (11)$$

Pero al comparar las Figuras 8 y 5 es evidente que si **r**, el tiempo entre cumplimientos de servicio, tiene distribución exponencial, entonces los tiempos de terminación deben representar por sí mismos un proceso de Poisson. El proceso de servicio es completamente análogo al proceso de llegada. Sobre esta base, la probabilidad de un cumplimiento en el intervalo **(t,t+δt).**

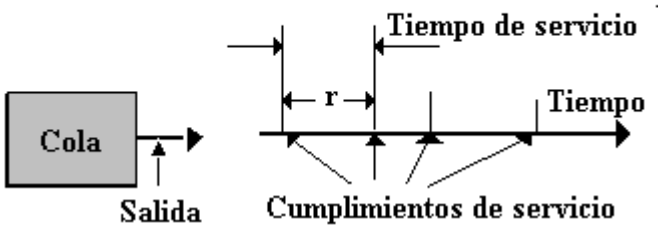


Figura 1.8. Cumplimientos de Servicio a la Salida de una Cola.

es **μδt+o(δt)**, mientras que la probabilidad de no cumplimiento en **(t,t+δt)** es **1-μδt+o(δt)**, independientemente de los cumplimientos pasados o futuros. El modelo de servicio con distribución exponencial tiene implícita la propiedad de independencia de los enunciados para el proceso de Poisson.

Antes de proseguir el proceso de formación de las colas, se incluye una propiedad más al proceso de Poisson. Supóngase que se mezclan m flujos de Poisson independientes, de tasas arbitrarias de llegada $\lambda_1, \lambda_2, ...\lambda_m$, respectivamente. Entonces el flujo compuesto es en sí un flujo de Poisson, con parámetro de llegada $\lambda + \sum_{i=1}^{n} \lambda_i$. Esta es una propiedad muy útil, y constituye una de las razones por las que a menudo se usan modelos de procesos de llegadas de Poisson para las llegadas. En el contexto de redes de conmutación de paquetes y conmutación de circuitos, esta situación ocurre al combinar estadística de paquetes y llamadas de varias fuentes de datos (terminales y teléfonos), cada una de las cuales genera paquetes o llamadas (según el caso) con cierta tasa de Poisson. Una prueba sencilla es la siguiente: sea $N^{(i)}(t, t+\Delta t)$ el número de eventos en un proceso de Poisson **i**, **i=1,2,...m** en el intervalo **(t,t+δt)**. Sea **N(t,t+δt)** el número total de eventos de flujo compuesto. Entonces:

$$prob.[N(t,t+\Delta t)=0] = \prod_{i=1}^{n} prob.[N^{(i)}(t,t+\Delta t)=0]$$

$$= \prod_{i=1}^{m} [1-\lambda_i \Delta t + o(\Delta t)] = 1-\lambda\Delta t + 0(\Delta t), \quad \text{(2-10)}$$

$\lambda = \sum_{i=1}^{m} \lambda_i$, ya que los procesos individuales son independientes. Un cálculo similar muestra que

$$prob.[N(t,t+\Delta t)=1] = \lambda\Delta t + o(\Delta t) \quad \text{(2-11)}$$

Esto prueba la relación deseada. Las sumas de procesos de Poisson conservan pues la distribución; es decir, cada una de ellas retiene la propiedad de Poisson. Esta propiedad se utilizará implícitamente en los ejemplos de análisis de colas y almacenamiento temporal.

1.1.1. Colas M/M/1

Consideremos en primer lugar una cola infinita donde llegan por término medio **λ** Clientes por unidad de tiempo, y son servidos por término medio μ clientes por segundo. La cola puede encontrarse en los estados {1,2,3,4,...k}. El número de transiciones **k→k+1** es entonces λp_k y el número de transiciones **k→k-1** es μp_k:

$$\begin{array}{ccccc} -\lambda p_0 \rightarrow & & -\lambda p_1 \rightarrow & & -\lambda p2 \rightarrow \quad ... \\ 0 & & 1 & & 2 \quad ... \\ \leftarrow \mu p_1 - & & \leftarrow \mu p_2 - & & \leftarrow \mu p3 - \end{array}$$

En el equilibrio, el número de transiciones en uno y otro sentido ha de ser igual, por término medio, so pena de que la cola se encuentre vacía permanentemente o crezca sin límite. Por tanto, si p_k es la probabilidad estacionaria de que el sistema se encuentre en el estado k:

$$\begin{aligned} \lambda p_0 &= \mu p_1 \\ \lambda p_1 &= \mu p_2 \\ \lambda p_2 &= \mu p_3 \\ &\cdots \\ \lambda p_{k-1} &= \mu p_k \end{aligned} \qquad (12)$$

Resolviendo la primera para p_1 podemos encontrar p_2 de la segunda, que podemos sustituir en la tercera para encontrar p_3 y así sucesivamente, obteniendo el resultado general:

$$p_k = \left(\frac{\lambda}{\mu}\right)^k p_0 \qquad (13)$$

Falta por encontrar p_0 para lo cual nos servimos de la condición:

$$\sum_{k=0}^{\infty} p_k = \sum_{k=0}^{\infty} (\frac{\lambda}{\mu})^k p_0 = p_0 \sum_{k=0}^{\infty} \left(\frac{\lambda}{\mu}\right)^k = 1 \qquad (14)$$

Como suponemos que el ritmo medio de servicio es mayor que el ritmo medio de llegada:

$$\sum_{k=0}^{\infty} \left(\frac{\lambda}{\mu}\right)^k = \frac{1}{1-\frac{\lambda}{\mu}} \qquad (15)$$

Finalmente:

$$p_k = \left(\frac{\lambda}{\varpi}\right)^k \left(1-\frac{\lambda}{\mu}\right) = (1-\rho)\rho^k \qquad (16)$$

El número medio de clientes en la cola no es otro que :

$$\overline{N} = \sum_{k=0}^{\infty} k p_k = (1-\rho) \sum_{k=0}^{\infty} k \rho^k \qquad (17)$$

Sabemos que:

$$\sum_{k=0}^{\infty} \rho^k = \frac{1}{1-\rho} \qquad (18)$$

Derivando la expresión anterior y multiplicando ambos lados por **ρ** obtenemos:

$$\sum_{k=0}^{\infty} k\rho^k = \frac{\rho}{(1-\rho)^2} \qquad (19)$$

En definitiva:

$$\overline{N} = \frac{\rho}{1-\rho} \qquad (20)$$

Obsérvese que cuando el ritmo de llegada se aproxima al ritmo de servicio la cola crece indefinidamente. Por otra parte, si llegan **λ** clientes a la cola por término medio, y esta tiene la longitud que acabamos de encontrar, el tiempo $\overline{T}$ viene dado por:

$$\overline{T} = \frac{\overline{N}}{\lambda} = \frac{\frac{1}{\mu}}{1-\rho} \qquad (21)$$

Supongamos ahora que el número de clientes en la cola es finito, de manera que aquellos que encuentran la cola llena son rechazados. Sea **M** el número máximo de clientes en cola. En este caso:

$$\begin{aligned} \lambda p_0 &= \mu p_1 \\ \lambda p_1 &= \mu p_2 \\ \lambda p_2 &= \mu p_3 \\ &\cdots \\ \lambda p_{M-1} &= \mu p_M \end{aligned} \qquad (22)$$

de donde:

$$p_k = \left(\frac{\lambda}{\mu}\right)^k p_0 \qquad (23)$$

Ahora sin embargo la condición de normalización, al extenderse la suma sobre un número finito de estados, es distinta. Encontramos que:

$$p_0 = \frac{1}{\sum_{k=0}^{M}\left(\frac{\lambda}{\mu}\right)^k} = \frac{1-\rho}{1-\rho^{M+1}} \qquad (24)$$

El número medio de clientes en la cola es:

$$\overline{N} = \sum_{k=0}^{M} kp_k = p_0 \sum_{k=0}^{M} kp^k \qquad (25)$$

Sabemos que:

$$\sum_{k=0}^{M} p^k = \frac{1-\rho^{M+1}}{1-\rho} \qquad (26)$$

Derivando esta expresión respecto a **ρ** y multiplicando por **ρ**, podemos sustituir ya en (25), resolviendo el problema.

1.1.2. Colas M/M/Q

Supongamos ahora un modelo de nodo algo más realista, donde llegan de media **λ** tramas por segundo, que pueden ser canalizadas por **Q** líneas de salida. Si una trama llega en un momento en que todas las líneas de salida se encuentran ocupadas, es descartada. La pregunta es ¿qué número de salidas están ocupadas por término medio?. Una representación de la cadena de estados en este caso es la siguiente:

$$\begin{array}{ccccccc} -\lambda p_0 \rightarrow & & -\lambda p_1 \rightarrow & & -\lambda p_2 \rightarrow & \ldots \\ 0 & & 1 & & 2 & \ldots \\ \leftarrow \mu p_1 - & & \leftarrow 2\mu p_2 - & & \leftarrow 3\mu p_3 - & \end{array}$$

Obsérvese que el número de transiciones por unidad de tiempo hacia un estado inferior depende del estado actual. Claramente, cuantas más líneas haya ocupadas en un momento dado, mayor es la probabilidad de que alguna quede libre en el instante siguiente.

