

## ANÁLISIS DE LAS CARACTERÍSTICAS DE FUNCIONAMIENTO ENTRE LOS SERVICIOS INTEGRADOS Y LOS SERVICIOS DIFERENCIADOS EN REDES DE COMUNICACIONES

MSc. N.Y. Gelvez García <sup>1</sup>

MSc. D. A. López Sarmiento <sup>2</sup>



### NANCY YANETH GELVEZ GARCÍA<sup>1</sup>

Ingeniera de Sistemas, Magister en Ciencias de Información y las Comunicaciones de la Universidad Distrital "Francisco José de Caldas", Docente Planta de Ingeniería de Sistemas de la ECCI, perteneciente al grupo de Investigación INDETECA. . nayag24@hotmail.com

### DANILO ALFONSO LÓPEZ SARMIENTO<sup>2</sup>

Ingeniero Electrónico, Magister en Teleinformática, Docente Planta de la Facultad de Ingeniería de la Universidad Distrital "Francisco José de Caldas", perteneciente al grupo de Investigación de Internet Inteligente.  
dalopezs@udistrital.edu.co



### RESUMEN

Los sistemas informáticos se basan en una red de datos que deben soportar una gama de aplicaciones cada vez más amplia por lo cual es importante emplear arquitecturas que facilitan el tráfico de datos dependiendo de la naturaleza de ellos, siendo las arquitecturas que se comparán en éste documento conocidas como servicios diferenciados e integrados. Cada arquitectura ofrece diferentes alternativas para garantizar una calidad de servicio, así que se observará el rendimiento de una y otra mediante la implementación de dos topologías sobre NS2 con el fin de verificar las bondades que ofrece el servicio diferenciado ante el servicio integrado en aspectos como jitter y delay.

**Palabras claves:** Calidad de servicio (QoS), NS2, RSVP, Servicios Diferenciados (Diffserv), Servicios integrados (Intserv).

## ABSTRACT

Computer systems are based on a data network that must support a range of increasingly wide applications so it is important to implement architectures that provide data traffic depending on the nature of them, with architectures that are compared in this paper known as differentiated services and integrated. Each architecture offers several alternatives to ensure quality of service, so performance is observed over and over through the implementation of two topologies on NS2 to verify the advantages offered differentiated service to the integrated services in areas such as jitter and delay.

**Key Words:** Quality of Service (QoS), NS2, RSVP, Differentiated Services (Diff-Serv), Integrated services (IntServ).

## 1. INTRODUCCIÓN

El desarrollo de las telecomunicaciones y la informática generó notables y crecientes cambios que formaron toda una rama que fusionó a estas dos áreas, naciendo la telemática [1]. Entonces se empieza a hablar de redes de datos y por consiguiente del tráfico que estas soportan. Este tráfico crece exponencialmente, pues la sociedad actual demanda nuevos servicios que requieren progresivamente mejor aprovechamiento de recursos y calidad de servicio [2]. Además genera una tendencia creciente, hacia la convergencia de una sola red, basada en el protocolo de internet IP [3]. IP se diseñó con una filosofía funcional: "hay que llegar sea como sea", lo que se traduce en que los paquetes IP llegaran a su destino en cuanto exista alguna forma de hacerlo [4]. A este tipo de servicio se le denomina "best-effort", o mejor esfuerzo, que ve imposibilitado garantizar el envío de

los paquetes, pues pueden ser encaminados por distintas rutas y a veces se pierden en nodos intermedios debido a la congestión de la red pero dada la gran cantidad de aplicaciones transportadas, existe un cierto interés en la implementación de mecanismo de calidad de servicio QoS [5].

Se realizan entonces simulaciones sobre el entorno NS2.26, teniendo como punto de partida una topología en la que se garanticen óptimas condiciones de comunicación, esto es, que el ancho de banda soportado por el medio sea suficiente y posteriormente una en la que el ancho de banda soportado por la topología sea inferior a lo requerido.

Para realizar el análisis correspondiente se presenta lo que existe en la frontera del conocimiento, se explica en detalle el funcionamiento de los servicios integrados

y diferenciados y finalmente se desarrolla la comparación entre ellos con el fin de determinar cuál de los dos presenta un desempeño superior.

