UNIVERSIDAD DE LOS ANDES
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA ELÉCTRICA

ESTUDIO DE MECANISMOS DE CALIDAD DE SERVICIO (QoS) PARA LA OPTIMIZACIÓN DE ENLACES DE BAJO ANCHO DE BANDA DE LA UNIVERSIDAD DE LOS ANDES

Br. Luis A. Romero S.

Mérida, Septiembre de 2009

UNIVERSIDAD DE LOS ANDES

FACULTAD DE INGENIERÍA

ESCUELA DE INGENIERÍA ELÉCTRICA

ESTUDIO DE MECANISMOS DE CALIDAD DE SERVICIO (QoS) PARA LA OPTIMIZACIÓN DE ENLACES DE BAJO ANCHO DE BANDA DE LA UNIVERSIDAD DE LOS ANDES

Trabajo de Grado presentado como requisito parcial para optar al título de Ingeniero Electricista

Br. Luis A. Romero S.
Tutor: Prof. Emigdio Malaver.
Asesor: Ing. Javier Contreras.

Mérida, Septiembre de 2009

UNIVERSIDAD DE LOS ANDES

FACULTAD DE INGENIERÍA

ESCUELA DE INGENIERÍA ELÉCTRICA

ESTUDIO DE MECANISMOS DE CALIDAD DE SERVICIO (QoS) PARA LA OPTIMIZACION DE ENLACES DE BAJO ANCHO DE BANDA DE LA UNIVERSIDAD DE LOS ANDES

Br. Luis A. Romero S.

Trabajo de Grado, presentado en cumplimiento parcial de los requisitos exigidos para optar al título de Ingeniero Electricista, aprobado en nombre de la Universidad de Los Andes por el siguiente Jurado.

Prof. Emigdio Malaver
C.I. 9.427.109

Prof. Zulima Barboza
C.I. 3.036.548

Prof. José R. Uzcategui M.
C.I. 13.803.000

DEDICATORIA

A Mi Familia

“El desarrollo del hombre depende fundamentalmente de la invención. Es el producto más importante de su cerebro creativo. Su objetivo final, es el dominio completo de la mente sobre el mundo material y el aprovechamiento de las fuerzas de la naturaleza a favor de las necesidades humanas”

Nikola Tesla

AGRADECIMIENTOS

A Dios todopoderoso, que sin su protección y consentimiento no habría podido alcanzar esta meta.

A la Ilustre Universidad de Los Andes a través de la Escuela de Ingeniería Eléctrica, por abrir sus puertas y permitir desarrollar mis estudios en esta prestigiosa Institución.

A mi Mamá y mi Papá, que sin su enseñanza, ejemplo y apoyo irrestricto, habría sido imposible llegar hasta el final.

A mis Hermanos Yolmer y Heidi, los cuales sirvieron de inspiración para superar cada obstáculo que se me presento durante el transcurso de la carrera.

A mi Abuela Tulia, por su cariño y apoyo durante todo este tiempo y mucho mas.

A mi Tío Homero y Tía Damaris, por haberme brindado tanta ayuda y aliento, sin el cual habría sido difícil culminar esta meta.

A Tía Elsy, que mientras estuvo con nosotros, fue punto de apoyo en el logro de este éxito. Dios te tenga en la gloria Tía.

A todos mis Primos por el cariño y ánimo en todo momento.

A mi Asesor el Ing. Javier Contreras por su importante contribución en este trabajo de grado, sin la cual, no habría podido ser realizado.

A mi Tutor el Profesor Emigdio Malaver por el respaldo en la elaboración de este trabajo.

A todo el personal de Red ULA por el soporte brindado durante el desarrollo experimental del trabajo, especialmente a Niassa, Alejandro, Cherry, José Daniel, Eduardo y Alejandra.

A todos mis compañeros de clase con quienes he compartido alegrías y tristezas durante tanto tiempo.

Y a todos mis amigos que de una u otra forma estuvieron presentes en todo momento.

Gracias a Todos.

Luis A. Romero S. Estudio de Mecanismos de Calidad de Servicio (QoS) para la Optimización de Enlaces de Bajo Ancho de Banda de la Universidad de Los Andes. Universidad de Los Andes. Tutor: Prof. Emigdio Malaver. Asesor: Ing. Javier Contreras. Junio 2009.

RESUMEN

La Universidad de Los Andes cuenta con una red en donde se encuentran integradas aplicaciones de voz, datos y videos, esta utiliza enlaces de red de área amplia *(WAN)* con bajo ancho de banda para su transmisión. A esta red se le conoce hoy en día como red convergente, estos tipos de redes presentan ciertas limitaciones debido al manejo de los diferentes tipos de tráfico que soporta, es por esta razón, que nace la idea de llevar a cabo el presente estudio, con el cual se permita administrar más eficientemente los diferentes servicios que allí convergen.

La respuesta a estas dificultades se encuentra en los Mecanismos de Calidad de Servicio (*QoS*), los cuales permite brindar a cada aplicación, de los recursos de red más apropiados para su buen funcionamiento. El objetivo fundamental de este trabajo está enmarcado en el estudio de estos mecanismos de *QoS* soportados por la red de la Universidad de Los Andes y la generación de un plan de implantación progresivo que permita el mejoramiento en la transmisión de diferentes tráficos de datos.

Se utilizara como metodología para este estudio, el uso de un laboratorio de redes en donde se simulará el tráfico de red soportado por la Universidad de Los Andes, recabando posteriormente la información necesaria para poder realizar la comparación, en eficiencia, de cada mecanismo simulado.

Descriptores: *QoS*, Flujo de Datos, Paquetes de Datos, *Jitter*, *Drop*, Latencia, Ancho de Banda, *IntServ*, *DiffServ*, Congestión, *IP*, *TCP*, Red, Tráfico, Red Convergente**.**

ÍNDICE GENERAL

LISTA DE FIGURAS

FIGURA **PP**

LISTA DE TABLAS

TABLA **PP**

LISTA DE GRÁFICOS

GRÁFICO **PP**

INTRODUCCIÓN

El crecimiento y evolución constante de aplicaciones y servicios, genera la necesidad de soportar diferentes tipos de tráficos sobre redes convergentes de voz, datos y video. Cada aplicación o servicio tiene diferentes requerimientos de red, lo que hace necesario la implementación de mecanismos que permitan administrarlos de forma individual. En una red de servicios convergentes el tráfico de mayor importancia para la organización, debe ser protegido del resto del tráfico.

Generalmente este tráfico es identificado y tratado con mayor prioridad evitando retrasos y posterior pérdida de información. En este sentido, es de vital importancia la implementación y administración de mecanismos de calidad de servicio (*Quality of Service "QoS"*) que garanticen el manejo eficaz de la creciente demanda de tráfico en una red.

Actualmente la Universidad de Los Andes cuenta con una red de servicios integrados sobre el Protocolo Internet (*Internet Protocol "IP"*), que conecta los Núcleos de Mérida, Táchira, Trujillo y Barinas, usando enlaces de tecnologías para redes de área amplia (*Wide Area Network "WAN"*) de dimensiones muy limitadas. Un análisis de los mecanismos de *QoS* que se pueden implementar en los equipos de comunicación y enlaces existentes en cada núcleo, así como un plan de implantación a corto plazo, permitirá una administración eficiente de trafico crítico en la red, optimizando el rendimiento de los enlaces *WAN* e incrementando la calidad de las aplicaciones y servicios percibidos por los usuarios.

El presente trabajo de grado se enmarca en el estudio de mecanismos de calidad de servicio que permitan un uso adecuado de la red con bajo ancho de banda, particularmente, en las redes convergentes. Las limitaciones que ofrecen estas redes, imponen grandes restricciones en la transmisión de paquetes de información a través de voz, datos y video. Estas restricciones se presentan en las redes convergentes por medio del elevado recurso del cual hacen uso (ancho de banda) y de cómo se manejan los parámetros que afectan los

niveles de calidad de servicio, representados por los retardos *(delays)* conocidos también como latencia *(latency)* , retardo variable *(jitter)* , la congestión y las pérdidas de paquetes *(drops)* entre otros.

Estos mecanismos permitirán gestionar la red en forma óptima, consiguiendo dar prioridad al enrutamiento de paquetes de datos enviados por ejemplo, por aplicaciones de voz sobre Protocolo de Internet *(VoIP)* o video conferencia; sobre aplicaciones de sólo transferencia de datos como *e-mail* o de acceso a una página *web.*

Cuando se habla de redes convergentes se hace referencia a la fusión de los servicios de datos, voz y video dentro de una sola plataforma, que basa su transmisión en el Protocolo de Internet. Los enlaces de bajo ancho de banda presentes en las redes convergentes, introducen problemas técnicos en las mismas, ya que la convergencia de diferentes servicios, traen como consecuencia, distintas características y requerimientos de red; por lo tanto, los datos que se transmiten en paquetes (unidad fundamental de transporte de información en una red), requieren de tanto ancho de banda como puedan tomar en determinado momento, la disponibilidad de estos datos para determinado usuario, dependerá en su mayor parte, del número de individuos que accedan a la red en ese instante, alimentando esto el consumo de ancho de banda en la red. Por ejemplo, la recepción de un *e-mail* (archivo de datos) con unos pocos segundos de retraso, no implica mayor importancia; a diferencia de las aplicaciones de voz, en donde se requiere un ancho de banda constante, con un porcentaje bajo de retardo en la transmisión.

Como es conocido, el tráfico de voz posee normalmente pequeños paquetes de datos, los cuales no pueden permitirse *delays* ni *jitters* en su recorrido por la red, situación que sí no se logra evitar, traerá como consecuencia, que la transmisión de voz se fraccione, resultando en una comunicación incomprensible; esta situación será extrapolable para el manejo del trafico de video, como por ejemplo, una video conferencia.

En lo que respecta a los paquetes que contienen archivos de datos, estos son de gran tamaño y pueden aceptar *delays* y *drops*; es factible retransmitir a su destino parte de un archivo de datos que haya presentado un *drop*, situación que no se puede permitir, en las transmisiones de voz y video. Como se puede notar, el tráfico de voz y video se hace sensible a los *delays* y *drops* en el tiempo de transmisión; por lo tanto, es de vital importancia, reducir a su mínima expresión este tipo de falla en el sistema, optimizando el uso de las redes con bajo ancho de banda, en lo cual, la calidad de servicio, juega un papel importante.

La calidad de servicio es la capacidad que tiene la red de proveer un mejor servicio para determinado tráfico de red, garantizando así, la transmisión de cierta cantidad de datos en un tiempo determinado. La meta de *QoS* es proveer un eficaz servicio de red, suministrando para esto; ancho de banda dedicado, controlando los *jitter*, manejando el tiempo transcurrido entre la transmisión y recepción de datos *(latency),* y mejorando las pérdidas en las características de la red.

Los objetivos generales del trabajo se enmarcan en el análisis del funcionamiento de estos mecanismos *QoS* y en la elaboración de un plan de implementación, que mejore el desempeño de la transmisión de voz, datos y video sobre enlaces de bajo ancho de banda, entre los núcleos extraurbanos de la Universidad de Los Andes.

Usando herramientas de monitorización de redes *(SmokePing y MRTG)* se recabará la información del tráfico sobre los enlaces *WAN*, parámetros de operación y rendimiento, para su posterior caracterización. En un ambiente de laboratorio se simulará el funcionamiento de los mecanismos de calidad de servicios soportados por los equipos de comunicación existentes en los núcleos *(Cisco Systems)* y contando con la información de caracterización de tráfico, parámetros de operación y rendimiento; se compara la eficiencia de cada mecanismo simulado.

El esquema de desarrollo para el trabajo de grado se estructuró en tres capítulos, dentro de los cuales se puede encontrar para el primer capítulo, referencia al marco teórico, en el cual se habla sobre el Protocolo *Internet*, las Redes Convergentes, la Gestión de Tráfico y sobre los Mecanismos de Calidad de Servicio. En el segundo capítulo se hace referencia a la parte experimental de este estudio, de cómo son desarrolladas las pruebas y de las herramientas utilizadas para las mismas. Finalmente el capítulo tres es enmarcado hacia los resultados obtenidos y de cuál es su impacto sobre el flujo de trafico *IP*.

CAPÍTULO 1
MARCO TEÓRICO

En el presente capítulo se hace mención acerca del Protocolo Internet, del formato del encabezado *IP*, se habla sobre el *RFC* y *DRAFT*, se citan también las Redes Convergentes, lo referente a la gestión de tráfico y de que herramientas son utilizadas para el monitoreo del mismo. Son tratados igualmente los parámetros que influyen en el comportamiento de la red de datos y todo lo referente a los Mecanismos de Calidad de Servicio.

1.1 PROTOCOLO INTERNET (*IP*)

1.1.1 Historia

Para poder hacer referencia al protocolo *Internet*, es necesario recordar cuál fue el punto de partida en la invención del mismo, y para esto, hay que remontarse a su origen. La *Internet* nace en los años sesenta en los Estados Unidos con la finalidad de que en caso de estallar una guerra nuclear se lograra preservar los sistemas de comunicación entre los diferentes organismos civiles y militares; haciéndolos independientes unos de otros, permitiendo esto, que en el caso de que fuese destruida una red de comunicación, las otras siguieran funcionando, facilitando el continuo flujo de información y haciéndolas así, independientes de un control central.

De esta forma en el año 1969 la *ARPA* (*Advanced Research Projects Agency*) una agencia dependiente del Departamento de Defensa de los Estados Unidos junto a *RAND Corporation*, *MIT* (*Massachusetts Institute of Technology)* y *UCLA* (*University of California)* deciden desarrollar esta tecnología de redes en los Estados Unidos. Para finales de ese año, ya se tenían 4 nodos en la red, siendo instalado el primero en la *UCLA*, para 1972 la red había aumentado a 37 nodos; a esta red se le denomino *ARPANET*. La transferencia de datos se debía basar en un mecanismo que permitiera manejar la destrucción parcial de la red; para poder hacer esto, se tomo la decisión de fraccionar los mensajes en pequeñas porciones de información llamadas paquetes, los cuales tendrían la dirección de destino pero, sin indicarles cual ruta debían tomar para llegar al mismo. Estos paquetes se encargarían de escoger la mejor vía disponible para llegar, y este punto de llegada, tomaría los paquetes individuales y los reensamblaría, para reconstruir el mensaje original sin importar la ruta tomada.

Para esto la red utilizaba un protocolo de intercambio de paquetes llamado *NCP* (*Network Control Protocol*) que permite las comunicaciones entre máquinas *(Host)* semejantes, en la misma red física; esto permitió que los usuarios de la red pudieran desarrollar diversas aplicaciones. Para el año 1973 un grupo de investigadores entre los que se destacan Vinton Cerf (*SRI "Stanford Research Institute"*) y Robert Kahn (*ARPA*) desarrollaron una nueva versión del protocolo, con mejores características que la anterior y la llamaron Protocolo de Control de Transmisión/Protocolo de Internet (*TCP/IP*), una de las ventaja que presentaba este protocolo, era que permitía la comunicación entre máquinas de diferente naturaleza.

TCP (*Transmission Control Protocol*) fracciona los mensajes en paquetes en el *Host* de origen y los reensambla en el *Host* destino, para obtener el mensaje original, mientras que *IP* (*Internet Protocol*) se encarga de encontrar la ruta destino. El uso extendido del *TCP/IP* en las redes vinculadas a *ARPANET* hizo que a finales de los años setenta y en los años ochenta con el auge del uso de las computadoras, se permitiera el fácil acceso a la creciente red. Este crecimiento, más el dominio público del *TCP/IP*, hicieron que la red se expandiera incontrolablemente, desembocando en lo que hoy conocemos como la *International Networking* (*Internet*).

El protocolo de *Internet* es “un protocolo no orientado a conexión usado tanto por el origen como por el destino para la comunicación de datos a través de una red de paquetes conmutados” (Wikipedia, 2009). Una red basada en este tipo de protocolo (*IP*) envía los datos en bloques llamados paquetes o datagramas, sin necesidad de realizar alguna configuración previa al envío de estos, sin importar si los equipos en los cuales se va hacer la transmisión, habían establecido comunicación anteriormente. Este protocolo provee un servicio de datagrama (fragmento de paquete) conocido como *best effort* (mejor esfuerzo; no ofrece *QoS*); el protocolo no presenta ningún mecanismo que determine si un paquete alcanza o no su destino, solo permite verificar la seguridad mediante *checksums* (suma de comprobación) de sus cabeceras y no de los datos enviados.

Las cabeceras *IP* contienen las direcciones de las máquinas (direcciones *IP*) de origen y destino, las mismas que serán utilizadas por los conmutadores de paquetes *(switches)* y los enrutadores *(routers)* para escoger la ruta en la red por la cual se reenviaran los paquetes. Este protocolo *IP* al no dar garantía en lo que respecta a la recepción de paquetes, permitirá que los mismos puedan llegar dañados, en otro orden con respecto a otros paquetes, duplicados o que no lleguen; para aumentar la seguridad y fiabilidad de *IP*, es que se utiliza en asociación el protocolo *TCP*, permitiendo una transmisión de paquetes más eficiente.

1.1.2 Formato del Encabezado *IP*

Como ya se comento anteriormente, todos los datos en la red *IP* viajan en unidades de información denominadas paquetes, al formato utilizado por los paquetes se le denomina datagrama, este consta de un encabezamiento (*Header*) y de los datos. El encabezamiento está formado por palabras de 32 bits; la estructura del paquete es mostrada en la figura 1.1.

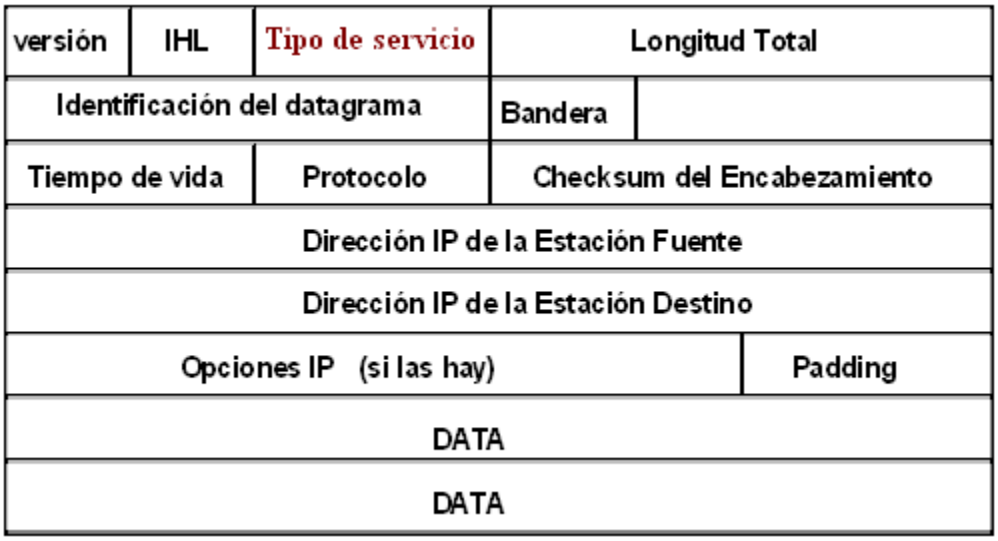

Figura 1.1 Encabezado del Paquete IP

Dentro del paquete *IP* se distinguen los siguientes campos:

Versión: se refiere a la versión del protocolo *IP* utilizada.

IHL (IP Header Length): es la longitud del encabezamiento *IP*, en palabras de 32 bits.

Tipo de Servicio*:* en este campo los protocolos superiores pueden dar mayor o menor prioridad al paquete. Permite establecer que "Calidad de Servicio" requiere el paquete.

Longitud Total: es el largo total del paquete *IP* en bytes, incluyendo encabezamiento y datos.

Identificación del Datagrama: este campo permite al destino, determinar a qué paquete pertenece el fragmento que ha llegado a él. Sirve para reensamblarlos, en el caso que *IP* los haya fragmentado.

Bandera: indican si el paquete de protocolo de capa superior fue fragmentado y si el paquete *IP* transporta el último fragmento.