El sistema que hemos de resolver ahora es el siguiente:

$$\begin{aligned} \lambda p_0 &= \mu p_1 \\ \lambda p_1 &= 2\mu p_2 \\ \lambda p_2 &= 3\mu p_3 \\ &\ldots \\ \lambda p_{k-1} &= k\mu p_k \end{aligned} \qquad (27)$$

de donde obtenemos la solución general:

$$p_k \frac{1}{k!}\left(\frac{\lambda}{\mu}\right)^k p_0 \qquad (28)$$

Como siempre, la probabilidad de que la cola se encuentre vacía se obtiene de la condición de normalización, es decir, del hecho de que la suma de las probabilidades para todos los estados es la unidad:

$$p_0 = \frac{1}{\sum_{k=0}^{Q} \frac{1}{k!}\left(\frac{\lambda}{\mu}\right)^k} \qquad (29)$$

En un modelo sencillo, el nodo dispondría de un buffer par almacenar aquellas llamadas que llegasen cuando todas las líneas estuviesen ocupadas. Inmediatamente que queda libre una línea, uno de los paquetes del buffer es tomado y colocado en esa línea. Si este proceso no altera significativamente la operación del nodo, se puede modelar el buffer como una cola de probabilidad de transición hacia estados superiores dada por λp_Q y probabilidad de transición hacia estados inferiores dada por **μQ**. Gráficamente:

$$\begin{array}{ccccccc} -\lambda p_Q \rightarrow & & -\lambda p_Q \rightarrow & & -\lambda p_Q \rightarrow & \ldots \\ 0 & & 1 & & 2 & \ldots \\ \leftarrow \mu Q - & & \leftarrow \mu Q - & & \leftarrow \mu Q - & \end{array}$$

Llamando $\overline{\lambda} = \lambda p_Q$ y $\overline{\mu} = \mu Q$,el número medio de tramas almacenadas en el buffer viene dado por:

$$\overline{N} = \frac{\overline{\rho}}{1-\overline{\rho}} \qquad (30)$$

con $\overline{\rho} = \frac{\overline{\lambda}}{\overline{\mu}}$.

1.2. Procesos de Markov

Constituyen una forma sencilla y útil de dependencia entre las variables aleatorias que forman el proceso estocástico.

Un proceso de Markov con un espacio de estado discreto se denomina cadena de Markov. La cadena de Markov de tiempo discreto es la más fácil de conceptualizar y comprender. Un conjunto de variables $\{X_n\}$ forma una cadena de Markov si la probabilidad que el nuevo estado sea x_{n+1} depende solo del estado actual x_n y no de los estados anteriores; es decir la dependencia se extiende hacia atrás solo una unidad de tiempo o bien la historia del proceso que afecta a su futuro queda resumida en su estado actual.

Analíticamente la propiedad de Markov puede escribirse como:

$$P[X(t_{n+1}) = x_{n+1} \mid X(t_n) = x_n, X(t_{n-1}) = x_{n-1}, \ldots, X(t_1) = x_1] =$$
$$= P[X(t_{n+1}) = x_{n+1} \mid X(t_n) = x_n]$$

donde $t_1 < t_2 < \ldots < t_n < t_{n+1}$ y x_i está incluida en algún espacio de estado discreto.

1.2.1. Procesos Semimarkovianos

En este caso lo que se hace es relajar la exigencia de falta de memoria en las distribuciones que definían los intervalos entre transiciones de los procesos de Markov haciendo que ahora pueda ser cualquier tipo de distribución.

Se observa, sin embargo que en los instantes de transición entre estados, el proceso se comporta exactamente como un proceso de Markov ordinario y se dirá que los procesos semimarkovianos tienen un proceso de Markov incluido en los instantes de transición.

1.2.2. Procesos Nacimiento-Muerte

Constituyen una clase muy importante dentro de los procesos de Markov y la condición suplementaria que presentan tanto en tiempo continuo como discreto es que las transiciones se producen solo entre estados vecinos. Esto es, si se elige el conjunto de los enteros como espacio de estado discreto (sin pérdida de generalidad), entonces el proceso de nacimiento-muerte requiere que si $X_n = i$, entonces $X_{n+1} = i - 1$, i ó $i + 1$ y ningún otro.

Los procesos Nacimiento-Muerte (N-M) son un tipo particular de proceso estocástico útil para modelar sistemas en los que los clientes llegan y completan su servicio de uno en uno. El estado del sistema estará representado por la variable aleatoria que corresponde al número de clientes en el mismo, k.

Sea un sistema caracterizado por el número k de elementos que hay en él y tal que estos elementos puedan nacer (llegar al sistema) o morir (salir del sistema). Si se denomina $P_k(t)$ la probabilidad de que haya k elementos en el instante t y se considera que en un intervalo suficientemente pequeño sólo puede variar el estado del sistema en un elemento en más o en menos, es decir que para que el instante $t + \Delta t$ haya k elementos es preciso que:

- O en el instante t hubiera k elementos y no se produjera ningún cambio.
- O en el instante t hubiera $k - 1$ elementos y se produjera una llegada.
- O en el instante t hubiera $k + 1$ elementos y se produjera una salida.

Además el estado del sistema nunca puede tomar valores negativos, es decir, el mínimo número de elementos que puede haber en el sistema es de cero.

Por ejemplo, un arribo a la cola (nacimiento) provoca el cambio de estado por + 1 y una partida después del servicio en la cola (muerte) provoca un cambio de estado –1.

En la Figura 1.9 se muestra la visión gráfica de los procesos descritos anteriormente.

Figura 1.9. Vista Gráfica de los Procesos Descritos.

PARTE 2. CONCEPTOS DE CALIDAD DE SERVICIO

1. INTRODUCCIÓN

Los sistemas informáticos actuales se basan en una red de datos, a la cual se pide que soporte, cada vez más, una amplia gama de aplicaciones. El Protocolo Internet (IP), utilizado en las redes de datos durante los últimos 30 años para intercambiar información entre ordenadores de diferentes fabricantes, se ha convertido, después de sucesivas mejoras, en el protocolo más usado. El desarrollo actual de redes IP se está enfocando hacia la Calidad de Servicio (QoS), la cual se requiere para permitir que los diferentes tipos de informaciones se transporten con distintas prioridades. El objetivo es prevenir que la congestión no llegue a ser un problema crítico en aplicaciones sensibles a los retardos, tales como la transmisión de voz y vídeo. Existen dos formas de cumplir este objetivo:

- Entrar en la carrera de la velocidad de conmutación y del ancho de banda. Esto implica una inversión a largo plazo para asegurar que el rendimiento de la red se mantenga con el cambio de requisitos. Este método puede llegar a ser muy costoso.

- Gestionar "inteligentemente" el ancho de banda disponible, compartiéndolo de forma óptima entre el tráfico de varias fuentes. No obstante, las redes y protocolos IP son independientes de las aplicaciones que transportan: no distinguen una aplicación de otra, Por esto se requiere que las funciones de QoS "sepan" reconocer las aplicaciones para asignarles la prioridad deseada.

Por lo anterior, el reto de los administradores y los arquitectos de las redes ha sido construir infraestructuras para soportar nuevas aplicaciones de voz, imágenes y datos basadas en IP junto con sus aplicaciones tradicionales orientadas a circuitos sobre WAN, compuestas por una variedad de medios.

Las siguiente secciones describe brevemente gran parte de las herramientas y mecanismos que se utilizan para conseguir la QoS de extremo a extremo y la utilización óptima del ancho de banda WAN.

2. DEFINICIÓN

La Calidad de Servicio se traduce como la capacidad de una red para entregar un servicio específico a un tipo concreto de tráfico, utilizando diferentes tecnologías de transporte, tales como Frame Relay, ATM, Jerarquía Digital Síncrona (SDH), 802.1P, etc.

La QoS también se define como la habilidad de un elemento de red (una aplicación, un enrutador, etc.) para proveer algún nivel de seguridad en la entrega de los datos de usuario, cumpliendo con un conjunto de parámetros que definen el desempeño de la red y forman parte del contrato entre el usuario y el proveedor de servicio. Las medidas de QoS cubren la calidad de transmisión, disponibilidad del servicio, latencia, retardos, etc. Las diversas aplicaciones tienen diferentes requerimientos, en particular, el tráfico de tiempo real (voz y video) no puede tolerar retardos en la transmisión o pérdida de paquetes; como consecuencia se da un tratamiento diferencial a cada tipo de tráfico, principalmente en el momento de congestión.

Para satisfacer los requerimientos de cada tipo de aplicación, se han definido clases básicas de QoS:

- Servicios Integrados. Los recursos de red son asignados y reservados de acuerdo a los requisitos de QoS de la aplicación, y están sujetos a políticas de gestión de ancho de banda.

- Servicios Diferenciados. El tráfico es clasificado y los recursos de red son asignados de acuerdo a políticas de gestión de ancho de banda y a la información de prioridad codificada en el proceso de clasificación. Para proporcionar QoS, los elementos de red dan un tratamiento preferencial a las clasificaciones identificándolas por los requerimientos de ancho de banda.

Estas dos clases son aplicadas a flujos individuales ó flujos agregados, de acuerdo a esto se definen dos o más maneras de caracterizar estas clases de QoS:

- Por flujo. Un flujo se define como un grupo de datos unidireccionales entre dos aplicaciones (fuente y destino), identificado por el protocolo de transporte, dirección de la fuente y el destino o por el número de puerto de la fuente y el destino.

- Por Unión de flujos. Una unión es simplemente dos o más flujos que tienen algún parámetro en común, por ejemplo, protocolo de transporte, dirección fuente y destino, número de puerto, número de prioridad o alguna otra información de autenticación.