## 2. ESTADO DEL ARTE

Las investigaciones que se han realizado durante años sobre los servicios integrados y diferenciados han permitido avanzar en las aplicaciones de variados dispositivos con el fin de suministrar mejor calidad de servicio y ampliar el campo de aplicación de los mismos. Algunas de las investigaciones más destacadas que se han realizado durante los últimos diez años se han convertido en punto de referencia para nuevas investigaciones en las que se ha profundizando en el conocimiento de estos servicios como es el caso de el documento "Comparación y utilización de los límites de servicios integrados y servicios diferenciados" [9].

Otros investigadores se han especializado en profundizar acerca de calidad de servicio realizando estudios sobre Ingeniería de tráfico en redes MPLS [10], Propuesta de Arquitectura MPLS/diffserv para proveer mecanismos de calidad de servicio (QoS) en el transporte de la telefonía IP [11], Propuesta de arquitectura de QoS en entorno inalámbrico 802.11e basado en diffserv con ajuste dinámico de parámetros [12], estudio y configuración de calidad de servicio para protocolos IPV4 e IPV6 en una red de fibra óptica WDM [13], Integración de MPLS y diffserv en una arquitectura para la provisión de calidad de servicio (QoS) [14].

## 3. MARCO TEÓRICO

Desde el punto de vista de una red la calidad de servicio (QoS) puede ser medida según las propiedades que presente, en términos de parámetros o recursos óptimos, que la red ofrece al usuario o a la aplicación. No hay una metodología concreta a seguir que nos proporcione una QoS óptima, para cada una de las aplicaciones. Lo que ocurre es que hay una serie de métricas que permiten analizar el rendimiento de la red. Algunos de estos parámetros de medida son:

- **RETARDO o DELAY:** tiempo que emplea un paquete para viajar desde la fuente al destino.
- **VARIACION DEL RETARDO O JITTER:** variación en cuanto a la cantidad de latencia entre paquetes que se reciben. Es un parámetro relevante en estudios de QoS con servicios multimedia.. Por tanto, su dimensionamiento correcto es fundamental.
- **THROUGHPUT:** cantidad de datos útiles del usuario recibidos en una comunicación dividida por la información transmitida. Para su cálculo hay muchos aspectos a tener en cuenta como: cabeceras de paquetes, paquetes de señalización, tiempos de espera, colisiones, retransmisiones de paquetes, etc.
- **PAQUETES PERDIDOS:** Hay muchas razones que pueden causar la pérdida de paquetes en entornos inalámbricos: enlaces de red saturados, colisiones,

**rotura de enlaces, etc. Normalmente se presenta como el porcentaje de paquetes perdidos dividido por el número total de paquetes enviados por la fuente.**

### 3.1 INTSERV.

Es un protocolo "valiente" que afronta cambiar internet para establecer calidad de servicio integral, o sea, entre extremos. No entra en ingeniería de tráfico, sino que se acopla a las rutas IP descubiertas por los algoritmos de encaminamiento tradicionales. Lo que hace es solicitar una cierta calidad de servicio y, si la red la admite, aprovecharla [18]. Intserv tiene como adicional al servicio "best effort", la posibilidad de soportar servicio garantizado y servicio controlado de carga [19]. La idea principal en la que basa intserv su funcionamiento, es el de flujo. Si los paquetes comparten recursos y una dirección IP de destino, así como los puertos de origen y destino, entonces se puede asumir que esos paquetes son todos parte de un aplicación stream de datos seguidos entre fuente y destino, con todo lo que ello implica [20].

### 3.2 DIFFSERV

En esta arquitectura, los paquetes son clasificados y marcados para recibir un trato particular en cuanto al envío en cada salto. Clasificación, marcado, política y operaciones de acondicionamiento, necesitan sólo ser implementadas en los bordes de la red o en los hosts.

Esta arquitectura logra escalabilidad al implementar complejas funciones de clasificación y condicionamiento sólo en los nodos del borde de la red, y aplicando conductas por salto a los agregados del tráfico que han sido apropiadamente marcados usando el campo DS en las cabeceras de IPv4.

Esta arquitectura sólo provee servicio diferenciado en una dirección del flujo de tráfico y es por ende asimétrica [22].