Offset de Fragmentación: es el número del fragmento del paquete original, me indica a que parte del paquete total pertenece el fragmento que se está recibiendo.

Tiempo de Vida: contiene un contador que se va decrementando hasta llegar a cero. Esto permite que el paquete sea descartado, y así se evita que hayan circulando por la red paquetes extraviados o en bucle indefinidamente.

Protocolo: este campo hace referencia de cual protocolo superior (TCP *o User Datagram Protocol "UDP"*) va a recibir los datos.

Checksum del Encabezamiento: se utiliza para detectar que los datos del encabezado del paquete *IP* no hayan sido modificados por errores en la transmisión.

Dirección IP de la Estación Fuente: corresponde a la dirección *IP* de la estación que envió el paquete.

Dirección IP de la Estación Destino: corresponde a la dirección *IP* de la estación que recibe el paquete.

Opciones IP: permite agregarle opciones al protocolo; como puede ser, seguridad.

Padding: es usado con la finalidad de que el tamaño en bits del encabezamiento sea múltiplo de 32.

Data: contiene los datos de las capas superiores.

1.1.3 Documentación Inherente (RFC/DRAFT)

Los *Request For Comment* (*RFC*) son una serie de documentos en donde se detalla prácticamente todo lo relacionado con la tecnología de la que se sirve Internet, como protocolos, recomendaciones, comunicaciones, etc. El *RFC* referido al protocolo *IP* es conocido como *RFC* 791, el cual por su extensión no se incluye en el presente trabajo, pero está disponible en detalle en la página web: *http://www.rfc-editor.org/* (versión en ingles) y *http://www.rfc-es.org/* (versión en español).

En lo que respecta a los *Internet Draft* estos son una serie de documentos de trabajo de la *Internet Engineering Task Force* (*IETF*), son una especie de borrador de *RFCs*, o sea,

documentos referidos a Internet no sometidos a revisión, los cuales pueden ser reemplazados o modificados por versiones posteriores, estos tienen una vida máxima de seis meses; si después de este vida útil no han sido actualizados, serán eliminados (*http://www.ietf.org/ID.html*.).

1.2 REDES CONVERGENTES

1.2.1 Historia

La convergencia en el acceso que aporta *IP*, permite ofrecer una interfaz única para gestionar todas las comunicaciones; la convergencia de datos, voz y video ha estado evolucionando y ganando impulso en los últimos años. En sus comienzos, si se hace memoria de cómo era la comunicación en el año 1960 y se compara en lo que es hoy en día, se puede notar que se han producido cambios asombrosos, se ha pasado de comunicaciones unidireccionales, a una experiencia en tiempo real y en línea. Si se hace un breve recuento de esta evolución; se tiene que para los años sesenta el *télex* era el medio de comunicación escrita en línea, a finales de esta década se desarrollo la *ARPANET* (antecesor de *internet*), en los años setenta se empezaron a construir las primera computadoras personales, en los años ochenta se estandarizó el *fax*, nació la telefonía celular y la fibra óptica, y aparecieron las primeras redes en *Narrowband* (banda angosta), en la década de los noventa, se inicio la miniaturización de dispositivos, arranco el correo electrónico y el *World Wide Web*, cobraron vigencia nuevos estándares (*Frame Relay, ATM* e *IP*) y surgieron las primeras redes privadas virtuales.

Si se hace referencia a la década actual, se puede notar que se ha extendido el uso de *Internet* y las aplicaciones para compartir y administrar información. La convergencia está haciendo posible que a través del protocolo *IP*, las diferentes redes, aplicaciones y procesos, interactúen para una experiencia única de usuario, lo que ha traído como consecuencia, la utilización de información de múltiples tecnologías, de forma rápida y

flexible. Hoy las redes convergentes y las aplicaciones basadas en *Internet*, están cambiando todo con respecto a la administración de red. Los administradores de red necesitan gestionar y monitorear una amplia variedad de elementos de red, comprender complejos eventos en la misma y responder rápidamente a estos. Efectivamente la gestión de esta tecnología integrada de red (datos, voz y video) es crucial en las operaciones de la misma, y su buen éxito, está en función de que la información se comparta de forma segura, correcta y a tiempo.

1.2.2 Particularidades

El término convergencia es generalmente usado en referencia a la integración de telefonía con servicios de datos y aplicaciones como video dentro de una red, figura 1.2. La convergencia de servicios de voz, datos y videos se está acercando de muchas maneras; estas tecnologías empleaban recursos separados en redes especializadas en el pasado, pero ahora pueden compartir recursos e interactuar entre sí, permitiendo ampliar de esta forma las características y beneficios de estos servicios.

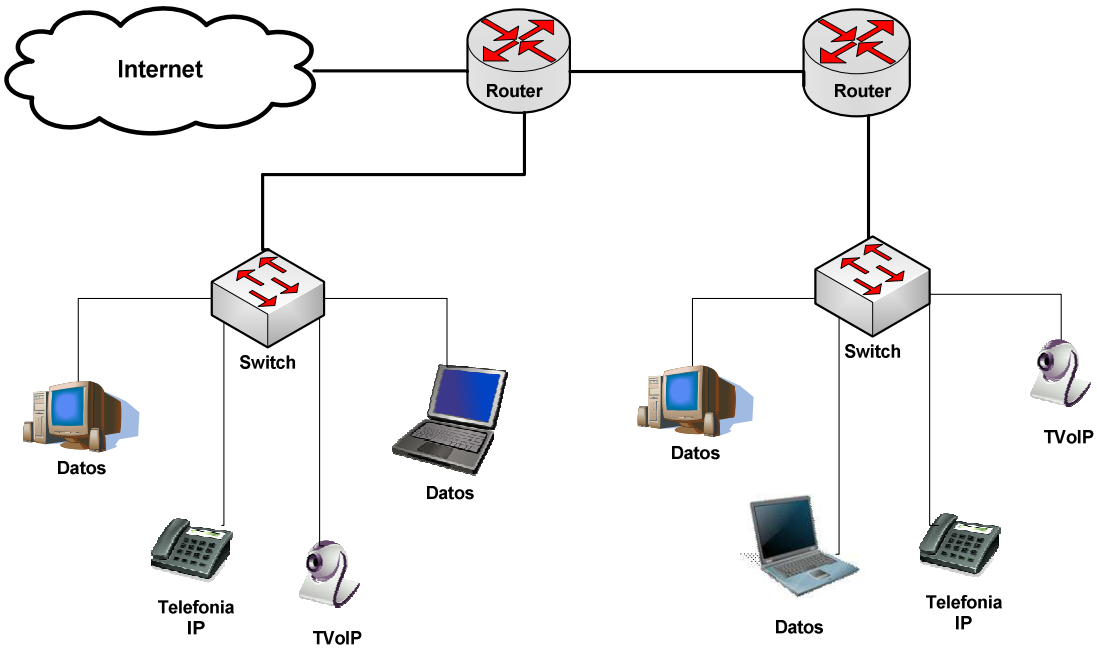

Figura 1.2 Esquema de una Red Convergente

Esta convergencia permite a los usuarios superar todas las barreras de distancia, tiempo y medios físicos, facilitando a las personas comunicarse entre sí, en cualquier lugar, en cualquier momento y mediante múltiples tipos de dispositivos; utilizando para esto, una infraestructura *IP* estándar. Con este mecanismo de fusión de información, se puede mejorar y acelerar significativamente la forma de trabajar, al permitir una más rápida toma de decisiones, transferencia y acceso a conocimientos a través de toda la red, en tiempo real.

La red convergente es una tecnología que está permitiendo unificar bajo un mismo criterio, tres redes bien diferenciadas. El origen de esta red se puede remontar, a la búsqueda de maximizar el uso del ancho de banda disponible. La implementación de estas redes implica la obtención de beneficios tales como:

- Se invierte en una única infraestructura física, en lo que antes debía hacerse para dos o tres redes.
- Esta permite la transmisión de voz, video conferencias y datos en tiempo real.
- Combina la flexibilidad de las redes de datos y de telefonía basadas en *IP*.
- Permite aplicar todas las herramientas de monitoreo, administración y seguridad existentes, para redes de datos.
- Presenta un despliegue más rápido y sencillo.
- Es escalable
- Facilidad para integrar nuevos servicios y tecnologías dentro de la misma plataforma de comunicaciones.
- Estas redes permiten el ahorro en ancho de banda, ya que al estar basadas en *IP*, permiten la compresión de voz y datos.
- Se reducen los costos en la operación y administración de las redes.

1.3 GESTIÓN DE TRÁFICO

1.3.1 Justificación

Se justifica la implementación de mecanismos de gestión de tráfico por varias razones; inicialmente se puede señalar, que con el crecimiento del tráfico que se presenta en las redes hoy día, resulta difícil solventar la congestión (esta se produce cuando los paquetes llegan a un puerto más rápido de lo que pueden ser transmitidos) incrementando solamente el ancho de banda; la inclusión de servicios de datos, voz y video en una misma plataforma, estiman requerimientos más estrictos en la gestión de tráfico de los que se podían tener anteriormente, en las redes con únicamente tráfico de datos. El estado actual de la transmisión en redes de datos, su carácter heterogéneo y altamente demandante de recursos, crea la necesidad de incluir mecanismos que permitan alcanzar altos niveles de utilización de la red.

La gestión de tráfico está en conexión directa con la utilización eficiente de recursos en situaciones de alta demanda. Algunos de los criterios utilizados para determinar el destino de un paquete en condición de congestión, se enmarca en la aplicación de reservas de canal o de políticas de calidad de servicio, que habrán de establecerse considerando las particularidades del tipo de tráfico a cursar (con alta sensibilidad al retardo o a la pérdida de paquetes).

En las redes en las que no existan herramientas de gestión de tráfico, el acceso a aplicaciones críticas, puede ser disminuido o inclusive inhabilitado por aplicaciones no críticas; como por ejemplo, usuarios descargando *(download)* o subiendo *(upload)* grandes archivos vía *http (hyper text transfer protocol)* o *ftp (file transfer protocol)* u observando aplicaciones multimedia *(Streaming)* vía *Internet*, esto hace que se ocupe el ancho de banda disponible, causando congestión en las redes y provocando así, el colapso de las aplicaciones criticas. La idea fundamental de la gestión de tráfico, se basa en poder

proporcionar una justa o equitativa distribución del ancho de banda, a las aplicaciones que se estén utilizando en la red.

1.3.2 Herramientas de Gestión de Tráfico

Las herramientas de gestión o monitoreo de tráfico son sistemas que permiten realizar medición y análisis de la red, basando su trabajo en el monitoreo de parámetros tales como, disponibilidad de ancho de banda, porcentaje de ocupación, retardo, pérdida de paquetes, *RTT (Round-Trip delay Time)*, etc; la mayoría de estos registros son presentados al usuario en forma gráfica. Estos son desarrollados para registrar mediante una interfaz visual, mediciones en determinados intervalos de tiempo o llevados a cabo constantemente. Para el estudio de los mecanismos *QoS* planteados en este trabajo, se hará uso de las herramientas *MRTG* y *SmokePing*.

MRTG (Multi Router Traffic Grapher). Es una aplicación de monitoreo de tráfico de carga en la red (ancho de banda), fue escrito en 1995, utiliza como lenguaje de programación *Perl* y C, trabaja bajo sistema operativo *UNIX* o *NT*. Este utiliza como interfaz visual una página *Web* con gráficos (figura 1.3), en el cual se detalla el tráfico entrante y saliente de algún dispositivo de red. A través de esta aplicación se puede consultar conmutadores de red, servidores, puntos de acceso, etc.

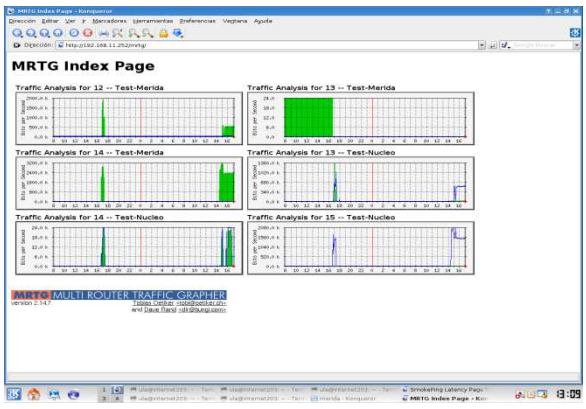

Figura 1.3 Interfaz Gráfica de la Aplicación *MRTG*.

Smokeping. Es una aplicación de monitoreo basada en *RRDTool*, está escrita en *Perl*; esta envía paquetes de prueba a la red y mide la cantidad de tiempo que se toman en viajar de un punto a otro y regresar, los datos obtenidos (*Jitter*, pérdidas de paquetes, *RTT*) son mostrados en gráficos, figura 1.4. Este puede enviar alertas cuando se presentan ciertas condiciones en la red; como por ejemplo, excesiva pérdida de paquetes en determinado enlace por un largo periodo de tiempo. Trabajando junto con *MRTG,* se puede observar, que efecto tiene la congestión de la red en la latencia y en la pérdida de paquetes.

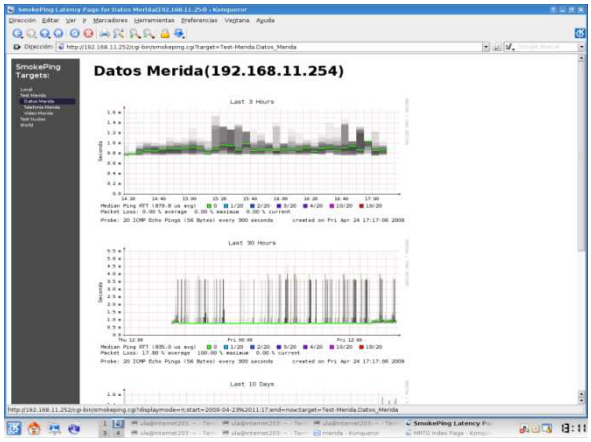

Figura 1.4 Interfaz Gráfica de la Aplicación *SmokePing*.

1.4 NECESIDAD DE LOS MECANISMOS DE CALIDAD DE SERVICIO (QoS).

La transmisión de voz, datos y video a través de la red, se ve afectada por parámetros como ancho de banda, retardo y pérdida de paquetes, los cuales influyen en el comportamiento del tráfico en los enlaces con bajo ancho de banda. Es por eso que se hace necesaria la implementación de alguna herramienta que permita hacer uso eficiente de la demanda de tráfico en la red, siendo esto posible, por medio de mecanismos de calidad de servicio.

1.4.1 Disponibilidad de Ancho de Banda

En la actualidad las aplicaciones de usuarios continúan dirigiendo el crecimiento y evolución de la red; por lo tanto, la demanda para soportar diferentes tipos de tráfico también se está incrementando, lo que repercute igualmente en el consumo de recursos como ancho de banda, figura 1.5. Los diversos tipos de aplicaciones con diferentes requerimientos de red, implican la necesidad de crear políticas administrativas, encargadas, de cómo las aplicaciones individuales son manejadas en la red. Se puede observar que el crecimiento del volumen de tráfico en las redes de conexión, hace difícil solventar la congestión con sólo incrementar el ancho de banda, lo cual implicaría un aumento en los costos por su utilización.

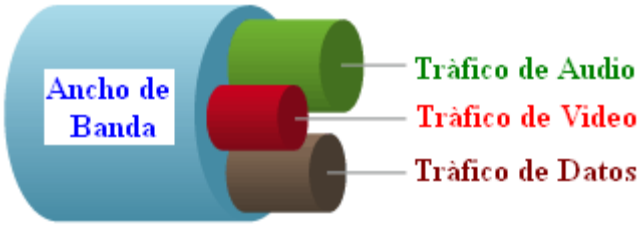

Figura 1.5 Representación de la Demanda de Ancho de Banda en la Red

La inclusión de nuevas aplicaciones sensibles al retardo, generan cambios en la distribución del tráfico y en la incidencia de fallas en los enlaces de red, que pueden traer como consecuencia, patrones de congestión impredecibles. Por otro parte, el fenómeno de la convergencia en las redes, ha concebido un cambio en el modelo de trabajo para suministrar además de transporte de datos, servicios de valor agregado. Las redes convergentes deben ofrecer políticas estrictas de calidad de servicio para poder proveer un soporte adecuado al usuario.

Los mecanismos de calidad de servicio tienen como fundamento, gestionar una variedad de parámetros y de funciones de tráfico en la red, que permitan garantizar un

ancho de banda adecuado y que facilite el desempeño de las aplicaciones en situación de congestión.

1.4.2 Definición de Retardo de Extremo a Extremo

El retardo extremo a extremo conocido como latencia o *delay*, es el tiempo tomado por un paquete en ser transmitido de un extremo de la red al otro, figura 1.6. También es definido como la suma de todos los retardos presentes en una red. Este es producido por la demora en la propagación y transmisión de los paquetes que viajan por la red; junto con el ancho de banda, define las características de desempeño de un enlace o canal específico (capacidad y velocidad en una red). El retardo extremo a extremo es medido estrictamente en términos de tiempo; por ejemplo, una red transcontinental *(WAN)* podría tener una latencia de 24 milisegundos; es decir, a un mensaje le tomaría 24 ms viajar de un extremo del continente a otro.

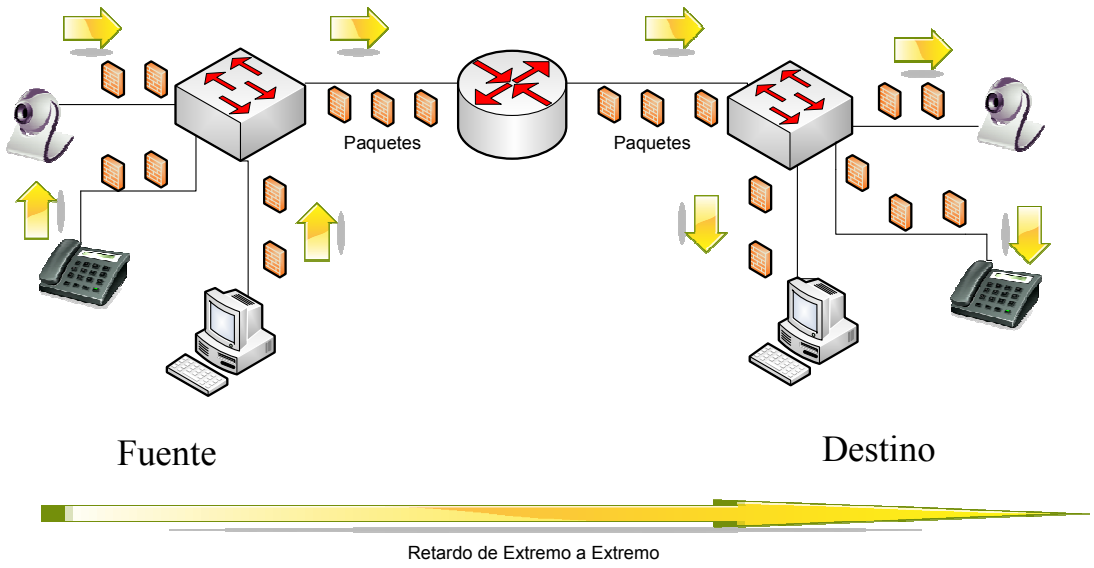

Figura 1.6 Representación del Retardo Extremo a Extremo

Se tienen cuatro tipos de retardo extremo a extremo, ellos son: retardo de procesamiento, retardo de colas, retardo de transmisión, retardo de propagación, figura 1.7.