Con el fin de implementar cualquier tipo de QoS y optimizar el ancho de banda, se definen responsabilidades específicas para la frontera como para el núcleo de la red; donde se destacan tres partes principales:

- QoS dentro de un elemento de red (gestión de colas, herramientas para la clasificación y vigilancia de tráfico, etc).

- Técnicas de señalización para coordinar la QoS de un extremo a otro entre los elementos de red.

- Políticas de QoS, gestión, y funciones para controlar y administrar el tráfico a través del dominio.

3. TECNOLOGÍAS Y ESPECIFICACIONES

Para llevar a cabo las metas de QoS, los elementos de red y el software de gestión, deben proveer la habilidad para garantizar el ancho de banda así como las características de retardo (latencia y jitter) por clase de tráfico o flujo. Para satisfacer estas metas de QoS se deben combinar la velocidad de clasificación de tráfico, el procesamiento en colas, y los mecanismos basados en políticas.

3.1. Herramientas de Gestión de Congestión.

Las herramientas de gestión de congestión ayuda a controlar sobrecargas de tráfico sin pérdidas excesivas de datos cuando un enrutador recibe más tráfico que el de su capacidad física de procesamiento, necesita almacenar los datos hasta que puedan ser procesados, y requiere de algoritmos de colas para seleccionar el tráfico y determinar el método de prioritización. Las herramientas de colas necesarias en la gestión de congestión son:

- Cola FIFO
- Cola de prioridad (PQ – Priority Queuing)
- Cola de Usuario (CQ – Custom Queuing)
- WFQ

Cada uno de estos algoritmos fueron diseñados para resolver un problema de tráfico específico y tienen un efecto particular sobre el desempeño de la red, como se describe a continuación.

3.1.1. Cola FIFO.

En las colas FIFO, cuando la red se congestiona se almacenan los paquetes entrantes y posteriormente, cuando este problema se haya superado, se envían los paquetes en el orden de llegada. En las colas FIFO las decisiones no se toman teniendo en cuenta prioridades de los paquetes, sino según el orden de llegada, el cual determina el ancho de banda y el espacio de buffer que se le asignará al paquete. A pesar de que las Colas FIFO fueron el primer paso para controlar el tráfico en las redes IP, actualmente se requieren de algoritmos más sofisticados que permitan dar a cada clase de servicio un tratamiento diferencial.

3.1.2. Cola de Prioridad (PQ – Priority Queuing)

PQ asegura que el tráfico importante sea procesado rápidamente en cada nodo del trayecto. PQ fue diseñado para dar prioridad estricta a tráfico importante de la red. Al tráfico se le asigna una prioridad de acuerdo al protocolo de red (IP, IPX, etc.), interfaz de entrada, tamaño del paquete, dirección fuente/destino, etc. En PQ, basándose en la prioridad asignada, cada paquete es colocado en una de cuatro colas: alta, media, normal o baja. Si los paquetes no son clasificados, caen en la cola normal (Figura 2.1). Durante la transmisión, el algoritmo da a las colas de prioridad alta un tratamiento preferencial absoluto sobre las colas de baja prioridad.

PQ es útil para asegurar que el tráfico de misión critica que pasa por varios enlaces WAN tenga un tratamiento prioritario. PQ actualmente, usa configuración estática y no se adapta automáticamente a los cambios de requerimientos de la red.

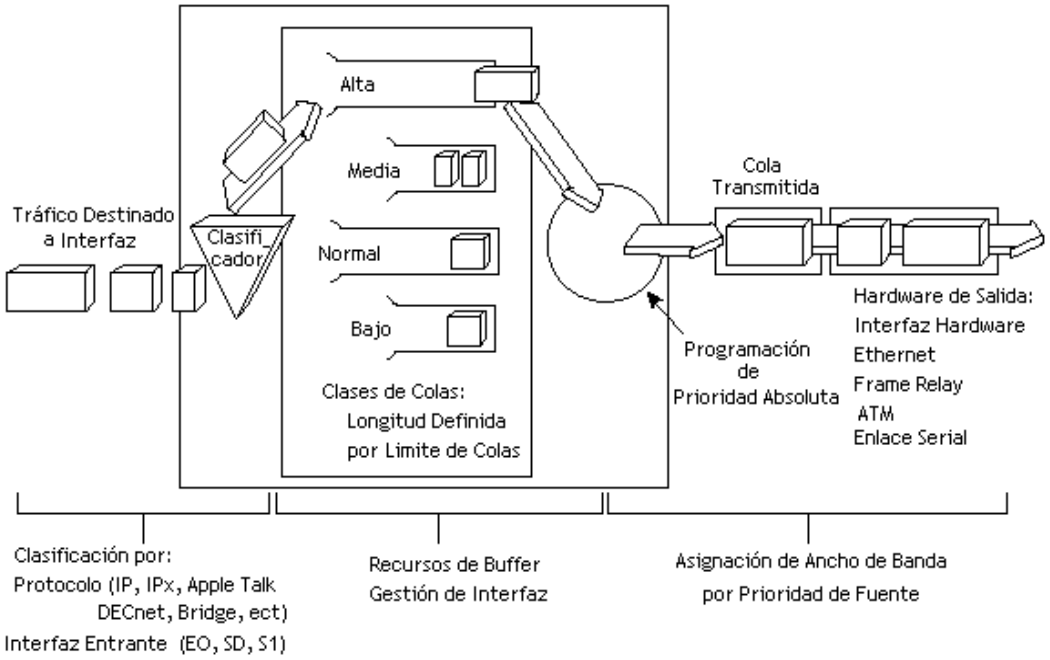


Figura 2.1. Funcionamiento PQ.

3.1.3. Cola de Usuario (CQ – Custom Queuing)

CQ fue diseñado para permitir que varias aplicaciones u organizaciones compartan la red entre aplicaciones con anchos de banda mínimo específicos o requerimientos de latencia. En este ambiente, el ancho de banda se comporte proporcionalmente entre aplicaciones y usuarios. CQ se usa para garantizar ancho de banda en un punto de congestión potencial, asegurando que un tráfico use una porción del ancho de banda disponible, y deje el resto para otro tráfico. CQ maneja el tráfico asignando una cantidad específica del espacio en cola para cada clase de paquetes y les da un tratamiento aleatorio.

3.1.4. WFQ

Es un algoritmo de gestión de congestión, que permite establecer prioridades para compartir equitativamente el ancho de banda entre las clases de tráfico. WFQ asigna un peso a cada flujo, el cual determina el orden de transmisión de los paquetes, si el peso es de un valor bajo será el primero en ser enviado. El peso se establece por direcciones IP, servicio, número de puerto o por el campo de precedencia de la cabecera IP. WFQ es diseñado para minimizar los esfuerzos de configuración y para adaptarse automáticamente a condiciones cambiantes del tráfico.

3.2. Clasificación del tráfico

La clasificación de tráfico permite diferenciar los servicios para después asignarles las restricciones y el tratamiento necesarios encada uno de los enrutadores. Con la clasificación, el enrutador mantiene cuatro colas independientes (Figura 2.2), dentro de las cuales se dividen los flujos entrantes del nivel 2, 3 o 4. Cada clasificación del tráfico es tratada como un flujo individual por los enrutadores. En general, la clasificación depende del nivel al que pertenece, por ejemplo, el flujo de nivel 2 son clasificados de acuerdo a prioridades, dirección MAC o por el puerto de entrada, él de nivel 3 se clasifica en base a dirección IP fuente/destino y el flujo de nivel 4 se clasifica según el número de puerto TCP/UDP en adición a la dirección IP, byte ToS, tipo de protocolo e interfaz o puerto de entrada. Una vez el tráfico es clasificado y puestos en cola, el usuario aplica una tasa limite y políticas de colas al tráfico, de esta manera , otras funciones no son afectadas y además se tendrá más control en el acceso a la red.

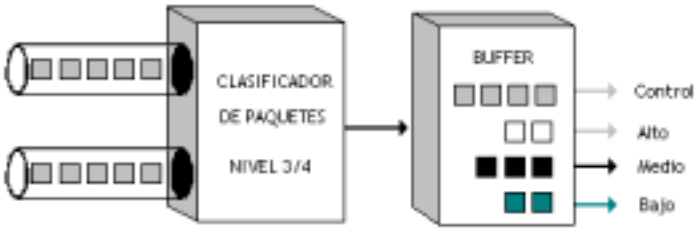


Figura 2.2. Clasificación del Tráfico.

3.3. Herramientas para Evitar la Congestión

Las herramientas para evitar la congestión constituyen políticas de gestión de congestión preventivas, que se dividen en tiempos de respuesta largo, mediano o corto. Las políticas a largo plazo incluyen la planeación para extender la capacidad de la red usando estimaciones o pronósticos de la demanda y la distribución del tráfico futuro. Las políticas a termino medio abarcan escalas de tiempo de minutos y días, sus acciones se ajustan a parámetros de enrutamiento y reconfiguración de la topología lógica de la red. las herramientas a corto tiempo incluye funciones de procesamiento a nivel de red, y entre ellas se encuentran:

- (RED - Random Early Detection) evita la congestión controlando el tamaño promedio de la cola. Durante la congestión (pero antes de que la cola se sature), el mecanismo RED marca los paquetes de acuerdo a un algoritmo probabilístico el cual tiene en cuenta el tamaño promedio de la cola. Los paquetes marcados son descartados cuando la congestión se incrementa. Los resultados de descartar un paquete es que la fuente detecta la acción y reduce su transmisión.