## 4. DESARROLLO

Se pretende construir una topología en la que existan 3 diferentes flujos de tráfico a través de una ruta en la que se encuentren 3 routers de por medio como se muestra en la figura 1, donde se pretende generar flujos entre los nodos 0-3, 1-4 y 2-5, siendo los nodos 0, 1 y 2 nodos emisores, 3, 4 y 5 nodos receptores y los nodos 6, 7 y 8 como enrutadores. Primero se implementará un escenario en el que se garantice un ancho de banda suficiente para soportar todos los flujos y luego un segundo escenario en el que la red troncal no sea capaz de soportar todos los flujos, generando congestión y pérdida de paquetes en la comunicación. Por cada uno de estos escenarios se realizarán 3 pruebas. La primera de ellas consiste en generar tráfico sobre la red sin implementar calidad de servicio. La segunda consiste en implementar la arquitectura de servicios diferenciados dando igual calidad de servicios a los 3 flujos de datos, mientras que para la tercera prueba, dándole prioridad a uno de los flujos de datos.

Todas las simulaciones se trabajarán con un mismo tráfico entre nodos y por ello todas ellas tendrán la siguiente estructura en cuanto a tráfico se refiere. El tráfico que se pretende evaluar será soportado por el protocolo de datagrama de usuario o UDP, el cual no está orientado a la conexión, por lo que con este no se proporciona ningún tipo de control de errores ni de flujo. La simplicidad del UDP hace que sea ideal para aplicaciones que requieren pocos retardos (por ejemplo, aplicaciones en tiempo real como pueden ser voz y video), lo que supone una necesidad de establecer para ellos, políticas que permitan a la mayoría, o bien, a la totalidad de los datos enviados, alcanzar con éxito su destino.

Para iniciar dicho protocolo se identifica y crea un agente UDP, se asigna a un nodo transmisor de datos e igualmente se identifica y crea un agente receptor de datos asociado a un nodo receptor, para posteriormente, conectar los agentes y generar así un flujo con una identificación propia. El agente receptor puede ser de tipo null o lossmonitor. Implementar el segundo, tiene como ventaja conocer la cantidad de bytes que ha recibido, la cantidad de paquetes que se han perdido y recibido y por ello se seleccionó este agente receptor.

Donde `udp1` y `null1` son los identificadores de los agentes de transmisión y recepción respectivamente y `1` el identificador que se le da al flujo (`$udp1 set fid_ 1`). Todos los flujos de datos a implementar serán de este tipo, así que basta con realizar un procedimiento igual para los flujos entre los nodos `1` y `4` y los nodos `2` y `5`, identificando estos flujos como `2` y `3` respectivamente.

Ahora se ha de indicar el tipo de tráfico que se va a generar. Para ello se incluirá en cada flujo un tráfico de tipo "tasa de bits constante" o CBR (constant bit rate), el cual permite introducir ciertos parámetros como el tamaño del paquete (`packet_size_`), la tasa de paquetes (`rate_`), el intervalo entre paquetes (`interval_`), el tamaño del paquete (`packetSize_`), un máximo de paquetes a transmitir (`maxpkts_`) o la inclusión o no de ruido (`random_`).

Para su creación se inicia la aplicación CBR mediante un identificador, se asigna a un agente creado anteriormente y se especifican los parámetros requeridos por la aplicación. Para todas las simulaciones se han de especificar paquetes de un tamaño igual a 1000 bytes con una tasa de envío de 2Mbps.

Otro procedimiento común para todas las simulaciones será el establecimiento de un tiempo de inicio y final por cada flujo y observar el funcionamiento de la red a través de las herramientas NAM, XGRAPH y reportes generados a través de los scripts realizados en el lenguaje AWK. Los tiempos de inicio y fin para cada flujo se especifican mediante

```
$ns at 0.5 "$cbr start"  
$ns at 1.0 "$ftp start"  
$ns at 1.0 "$ftp start"  
$ns at 100.0 "$cbr stop"  
$ns at 110.0 "$ftp stop"  
$ns at 100.0 "$cbr stop"
```

El resto de parámetros depende de la prueba implementada y se explicarán a continuación.

#### 4.1 ESCENARIO UNO (ANCHOS DE BANDA GARANTIZADOS)

Para este escenario cada emisor enviará una taza de bits de 2Mbps, por lo que la red troncal deberá garantizar una capacidad de al menos 6Mbps. Para nuestras pruebas hemos escogido 7Mbps para así garantizar los recursos suficientes para el flujo total de datos en la red. Los enlaces de los hosts deben entonces soportar una capacidad de 2Mbps. Esta topología es mostrada en la figura 1.



Figura 1. Topología de la red con garantía de recursos para el tráfico total de datos

los creamos y luego establecemos el enlace entre ellos.