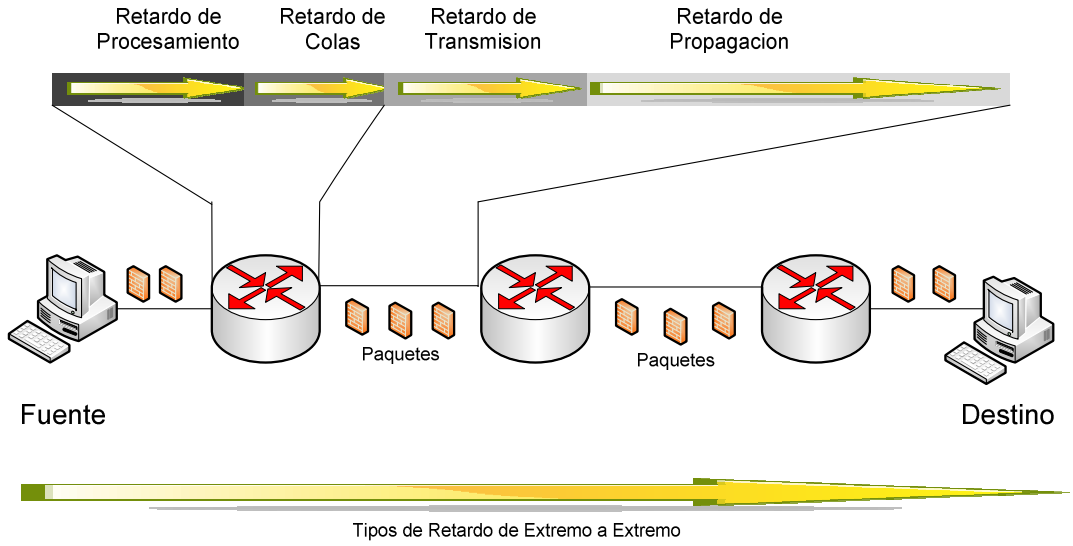

Figura 1.7 Clasificación del Retardo Extremo a Extremo

Retardo de Procesamiento. Es el tiempo que le toma a un *router* recibir un paquete de la interfaz de entrada, examinarlo y ponerlo en la cola de salida. El retardo de procesamiento depende de varios factores; tales como, velocidad y porcentaje de utilización del *CPU*, modo de conmutación *IP*, arquitectura del *router* y características de configuración en la interfaz de entrada y salida.

Retardo de Colas. Es el tiempo que el paquete espera en la cola de salida del *router*. Este retardo depende del número y tamaño de los paquetes en cola, del ancho de banda de la interfaz y de los mecanismos de cola.

Retardo de Transmisión. Es el tiempo que toma colocar una trama (*frame)* en el medio físico de transmisión, para ser enviada. Por ejemplo, para transmitir 1024 bits utilizando *Fast Ethernet* (100 Mbps), se necesitan 10,24 µs.

Retardo de Propagación. Es el tiempo que se lleva transmitir un paquete; usualmente esto depende de la distancia del enlace físico.

1.4.3 Efectos de Retardo de Extremo a Extremo

Los retardos tienen un impacto significativo en la entrega de datos en la red y en el rendimiento de las aplicaciones, dependiendo del tipo de aplicación y de cómo funcionan; estos pueden influir negativamente en el uso eficiente del ancho de banda. Cuando hay un aumento en la latencia, los paquetes tienen que esperar más tiempo en cola, lo que hace que el buffer se sature y estos deban ser descartados, lo cual aumenta el tiempo de transmisión e influye drásticamente en aplicaciones sensibles al retardo, como por ejemplo, voz y video.

Es importante mencionar también, el efecto que causa la variación del retardo en el tiempo de arribo de los paquetes a su destino, estando estos presentes en el mismo caudal de datos; a esta variación se le conoce como *jitter,* figura 1.8. Debido a la influencia que esta tiene en el tiempo de transmisión de la información, es importante tomarla en cuenta. Entre las principales causas del *jitter* tenemos la congestión en la red, también el hecho de que algunos paquetes fluyen en la red siguiendo caminos físicos diferente, otra situación significativa, se da cuando hay un aumento en el consumo del ancho de banda, lo cual genera un crecimiento exponencial de este *(jitter);* marcando esto, influencia en las aplicaciones sensibles al retardo.

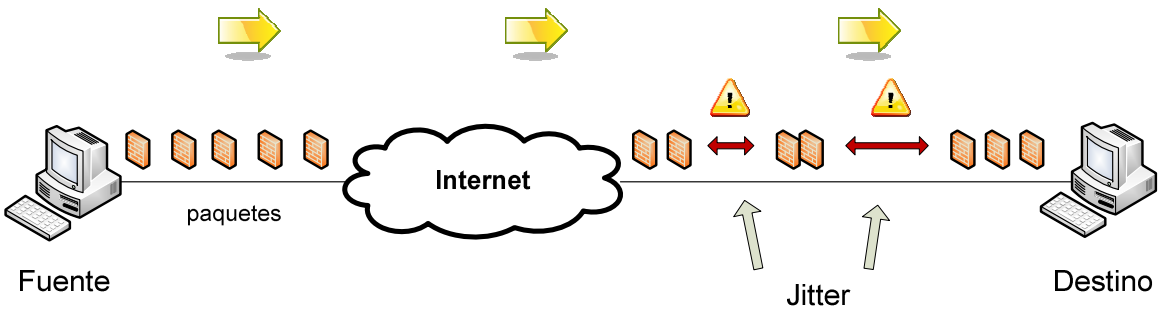

Figura 1.8 ***Jitter***

1.4.4 Pérdidas de Paquetes IP en Flujo de Tráfico en Redes Convergentes

Se define a las pérdidas de paquetes como la imposibilidad de la red *IP* de entregar paquetes de datos a su destino, debido a los errores en la transmisión o a la sobrecarga de los *routers* en la red; cuando un paquete llega a una cola y esta se encuentra llena, el paquete es descartado. Además de lo mencionado anteriormente, se pueden descartar paquetes debido a una rotura en un enlace físico, evitando la transmisión del mismo o un paquete corrupto, producido por ruido detectado por un sistema de checksum, figura 1.9.

En las redes convergentes que operan en base de entrega al mejor esfuerzo *(best effort)*, es probable que se produzcan situaciones de congestión en alguna interfaz, produciendo a su vez pérdidas de paquetes; cuando esto sucede, la mayoría de las aplicaciones que usan *TCP* experimentan desaceleración debido a que *TCP* ajusta los recursos en la red, trayendo dificultades a las aplicaciones de desempeño critico. Es de suma importancia que no ocurran estas situaciones de pérdida de información, de que exista un adecuado ancho de banda disponible y que los retrasos en los envíos de los paquetes de datos, sean mínimos; es por ello que surge la necesidad de aplicar calidad de servicio a nivel de transporte de datos, con el fin de otorgar preferencia a datos sensibles.

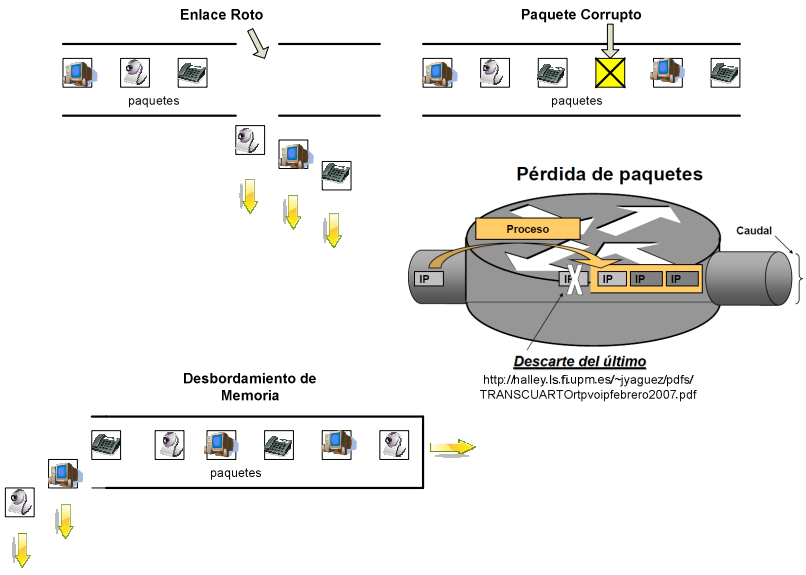

Figura 1.9 Pérdidas de Paquetes

1.5 MECANISMOS DE CALIDAD DE SERVICIO *(QoS)*

1.5.1 Definición de *QoS*

En sus comienzos la *Internet* ofrecía mayoritariamente aplicaciones con tráfico de red basados en servicios web, correo electrónico, acceso remoto y transmisión de ficheros; los cuales no presentaban grandes requerimientos en lo que respecta a ancho de banda, perdidas de paquetes, *jitters* o latencia. De este modo el tráfico generado por estas aplicaciones, era tratado de igual forma por medio de una sola clase de servicio, llamada *Best Effort*, aquí cada paquete de información compite por el ancho de banda disponible para poder alcanzar su destino.

Como ya es conocido, el auge de *Internet* y su éxito comercial han hecho que el número de aplicaciones que envían datos a la red, haya tenido un crecimiento extraordinario; anexando a esto, la popularización de las redes convergentes. De esta manera es que surge la necesidad de aplicar mecanismos que permitan hacer un uso optimo de la red con un determinado nivel de exigencia, de allí es que nace lo que se conoce con el nombre de "Calidad de Servicio *(QoS)*"

Cisco System define la calidad de servicio como "la habilidad de la red para proveer un mejor servicio a determinados usuarios o aplicaciones en detrimento de otros usuarios o aplicaciones" (Cisco System, 2004). La meta principal de la calidad de servicio es proveer un mejor y predecible servicio de red, ejecutando controles sobre parámetros como pérdidas de paquetes, retardos, variación de retardos y haciendo un manejo eficiente del ancho de banda.

1.5.2 Modelo para *QoS*

Existen tres modelos diferentes para la implementación de Calidad de Servicio en la red; el primero es el llamado *Best Effort* (mejor esfuerzo) el cual no ofrece garantía en la entrega de paquetes de datos, es el que se aplica por defecto en *Internet*. El segundo es conocido como *Integrated Service "IntServ"* (Servicio Integrado), este modelo fue introducido como complemento en la transferencia de archivos por *Best Effort*; por medio de él, se ajusta el ancho de banda y se dan garantías en el retardo para las aplicaciones que la requieran. Este modelo provee a las aplicaciones de un nivel garantizado de servicio, negociando parámetros de red.

Por último, el tercer modelo de Calidad de Servicio aplicado a la red, es el definido como *Differentiated Service "Diffserv"* (Servicios diferenciados), el cual surgió para poder proveer gran escalabilidad y flexibilidad en la implementación de *QoS* en la red. Algo que diferencia a este mecanismo del modelo *IntServ* es que la red puede reconocer los paquetes de datos y provee de los servicios apropiados a los mismos. A continuación se hace una descripción más detallada de estos modelos.

Best Effort. El modelo se refiere a la conectividad básica sin ningún tipo de garantía en la entrega de paquetes, todos los paquetes son tratados de la misma forma, con la misma prioridad; por ejemplo, un mensaje importante de voz, es tratado de igual manera que una fotografía digital adjunta en un e-mail. Con este modelo el protocolo *IP* hace el "mejor esfuerzo" para enviar los datagramas a su destino, pero sin la certeza de que estos paquetes no lleguen corruptos, duplicados o reordenados; a demás de esto, no existe un control en lo que respecta al uso del ancho de banda, retardo, *jitter* o pérdidas de paquetes que experimente la información. Las aplicaciones envían los datos de la forma que mejor puedan, sin informar a la red, ni solicitar confirmación de arribo de los paquetes al destino, lo que lo hace poco confiable. Se puede decir que la ventaja que tiene este modelo radica en su alta escalabilidad y de que no requiere de algún mecanismo especial para su implementación.

Integrated Service (IntServ). Algunas aplicaciones requieren de un dedicado y consistente ancho de banda en la red, para poder proveer al usuario, de un correcto funcionamiento de las mismas. *IntServ* (Servicios Integrados) fue introducido para garantizar un predecible comportamiento de la red antes de que estas aplicaciones entren en funcionamiento. En este modelo, la aplicación envía un mensaje de señalización a la red, con el fin de solicitar un tipo de servicio, que le suministre el ancho de banda y el retardo máximo tolerable para los paquetes a ser enviados.

La aplicación solamente enviará esta información en el momento en que reciba la confirmación por parte de la red. Los recursos de red son mantenidos hasta que la aplicación finaliza o hasta que el ancho de banda requerido por esta, exceda el límite que se había reservado para dicha aplicación. El modelo *IntServ* basa su operación en el Protocolo de Reservación de Recursos *(Resource Reservation Protocol "RSVP")* para señalizar y reservar la calidad de servicio requerida en la red. El *RSVP* trabaja con los protocolos de *routing* e instala el equivalente a listas de acceso dinámicas en los enrutadores *(routers)* que atraviesa, ver figura 1.10.

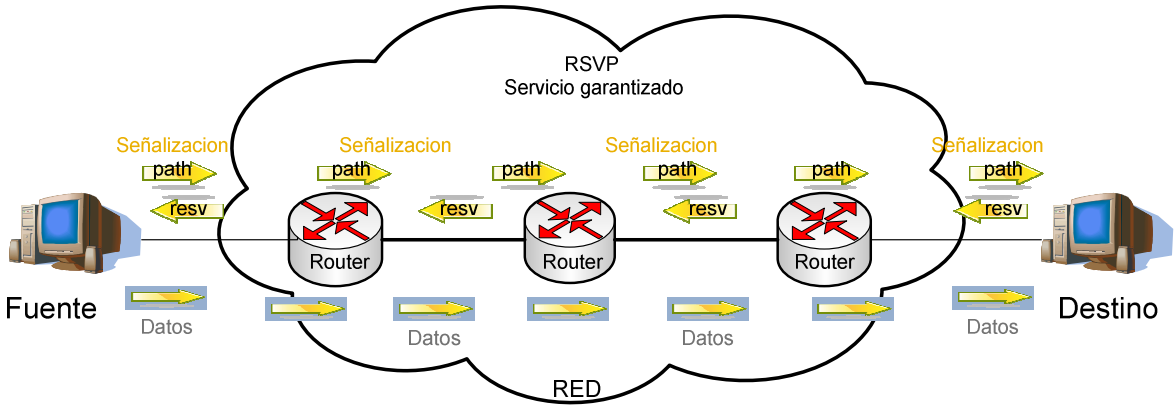

Figura 1.10 Modelo *IntServ* (protocolo *RSVP*)

Un enrutador *RSVP* solicita reservar ancho de banda al siguiente, y este lo hace con el que le sigue y así sucesivamente. Cuando el último lo concede, todos se lo conceden a su enrutador anterior, dejando un ancho de banda reservado en todo el camino. Para realizar

esto, se envían paquetes de *path message (path y resv)*, con la finalidad de descubrir todos los caminos posibles al destino e identificar el mejor, y de esta manera, notificar al resto de los enrutadores el camino elegido. Entre las desventajas que se pueden encontrar en este modelo, se puede hacer referencia, a que su implementación es compleja, además, se tiene que todos los nodos que forman la red, incluidos los sistemas de los extremos (*Pc`s* o servidores), deben entender perfectamente el protocolo *RSVP*, también se tiene su falta de escalabilidad para grandes volúmenes de información, y que los servicios suministrados son poco flexibles.

*Differentiated Service (DiffServ).*Los Servicios Diferenciados *(DiffServ)* fueron diseñados para superar las limitaciones de los modelos *Best Effort* e *IntServ*, este incluye un conjunto de herramientas de clasificación y marcado en el origen, además de mecanismos de cola. Con este modelo el tráfico de red es dividido en clases o requerimientos de servicio. A cada clase se le puede asignar un nivel diferente de prioridad. Cuando un paquete fluye por la red, cada enrutador identifica la clase y el tipo de servicio a que va destinado según su categoría (que viene marcada en la cabecera del paquete), donde serán tratados de igual modo si pertenecen a la misma clase. Con *Diffserv* se pueden escoger diferentes niveles de servicio; por ejemplo, al tráfico de voz generado por telefonía *IP*, se le da usualmente trato preferencial sobre otros tipos de aplicación como *e-mail*, páginas *web*, etc, figura 1.11.

En la implementación de este modelo se pueden conseguir distintas ventajas; como por ejemplo, los enrutadores pueden operar más rápido, ya que se limita la complejidad de la clasificación y el encolado; también se minimiza el tráfico de señalización y el almacenamiento. Los enrutadores en *Diffserv* solo se interesan en el comportamiento por salto *(Per Hop Behavior "PHB")* marcado en la cabecera del paquete. También es importante señalar otro mecanismo de marcado de paquetes que se aplica en *Diffserv*, el llamado *DiffServ Code Point (DSCP)*, el cual permite clasificar los paquetes y definir qué tipo de mecanismo se les ha de aplicar en los nodos internos de la red; por ejemplo, el tráfico procedente de distintos flujos con requisitos similares de *QoS*, se marca con el mismo *DSCP*, y así de esta manera, los flujos se agregan a una cola común o reciben el mismo trato. Esta arquitectura permite al modelo un rendimiento óptimo en ambientes de bajo ancho de banda y un mejor desempeño que el realizado por el modelo *IntServ*.

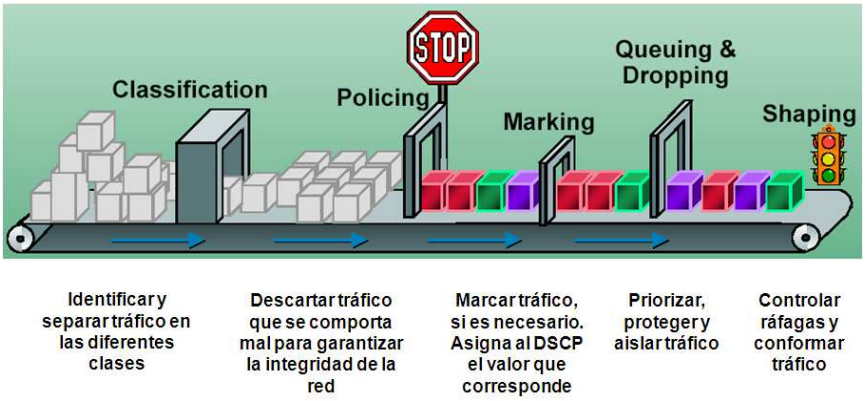

Figura 1.11 Implementación de *DiffServ* en los *Routers* (Pontificia Universidad Católica de Chile, Escuela de Ingeniería, 2009*)

1.5.3 Tipos de Mecanismos de *QoS*

Los mecanismos de Calidad de Servicio son usados para implementar eficientes políticas en el manejo de flujo de datos de la red. Desde el instante que un paquete circula por la red, este es clasificado y usualmente marcado con un tipo de identificador de clase, desde ese momento el paquete es tratado por una variedad de mecanismos de Calidad de Servicio *(IP QoS)* de acuerdo a la clasificación que presente; las herramientas de red podrán enviar, retardar, comprimir, fragmentar o descartarlo estos paquetes. Los siguientes son los mecanismos utilizados para la implementación de *QoS* en la red.

Mecanismos de Clasificación de Paquetes. La clasificación consiste en la identificación y división del tráfico de red en diferentes clases. En las redes donde son aplicados los *QoS*, todo el tráfico es clasificado en la entrada de la interfaz. Con la clasificación se puede particionar el tráfico de red en múltiples niveles de prioridad o clase de servicio, figura 1.12. La clasificación de paquetes es realizada basándose en factores, como:

- *DSCP* (Punto de Código de Servicios Diferenciados)
- *IP Precedence* (Precedencia *IP*)

- *Source Address* (Dirección Fuente)
- *Destination Address* (Dirección Destino)

BIT	0 (LSB)	1	2	3	4	5	6	7 (MSB)
Sin Diffserv	**IP Precedente** 0: Routine 1: Priority 2: Inmediate 3: Flash 4: Flash Override 5: Critical 6: Internetwork control 7: Network control			**TOS** Flags for throughput, delay and reliability				**No usado**
Con Diffserv	**DSCP** 8 clases de servicio compatibles con IP Precedente: los bits 3-5 son 000 64 clases en total Existen 12 valores específicos que marcan la prioridad a la hora de descartar paquetes (cuatro clases, tres prioridades) Se escribe como AFxy, donde x es la clase "y" la prioridad, cuanto menor sea, menos se descartará.						**ECN** Explicit Congestion Notification. Se marcan cuando el flujo esta afectado por congestión.	