- (WRED -Weighted Random Early Detection) combina las capacidades de RED con los niveles de precedencia IP separando umbrales y pesos ó prioridades, así diferentes niveles de QoS son implementados para cada clase de tráfico; por ejemplo, el tráfico estándar es descartado más frecuentemente que el tráfico premium durante periodos de congestión.

- D-WRED – Distributed Weighted Random Early Detection) es una versión avanzada de WRED que corre sobre procesos distribuidos y provee más funcionalidades, tales como, definición de umbrales máximo y mínimo de la longitud de la cola y aumenta las capacidades de descarte para cada clase de tráfico.

3.4. Herramientas de Vigilancia y filtrado de tráfico

Para gestionar el tráfico y la congestión de la red se usan herramientas de vigilancia y filtrado de tráfico. Las herramientas de vigilancia hace un seguimiento al tráfico introducido por el usuario en la red, para verificar que no exceda el perfil pactado, y las herramienta de filtrado establece márgenes máximos al tráfico a ráfagas, suele utilizarse para fijar una QoS entre el operador y el usuario; entre tanto el usuario respete lo establecido, el operador se compromete a no descartar sus paquetes. Los mecanismos más utilizados son la Configuración Genérica de tráfico (GTS – Generic Traffic Shaping) y Conformación de Tráfico Frame Relay (FRTS – Frame Relay Traffic Shaping).

3.4.1. GTS

GTS provee un mecanismo de control del flujo sobre una interfaz particular, conocido como buffer con créditos. El GTS .(Figura 2.3) se aplica por interfaz usando una lista de acceso para seleccionar el tráfico a conformar, y trabaja con una variedad de tecnologías de nivel 2, incluyendo Frame Relay, ATM y Ethernet.

El mecanismo de control de flujo para evitar congestión reduce el tráfico saliente restringiéndolo a una tasa de bit particular, mientras se almacenan en cola las ráfagas del dicho tráfico. De esta manera, el tráfico que pertenece aun perfil puede filtrarse para satisfacer requerimientos downstream, eliminando cuellos de botella en la topología.

El algoritmo de buffer con créditos intenta compensar al usuario que alterna intervalos de tráfico con otros de inactividad, frente al que esta siempre transmitiendo: cuando el nodo no envía datos a la interfaz éste va sumando créditos hasta un máximo igual a la capacidad del buffer; los

créditos acumulados pueden utilizarse después para enviar ráfagas con un caudal mayor de lo normal; cuando se agotan los créditos el caudal vuelve a su valor normal, y funciona como un buffer agujereado.

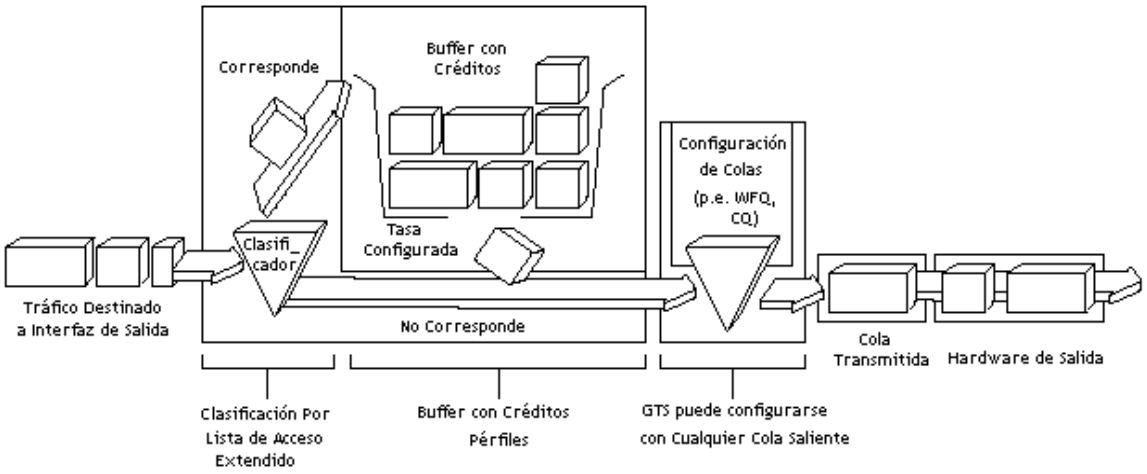


Figura 2.3. Funcionamiento de GTS.

3.4.2. FRTS

FRTS provee parámetros que son útiles para controlar la congestión del tráfico en la red. FRTS proporciona un mecanismo para compartir el medio entre múltiples VCs y configura una tasa aplicada (tasa pico configurada para limitar el tráfico saliente) en base a circuitos virtuales, que puede ser la Tasa de Información Negociada (CIR – Committed Information Rate) o algún otro valor definido, tal como la tasa de Información Excedida (EIR – Excess Information Rate). La definición de la tasa aplicada elimina los cuellos de botella en las redes Frame Relay y limita la tasa con que los datos son enviados a los circuitos virtuales. Además esta tasa permite controlar la velocidad de transmisión de los enrutadores teniendo en cuenta el CIR o el EIR, finalmente, con el uso de esta tasa se preasigna el ancho de banda de cada VC, creando una red virtual.

4. PROTOCOLOS DE QOS

Para realizar un uso eficiente de la red y a su vez garantizar calidad de servicio, es necesario desarrollar todo un conjunto de protocolos que permita una buena gestión y control de todos los recursos y que faciliten la adaptación a las necesidades de los tipos de QoS. Así dentro de los protocolos que proporcionan QoS a nivel de red y transporte son RSVP, Servicios Diferenciados, MPLS y SBM.

4.1. RSVP

El Protocolo de Reservación de Recursos (RSVP - Resource reSerVation Protocol) es un protocolo de señalización que proporciona un control para la reserva de los recursos, orientado fundamentalmente a redes IP. Es un componente clave de la arquitectura de los Servicios Integrados (IntServ – Integrated Service) en Internet, en la que se define el funcionamiento y la forma de petición e intercambio de información entre cada elemento de la red para realizar un control de la calidad de servicio .

RSVP es hasta el momento, la más compleja de todas las tecnologías de QoS para las aplicaciones y para los distintos elementos de la red. Como resultado, representa el mayor estándar creado desde el servicio de mejor esfuerzo de las redes IP, que proporciona el mayor nivel de QoS en términos de servicio garantizado y granularidad de asignación de recursos.

RSVP define un modelo de asignación de QoS en el que cada receptor (para una sesión) es responsable de elegir su propio nivel de reservación de recursos, iniciando la reserva y manteniéndola activa tanto tiempo como desee. Este modelo consiste en una solución distribuida que permite a múltiples receptores heterogéneos efectuar reservas específicamente dimensionadas según sus propias necesidades. Además, para mantener el control, el receptor puede enviar sus especificaciones a la fuente encargada de solicitar las reservas de la red. En definitiva, RSVP permite que las aplicaciones soliciten una calidad de servicio específica a la red. como consecuencia, RSVP se considera como un protocolo de control, en vez de un protocolo de encaminamiento y su tarea consiste en establecer y mantener las reservas de recursos en un árbol de distribución, con independencia de cómo se hayan creado.

El modelo de Servicios integrados ha considerado la existencia de varias clases de QoS, si bien actualmente sólo dos de éstas han sido formalmente especificadas como compatibles con RSVP:

- Servicios garantizados: Este servicio proporciona un nivel de ancho de banda y un límite en el retardo, garantizando la no existencia de pérdidas en colas. Está pensado para aplicaciones con requerimientos en tiempo real, tales como aplicaciones de audio y vídeo. Cada enrutador caracteriza el servicio garantizado para un flujo específico asignando un ancho de banda y un espacio en buffer.

- Servicio de Carga Controlada: A diferencia del servicio garantizado, este no ofrece garantías en la entrega de los paquetes. Así, será adecuado para aquellas aplicaciones que toleren una cierta cantidad de pérdidas y un retardo mantenidos en un nivel razonable. Los enrutadores que implementen este servicio deben verificar que el tráfico recibido siga las especificaciones exigidas, y cualquier tráfico que no las cumpla será reenviado por la red, como tráfico de mejor esfuerzo.

Los servicios integrados se implementan utilizando cuatro componentes básicos: un protocolo de señalización, RSVP; una rutina de admisión de control, un clasificador y un programador de paquetes. De esta manera, cualquier aplicación que requiera de QoS bien sea garantizado o de carga controlada requerirá inicializar los trayectos a utilizar, reservando los recursos previamente

al envío de la información. Las rutinas de control de admisión decidirán si la petición de reserva de recursos podrá ser admitida. Cuando un enrutador recibe un paquete, el clasificador realiza una división en base a varios campos, ubicando éste en una cola específica en función del resultado de la clasificación. El programador de paquetes los trata de manera que se cumpla los parámetros de QoS acordados para cada conexión.

Para tomar decisiones de QoS asociadas a los paquetes de una aplicación, RSVP interactúa con el clasificador y el programador de paquetes instaladas en los nodos. Primero consulta a los módulos las decisiones locales para saber si la QoS deseada puede ser provista (bien mediante decisiones basadas en recursos o bien mediante decisiones basadas en políticas) y, en consecuencia, establece los parámetros requeridos en el clasificador y en el programador del paquete.