Figura 1.12 Reparto de los *bits* en el *byte* "Tipo de Servicio" en la cabecera del paquete *IP* (Arribas V., Francisco, 2009)

Mecanismo de Marcado de Paquetes. Este mecanismo se encarga de marcar cada paquete como miembro de un determinado tipo de clase en la red, lo que hace que los enrutadores a través de todo el trayecto de red, logren reconocer rápidamente la clase a la cual pertenece el paquete, dándole el tratamiento correspondiente. Los mecanismos *QoS* configuran los *bits* en el campo *DSCP* o *IP Precedence* en la cabecera del paquete *IP*, de acuerdo a la clase a la que pertenece el paquete, figura 1.13.

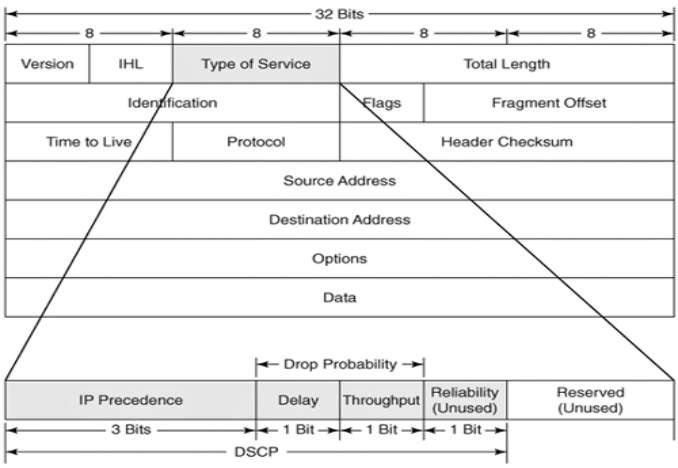

Figura 1.13 Campos *DSCP* e *IP Precedence* en la Cabecera del Paquete *IP*

Estos bits también son usados por otros mecanismos para determinar cómo son tratados los paquetes cuando estos arriban; por ejemplo, si un paquete esta marcado como paquete de voz de alta prioridad, estos en la mayoría de los casos, no serán descartados por algún mecanismo de evasión de congestión y se le dará preferencia por encima de otro. De igual modo, si el paquete es marcado como paquete de datos de baja prioridad, este será descartado cuando ocurra alguna situación de congestión

Mecanismos de Administración de Congestión. El control o administración de congestión se basa en la creación de colas, asignación de paquetes a dichas colas basándose en la clasificación de los mismos, y la distribución de paquetes en la cola para su transmisión. Dependiendo del tipo de cola, se le dará un determinado tratamiento a través del algoritmo de encolamiento, basándose en la clase de paquete.

Usualmente a las colas de paquetes marcados con alta prioridad, se les da un trato preferencial. Este manejo de congestión es implementado en la salida de la interfaz en las redes con calidad de servicio, usando los mecanismos de encolamiento correspondientes para el manejo de tráfico de salida. Cada algoritmo de encolamiento fue diseñado para resolver un determinado problema en el tráfico, y este tendrá un efecto en el funcionamiento de la red. La característica de control de congestión dentro de la calidad de servicio, ofrece un variado número de algoritmos de encolamiento, entre los que se destacan:

FIFO "First In, First Out". Este tipo de cola no contiene el concepto de prioridad o clase de tráfico. Por defecto los enrutadores se basan en este mecanismo, figura1.14.

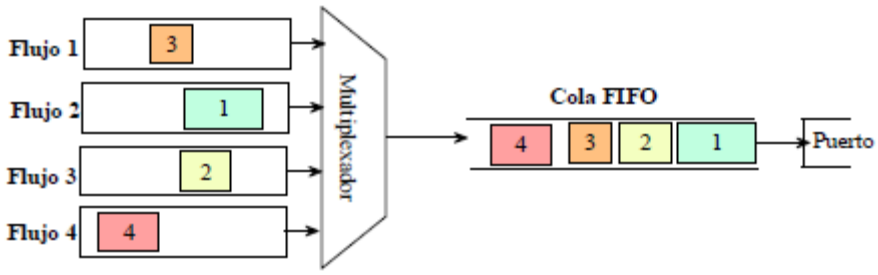

Figura 1.14 Representación del Algoritmo de Encolamiento *FIFO* (Llamas Ricardo, 2009)

PQ "Priority Queuing". Los paquetes que pertenecen a la clase de prioridad de un tráfico, son enviados antes que todo el tráfico de más baja prioridad, para asegurar la entrega oportuna de estos, figura 1.15.

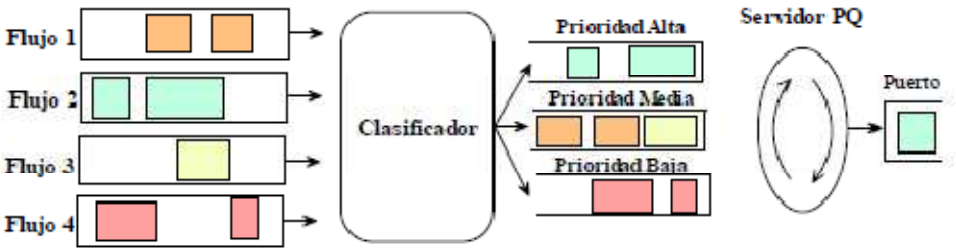

Figura 1.15 Representación del Algoritmo de Encolamiento *PQ* (Llamas Ricardo, 2009)

CQ "Custom Queuing". El ancho de banda se asigna proporcionalmente para cada clase de tráfico. Este permite especificar el número de bytes o de paquetes que se almacenan en la cola, figura 1.16.

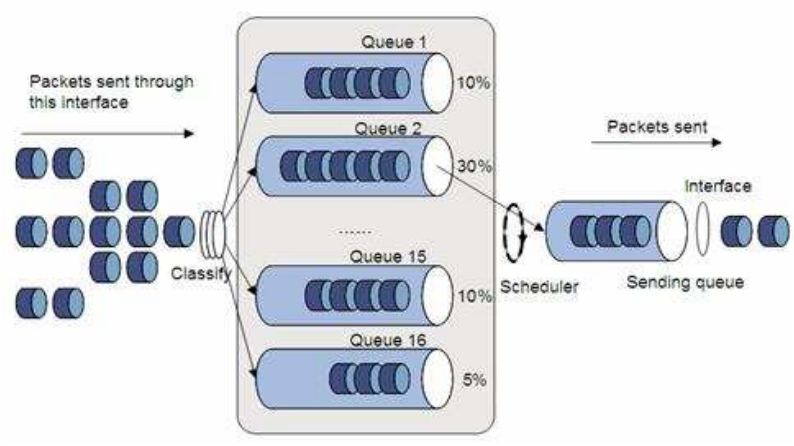

Figura 1.16 Representación del Algoritmo de Encolamiento *CQ* (H3C, 2009)

WFQ "Weighted Fair Queuing". Este asigna una ponderación a cada flujo, de esta forma determina el orden de tránsito en la cola de paquetes, divide el ancho de banda a través de las colas de trafico basadas en pesos, figura 1.17.

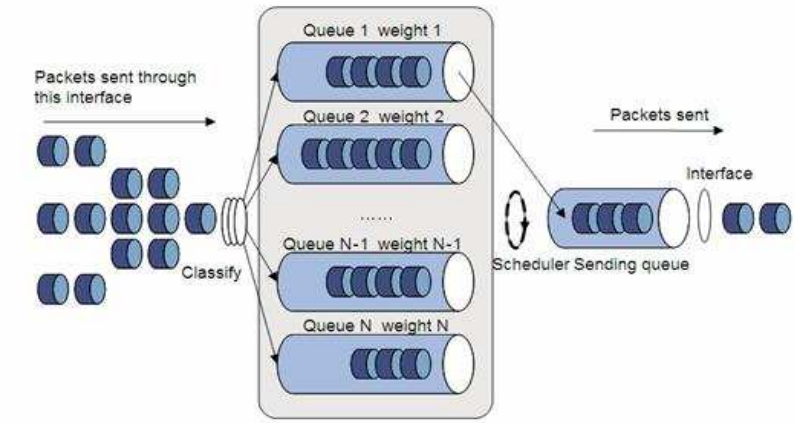

Figura 1.17 Representación del Algoritmo de Encolamiento *WFQ* (H3C, 2009)

CBWFQ "Class Based Weighted Fair Queuing". Este es una extensión de WFQ para brindar soporte a las clases de tráfico definidas por el usuario. En esta se especifican las clases de tráfico basadas en criterios de coincidencias que incluyen, protocolos, listas de control de acceso e interfaces de entrada, figura 1.18.

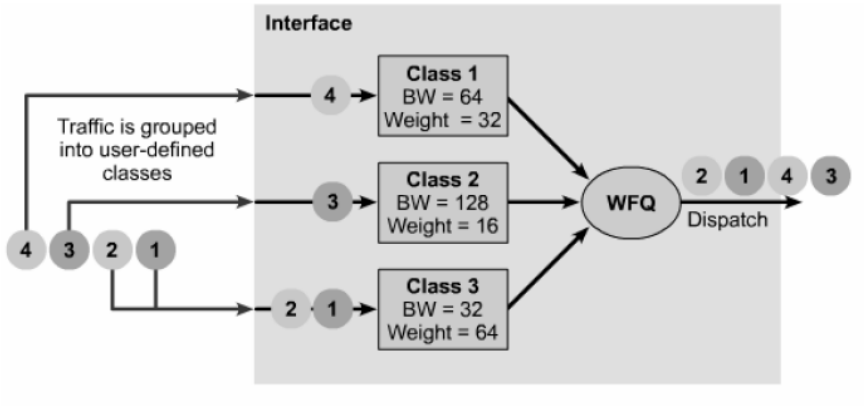

Figura 1.18 Representación del Algoritmo de Encolamiento *CBWFQ* (Montes de Oca, Faustino, 2009)

LLQ "Low Latency Queuing". El encolamiento de baja latencia es una mezcla de *PQ* y *CBWFQ*; es actualmente el método de encolamiento recomendado para voz sobre *IP (VoIP)* y también trabaja apropiadamente con tráfico de video. Permite dar preferencia absoluta a determinadas colas prioritarias, figura 1.19.

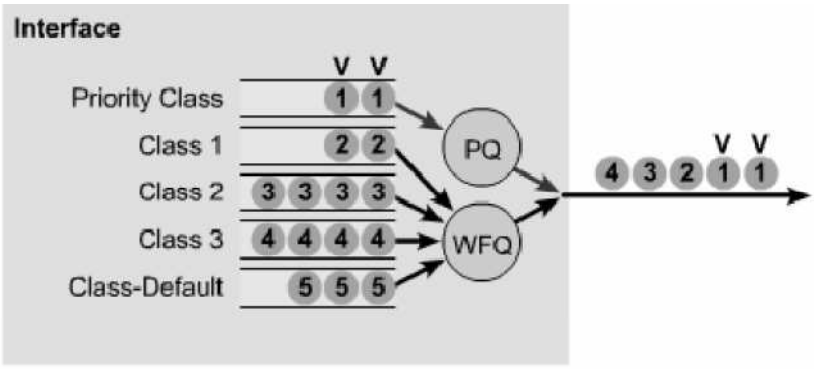

Figura 1.19 Representación del Algoritmo de Encolamiento *LLQ* (Montes de Oca, Faustino, 2009)

Mecanismo de Prevención de Congestión. Las metodologías de evasión o prevención de congestión se basan en la manera que los protocolos operan, con el fin de no llegar a la congestión de red. Los mecanismos de prevención de congestión monitorean la carga de tráfico en la red con la finalidad de anticipar y evitar la congestión en los comunes cuellos

de botella en la red; esta evasión se consigue a través del descarte de paquetes antes de que las colas se llenen. Los mecanismos son implementados normalmente en la salida de la interfaz y se usan técnicas como *RED (Random Early Detection)* y *WRED (Weighted Random Early Detection)* para tal fin.

Mecanismos de Eficiencia de Enlace. Este mecanismo utiliza técnicas de compresión y fragmentación. Comprime el contenido de todo el tráfico o solo las cabeceras, para emplear más eficientemente el ancho de banda.

Mecanismos Traffic Policing and Shaping. Estos mecanismos son usados frecuentemente para condicionar el tráfico antes de ser transmitido o recibido en la red. A través de *Traffic Policing* se descartan los paquetes que hacen superar el ancho de banda disponible, evitando que se afecten a otros flujos. Con *Traffic Shaping* se encolan todos los paquetes que hacen que el ancho de banda aumente por encima de un valor establecido, evitando que sean descartados por cuellos de botella posteriores.

1.5.4 Clasificación y Priorización de Tráfico en Redes Convergentes.

Se sabe que las aplicaciones existentes en una red convergente generan tráfico a ritmo variable y se requiere que este tráfico sea transportado al nivel que las aplicaciones lo han generado. La capacidad de un dispositivo de red para enviar tráfico constituye un recurso de red fundamental. Mecanismos como clasificación y priorización de tráfico de red, proporcionan un servicio mejorado a los usuarios, al mismo tiempo que permiten al administrador de la red, gestionar los recursos de forma eficaz.

Con la clasificación del tráfico se consigue dividir el mismo en diferentes categorías; la priorización de tráfico tomando en cuenta la clasificación de los mismos, permite dar un tratamiento diferente a cada flujo de datos, para asegurar que el tráfico perteneciente a

aquellas clases con requerimientos de menor retardo, sea reenviado antes que el tráfico que no es sensible al retardo.

1.5.5 Impacto de los Diferentes Mecanismos de *QoS* sobre los Paquetes *IP*.

En las redes convergentes se hace necesario poder manipular en forma eficiente las distintas aplicaciones que son utilizadas por los usuarios; la aplicación de mecanismos de calidad de servicio permite gestionar esa demanda de tráfico. Esta se basa en un conjunto de políticas administrativas y patrones de uso, que son aplicados entre otros, a los paquetes que componen el caudal de datos que fluye por la red. El impacto principal de los mecanismos *QoS* se da en la garantía de que el rendimiento de las aplicaciones criticas en la red, sea llevado a cabo en forma óptima y que los parámetros que afectan este rendimiento, como el retardo, *jitter*, pérdidas de paquetes, congestión y uso de ancho de banda, sean manejados, aplicando correctas políticas de administración de red.

CAPÍTULO 2
EXPERIMENTACIÓN

El siguiente capítulo describe el escenario en donde se lleva a cabo la parte experimental del trabajo de grado, se dan a conocer el desarrollo de las pruebas de los diferentes mecanismos *QoS* y de que herramientas se utilizan para las mismas.

2.1 DESCRIPCIÓN DE LOS ESCENARIOS PARA LAS PRUEBAS INDIVIDUALES

El escenario planteado para la realización de las pruebas de los mecanismos *QoS* se traslado a uno de los laboratorios de redes de que dispone Red ULA. Allí se logro instalar un sistema compuesto por varios dispositivos de red que tuvieron como finalidad lograr simular el tráfico en la red que fluye entre el Núcleo de Mérida y el Núcleo Universitario Rafael Rangel (NURR) pertenecientes a la Universidad de Los Andes. Este tenía como componente principal dos *routers*, uno ubicado en el Núcleo Mérida y el otro en el NURR. A través de estos enrutadores se creó una conexión principal o troncal por medio de la cual se envían y enrutan todos los paquetes *IP*.

El router principal se ubicó en el Núcleo Mérida, este presentaba un puerto de salida hacia *Internet* que permitía establecer conexión con el teléfono receptor de llamadas bajo

protocolo *Internet (IP)*; se realizo de este modo con la finalidad de asignar un *IP* pública al teléfono, lo cual facilitaba realizar la llamada entre los núcleos y simular llamadas hacia otros sitios en Internet; que es la condición mas critica para el tráfico de voz. Al Núcleo Mérida se le coloco un switch para interconectar los diferentes segmentos de red, se aplicó la configuración en estrella para tal fin; a cada segmento se le asignó una red de área local virtual *(Virtual Local Area Network "VLAN")*, tres en total. Estas fueron configuradas para poder disponer de redes lógicamente independientes dentro de la misma red física con la cual se trabajo. A las redes de video, datos y telefonía les fue asignada una *VLAN* respectivamente para poder desarrollar las pruebas, mismas que fueron monitoreadas a través de un equipo instalado en el switch ubicado en Núcleo Mérida.

Este monitor de red se encargo de hacer los registros del tráfico que circulaba por el troncal que interconecta a los núcleos. En lo que respecta al Núcleo Universitario Rafael Rangel se aplicó la misma configuración de red que se realizó en Núcleo Mérida, con la diferencia que en el *switch* de este núcleo no se instalo monitor de red y que la red de telefonía *IP* utilizada por este, uso una aplicación para *VoIP* como *Xlite*, instalada en el computador y no desde un teléfono *IP*, figura 2.1.

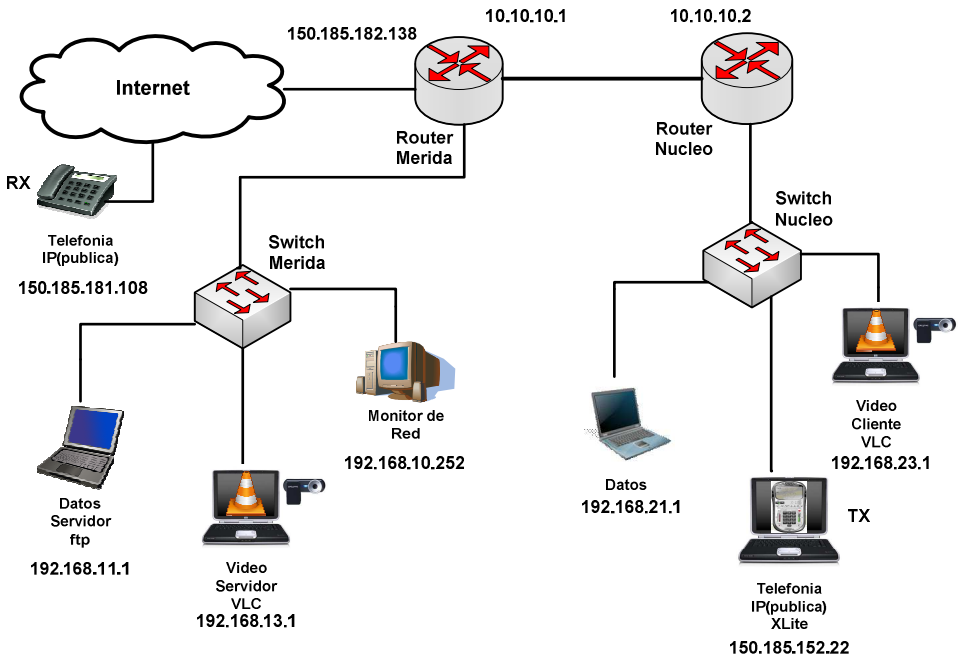

Figura 2.1 Configuración de Red Aplicada en el Laboratorio

La transferencia de los paquetes de información se realizó en forma unidireccional desde Núcleo Mérida al NURR para datos y video, y en forma bidireccional para telefonía *IP*.

Para realizar el estudio de los mecanismos *QoS*, se hizo necesario tener registro de un proceso estadístico acerca del comportamiento del tráfico existente entre el Núcleo Mérida y el *NURR* en tiempo real, para poder tener una idea exacta de sobre que parámetros de tráfico se basará el estudio; llevándose esto a cabo, a través de la caracterización del tráfico de red. Con los datos obtenidos por medio de la caracterización, se pudo tener un punto de partida en la aplicación de los mecanismos de Calidad de Servicio, debido a que permite poder calibrar los parámetros necesarios para tener una transferencia de paquetes más acorde con las necesidades de cada servicio.

Esta caracterización de tráfico fue elaborada por el sistema de monitoreo de Red ULA a través del protocolo de red *NetFlow*, con un registro estadístico llevado a cabo durante tres días, en el troncal que comunica el Núcleo Mérida con el Núcleo Universitario Rafael Rangel de la Universidad de Los Andes. La figura 2.2 muestra el registro gráfico de esta caracterización.

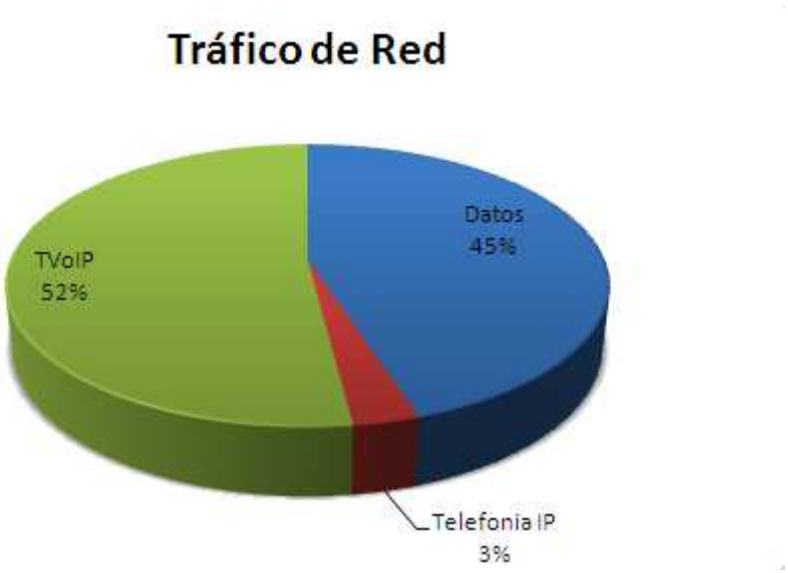

Figura 2.2 Registro Gráfico de la Caracterización.

Como muestra el gráfico, el tráfico que se registro durante este periodo de tiempo, se distribuyo de la siguiente manera, para flujo de Datos 45%, Telefonía *IP* 3% y *TVoIP* 52%. Se debe aclarar que en el momento de realizar este monitoreo, se registro actividad sobre una aplicación de *TVoIP*, situación que es poco frecuente, si se toma en cuenta, el tipo de información que se transfiere diariamente entre los núcleos. En los casos que se presenta, esta tiende a ocupar similar ancho de banda; por lo tanto, se tomo como patrón ese consumo.

2.2 DESARROLLO DE PRUEBAS INDIVIDUALES DE MECANISMOS QoS

En la primera parte de este desarrollo, se hicieron pruebas para la calibración de cada aplicación, sobre el enlace configurado; con la finalidad de obtener los requerimientos de ancho de banda mínimos para su buen funcionamiento. Estas se realizaron en forma individual, empleando para esto, un flujo de paquetes por tipo de aplicación, enviados de un núcleo a otro y monitoreando su comportamiento. A través de estas, se consiguieron los siguientes parámetros, Telefonía *IP* 90 Kbps, *TVoIP* 1200 Kbps y 700 Kbps para datos.

Para llevar a cabo las pruebas de los mecanismos *QoS*, se debieron programar los enrutadores *Cisco* 2811, configurándolos a través del sistema de interconexión de red *“IOS” (Internetwork Operating System)* , procediendo de la siguiente manera; primero se definieron las listas de acceso *(Access List “ACL”)* por cada rango de direcciones *IP* que se utilizaron en la red, luego se crearon los *Class-Map* con los cuales se establecieron las clases de tráfico y finalmente se configuró el *Policy-Map*, en donde se especifica que tratamiento deben recibir los paquetes de cada una de las clases creadas, a través de esta, se puede aplicar diferentes políticas, como por ejemplo, *CBWFQ*, *WRED* o *LLQ*; toda esta configuración se hace bajo el modelo de calidad de servicio *DiffServ*. A continuación se muestra la sintaxis de los comandos *class-map*, *policy-map* y un ejemplo de la lista de acceso.

Access List:

access-list 10 permit 192.168.10.0 0.0.0.255

access-list 10 permit 192.168.11.0 0.0.0.255

access-list 10 permit 192.168.12.0 0.0.0.255

access-list 10 permit 10.10.10.0 0.0.0.3

access-list 10 deny any

Class-Map:

class-map [match-any/match-all] *class-name*

Policy-Map:

policy-map *policy-map*

class class-name

2.2.1 Mecanismo de Administración de Congestión *FIFO*

Este mecanismo de gestión de cola es uno de los que está presente por defecto en los *routers*; como ya se explicó, en *FIFO*, el primer paquete que arriba al enrutador, es el primer paquete en salir. Este mecanismo es activado colocando en inactividad *(no fair queue)* a otras colas, como por ejemplo *WFQ*, que por defecto trae la interfaz serial. Aquí no hay mecanismos de diferenciación o clasificación de paquetes y se configuró de la siguiente manera, figura 2.3.

```
class-map match-all VIDEO-MERIDA
 match access-group name video-merida
class-map match-all DATOS-MERIDA
 match access-group name datos-merida
class-map match-all VOZ-MERIDA
 match access-group name voz-merida

interface Serial0/0/0
 description CONEXION a NUCLEO
 bandwidth 2000
 ip address 150.185.152.17 255.255.255.252
secondary
 ip address 10.10.10.1 255.255.255.252
 ip flow ingress
 ip flow egress
 ip nat inside
 ip virtual-reassembly
 encapsulation ppp
 no fair-queue
 clock rate 2000000
!
interface Serial0/0/1
 no ip address
 shutdown
 no fair-queue
 clock rate 2000000
```

Figura 2.3 Configuración para *FIFO*

2.2.2 Mecanismo de Administración de Congestión *WFQ*

Este mecanismo de gestión de cola es otro de los que viene habilitado en muchos casos en el router por defecto, dándose esto sólo en aquellas interfaces que tiene un ancho de banda menor a 2 *Mbps*. En este caso se usó el comando *fair-queue* para habilitarlo en la interfaz. Se asignaron un número de colas dinámicas a cada flujo de datos, el valor utilizado es el que existe por defecto (256), tomando en cuenta como base, el ancho de banda de la interfaz, que para este estudio, es de 2048 *Kbps* (troncal); partiendo de la recomendación que hace Cisco de asignar 256 colas dinámicas para enlaces mayores de 512 *Kbps*. El número de mensajes máximo se estableció en 64 (por defecto) antes de que el router comience a descartar los paquetes, figura 2.4.

```
class-map match-all VIDEO-MERIDA
 match access-group name video-merida
class-map match-all DATOS-MERIDA
 match access-group name datos-merida
class-map match-all VOZ-MERIDA
 match access-group name voz-merida

interface Serial0/0/0
 description CONEXION a NUCLEO
 bandwidth 2000
 ip address 150.185.152.17 255.255.255.252 secondary
 ip address 10.10.10.1 255.255.255.252
 ip flow ingress
 ip flow egress
 ip nat inside
 ip virtual-reassembly
 encapsulation ppp
 fair-queue
 clock rate 2000000
!
interface Serial0/0/1
 no ip address
 shutdown
 no fair-queue
 clock rate 2000000
GW-Merida#sh queueing
Current fair queue configuration:

  Interface             Discard    Dynamic  Reserved
Link     Priority
                        threshold  queues   queues
queues   queues
  Serial0/0/0           64         256      0         8
    1
```

Figura 2.4 Configuración para *WFQ*

2.2.3 Mecanismo de Administración de Congestión *CBWFQ-WFQ*

Otro de los mecanismos de gestión de cola utilizado en el estudio realizado es el llamado *CBWFQ-WFQ*, aquí se definieron las clases de tráfico como voz (telefonía), video y *class default* asignándola al resto del tráfico (datos), también se reservo una cola para cada clase, asignándole valores por defecto. En el caso del peso *(weight)* y el límite máximo de paquetes admitidos por cola *(discard threshold)* se colocaron 256 y 64 respectivamente. El número de colas dinámicas se estableció por defecto en 1024, además de esto el ancho de banda asignado a cada clase fue de 90 *Kbps* para voz (telefonía), 1200 *Kbps* para video y 512 *Kbps* para el resto del caudal (datos).Todo esto se hizo tomando en consideración la “regla de 75%”, la cual dice que la cantidad total de ancho de banda distribuido en todas las clase, no debe exceder el 75% del ancho de banda disponible por la interfaz, que en la red configurada es de 2048 *Kbps*, figura 2.5.

```
class-map match-all VIDEO-MERIDA
 match access-group name video-merida
class-map match-all DATOS-MERIDA
 match access-group name datos-merida
class-map match-all VOZ-MERIDA
 match access-group name voz-merida
!
!
policy-map WFQ
 class VOZ-MERIDA
  bandwidth 90
 class VIDEO-MERIDA
  bandwidth 1200
 class class-default
  fair-queue 512


GW-Merida#sh queueing
Current fair queue configuration:

  Interface             Discard    Dynamic  Reserved
Link     Priority
                        threshold  queues   queues
queues   queues
  Serial0/0/0             64         1024     256
    8      1
```

Figura 2.5 Configuración para CB*WFQ-WFQ*

2.2.4 Mecanismo de Administración de Congestión *CBWFQ-FIFO*

En este caso se utilizó nuevamente el mecanismo de administración de congestión *CBWFQ* y dentro de las colas formadas por este mecanismo, se aplicó *FIFO*; lo cual es una diferencia, si lo comparamos con el mecanismo anterior, al que se le aplico *WFQ*. En este mecanismo se definieron 2 tipos de clases de tráfico, una para voz (telefonía) y la otra para video; la tercera clase de tráfico que se debió configurar, era la de datos, pero en este caso no fue necesario, debido a que el router la asumía por defecto para el tráfico restante. Se le asigno un máximo de paquetes admitidos por cola de 64 para ambas clases, este valor es una cifra por defecto. El ancho de banda configurado fue de 90 *Kbps* para audio y de 1200 *Kbps* para video. Se tomo en cuenta nuevamente la “regla del 75%”, figura 2.6.

```
class-map match-all VIDEO-MERIDA
 match access-group name video-merida
class-map match-all DATOS-MERIDA
 match access-group name datos-merida
class-map match-all VOZ-MERIDA
 match access-group name voz-merida
!
!
policy-map CBWFQ-FIFO
 class VOZ-OUT
  bandwidth 90
 class VIDEO-OUT
  bandwidth 1200
!
!
!

GW-Merida#sh policy-map CBWFQ-FIFO
  Policy Map CBWFQ-FIFO
    Class VOZ-OUT
      Bandwidth 90 (kbps) Max Threshold 64 (packets)
    Class VIDEO-OUT
      Bandwidth 1200 (kbps) Max Threshold 64 (packets)
```

Figura 2.6 Configuración para *CBWFQ-FIFO*

2.2.5 Mecanismo de Marcado de Paquetes y Prevención de Congestión *DSCP-Based WRED*

Para este mecanismo de evasión de congestión *(WRED)* se utilizó como selector de descarte, el mecanismo de marcado de paquetes *DSCP*, con el cual se asignaron diferentes prioridades a los paquetes de datos en cada tipo de tráfico, permitiendo esto dar mayor probabilidad de descarte al tráfico con niveles de prioridad bajo. Los valores que se utilizaron en este marcado de los paquetes fueron los siguientes; para el tráfico de voz “AF11”, para el tráfico de video “AF22” y para el resto del tráfico (datos) “AF33”.La probabilidad de descarte de paquetes aumenta a medida que este valor sea más alto, como se muestra a continuación:

AF11 = Baja probabilidad de descarte

AF22 = Media probabilidad de descarte

AF33 = Alta probabilidad de descarte

El ancho de banda asignado se mantiene en 90 *Kbps* para voz y 1200 *Kbps* para video, el trafico restante se asigno por defecto *(class-default)*, figura 2.7.

```
class-map match-any VOZ-OUT
 match access-group name voz-merida
 match ip dscp af11
class-map match-any DATOS-OUT
 match access-group name datos-merida
 match ip dscp af33
class-map match-any VIDEO-OUT
 match access-group name video-merida
 match ip dscp af22
!
!
policy-map DSCP-BasedWRED
 class VOZ-OUT
  bandwidth 90
  random-detect dscp-based
  set ip dscp af11
 class VIDEO-OUT
  bandwidth 1200
  random-detect dscp-based
  set ip dscp af22
 class class-default
  fair-queue
  random-detect dscp-based
  set ip dscp af33
```

Figura 2.7 Configuración para *DSCP-Based WRED*

2.2.6 Mecanismo de Marcado de Paquetes y Prevención de Congestión *IP Precedence-Based WRED*

Este caso es similar al anterior, se utiliza *WRED* como mecanismo de evasión de congestión, pero esta vez se marcan los paquetes con *IP Precedence*, con lo cual se especifica la prioridad que se le da al paquete en el descarte. El tráfico de voz fue configurado con *IP precedence* 5 y un ancho de banda de 90 *Kbps*, el tráfico de video con *IP precedence* 3 y un ancho de banda de 1200 *Kbps* y por último al resto del tráfico (class-default) se le asigno *IP precedence* 1 tomando el ancho de banda restante, figura 2.8. Los indicadores en el nivel de precedencia tienen los siguientes parámetros:

IP Precedence 5 = Critico

IP Precedence 3 = Urgente

IP Precedence 1 = Prioridad

```
class-map match-any VOZ-OUT
 match access-group name voz-merida
 match ip precedence 5
class-map match-any DATOS-OUT
 match access-group name datos-merida
 match ip precedence 0  1
class-map match-any VIDEO-OUT
 match access-group name video-merida
 match ip precedence 3  4
!
!
policy-map IP-PRECEDENCE-BasedWRED
 class VOZ-OUT
  bandwidth 90
  random-detect
  set ip precedence 5
 class VIDEO-OUT
  bandwidth 1200
  random-detect
  set ip precedence 3
 class class-default
  fair-queue
  random-detect
  set ip precedence 1
```

Figura 2.8 Configuración para *IP Precedence-Based WRED*

2.2.7 Mecanismo *Traffic Policing and Shaping*

A través de estos mecanismos de calidad de servicio se puede limitar el rango o la cantidad de ancho de banda utilizado por las clases de tráfico configuradas en la red. Los tráficos se configuraron de la siguiente manera, para voz (telefonía) se asigno 90 Kbps, para video se asignó 1200 Kbps y al resto del tráfico por defecto (class-default), figura 2.9.

```
class-map match-any VOZ-OUT
 match access-group name voz-merida
class-map match-any DATOS-OUT
 match access-group name datos-merida
class-map match-any VIDEO-OUT
 match access-group name video-merida
!
!
policy-map CB-SHAPING
 class VOZ-OUT
  shape average 90000
 class VIDEO-OUT
 shape average 1200000
 class class-default
  fair-queue
```

Figura 2.9 Configuración para *Traffic Policing and Shaping*

2.2.8 Mecanismo de Administración de Congestión *LLQ*

El encolamiento de baja latencia, garantiza estricta prioridad a las colas de paquetes sensibles a los retardos. La configuración planteada en este mecanismo se realizo de la siguiente manera, para el tráfico de voz se fijo una prioridad en ancho de banda de 90 Kbps con *IP precedence* de valor 5, para el tráfico de video se asignó 1200 Kbps y un *IP precedence* de valor 3 y al resto del tráfico se fijaron valores por defecto; o sea, *class-default* con *IP precedence* 1 y *fair queue* como encolamiento, figura 2.10.

```
class-map match-any VOZ-OUT
 match access-group name voz-merida
class-map match-any DATOS-OUT
 match access-group name datos-merida
class-map match-any VIDEO-OUT
 match access-group name video-merida
!
!
policy-map LLQ-OUT
 class VOZ-OUT
  priority 90
  set ip precedence 5
 class VIDEO-OUT
  bandwidth 1200
  set ip precedence 3
 class class-default
  set ip precedence 1
  fair-queue
  random-detect
```

Figura 2.10 Configuración para *LLQ*

2.3 HERRAMIENTAS UTILIZADAS EN EL DESARROLLO DE LAS PRUEBAS DE MECANISMOS *QoS*

2.3.1 Aplicaciones de Red (voz, datos y video)

Las pruebas de simulación que se llevaron a cabo, utilizaron como aplicaciones para generar el tráfico en la red, las siguientes herramientas.

VLC media player .Este es un reproductor multimedia que soporta un variado número de *codecs* de audio y video, así como diferentes tipos de archivos, *DVD*, *VCD* y varios protocolos *streaming*. Este permite ser utilizado como servidor en *unicast* o *multicast*. *VLC* se utilizó como fuente para generar tráfico de video entre los núcleos; para ello se configuraron dos *Laptop Dell Inspiron* 1300 bajo plataforma *Windows XP*. Uno se estableció como servidor (Núcleo Mérida) y el otro como cliente (NURR); como generador de tráfico de video se utilizo una película en formato DVD, figura 2.11.

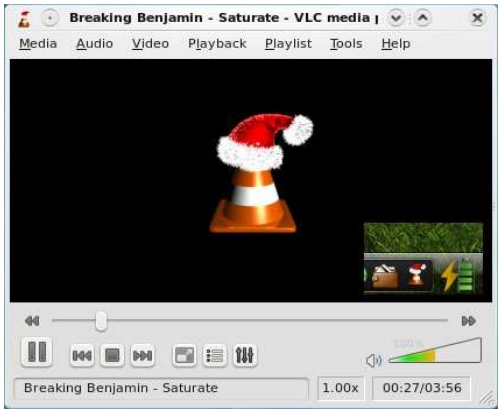

Figura 2.11 Interfaz Visual de *VLC*

Xlite. Esta es una aplicación conocida como *SoftPhone*, la cual permite mezclar voz y video en una llamada a través de telefonía *VoIP*. La misma se instalo en *un Laptop Dell Inspiron* 1300 bajo plataforma *Windows XP* en el NURR, figura 2.12.

Figura 2.12 Interfaz Visual de *Xlite*

Dialog 4422 IP Office. Teléfono *IP* utilizado en aplicaciones de *VoIP,* fabricado por *Ericsson* (ver figura 2.13).

Figura 2.13 Teléfono *IP Dialog 4422 IP Office*

Servidor ftp. La generación del tráfico de datos se realizó desde un *Laptop Aspire One* de *Acer*, esta máquina fue configurada para actuar como servidor de descarga de archivos hacia el NURR, opero bajo plataforma *Linux Debian* y se utilizó para la descarga un archivo de imagen *ISO*, figura 2.14.

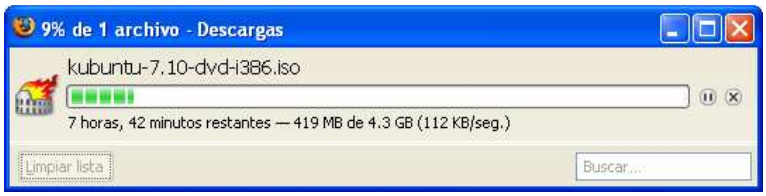

Figura 2.14 Archivo de imagen *ISO*

2.3.2 Analizadores de Tráfico

El análisis de tráfico empleado para la caracterización se hizo a través del protocolo de red *NetFlow*, desarrollado por *Cisco System*. Este se basa en la recolección de información de tráfico *IP* en la red. El método para el registro del tráfico es el siguiente; en el enrutador de borde *Cisco* 7200, se activa una aplicación que permite a estos equipos enviar el protocolo de comunicación *NetFlow*, para posteriormente comenzar a suministrar los flujos de tráfico a un equipo designado para su captura. En este equipo se almacena el tráfico durante un año, aproximadamente. Con esta herramienta es posible ver *IP* origen y destino, puertos, número de *Bytes* y hora en el registro de los flujos. Con los parámetros nombrados anteriormente, se procesa la data por puertos o protocolos y se van caracterizando los datos, para poder construir el patrón de tráfico de la Universidad de Los Andes.

CAPÍTULO 3
RESULTADOS

El capítulo que a continuación se presenta, describe los resultados obtenidos a través de la implementación de los diferentes mecanismos *QoS*, estos resultados son expuestos en forma gráfica y por medio de tablas.

3.1 IMPACTO DE LOS DIFERENTES MECANISMO *QoS* SOBRE EL FLUJO DE TRAFICO *IP*

La data recabada fue obtenida por medio del monitor de red, utilizando para esto, la aplicación *Smokeping* y ejecutando en el *router* el comando *show interface.* Los parámetros latencia, *jitter*, perdidas de paquetes y *RTT* fueron tomados para periodos de tiempo de 20 minutos. El porcentaje de utilización de *CPU* y memoria se llevo a cabo para periodos de 5 minutos. Los parámetros de latencia se obtuvieron dividiendo *RTT* entre 2, debido a que *Smokeping* no grafica directamente este valor.