El clasificador de paquetes determina la ruta del paquete y el programador toma las decisiones de envío para alcanzar la QoS deseada, negociando, si es necesario, con aquellos nodos que tengan capacidad propia de gestión de QoS, para proporcionar la calidad solicitada por RSVP.

4.1.1. Tipos de Mensajes

Existen dos tipos fundamentales de mensajes en RSVP, Resv y Path. Una aplicación solicita participar en una sesión RSVP como emisor, enviando un mensaje Path en el mismo sentido que el flujo de datos, por las rutas uni/multicast proporcionadas por el protocolo de enrutamiento. Al recibir este mensaje, el receptor transmite un mensaje Resv, dirigido hacia la fuente de los datos, siguiendo exactamente el camino inverso al de los mismos, en el cual se especifica el tipo de reserva a realizar en todo el camino.

En general, sin especificar tipos de QoS un mensaje Path, contiene Parámetros que describen el formato de los paquetes que la fuente generará, información sobre la QoS y propiedades de la aplicación, necesaria para poder encaminar los mensajes Resv.

En el caso de los mensajes Resv, se especifican los recursos a reservar para la sesión (requisitos de ancho de banda y de retardos), e indican qué subconjunto de paquetes de la sesión han de recibir la QoS, para ello utilizan cualquier campo de las cabeceras de protocolo o de las cabeceras de aplicaciones (discriminar por dirección de origen, por aplicación de destino, por el puerto, por el protocolo, etc.).

4.1.2. Descripción de Problemas

Hay dos problemas fundamentales que afectan el funcionamiento del protocolo RSVP, la escalabilidad y el enrutamiento.

La cantidad de información relacionada con el estado de cada conexión se incrementa proporcionalmente al número de flujos manejado. Todo ello obliga a que los enrutadores dispongan de capacidad extra de almacenamiento, además de tener que procesar toda la

información de sobrecarga relacionada al propio funcionamiento del mecanismo. Todo ello crea problemas de escalabilidad y enrutamiento en el núcleo de la red.

Otro problema que plantea esta técnica es el hecho de obligar a trabajar con enrutadores que incorporen el protocolo de señalización RSVP, el control de admisión, clasificador y programador de paquetes obligando a cambios globales en todos los elementos de red.

4.2. DIFFSERV

Dados los problemas presentados por lel protocolo RSVP y en general por los servicios integrados, se presenta otra técnica denominada Servicios Diferenciados (DiffServ – Differentiated Services), que permite una mejor implementación y control de establecimiento de diferentes calidades de servicio en una misma red. DiffServ es un protocolo de QoS propuesto por IETF que distingue diferentes clases de servicios marcando los paquetes IP. A diferencia de RSVP no especifica un sistema de señalización, consiste en un método para marcar o etiquetar paquetes, permitiendo a los enrutadores modificar su comportamiento de envío. Cada tipo de etiqueta representa una determinada clase de QoS y al tráfico con la misma etiqueta se le da un mismo tratamiento.

Para proporcionar los diferentes niveles de servicio, DiffServ modifica la estructura del campo ToS del protocolo IPv4, y lo redefine como campo Diffserv Codepoint (DSCP), que comsiste de 8 bits, entre los cuales, los 2 últimos son reservados para futuras aplicaciones, y con los 6 bits restantes se consiguen 64 combinaciones: 48 para el espacio global y 16 para uso local.
Para que un usuario reciba los servicios diferenciados por parte de su proveedor de acceso, debe haber acordado de antemano un cierto nivel de servicio estático o dinámico, según si la negociación se hace de forma cuasipermanente (ejemplo mensualmente) o de forma dinámica según las necesidades de cada momento (en este caso los clientes deben usar protocolos de señalización como RSVP).

Los usuarios marcan los paquetes de forma individual para indicar el tipo de servicios que desean recibir, o bien son los enrutadores de frontera los encargados de utilizar el campo ToS de los paquetes que ingresan en el Dominio DS. Una vez en la red, los paquetes son clasificados, inspeccionados modifican su estadística si fuere preciso. Para clasificar el tráfico se proponen básicamente tres opciones de marcas:

1. Ninguna (None): ofrece el servicio del mejor esfuerzo convencional.
2. Asegurado y dentro del Perfil (Assured and in profile): definida en el SLA entre el cliente y el proveedor de servicio.
3. Asegurado y fuera de perfil (Assured and out of profile): no cumplirá lo definido en el SLA entre el cliente y el proveedor de servicio.

Una vez los paquetes son clasificados, se define un proceso de envío conocido como PHB que describe el tratamiento recibido por los paquetes a lo largo de su transmisión para entregar la niveles de QoS específicos. Los tratamientos se definen de acuerdo a tipos de políticas aplicados,

conformación del tráfico, posibles remarcados en el campo DS, encolamientos y gestión del tráfico. Existen varios tipos de PHBs:

B) Envío Acelerado (EF - Expedited Forwarding): Tiene un solo valor de DiffServ (codpoint). Minimiza el retardo, el jitter y asegura baja pérdida de paquetes, proporcionando el mayor nivel de QoS.
C) Envío Asegurado (AF - Assured): Define cuatro clases de probabilidades de tráfico, con 3 variaciones en cada una. La probabilidad de entrega no es tan alta como en EF, provocando retardos.
D) Envío por Defecto (DE - Default Fordwarding): Funciona como el servicio de mejor esfuerzo, tradicional.

En la siguiente Figura 2.4 se comprueba el funcionamiento de los PHBs en los enrutadores visualizando como se clasifica, marca y condiciona el tráfico de acuerdo a unos criterios de políticas predeterminadas. El tráfico será marcado y transportado de acuerdo a las marcas.

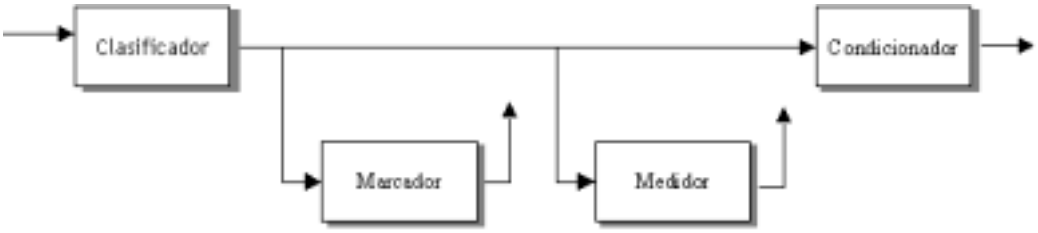


Figura 2.4. Arquitectura de Servicios Diferenciados

El clasificador de tráfico selecciona los paquetes aplicando un filtro que define las condiciones que la cabecera IP debe cumplir para ser aceptado en un PHB. Si el filtro acepta el tráfico su perfil es adicionado para que el filtro sea aplicado. En general hay dos tipos de casificadores:

- Comportamiento Agregado (BA- Behaviour Aggregate): Usa sólo el valor DSCP como índcie en la tabla de los PHBs.
- Multicampo: Usa otra información de la cabecera IP, como la dirección del destino, dirección de la fuente, número de puerto, etc.

Despues que los paquetes son clasificados, el medidor calcula el nivel de tráfico y lo compara con el perfil especificado en el SLA que incluye aspectos como el promedio de la tasa de datos, tasa máxima, etc. El medidor vigila el nivel del tráfico para cada clase de servicio y toma un número de acciones si el nivel excede los parámetros asociados.

Cunado los datos han sidos clasificados y la tasa es determinada, el PHB seleccionado es codificado en el byte DS. La razón por la cual el PHB no es marcado directamente por el clasificador es que éste es independiente de la tasa de los datos.

Finalmente, el condicionador altera los paquetes para que cumplan las reglas de los servicios, mediante funciones de etiquetado, modelado y monitorización .

Estas funcionalidades están activas en cada enrutador habilitado para ofrecer Diffserv, aunque no todas las funciones se utilizan al mismo tiempo. Los enrutadoees de frontera utilizan más estas funciones que los enrutadores del núcleo.

4.3. Gestión de Ancho de Banda de Subred (SBM - Subnet Bandwidth Management)

Hasta ahora se ha estudiado cómo obtener QoS extremo a extremo entre el emisor y el receptor, esto significa que cada enrutador a lo largo de la ruta debe soportar la tecnología de QoS que se esté usando, tal y como se vio en la descripción de los anteriores protocolos de QoS, pero también hay que tener en cuenta la posibilidad de conseguir QoS en los nodos finales (top-to-bottom). Para ello es necesario que :

1. La fuente y destino deben permitir la obtención de QoS, siendo necesario que las aplicaciones la permitan explícitamente o que la permita el sistema implícitamente. Cada capa OSI, desde la aplicación a las capas inferiores, deben utilizar también QoS para asegurar que las peticiones de alta prioridad sean tratadas desde la fuente o el destino.

2. Si los sistemas finales se conectan a una red de área local (LAN), éstas deben permitir QoS, de forma que las tramas de alta prioridad sean tratadas con prioridad mientras circulan por la red (ejemplo de nodo-a-nodo, nodo-a-enrutador, enrutador-a-enrutador). De esta forma se proporciona QoS en la capa de enlace del modelo OSI, mientras que los protocolos anteriores ofrecían QoS en otras capas superiores.