Hay que tener en cuenta, que para la toma del valor de latencia, se puede presentar un margen de error en el resultado, debido a que es posible que el canal de transmisión se encuentre saturado en un sentido y no en el otro, al mismo nivel; o sea, que los paquetes que viajan desde Núcleo Mérida a NURR, necesariamente tengan el mismo retardo, que los paquetes que lo hacen en sentido contrario.

Tabla 3.1 Parámetros Registrados en Núcleo Mérida y NURR (*FIFO)*

Tipo de Mecanismo QoS	Clase de Trafico	Ancho de Banda (Kbps)	Latencia (ms)	Jitter (ms)	Perdidas de Paquetes (%)	RTT (ms)	Uso CPU (%)	Uso Memoria RAM (%)
FIFO (Mérida)	Datos	2048	0.21	0.54	0	0.42	7	17.46
	Telefonía		0.42	1.54	0	0.84	7	17.46
	Video		0.26	0.75	0	0.51	7	17.46
FIFO (NURR)	Datos	2048	256.9	310	6	513.7	4	16.05
	Telefonía		254.8	330	8	509.5	4	16.05
	Video		257.7	390	0	515.4	4	16.05

Tabla 3.2 Parámetros Registrados en Núcleo Mérida y NURR (*WFQ)*

Tipo de Mecanismo QoS	Clase de Trafico	Ancho de Banda (Kbps)	Latencia (ms)	Jitter (ms)	Perdidas de Paquetes (%)	RTT (ms)	Uso CPU (%)	Uso Memoria RAM (%)
WFQ (Mérida)	Datos	2048	0.21	0.6	0	0.42	7	17.43
	Telefonía		0.43	0.9	0	0.85	7	17.43
	Video		0.24	1.84	0	0.49	7	17.43
WFQ (NURR)	Datos	2048	4.20	1.7	0	8.4	4	16.06
	Telefonía		4.75	2.3	0	9.5	4	16.06
	Video		5.4	3	0	10.48	4	16.06

Tabla 3.3 Parámetros Registrados en Núcleo Mérida y NURR (*CBWFQ-WFQ)*

Tipo de Mecanismo QoS	Clase de Trafico	Ancho de Banda (Kbps)	Latencia (ms)	Jitter (ms)	Perdidas de Paquetes (%)	RTT (ms)	Uso CPU (%)	Uso Memoria RAM (%)
CBWFQ WFQ (Mérida)	Datos	2048	0.23	0.74	0	0.46	9	17.52
	Telefonía		0.44	1.47	0	0.88	9	17.52
	Video		0.24	0.48	0	0.49	9	17.52
CBWFQ WFQ (NURR)	Datos	2048	39.25	13	0	78.50	5	16.10
	Telefonía		44.20	46	0	88.40	5	16.10
	Video		47.75	15	0	95.50	5	16.10

Tabla 3.4 Parámetros Registrados en Núcleo Mérida y NURR (*CBWFQ-FIFO)*

Tipo de Mecanismo QoS	Clase de Trafico	Ancho de Banda (Kbps)	Latencia (ms)	Jitter (ms)	Perdidas de Paquetes (%)	RTT (ms)	Uso CPU (%)	Uso Memoria RAM (%)
CBWFQ FIFO (Mérida)	Datos	2048	0.21	0.65	0	0.41	9	20.77
	Telefonía		0.42	0.85	0	0.83	9	20.77
	Video		0.23	0.68	0	0.46	9	20.77
CBWFQ FIFO (NURR)	Datos	2048	3150	3100	30.01	6300	4	15.89
	Telefonía		3150	3300	30.01	6300	4	15.89
	Video		3150	3200	30.01	6300	4	15.89

Tabla 3.5 Parámetros Registrados en Núcleo Mérida y NURR (*DSCP-Based WRED)*

Tipo de Mecanismo QoS	Clase de Trafico	Ancho de Banda (Kbps)	Latencia (ms)	Jitter (ms)	Perdidas de Paquetes (%)	RTT (ms)	Uso CPU (%)	Uso Memoria RAM (%)
DSCP-Based WRED (Mérida)	Datos	2048	0.21	0.76	0	0.43	9	20.77
	Telefonía		0.43	1.18	0	0.86	9	20.77
	Video		0.24	0.81	0	0.48	9	20.77
DSCP-Based WRED (NURR)	Datos	2048	43.80	53	0	87.60	4	15.87
	Telefonía		30.60	82	0	61.20	4	15.87
	Video		41.10	124	0	82.20	4	15.87

Tabla 3.6 Parámetros Registrados en Núcleo Mérida y NURR (*IP Precedence-Based WRED)*

Tipo de Mecanismo QoS	Clase de Trafico	Ancho de Banda (Kbps)	Latencia (ms)	Jitter (ms)	Perdidas de Paquetes (%)	RTT (ms)	Uso CPU (%)	Uso Memoria RAM (%)
IP Precedence-Based WRED (Mérida)	Datos	2048	0.23	0.55	0	0.46	9	20.79
	Telefonía		0.43	1.53	0	0.86	9	20.79
	Video		0.23	0.83	0	0.47	9	20.79
IP Precedence-Based WRED (NURR)	Datos	2048	49.55	21	0	99.10	4	15.90
	Telefonía		45.50	41	0	91	4	15.90
	Video		47.30	94	0	94.60	4	15.90

Tabla 3.7 Parámetros Registrados en Núcleo Mérida y NURR (*Traffic Policing and Shaping)*

Tipo de Mecanismo QoS	Clase de Trafico	Ancho de Banda (Kbps)	Latencia (ms)	Jitter (ms)	Perdidas de Paquetes (%)	RTT (ms)	Uso CPU (%)	Uso Memoria RAM (%)
Policing and Shaping (Mérida)	Datos	2048	0.21	0.47	0	0.42	9	20.77
	Telefonía		0.42	1.06	0	0.83	9	20.77
	Video		0.23	0.74	0	0.46	9	20.77
Policing and shaping (NURR)	Datos	2048	5.65	5.2	0	11.30	4	15.88
	Telefonía		5	0.1	0	10	4	15.88
	Video		5.5	0.7	0	11	4	15.88

Tabla 3.8 Parámetros Registrados en Núcleo Mérida y NURR (*LLQ)*

Tipo de Mecanismo QoS	Clase de Trafico	Ancho de Banda (Kbps)	Latencia (ms)	Jitter (ms)	Perdidas de Paquetes (%)	RTT (ms)	Uso CPU (%)	Uso Memoria RAM (%)
Policing and Shaping (Mérida)	Datos	2048	0.22	0.63	0	0.44	9	20.67
	Telefonía		0.43	1.47	0	0.86	9	20.67
	Video		0.25	0.76	0	0.5	9	20.67
Policing and shaping (NURR)	Datos	2048	45.15	128	0	90.30	4	15.86
	Telefonía		43.95	47	0	87.90	4	15.86
	Video		46.20	40	0	92.40	4	15.86

Los siguientes gráficos muestran los parámetros obtenidos a través del mecanismo de administración de congestión *FIFO*, el cual se toma como base para realizar la comparación con los otros mecanismos *QoS* aplicados en la red, debido a que este mecanismo es el implementado por defecto en los *routers Cisco.*

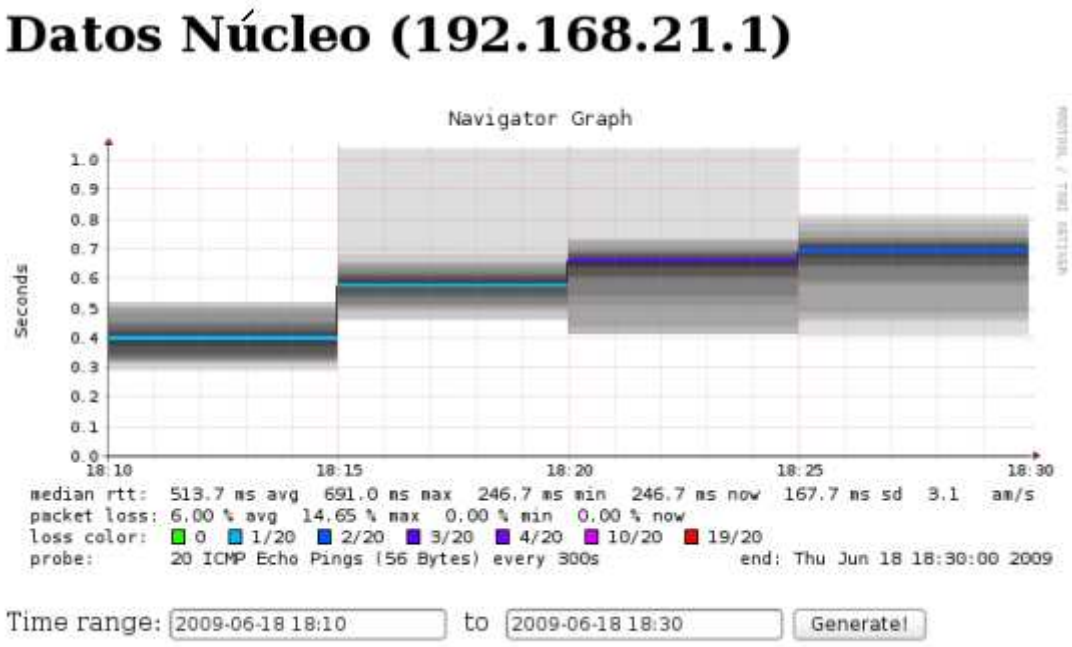

Gráfico 3.1 ***FIFO Smokeping*** **Datos Núcleo**

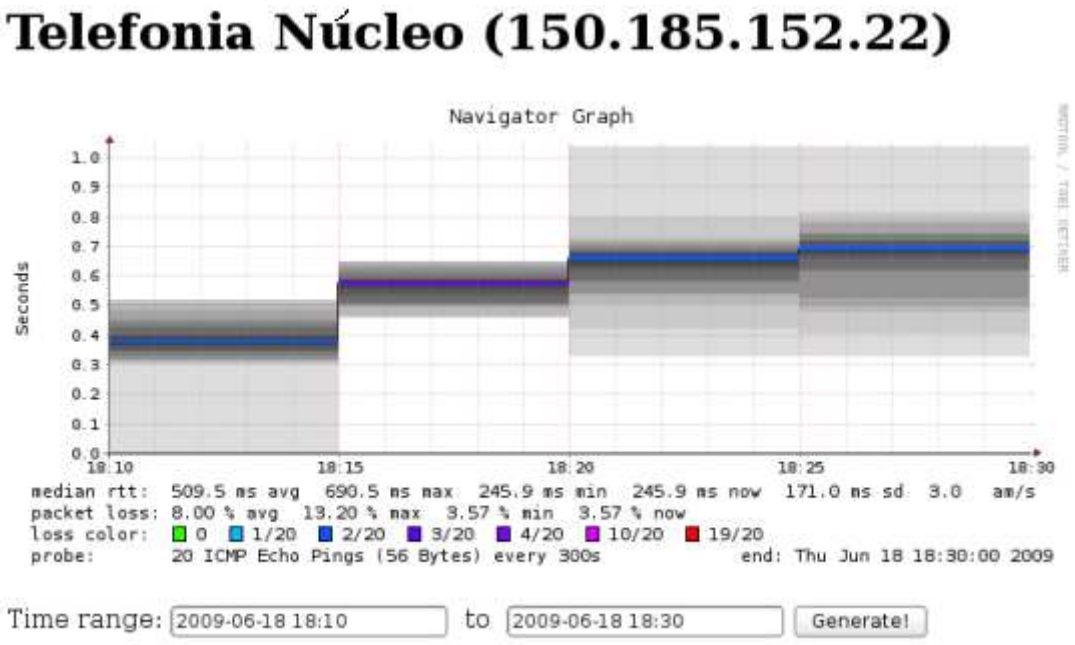

Gráfico 3.2 ***FIFO Smokeping*** **Telefonía Núcleo**

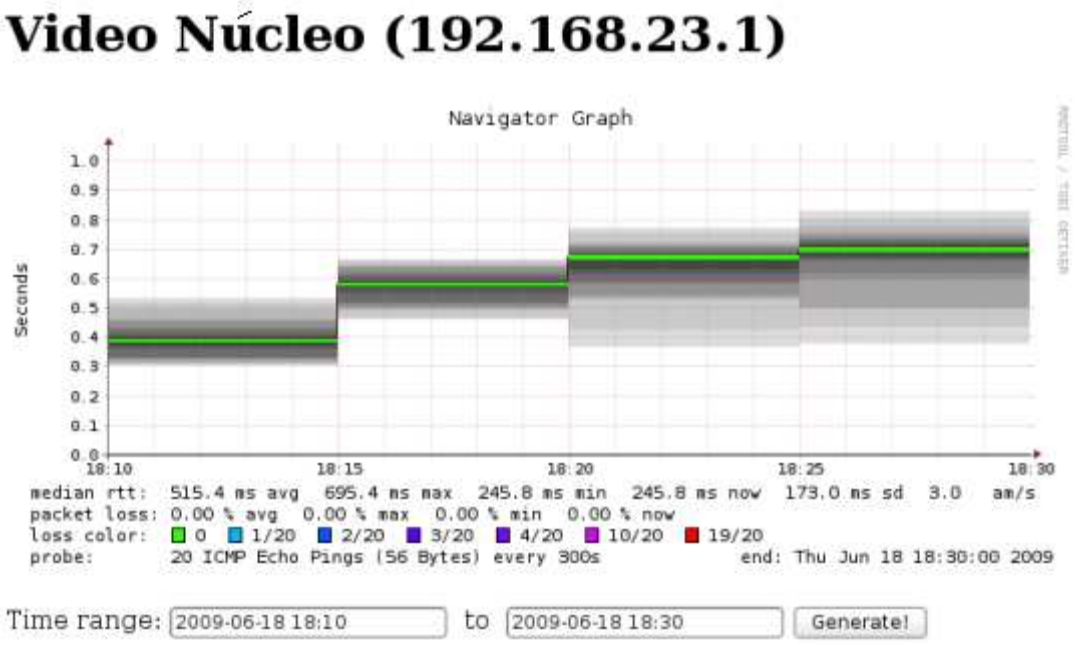

Gráfico 3.3 ***FIFO Smokeping*** **Video Núcleo**

Tomando en cuenta los valores de los parámetros obtenidos a través del mecanismo *FIFO*, por medio de la aplicación de red *Smokeping* y conociendo cuales son los

requerimientos mínimos necesarios recomendados por *Cisco System* (tabla 3.9), para un correcto funcionamiento en aplicaciones de *VoIP* y *TVoIP*, se llega a la conclusión que el mecanismo que suministró un adecuado rendimiento a la red simulada, fue *WFQ "Weighted Fair Queuing"*, pudiéndose apreciar la efectiva disminución en los tiempos de registro de la latencia, *jitter* y *RTT*, además de la reducción en el porcentaje de pérdidas de paquetes. En lo que respecta al uso de la memoria *RAM* y *CPU* en el *router*, todos los mecanismos manifestaron un comportamiento similar, el cual no tiene influencia significativa en el resultado del estudio.

Tabla 3.9 Requerimientos Mínimos QoS

	Voz	Video	Datos
Latencia	≤ 150 ms	150 ms	variable
Jitter	≤ 30 ms	30 ms	variable
Perdidas	≤ 1 %	1 %	variable

En el gráfico se muestran los parámetros obtenidos a través de la aplicación *Smokeping* para mecanismo *WFQ*.

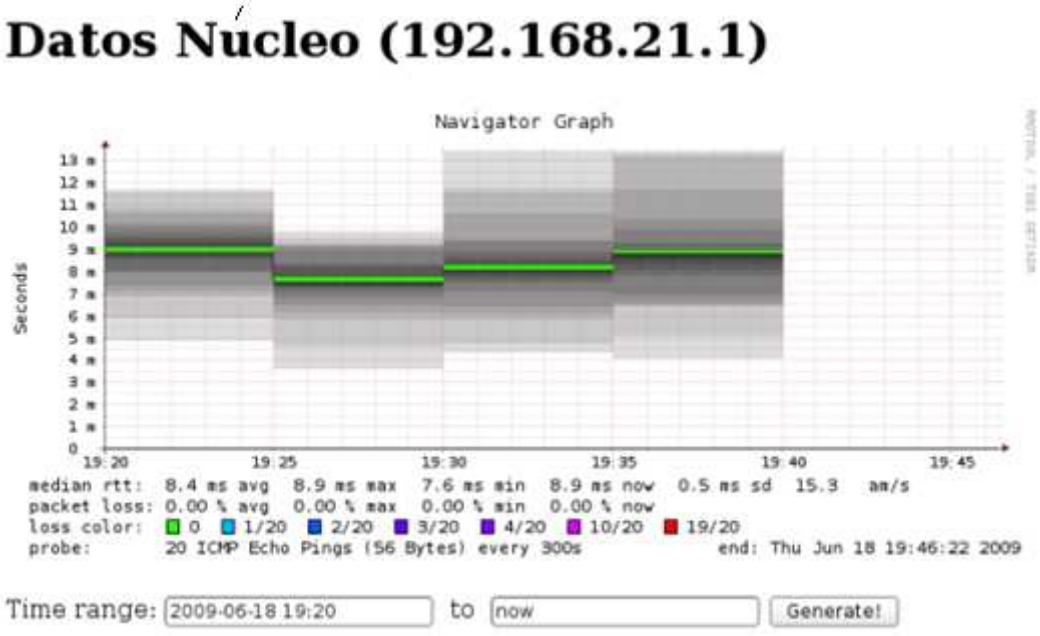

Gráfico 3.4 *WFQ Smokeping* Datos Núcleo

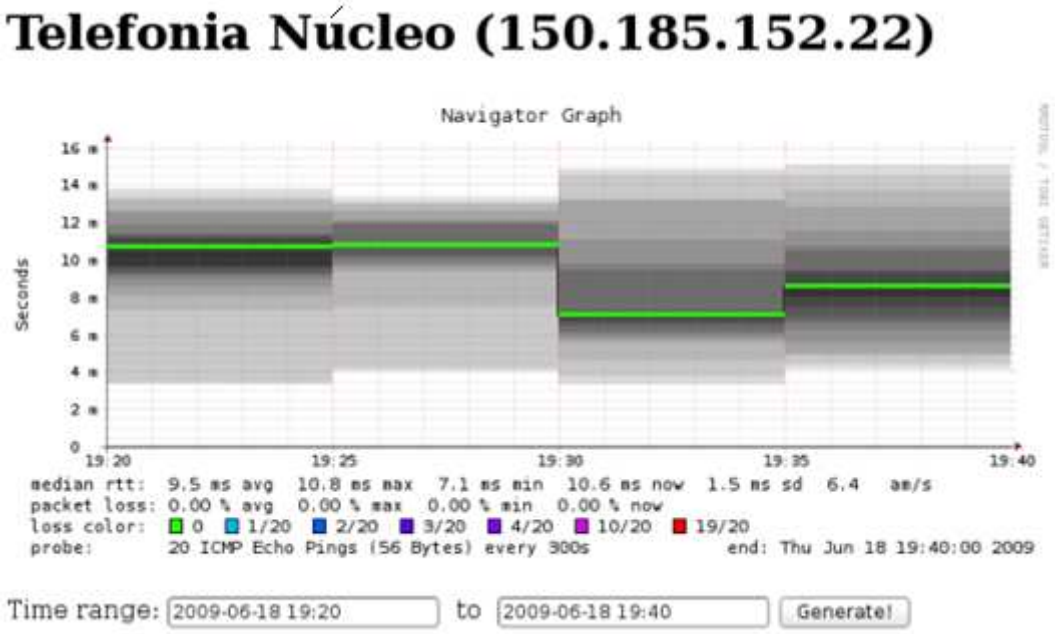

Gráfico 3.5 ***WFQ Smokeping*** **Telefonía Núcleo**

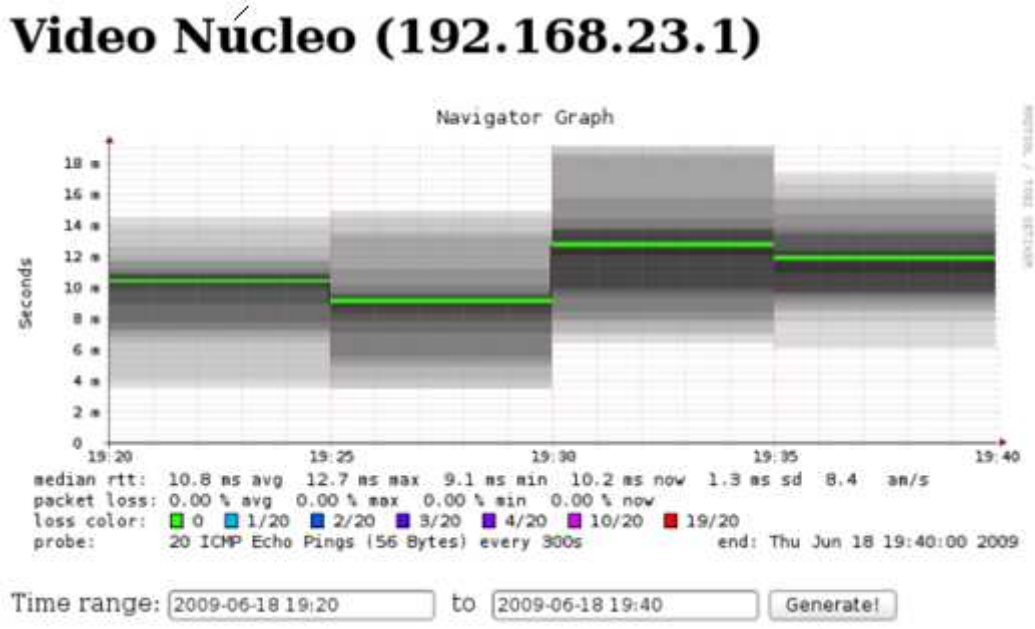

Gráfico 3.6 ***WFQ Smokeping*** **Video Núcleo**

Hay que acotar que durante la prueba, al generar el tráfico en la red (voz, datos y video), la aplicación de video en el cliente (NURR) presentó ciertos inconvenientes, con respecto a fallas en la recepción de la imagen, la cual se detenía por algunos instantes; eso

se presentó al comienzo de la transmisión, lo cual quedo solventado al transcurrir cierto tiempo. Por esta razón se presenta cierta ambigüedad en los resultados obtenidos, en cuanto a los parámetros del video registrados a través del *Smokeping*, los cuales no muestran paquetes descartados, ni aumento en la latencia que indiquen problemas con la transmisión de video.