Existen algunas tecnologías creadas para proporcionar QoS en la capa de enlace, como ATM, pero ésta es una tecnología imposible de implementar por algunas empresas, debido a su costo y a su complejidad. Todas estas empresas, por el contrario, utilizan otras tecnologías más comunes para sus LANs, tales como Ethernet, que originalmente no fueron diseñadas para ofrecer QoS. Ethernet proporciona, simplemente, un servicio análogo al prestado por IP, el servicio del mejor esfuerzo, en el que existe la posibilidad de que se produzcan retardos y variaciones (jitter) que pueden afectar las aplicaciones de tiempo real. Por todas estas cosas, IEEE ha redefinido el estándar Ethernet y otras tecnologías de la capa de enlace para proporcionar QoS, mediante diferenciación del tráfico.

Los estándares de IEEE 802.1p, 802.1q y 802.1D definen cómo los conmutadores Ethernet pueden clasificar las tramas para entregar en primer lugar el tráfico considerado crítico. El grupo de trabajo del IETF ha especificación el mecanismo de Capas de Enlace Específicos sobre Servicos Integrados (ISSL – Integrated Services over Specific Link Layers) que se encarga de definir cómo relacionar los distintos protocolos de QoS de capas superiores con las diferentes tecnologías de la capa 2, como Ethernet. Entre otras cosas, el ISSL ha desarrollado el protocolo de Gestión del ancho de banda de la subred (SBM - Subnet Bandwidth Manager) para aplicarlo con LANs 802. SBM es un protocolo de señalización que permite la comunicación y coordinación entre nodos de la red y su relación con protocolos de QoS de capas superiores. Un requisito fundamental en SBM es que todo el tráfico debe pasar por lo menos por un conmutador que utilice SBM.

4.3.1. Componentes de SBM

Los principales componentes lógicos de SBM son:

- Distribuidor de ancho de banda (BA - Bandwidth Allocator): Gestiona la asignación de los recursos y realiza el control de admisión de acuerdo a su disponibilidad y al resto de criterios definidos en la política del servicio.
- Módulo del cliente (RM - Requestor Module): Reside en cada estación final. La relación entre el RM y los parámetros de protocolos de QoS superiores son definidas de acuerdo a una política determinada.

En la siguiente Figura 2.5 se observar cómo la localización del BA determina el tipo de configuración del SBM en uso: centralizado o distribuido. Además, cuando existe más de un BA por segmento de red, uno de ellos será elegido como SBM Designado(DSBM - Designated SBM).

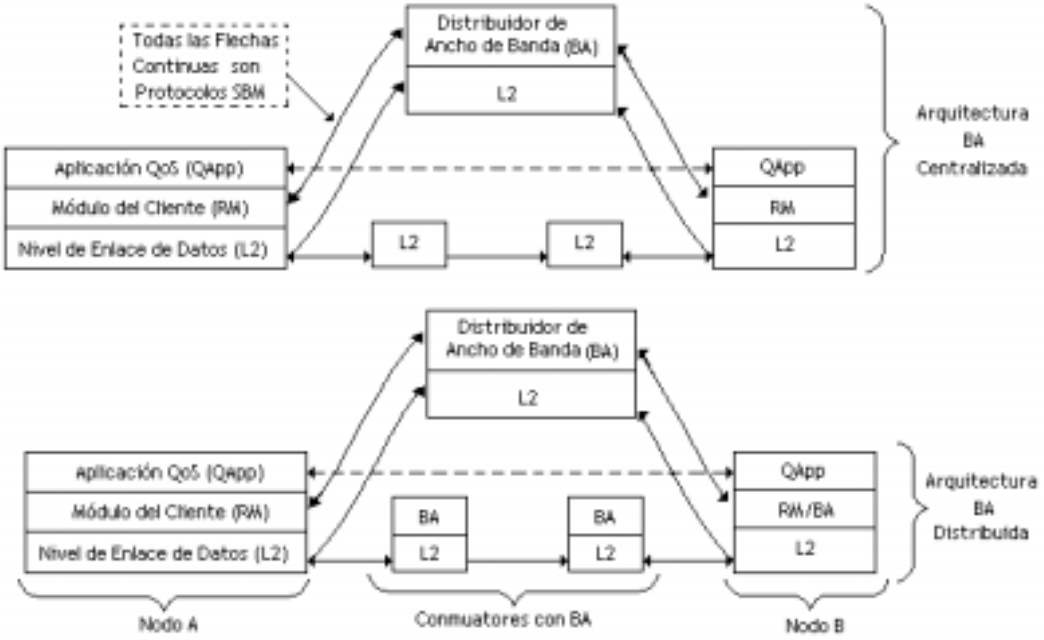


Figura 2.5. Componentes de SBM.

4.3.2. Funcionamiento de SBM

Este protocolo utiliza un mecanismo de señalización (usualmente RSVP) entre RM y BA para iniciar las reservas, consultar al BA los recursos disponibles y modificar las reservas. SBM también se usa entre las Aplicaciones de QoS (QoSpp – QoS Enabled Application) y el RM, pero es necesario involucrar Aplicaciones de programación de Interfaz (API - Applications Programming Interface) para simplificar el intercambio de primitivas funcionales. A continuación se describe de forma genérica el procedimiento del control de admisión en SBM:

1. El DSBM inicializa: consigue la disponibilidad de los recursos.

2. El cliente DSBM (cualquier nodo extremo o enrutador RSVP): busca el DSBM en los segmentos agregados a cada interfaz. (esta tarea esta monitorizada con el campo "AllSBMAddress", estando reservada como dirección IP multidifusión la 224.0.0.17).

3. El cliente envía un mensaje PATH con el campo "DSBMLogicalAddress", en lugar de la dirección de destino RSVP.

4. Una vez recibido el PATH, el DSBM indica su estado en el conmutador, almacenando la dirección de origen de capa 2 y capa 3 (L2/L3) y la pone en el mensaje, encaminándolo al próximo conmutador.

5. Cuando el mensaje es un RSVP RESV, éste se envía hasta llegar al primer enrutador.

6. DSBM evalúa la petición y si los recursos solicitados están disponibles se lo indica al emisor.

Como se observa, es un proceso muy parecido al ocurrido en los enrutadores RSVP. Por otro lado, cualquier DSBM puede añadir un objeto denominado TCLASS a los mensajes Resv o Path del protocolo RSVP. Este objeto contiene información de prioridad basada en la norma 802.1p. De esta manera la información de clase de servicio de las redes IEEE 802 puede ser transmitida por la red.

5. ARQUITECTURAS DE QOS

Excepto para el caso de SBM que utiliza RSVP para la señalización, el resto de protocolos se estudió de forma independiente; sin embargo, aunque los protocolos de QoS aquí señalados son diferentes, no se excluyen unos a otros, todo lo contrario, en la realidad se complementan a la hora de su aplicación para obtener los niveles de calidad requeridos en una determinada red. Esta complementación forma una gran variedad de arquitecturas en las que los protocolos trabajan conjuntamente para proporcionar QoS extremo a extremo a través de múltiples proveedores de servicio. La Figura 2.6 muestra en forma general cómo es posible mezclar los distintos protocolos.

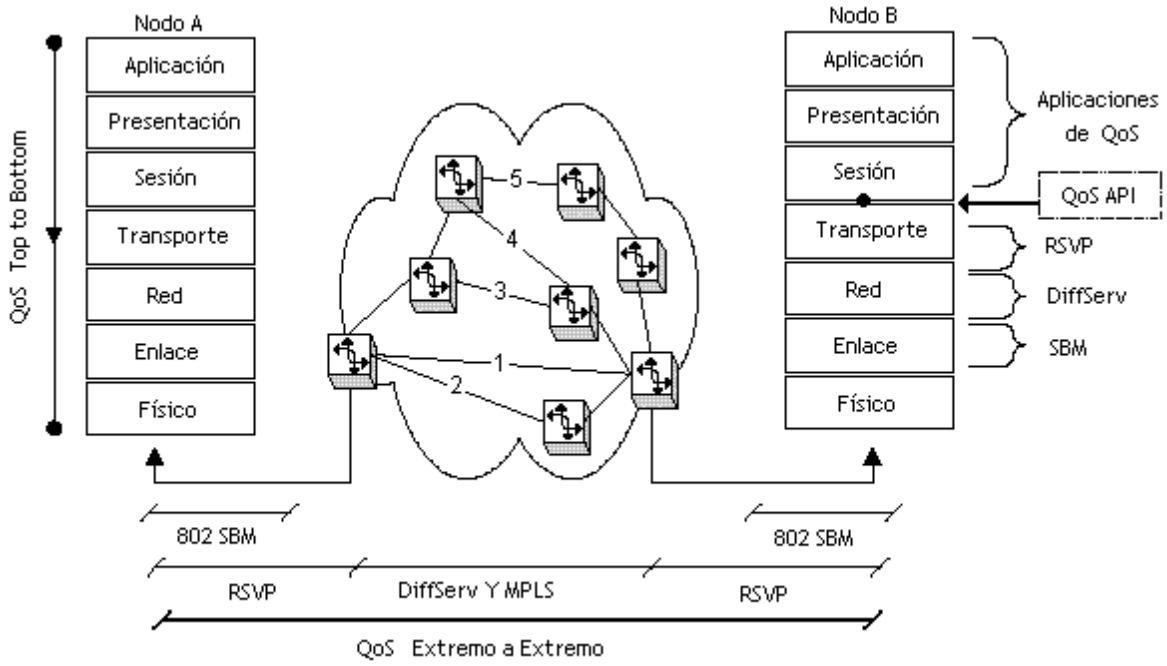


Figura 2.6. Arquitectura de QoS.