Estas discrepancia entre los datos recabados por *Smokeping* y el comportamiento que tuvo la aplicación de video, se puede atribuir a la forma en que el monitor de red *Smokeping* recaba los datos del enlace; este envía pequeños paquetes *ICMP* múltiples veces y registra los tiempos de respuesta, o sea, envía los paquetes desde el emisor al receptor y luego espera que estos retornen de nuevo al emisor. Todo este proceso toma un tiempo y este depende de que tan congestionado este el enlace; por lo tanto, si la transmisión de video cesa como consecuencia de un manejo ineficiente de las colas destinadas a video, los paquetes *ICMP* enviados por *Smokeping* no evidenciaran saturación del canal, viajando de este modo en un menor tiempo.

El rendimiento más bajo se manifestó en el mecanismo de administración de congestión *CBWFQ-FIFO "Class Based Weighted Fair Queuing- First In First Out"*, registrándose a través de él, valores bastantes elevados, aún mucho mayores que el mecanismo de congestión por defecto *FIFO.* Los siguientes gráficos muestran los parámetros obtenidos a través de la aplicación *Smokeping*.

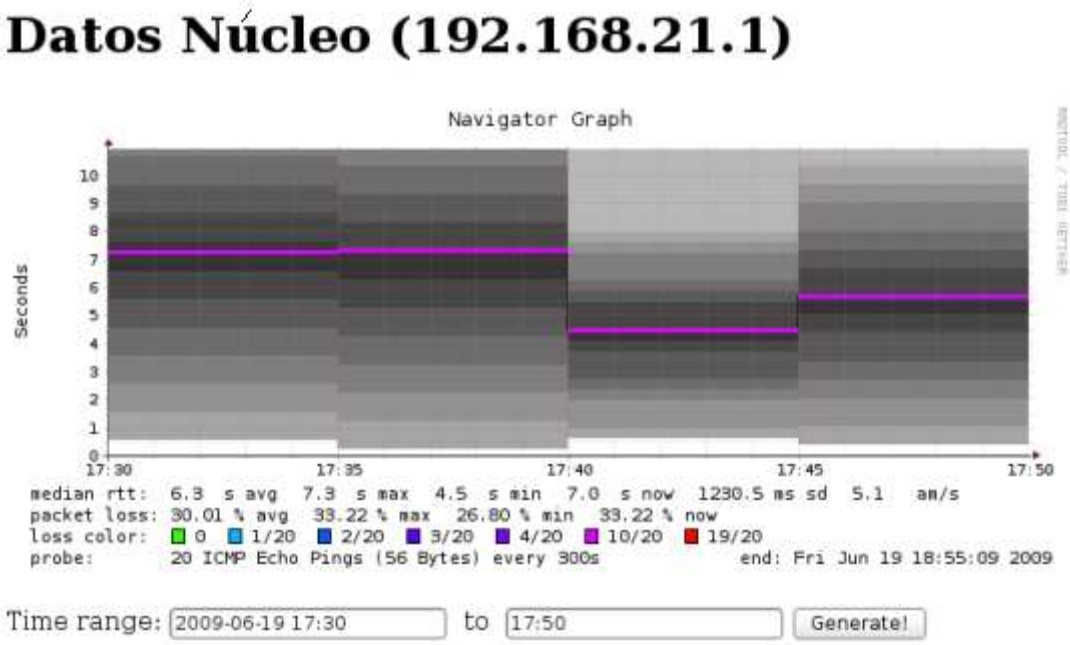

Gráfico 3.7 ***CBWFQ-FIFO Smokeping*** **Datos Núcleo**

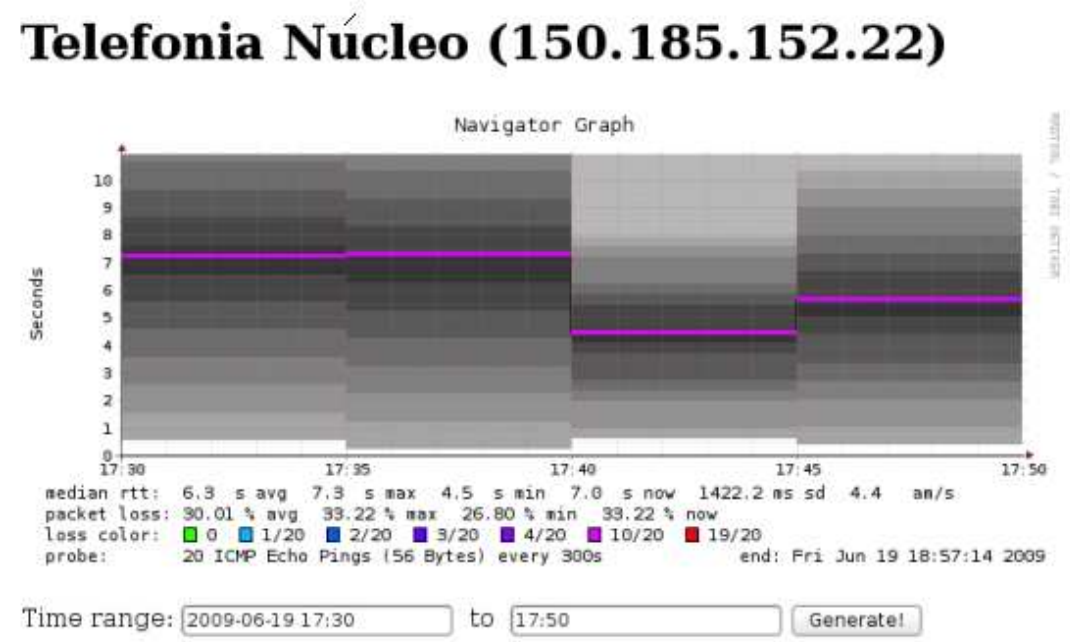

Gráfico 3.8 ***CBWFQ-FIFO Smokeping*** **Telefonía Núcleo**

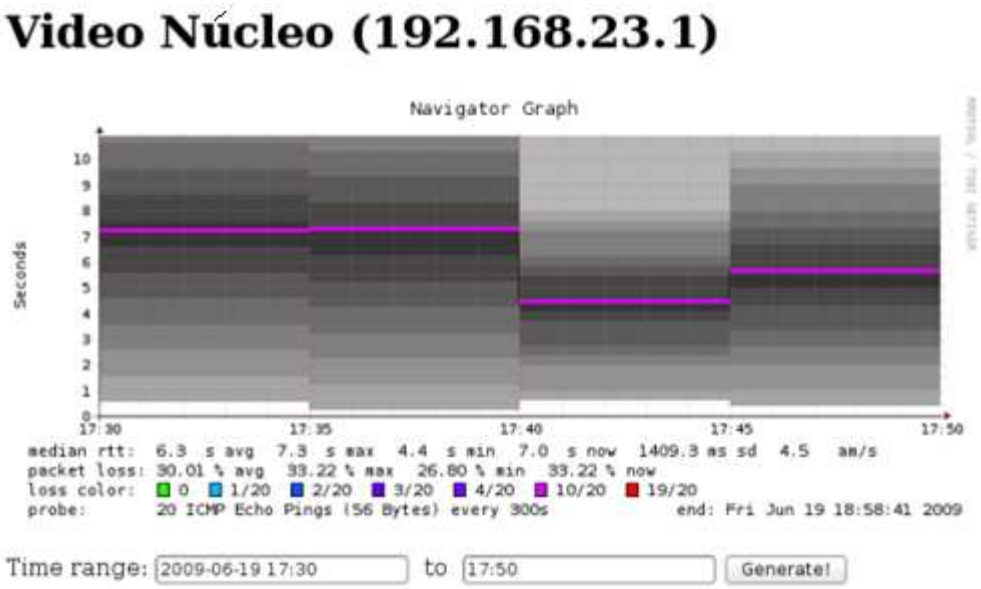

Gráfico 3.9 ***CBWFQ-FIFO Smokeping*** **Video Núcleo**

Es importante destacar que la aplicación del mecanismos *Traffic Shaping and Policing*, también obtuvo buenos registros, en lo que respecta a *jitter*, latencia, *RTT* y perdidas de paquetes; estando todos estos, por debajo de los requerimientos mínimos *QoS* (tabla 3.9) y por debajo también del mecanismo base *FIFO*. Los siguientes gráficos muestran el comportamiento que tuvo este mecanismo en la red.

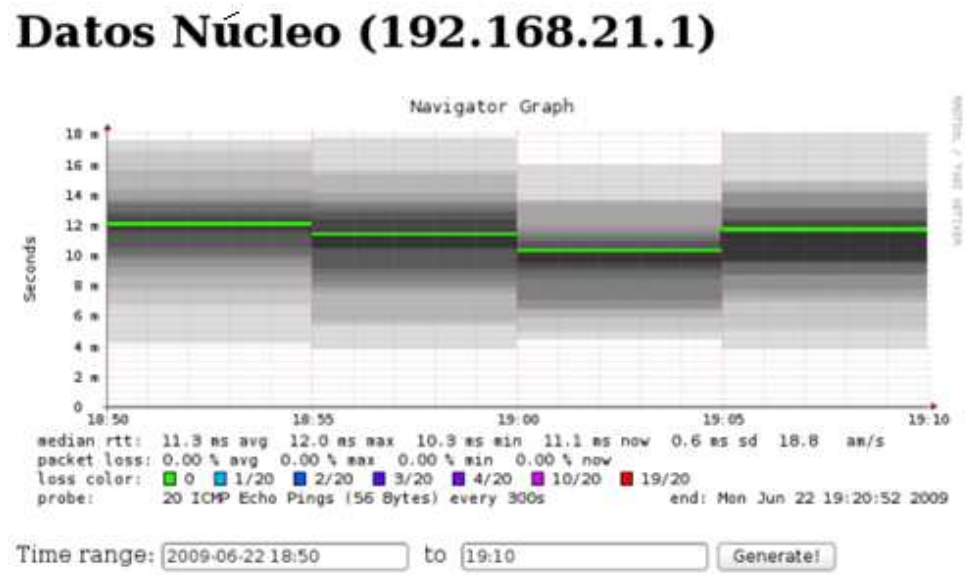

Gráfico 3.10 Traffic ***Shaping and Policing Smokeping*** **Datos Núcleo**

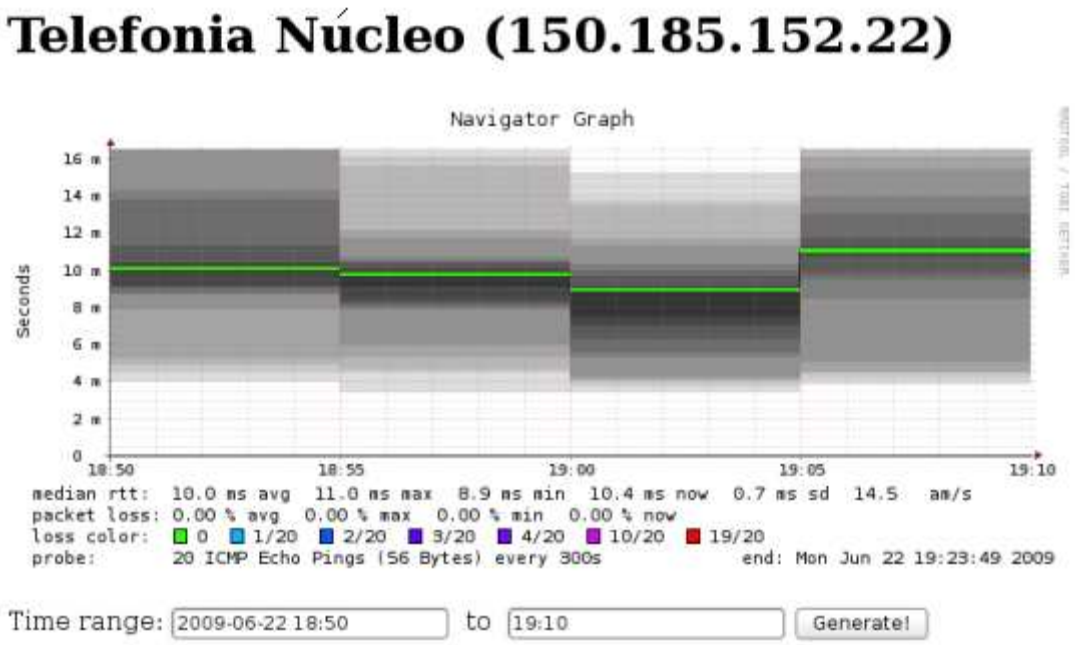

Gráfico 3.11 ***Traffic Shaping and Policing Smokeping*** **Telefonía Núcleo**

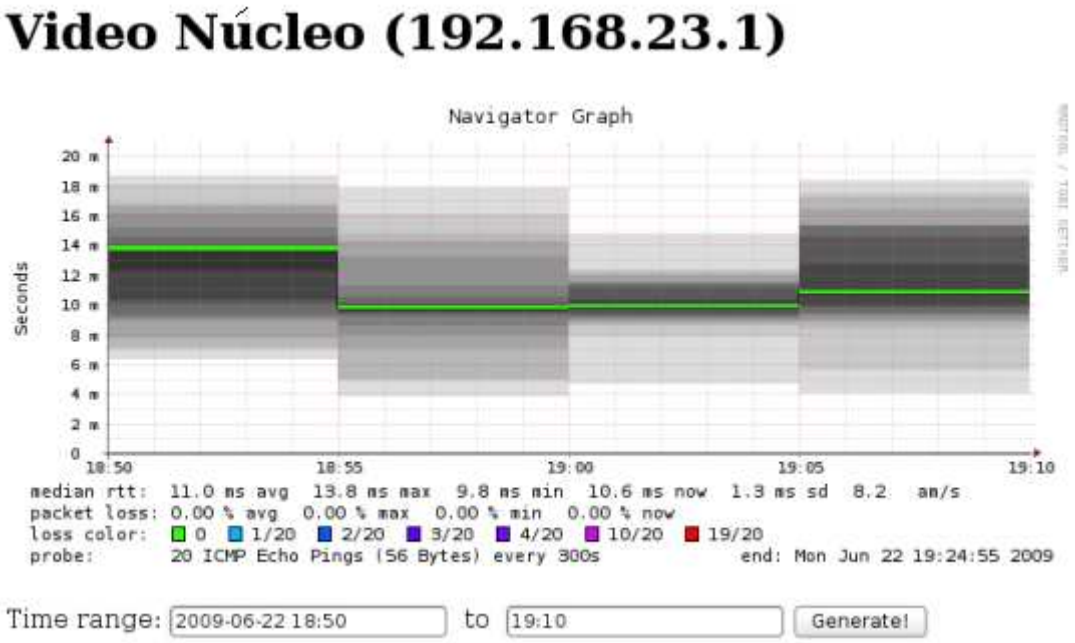

Gráfico 3.12 ***Traffic Shaping and Policing Smokeping*** **Video Núcleo**

Finalmente hay que destacar que de los mecanismos aplicados, los que mejor desempeño presentaron hacia la aplicación de video, fueron *Low Latency Queuing*

“LLQ” e *IP Precedence Based Wred*. Destacándose *LLQ*, situación esta que no se reflejó en los datos obtenidos por *Smokeping*, los cuales mostraron un comportamiento por encima de lo que se debe esperar para este tipo de administrador de congestión, pero dentro de los parámetros correctos (*Jitter*, Latencia, perdidas) para una transmisión de video sobre *IP* (tabla 3.9); sabiendo que es uno de los recomendados para aplicaciones de telefonía *IP* y videoconferencia. Pero esta premisa se pudo comprobar a través del comando “show interface” en el *router* (anexos), el cual logro verificar que *LLQ* presentara el menor número de paquetes descartados en la interfaz de salida, motivo por el cual, la aplicación de video funciono en forma correcta.

CONCLUSIONES

El objetivo principal del trabajo de grado, referente al análisis del funcionamiento de los mecanismos de calidad de servicio (clasificación y marcado de paquetes, manejo de congestión, evasión de congestión, *traffic policing and shaping*) en enlaces con bajo ancho de banda, fue logrado satisfactoriamente, con la salvedad de algunas discrepancias encontradas en los resultados obtenidos con la aplicación de video.

Estas discrepancias entre los datos tabulados recabados por el *Smokeping* y el comportamiento obtenido en tiempo real por la aplicación *VLC* instalada en el cliente de video (NURR), como se explicó en los resultados; puede tener relación con la forma en que *Smokeping* realiza las pruebas y también, al ambiente controlado presente en el laboratorio. De igual modo se comentó que la aplicación de tráfico de video encontró su mejor o más eficiente comportamiento, cuando se aplicó el mecanismo de administración de congestión *LLQ* y el mecanismo de marcado de paquetes y prevención de congestión *IP Precedence Based Wred*, pudiéndose constatar la reputación que tiene *LLQ* con aplicaciones de videoconferencia y telefonía *IP*.

En lo que respecta a la telefonía *IP*, esta no presento problemas significativos en el desarrollo de las pruebas, la transmisión realizada se llevo a cabo sin interrupciones, para todos los mecanismos. Igualmente sucedió con el tráfico de datos, el cual no se vio afectado de manera trascendente. Es importante notar, que a través del flujo de datos, se pudo constatar el comportamiento dinámico de los mecanismos de calidad de servicio, demostrado esto, en el momento en que una de las aplicaciones de tráfico dejaba de funcionar o era detenida, el caudal de datos se apropiaba del ancho de banda dejado por la aplicación que se encontraba funcionando.