Para comprender aún mejor las arquitecturas descritas a continuación se describe brevemente cómo pueden trabajar juntas las distintas tecnologías de QoS para proporcionar calidad de servicio extremo a extremo.

5.1. RSVP Y DIFFSERV EXTREMO A EXTREMO

RSVP proporciona recursos para el tráfico de la red, mientras que Diffserv simplemente marca y clasifica el tráfico. RSVP es más complejo y demanda más actividad a los enrutadores que Diffserv, por eso, normalmente se utiliza DiffServ en el backbone.

A pesar de estas diferencias, Diffserv y RSVP se complementan perfectamente para ofrecer QoS extremo a extremo. Los nodos finales pueden utilizar peticiones RSVP con alta granularidad. Los enrutadores situados a la entrada del backbone de la red pueden asociar esas reservas RSVP a una determinada clase de servicio, indicada por un byte DS y acordada en los acuerdos de servicios (SLAs).

5.2. Servicios Integrados Extremo a Extremo sobre Redes DiffServ

En esta interacción los dominios de DiffServ se ven como elementos de red usados para implementar Servicios Integrados extremo a extremo. Dentro de la estructura, los servicios integrados (incluye RSVP) son considerados como la tecnología que solicita servicios y controla la

admisión del tráfico y DiffServ es la tecnología que permite implementar la QoS en las redes a gran escala.

En esta arquitectura, los enrutadores de los puntos de entrada y salida de la red se conocen como Enrutadores de Límite (ER - Edge Routers) en el ambiente de Servicos integrados, y en las regiones DiffServ se conocen como Enrutadores de Frontera (BR - Border Routers). De acuerdo a esta configuración, el trayecto extremo a extremo de Servicios Integrados se conforma de saltos a través de regiones DiffServ. Las regiones DiffServ se consideran como enlaces virtuales entre los nodos de Servicio Integrado (figura 2.7).

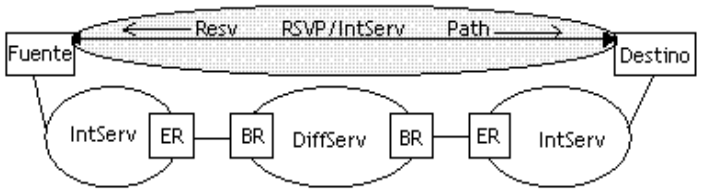


Figura 2.7. Estructura de Servicio Integrado sobre Diffserv

Los fundamentos de esta arquitectura son:

- Los usuarios residen en las regiones de Servicos Integrados de la red.
- Aunque RSVP no es parte integral de Servicios Integrados, esta arquitectura asume a RSVP como el mecanismos para reservar recursos.
- Los mensajes de señalización viajan extremo a extremo entre la fuente y el destino.
- El estilo de QoS de Servicos Integrados es codificado en el DCSP a la entrada de la región DiffServ.
- Los nodos internos de la región Diffserv pueden soportar RSVP y participar en la señalización, pero esta característica es opcional.
- El control de admisión en las partes que soportan Servicios Integrados se realiza salto por salto de acuerdo a los recursos locales. En la frontera entre las regiones de Servicios ntergados y DiffServ hay dos formas de implementar el control de admisión: en el caso en que una región DiffServ no use RSVp, el enrutador ER actúa como el agente para controlar la admisión de la región DiffServ, en caso contrario, los BR soportarán RSVP y actuaran como el agente para estas regiones.
- Control de admisión basado en recursos: El uso de control de admisión explícito (un procedimiento de solicitud-respuesta) proporcionado por InServ/RSVP, ayuda a usar eficientemente los recursos de la red.
- Control de admisión basado en políticas: los elementos de red con RSVP pueden identificar aplicaciones y usuarios usando una solicitud de servicio y aplicando políticas basadas en control de admisión. Las políticas DiffServ típicamente se basan en filtros de paquetes (prefijo de direcciones/número de puertos) y puesto que se usa señalización es más útil soportar control de admisión basado en políticas.
- Condicionamiento del Tráfico: los elementos con Servicios Integrados condicionan el tráfico a una granularidad por flujo. Si el condicionamiento es una precondición para que los paquetes

entren a las regiones DiffServ se proveen garantías cuantitativas extremo a extremo, usando sólo control de tráfico agregado en la región DiffServ.

5.3. MPLS PARA RSVP

Existe una propuesta del IETF de usar un objeto en RSVP, denominado, EXPLICIT_ROUTE, para predeterminar caminos que puedan ser usados por flujos de RSVP. Estos flujos usan caminos virtuales establecidos a través de enrutadores MPLS. Incluso sin el citado objeto, es posible para MPLS asignar etiquetas de acuerdo al campo flowsepc de RSVP. Esto simplifica el soporte RSVP sobre los enrutadores MPLS, los cuales para referenciar las etiquetas no necesitan gestionar el estado RSVP.

5.4. MPLS PARA DIFFSERV

Al ser DiffServ y MPLS similares, asociar el tráfico DiffServ sobre trayectos de MPLS es bastante sencillo. Para soportar el modelo de DiffServ, un operador de red MPLS necesita asignar una serie de recursos para cada clase Diffserv transmitida en cada enrutador MPLS y asignar, a su vez, etiquetas.

6. QOS BASADA EN POLÍTICAS

La calidad de servicio basada en políticas permite al administrador de red asignar anchos de banda y priorizar el tráfico en función de un conjunto de políticas administrativas y patrones de uso. Esta capacidad para controlar los flujos es importante debido al constante aumento del tráfico de las redes. Construida sobre estándares emergentes, como RSVP, IEEE 802.1p y 802.1Q, QoS basado en políticas es una capacidad de alto nivel que añade agrupación, asignación de perfiles de QoS y mapeo en colas asociadas a puertos específicos de enrutadores o conmutadores. Esta capacidad de alto nivel se traduce en un mayor control para el administrador de red, y para esto, especifica cómo las reglas son definidas, construidas, almacenadas, accesadas y aplicadas.

Los beneficios de proveer QoS basada en políticas son los siguientes:

- Define dónde, cuando y cómo la QoS es aplicada al tráfico.
- Proporciona un servidor de políticas para configurar políticas de QoS extremo a extremo.
- Provee protección al desempeño de las aplicaciones de misión crítica.
- Garantiza niveles de QoS específicos, que están de acuerdo a los requerimientos de los usuarios.

Para comprender el funcionamiento de QoS basada es políticas, primero es necesario tener claro el concepto de política. Las políticas no son más que un conjunto de reglas que describen las acciones que ocurrirán cuando una condición específica ocurre (figura 2.8).

Figura 2.8. Definición de Políticas.

Las políticas determinan cómo los recursos de red son asignados a las aplicaciones, y para ello se especifican grupos de tráfico y se definen perfiles de QoS, los cuales indican el nivel de servicio que los grupos recibirán. Básicamente, los perfiles definen un ancho de banda máximo y mínimo, al igual que fijan una prioridad relativa para el tráfico. Como consecuencia, las políticas deben ser verificables y sin ambigüedades de tal forma que sólo una (o un conjunto) de reglas sea apropiada para un conjunto de condiciones específicas.

En las políticas se dan jerarquías o niveles, donde una política puede comprender varias reglas y las reglas comprenden condiciones y acciones; haciendo posible que una política pueda referenciar a otras formando grupos de políticas (figura 2.9). Con esta jerarquía se definen políticas simples o complejas, en las políticas simples las acciones y/o condiciones dependen de su conjunto de reglas y las políticas complejas son aquellas en que las acciones y/o condiciones dependen de otras políticas o grupos de políticas.

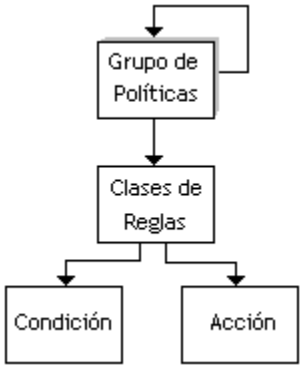


Figura 2.9. Jerarquía de Políticas.

6.1. Estructura de la QoS basada en políticas

La estructura es un modelo de control de políticas escalable, en el cual se identifican dos componentes primarios que desarrollan funciones específicas. La estructura esta compuesta del Punto de Aplicación de Políticas (PEP- Policy Enforcement Point) y un Punto de Decisión de Políticas (PDP- Policy Decision Point), el PEP es quien hace cumplir las políticas y el PDP toma las decisiones de acuerdo a las políticas recibidas desde el almacén de políticas y, quizás, de otras locaciones tales como servidores de autenticación (figura 2.10)

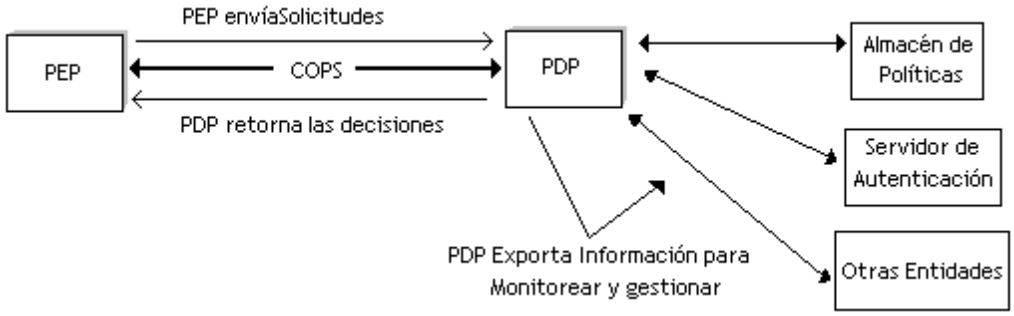


Figura 2.10. Estructura de la QoS basada en Políticas

La separación del punto PEP y PDP es funcionalmente lógica no física, en otras palabras, esta separación no implica tener que construir estos dos puntos en dispositivos físicos diferentes. Actualmente, en la estructura se describe un subcomponente del PEP llamado PDP Local (LPDP – Local PDP) que permite al PEP tomar algunas decisiones. Sin embargo, para evitar violaciones de seguridad, un PEP siempre esta diseñado para formular solicitudes al PDP y tomar decisiones acerca de las políticas. Las solicitudes pueden contener una o más elementos en adición a la información de control de admisión. El PDP retorna la decisión sobre la política y el PEP la aplica apropiadamente.