RECOMENDACIONES

Partiendo de los resultados obtenidos y tomando en cuenta la importancia que tiene la transmisión de video y audio sin interrupción, como es el caso de una transmisión de *TVoIP*, en la cual el flujo continuo de paquetes se hace imprescindible; se realizan las siguientes sugerencias.

Para un enlace con ancho de banda de 2048 Kbps como el encontrado en el troncal Núcleo Mérida y NURR, la implementación del mecanismo de calidad de servicio *Low Latency Queueing "LLQ"* permitirá obtener resultados óptimos en la transmisión de video, sin desmejorar el flujo de datos y telefonía en ese enlace; consiguiendo de este modo, que una comunicación bidireccional como la planteada anteriormente, se realice de manera constante. Los parámetros utilizados para la configuración de este mecanismo *QoS*, pueden ser los mismos aplicados en este estudio o serian susceptibles de modificaciones, adaptándolos a los requerimientos del caso.

Es importante comparar el monitoreo de la red en igual cantidad de tiempo, antes de la aplicación del mecanismos *QoS* y posterior a esta, para verificar el comportamiento de la misma y de ser necesario, calibrar estos mecanismos para un correcto funcionamiento.

REFERENCIAS

Alestra (2007).Convergencia en Movimiento. Recuperado el 15 de marzo del 2009, de http://www.alestra.com.mx/alestra_htmls/empresa/hablemos/hablemos50-4.htm

Álvarez M., Sebastián A. & Gonzales V. Agustín J. (2004).Estudio y Configuración de Calidad de Servicio para Protocolos IPV4 e IPV6 en una Red de Fibra Óptica WDM. Recuperado el 11 de Marzo de 2009, de http://www.scielo.cl/pdf/rfacing/v13n3/art15.pdf

Arribas V., Francisco (2009).Cisco Certified Internetwork Expert CCIE, V2. Recuperado el 02 de Marzo de 2009, de http://www.francisco-valencia.es/Documentos/CCIE.pdf

Balliache, Leonardo (2009).Servicios QoS. Recuperado el 05 de Abril de 2009, de http://www.opalsoft.net/qos/Spanish-QOS.htm

Baños C., María D. (2009). Nuevas Técnicas de Control y Gestión de Tráfico en Internet para Proporcionar Calidad de Servicio Extremo a Extremo. Recuperado el 18 de Marzo de 2009, de http://repositorio.bib.upct.es/dspace/bitstream/10317/795/1/mdcb.pdf

Berlingeri, Adriana, Flores Mauro & Alonzo Maximiliano (2009). Seguridad en Redes Convergentes. Recuperado el 10 de Abril de 2009, de http://www.deloitte.com/dtt/article/0,1002,cid%253D222328,00.html

Camp, Ken (2009). The Definitive Guide to Converged Network Management. Recuperado el 20 de Marzo de 2009, de http://nexus.realtimepublishers.com/dgcnm.php

Carlos Vicente (2009). Análisis de Rendimiento. Recuperado el 01 de Abril de 2009, de http://ws.edu.isoc.org/workshops/2008/walc/presentaciones/performance_concepts.pdf

Cisco System (2004). Implementing Cisco Quality of Service “QoS” V2.1, pag 1-20.

Cisco System (2009). Quality of Service (QoS).Recuperado el 15 de Marzo de 2009, de http://www.cisco.com/en/US/products/ps6558/products_ios_technology_home.html

Delgado H., Julio C. (2009). Algoritmos de gestión de tráfico en redes de servicios múltiples: una aproximación. Recuperado el 01 de Abril de 2009, de http://www.uvmnet.edu/investigacion/episteme/numero6-06/reportes/a_algoritmos.asp

Gerometta, Oscar (2009). Mis Libros de Networking . Recuperado el 10 de Junio de 2009, de http://librosnetworking.blogspot.com/2008/04/elementos-bsicos-de-qos.html

Global Crossing (2009). Rendimiento de Red IP. Recuperado el 16 de Marzo de 2009, de http://www.globalcrossing.com/sp/network/network_performance_methodology_sp.aspx

H3C (2009). QoS Introduction. Recuperado el 18 de Abril de 2009, de http://www.h3c.com/portal/Products___Solutions/Technology/QoS/Technology_Introduction/200701/195599_57_0.htm

IRISTEL (2009). Recuperado el 22 de Marzo de 2009, de http://www.iristel.ca/support.php/act/view/ids/43

Llamas Ricardo, A. (2009). Estudio e Implementación de Mecanismos de Calidad de Servicio sobre una Arquitectura de Servicios Diferenciados. Recuperado el 03 de Marzo de 2009, de http://repositorio.bib.upct.es:8080/dspace/bitstream/10317/184/1/pfc908.pdf

Mejías Fajardo, Ángela M. (2004). Redes Convergentes. Recuperado el 09 de Marzo de 2009, de http://redalyc.uaemex.mx/redalyc/pdf/911/91101407.pdf

Montes de Oca, Faustino (2009). Redes de Computadora, Control de Congestión, QoS. Recuperado el 18 de Marzo de 2009, de http://www.ie.itcr.ac.cr/faustino/Redes/Clase10/QoS.pdf

Montevideo Libre (2007). Monitoreo. Recuperado el 05 de Abril de 2009, de http://www.montevideolibre.org/manuales:libros:wndw:capitulo_6:monitoreo

Murazzo, María A. (2001). Interoperabilidad de los Mecanismos de QoS en Internet. Recuperado el 03 de Marzo de 2009, de http://www.unsj-cuim.edu.ar/portalzonda/Congreso/papers/2001/UI22.pdf

Oetiker's, Tobi (2009). Tobi Oetiker's Toolbox. Recuperado el 01 de Mayo de 2009, de http://tobi.oetiker.ch/hp/

Pontificia Universidad Católica de Chile, Escuela de Ingeniería (2009). Calidad de Servicio (QoS). Recuperado el 17 de Marzo de 2009, de www2.ing.puc.cl/~iee3542/amplif_4.ppt

RFC Editor (2009). Request for Comments. Recuperado el 20 de Febrero de 2009, de http://www.rfc-editor.org/

The Internet Engineering Task Force (2009). Internet Draft. Recuperado el 20 de Febrero de 2009, de http://www.ietf.org/ID.html

Tovar, Alejandro T. (2009). Gestión del Nivel de Servicio y QoS. Recuperado el 09 de Marzo de 2009, de http://calypso.unicauca.edu.co/gntt/atoledo/QoS/05-Gestion%20del%20nivel%20de%20servicio%20y%20de%20la%20QoS-2.pdf

Universidad Yacambu (2009).Capítulo II. Recuperado el 02 de Mayo de 2009, de http://es.geocities.com/yvillasana2005/teg/Tesis/TrabajoEspecialdeGrado/CapituloII.htm

Wikipedia (2009). Internet Draft. Recuperado el 20 de Febrero de 2009, de http://en.wikipedia.org/wiki/Internet_Draft

Wikipedia (2009). Latencia. Recuperado el 28 de Marzo de 2009, de http://es.wikipedia.org/wiki/Latencia

Wikipedia (2009). Protocolo Internet. Recuperado el 19 de Febrero de 2009, de http://es.wikipedia.org/wiki/Protocolo_de_Internet

Wikipedia (2009).Protocolo Internet. Recuperado el 05 de Marzo de 2009, de http://es.wikipedia.org/wiki/Protocolo_de_Internet

Yaguez G., Javier. (2009). Aplicaciones Multimedia en Tiempo Real en Internet. Recuperado el 15 de Abril de 2009, de http://halley.ls.fi.upm.es/~jyaguez/pdfs/TRANSCUARTOrtpvoipfebrero2007.pdf

ANEXOS

Tabla 3.10 Resumen de Parámetros Registrados en las Pruebas de Laboratorio (Mérida)

Tipos de Mecanismos QoS	Clases de Trafico	Latencia (ms)	Jitter (ms)	Perdidas Paquetes (%)	RTT (ms)	Uso de CPU (%)	Uso Memoria RAM (%)	Ancho de Banda (Kbps)
FIFO	Datos	0.21	0.54	0	0.42	7	17.46	2048
	Telefonía	0.42	1.54	0	0.84	7	17.46	2048
	Video	0.26	0.75	0	0.51	7	17.46	2048
WFQ	Datos	0.21	0.6	0	0.42	7	17.43	2048
	Telefonía	0.43	0.9	0	0.85	7	17.43	2048
	Video	0.24	0.84	0	0.49	7	17.43	2048
CBWFQ WFQ	Datos	0.23	0.74	0	0.46	9	17.52	2048
	Telefonía	0.44	1.47	0	0.88	9	17.52	2048
	Video	0.24	0.78	0	0.49	9	17.52	2048
CBWFQ FIFO	Datos	0.21	0.65	0	0.41	9	20.77	2048
	Telefonía	0.42	0.85	0	0.83	9	20.77	2048
	Video	0.23	0.68	0	0.46	9	20.77	2048
DSCP Based Wred	Datos	0.21	0.76	0	0.43	9	20.77	2048
	Telefonía	0.43	1.18	0	0.86	9	20.77	2048
	Video	0.24	0.81	0	0.48	9	20.77	2048
IP Precedence Based Wred	Datos	0.23	0.55	0	0.46	9	20.79	2048
	Telefonía	0.43	1.53	0	0.86	9	20.79	2048
	Video	0.23	0.83	0	0.47	9	20.79	2048
Traffic Policing and Shaping	Datos	0.21	0.47	0	0.42	9	20.77	2048
	Telefonía	0.42	1.06	0	0.83	9	20.77	2048
	Video	0.23	0.74	0	0.46	9	20.77	2048
LLQ	Datos	0.22	0.63	0	0.44	9	20.67	2048
	Telefonía	0.43	1.47	0	0.86	9	20.67	2048
	Video	0.25	0.76	0	0.5	9	20.67	2048

Tabla 3.11 Resumen de Parámetros Registrados en las Pruebas de Laboratorio (NURR)

Tipos de Mecanismos QoS	Clases de Trafico	Latencia (ms)	Jitter (ms)	Perdidas Paquetes (%)	RTT (ms)	Uso de CPU (%)	Uso Memoria RAM (%)	Ancho de Banda (Kbps)
FIFO	Datos	256.9	310	6	513.7	4	16.05	2048
	Telefonía	254.8	330	8	509.5	4	16.05	2048
	Video	257.7	390	0	515.4	4	16.05	2048
WFQ	**Datos**	**4.20**	**1.7**	**0**	**8.4**	**4**	**16.06**	**2048**
	Telefonía	**4.75**	**2.3**	**0**	**9.5**	**4**	**16.06**	**2048**
	Video	**5.4**	**3**	**0**	**10.80**	**4**	**16.06**	**2048**
CBWFQ WFQ	Datos	39.25	13	0	78.50	5	16.10	2048
	Telefonía	44.20	46	0	88.40	5	16.10	2048
	Video	47.75	15	0	95.50	5	16.10	2048
CBWFQ FIFO	**Datos**	**3150**	**3100**	**30.01**	**6300**	**4**	**15.89**	**2048**
	Telefonía	**3150**	**3100**	**30.01**	**6300**	**4**	**15.89**	**2048**
	Video	**3150**	**3100**	**30.01**	**6300**	**4**	**15.89**	**2048**
DSCP Based Wred	Datos	43.80	53	0	87.60	4	15.87	2048
	Telefonía	30.60	82	0	61.20	4	15.87	2048
	Video	41.10	124	0	82.20	4	15.87	2048
IP Precedence Based Wred	Datos	49.55	21	0	99.10	4	15.90	2048
	Telefonía	45.50	41	0	91	4	15.90	2048
	Video	47.30	94	0	94.60	4	15.90	2048
Traffic Policing and Shaping	**Datos**	**5.65**	**5.2**	**0**	**11.30**	**4**	**15.88**	**2048**
	Telefonía	**5**	**0.1**	**0**	**10**	**4**	**15.88**	**2048**
	Video	**5.5**	**0.7**	**0**	**11**	**4**	**15.88**	**2048**
LLQ	Datos	45.15	128	0	90.30	4	15.86	2048
	Telefonía	43.95	47	0	87.90	4	15.86	2048
	Video	46.20	40	0	92.40	4	15.86	2048

Tabla 3.12 Tabla Comparativa Entre Mecanismo Más Eficiente y Menos Eficiente

Tipos de Mecanismos QoS	Clases de Trafico	Latencia (ms)	Jitter (ms)	Perdidas Paquetes (%)	RTT (ms)	Uso de CPU (%)	Uso Memoria RAM (%)	Ancho de Banda (Kbps)
CBWFQ FIFO	**Datos**	**3150**	**3100**	**30.01**	**6300**	**4**	**15.89**	**2048**
	Telefonía	**3150**	**3100**	**30.01**	**6300**	**4**	**15.89**	**2048**
	Video	**3150**	**3100**	**30.01**	**6300**	**4**	**15.89**	**2048**
WFQ	**Datos**	**4.20**	**1.7**	**0**	**8.4**	**4**	**16.06**	**2048**
	Telefonía	**4.75**	**2.3**	**0**	**9.5**	**4**	**16.06**	**2048**
	Video	**5.4**	**3**	**0**	**10.80**	**4**	**16.06**	**2048**

Comando de Interfaz (router) "*Show Interface*" para paquetes descartados.

FIFO

```
Last clearing of "show interface" counters 01:31:58
Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 1336
Queueing strategy: fifo
Output queue: 26/40 (size/max)
5 minute input rate 194000 bits/sec, 206 packets/sec
5 minute output rate 1807000 bits/sec, 247 packets/sec
```

WFQ

```
Last clearing of "show interface" counters 00:44:40
Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 1166
Queueing strategy: weighted fair
Output queue: 51/1000/64/1166 (size/max total/threshold/drops)
   Conversations  1/5/256 (active/max active/max total)
   Reserved Conversations 0/0 (allocated/max allocated)
   Available Bandwidth 1500 kilobits/sec
5 minute input rate 123000 bits/sec, 159 packets/sec
5 minute output rate 1754000 bits/sec, 205 packets/sec
```

CBWFQ-WFQ

```
GW-Merida#sh queue serial 0/0/0
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 1227
  Queueing strategy: Class-based queueing
  Output queue: 57/1000/64/1227 (size/max total/threshold/drops)
     Conversations  2/7/256 (active/max active/max total)
     Reserved Conversations 2/2 (allocated/max allocated)
     Available Bandwidth 710 kilobits/sec
```

CBWFQ-FIFO

```
Last clearing of "show interface" counters 00:09:56
Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 765
Queueing strategy: Class-based queueing
Output queue: 45/1000/64/765 (size/max total/threshold/drops)
   Conversations  1/9/256 (active/max active/max total)
   Reserved Conversations 2/3 (allocated/max allocated)
   Available Bandwidth 710 kilobits/sec
5 minute input rate 113000 bits/sec, 139 packets/sec
5 minute output rate 1937000 bits/sec, 204 packets/sec
```

DSCP-Based WRED

```
Last clearing of "show interface" counters 00:16:33
Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 546
Queueing strategy: Class-based queueing
Output queue: 72/1000/64/546 (size/max total/threshold/drops)
   Conversations  2/14/256 (active/max active/max total)
   Reserved Conversations 2/2 (allocated/max allocated)
   Available Bandwidth 710 kilobits/sec
5 minute input rate 119000 bits/sec, 155 packets/sec
5 minute output rate 1942000 bits/sec, 206 packets/sec
```

IP Precedence-BasedWRED

```
Last clearing of "show interface" counters 00:21:11
Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 599
Queueing strategy: Class-based queueing
Output queue: 30/1000/64/599 (size/max total/threshold/drops)
   Conversations  1/23/256 (active/max active/max total)
   Reserved Conversations 2/2 (allocated/max allocated)
   Available Bandwidth 710 kilobits/sec
5 minute input rate 114000 bits/sec, 141 packets/sec
5 minute output rate 1940000 bits/sec, 207 packets/sec
```

Traffic Policing and Shaping

```
Last clearing of "show interface" counters 00:29:24
Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 2740
Queueing strategy: Class-based queueing
Output queue: 56/1000/64/2740 (size/max total/threshold/drops)
   Conversations  2/9/256 (active/max active/max total)
   Reserved Conversations 0/0 (allocated/max allocated)
   Available Bandwidth 2000 kilobits/sec
5 minute input rate 123000 bits/sec, 159 packets/sec
5 minute output rate 1945000 bits/sec, 205 packets/sec
```

LLQ

```
Last clearing of "show interface" counters 00:12:11
Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 458
Queueing strategy: Class-based queueing
Output queue: 52/1000/64/458 (size/max total/threshold/drops)
   Conversations  2/8/256 (active/max active/max total)
   Reserved Conversations 1/1 (allocated/max allocated)
   Available Bandwidth 710 kilobits/sec
5 minute input rate 120000 bits/sec, 156 packets/sec
5 minute output rate 1940000 bits/sec, 206 packets/sec
```

GLOSARIO DE TÉRMINOS

A

ARPA Advance Reseach Proyecs Agency

ARPANET Advance Reseach Projects Agency Network

ACL Access List (Lista de Acceso)

AF Assured Forwarding (Envió Asegurado)

ATM Asynchronous Transfer Mode (Modo de Transferencia Asíncrona)

B

Bit Binary Digit (Digito Binario, unidad mínima de almacenamiento)

Byte Unidad de Información compuesta por 8 bits

C

CPU Central Processing Unit (Unidad Central de Procesamiento)

Colas Agrupación de Paquetes de Datos

CQ Custom Queuing (Encolamiento Personalizado)

CBWFQ Class Based WFQ (WFQ Basado en Clases)

D

DiffServ Differentiated Service (Servicios Diferenciados)

DSCP DiffServ Code Point (Punto de Código de Servicios Diferenciados)

Dynamic Queues Colas Dinámica

F

Frame Relay Técnica de Transmisión de Datos

FTP File Transfer Protocol (Protocolo de Transferencia de Archivos)

Frame Trama de dato

FIFO First In, First Out (Primero en Entrar, Primero en Salir)

H

HTTP Hiper Text Transfer Protocol (Protocolo de Transferencia de Híper Texto)

I

IOS Internetwork Operating System (Sistema Operativo de Interconexión)

IP Internet Protocol (Protocolo de Internet)

IHL IP Header Length (Longitud del Encabezado IP)

IETF Internet Engineering Task Force (Grupo de Trabajo en Ingeniería de Internet)

IntServ Integrated Service (Servicios Integrados)

ICMP Internet Control Message Protocol (Protocolo de Mensajes de Control de Internet)

L

LSB Less Significant Byte (Byte Menos Significativo)

LLQ Low Latency Queuing (Encolamiento de Baja Latencia)

M

MRTG Multi Router Traffic Grapher

MSB Most Significant Byte (Byte Mas Significativo)

N

NetFlow Protocolo de Red para Monitoreo de Trafico

NarrowBand Banda Angosta, Poco ancho de Banda

NCP Network Control Protocol (Protocolo de Control de Red)

P

PHB Per Hop Behavior (Comportamiento por Salto)

PQ Priority Queuing (Encolamiento Prioritario)

Paquete de Datos Unidad de Transporte de Información

Q

QoS Quality of Service (Calidad de Servicio)

R

RFC Request for Comments

RTT Round Trip Delay Time

RRDTool Round Robin Database Tool

Router Enrutador de Red

RSVP Resource Reservation Protocol (Protocolo de Reservación de Recursos)

RED Random Early Detection (Detección Temprana de Congestión)

Reserved Queues Encolamiento Reservado

S

Switch Conmutador de Red

T

TOS Type of Service (Tipo de Servicio)

Threshold Discard Umbral de Descarte de Paquetes

TCP Transmission Control Protocol (Protocolo de Control de Transmisión)

U

UDP User Datagram Protocol (Protocolo de Datagrama de Usuario)

V

VoIP Voice over IP (Voz sobre Protocolo IP)

VLAN Virtual Local Ares Network (Red de Área Local Virtual)

VLC Aplicación Multimedia de Video

W

WAN Wide Area Network

WFQ Weighted Fair Queuing (Planificacion Equitativa por Pesos)

WRED Weighted Random Early Detection (Deteccion Temprana por pesos)

Xlite Aplicación para Telefonía IP