Es probable que en un dominio de red existan muchos PEPs, un PDPs y un almacén de políticas, sin embargo, en una red puede existir varios PEP y dos o mas PDPs o almacenes de políticas, distribuidas a lo largo de la red, esto proporciona tolerancia a fallas pero podría complicar la administración por parte del proveedor.

Además de las funciones de aplicación y la toma de decisiones definidas y asignadas a los puntos PEP y PDP respectivamente, se describen funciones básicas importantes en el sistema de políticas:

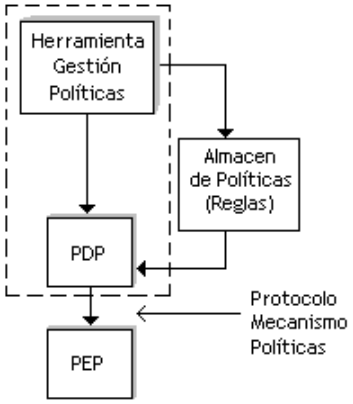
- **Toma de decisiones:** Esta función compara el estado actual de la red con un estado deseable descrito por una política específica de aplicación, y decide cómo hacer que la red llegue a ese estado. El (COPS - Common Open Policy Server) fue diseñado para este propósito, y funcionan en modo cliente/servidor. Cuando la central recibe una petición se pone en contacto con el servidor para su validación.

1. **Aplicación**: Este define un estado de políticas deseado a través de un conjunto de comandos de gestión, que son aplicados a los elementos de red. Los comandos cambian la configuración de los dispositivos.

2. **Medidas:** Examinan activa o pasivamente la red y sus componentes, para verificar si las políticas se satisfacen.

Sobre los conceptos de la estructura de políticas se construye una arquitectura que describe cómo las herramientas de gestión de políticas se relacionan con los otros componentes de la

estructura. La Figura 2.11 ilustra la arquitectura que incluye las herramientas de gestión y las prescripciones de políticas.



la arquitectura de QoS basada en políticas cumple con una serie de funciones:

- **Clasificación**: Para aplicar las políticas al tráfico, primero, los paquetes son inspeccionados y almacenados en grupos de tráficos según la información de la cabecera del paquete IP. La clasificación de los paquetes da flexibilidad para identificar los grupos de tráfico y para implementar políticas basadas en una variedad de criterios.

- **Decisiones de marcado (etiquetado) de tráfico**: Después de clasificar los paquetes, éstos son marcados con códigos que indican el manejo que deberán recibir por parte de los enrutadores. Una vez los paquetes son marcados, se les da los niveles de prioridad definidos en el perfil de QoS.

Finalmente, la estructura y la arquitectura de la QoS basada en políticas son definidas para identificar las funcionalidades requeridas, para determinar cómo las responsabilidades son distribuidas, y los protocolos para garantizar el cumplimiento de las políticas.

7. IMPLEMENTACIÓN DE LA CALIDAD DE SERVICIO

Las especificaciones técnicas por sí mismas no suministran Calidad de Servicio, sólo hacen posible la asignación de valores relativos de prioridades. La QoS sólo se puede alcanzar implementándolas de una forma coordinada en red, en el equipo, en la gestión de red, en las estaciones de trabajo y en las aplicaciones. Por analogía con el enrutamiento, que necesita la definición de un esquema de direccionamiento y el protocolo de cálculo de ruta, la implementación de la QoS requiere el acuerdo con la topología funcional y los métodos de señalización utilizados. La QoS se debe elegir de acuerdo con las limitaciones de la aplicación y con las características físicas de la red.

Por ejemplo, para una conexión a larga distancia y de baja velocidad, el ancho de banda debe ser supervisado atentamente, mientras que en una red troncal de alta velocidad se necesita reducir los retardos. En todos los casos, las aplicaciones a utilizar y sus limitaciones se tienen que identificar o evaluar antes de implementar la QoS. Esto permite identificar las partes de una red que necesitan mejorarse dentro del esquema de una actualización por pasos.

7.1. Implementación en las aplicaciones

Actualmente, la mayoría de las aplicaciones ignoran la Calidad de Servicio. La próxima generación puede utilizar las nuevas API en sistemas operativos. Sin embargo, si todos los usuarios de una red piden tratamiento prioritario, puede ser imposible satisfacer sus peticiones. Por lo tanto, es necesario asegurar que las peticiones están bien fundadas; éste es el papel de las funciones de supervisión.

7.2. Implementación en las estaciones de trabajo de la red

Las estaciones de trabajo pueden utilizar todas las técnicas de QoS. Sin embargo, sólo las nuevas interfaces de red, construidas desde principios de 1999, soportan el 802.1Q/P, así como su nuevo formato de trama. Una petición QoS a nivel de estación de trabajo puede tener dos orígenes: o llega desde la aplicación, o bien está implícita en la naturaleza de la estación de trabajo (servidor, pasarela, etc.).

7.3. implementación en el equipo

Para implementar QoS, el equipo debe suministrar todas o algunas de las siguientes funciones:

- Clasificación.

  Cuando las aplicaciones y las estaciones de trabajo no utilizan tecnologías de QoS, el equipo debe de ser capaz de identificar el tipo de tráfico entrante (aplicación, usuario, destino, etc.). Las diferentes capas en las que está basada la clasificación son las capas 1, 2 y 3 del modelo OSI, estas se pueden usar para identificar exactamente la estación de trabajo de la red; no obstante, esta información no indica qué aplicaciones se están utilizando. Para conseguir esto, es necesario mirar detenidamente las tramas. Determinadas aplicaciones se pueden identificar con certeza en la capa 4 del modelo. En casos más complejos, tales como la voz y la distinción entre varios servidores de web en la misma máquina, es necesario referirse a la capa 7. Los resultados de la clasificación que se encuentran disponibles para otros módulos son: proceso de baja prioridad, garantía de ancho de banda, discriminación de servicio.

- Control del tráfico entrante y saliente.
  Se utiliza un control en la entrada para excluir cualquier tráfico que pudiese sobrepasar los parámetros contratados. En la salida se asegura que el tráfico se ajusta a esos parámetros, los cuales pueden relacionarse con el ancho de banda, la prioridad, etc.

- Gestión de colas.
  Esta función distribuye el tráfico a las colas de acuerdo con los resultados de la clasificación. También maneja el flujo de tráfico desde las colas a los puertos de salida de acuerdo a los principios que aseguran la QoS pedida (WFQ, CBQ, FIFO, etc.). Además, en caso de desbordamiento de una cola, se implementan mecanismos de supresión de trama (WRED, RED, etc.).

- Modificación de los campos asociados a la QoS y ajuste a las tecnologías de QoS.
  Esta función modifica los valores de determinados campos relacionados con la QoS de acuerdo a los resultados de la clasificación. Por ejemplo, en una red que utiliza centrales de nivel 2 en la periferia y centrales de nivel 3 para la red troncal, esta función verifica el tráfico con una codificación de nivel 2 y con una codificación de nivel 3, o más sencillamente, asigna la prioridad deseada a un determinado tipo de tráfico.

- Conservación de la QoS mediante codificación de los datos de las cabeceras de trama.
  Después de la clasificación, esta función pone una marca de QoS en la cabecera de cada trama. En la práctica, la clasificación es muy difícil después de que los datos hayan sido codificados.

- Mantenimiento de la QoS cuando se aplica la Traducción de Dirección de Red.
  Esta función pone una marca de QoS en la cabecera de cada trama después de la clasificación. La información original de enrutamiento no puede ser mayor que la utilizada después de la traducción.

- Autenticación de la petición de reserva .
  Esta función verifica la validez de todas las peticiones con un servidor de seguridad usando, por ejemplo, autenticación remota.

7.4. Implementación en la administración de la red

La administración de la red es indispensable para simplificar el despliegue de la QoS. Esto permite que la configuración del equipo esté centralizada de forma que se reduzcan los errores originados por una configuración manual. También proporciona gestión centralizada de la hora; por ejemplo, a un determinado tipo de tráfico sólo se le puede dar prioridad de 8 de la mañana a 5 de la tarde, de lunes a viernes.

También es necesario incluir una herramienta de administración para manejar globalmente la QoS y facilitar su implementación basada en un servicio de directorio que permite al equipo tomar decisiones autónomas. Estas herramientas ayudan a que la QoS esté coordinada con otros

servicios de la red, tales como la seguridad y el direccionamiento. Así, un usuario podrá ser autentificado antes de obtener una dirección IP del servicio de directorio, lo cual proporcionará a las centrales información sobre el nivel de QoS requerida por este usuario y sus aplicaciones.