Esto se ha de hacer para cada nodo y cada enlace de la topología. La inicialización del protocolo de transporte así como la aplicación se describió en la sección 4.1. Alcanzados estos pasos se tiene implementada entonces una red con flujo de datos, sin garantizar en ella, política alguna de calidad de servicios.

#### B. Red con calidad de servicios diferenciados con prioridad igual para los 3 flujos

Para la implementación de servicios diferenciados sobre la topología se deben configurar los routers como frontera o núcleo, donde los primeros son los encargados de marcar el paquete y los segundos los encargados de la priorización de los paquetes de acuerdo a la marca establecida por los primeros.

#### A. Red sin calidad de servicio (QoS)

Inicialmente se han de construir los enlaces que corresponden a la topología mostrada en la figura 2. En ellos se especifica su capacidad (en bits por segundo) por cada enlace y además, para efectos del simulador se deben indicar la capacidad de transmisión (duplex, simplex), el origen, el destino y el manejo que se le ha de dar a la cola.

Para este último parámetro se selecciona DropTail que tiene como característica que si la capacidad del buffer de la cola de salida es excedida, entonces el último paquete que llegue es descartado, (disciplina FIFO, first input first output). Para conectar por ejemplo el nodo 6 (identificado como routero) con el nodo 7 (router1), primero

Los enlaces entre los routers deben ser entonces modificados para implementar esta clase de servicio, esto es, los enlaces entre los nodos 6, 7 y 8. Los nodos 6 y 8 se indicaran como routers de frontera (redgeo y redgeo1 respectivamente) mientras el 7 como router de núcleo (rcore). Para especificar el enlace entre los diferentes routers se deben indicar entonces: el ancho de banda soportado por el enlace, el retardo del enlace y el manejo de cola del enlace (dsRED/edge o dsRED/core para indicar si es de frontera a núcleo o viceversa).

El resto de enlaces no varían en su declaración de acuerdo a lo indicado en la sección anterior. Ahora se declaran las

colas entre los enlaces de los nodos routers, dándole un identificador a cada uno de ellos. Para el caso, se deben declarar 4 colas.

Creadas las colas se deben configurar las políticas propias del protocolo por cada flujo. Inicialmente se ha de configurar por cola, el tamaño promedio de paquetes, el número de colas físicas y el número de colas virtuales. El tamaño medio de paquetes para nuestro caso corresponde a 1000 bytes y el número de colas físicas es 1 mientras las virtuales son 2.

Para cada flujo se adicionará una política tokenbucket la cual depende de unos valores CIR (committed information rate), tasa promedio de ancho de banda que se le asigna a un determinado flujo en unidad de tiempo  $T$ , que se halla además con la relación del parámetro CBS (committed burst size) e iguales a 2000000 y 1000 respectivamente lo cual da un tiempo  $T=1000/2000000=0,5ms$ .

Luego se define el policer propiamente dicho. Un TokenBucket tiene dos precedencias de descarte, representadas con dos DSCPs: 10 y 11. Esto quiere decir que como resultado del policing, un paquete puede ser "clasificado" en uno de dos niveles de conformidad. Los perfiles de tráfico esperados en cada uno de los niveles de conformidad se especifican con los valores CIR y CBS.

Luego se define un mecanismo para manejar paquetes entrantes. La secuencia indica que un paquete con un determinado PHB debe ser encolado en la cola física X y en la cola virtual Y.

Finalmente restan solo 3 líneas en las que se indican, por línea, la cola física, la cola virtual, el nivel mínimo de ocupación media de la cola, el nivel máximo de ocupación media de la cola y la probabilidad máxima de descarte.

Para el caso del router core, no es necesario establecer política alguna, pues él hace el papel de distribuir el paquete que recibe, de acuerdo a la política anteriormente instaurada.

Solo basta agregar el protocolo, el tipo de tráfico y el inicio y final de la simulación, explicado en la sección 4.1.

### C. Red con calidad de servicios diferenciados con prioridad para uno de sus flujos

El flujo para el cual se dará prioridad, será al identificado como el 1, esto es, el tráfico entre los nodos pco y pc3. Esta prioridad se indica mediante los parámetros CIR y CBS en cada uno de los flujos. Una cantidad mayor en el parámetro CIR aumentara la prioridad al flujo de datos al cual se le asigne este parámetro y viceversa. Entonces para dar prioridad al flujo 1 sobre los flujos 2 y 3, se aumenta el CIR del flujo 1 y se disminuye el CIR de los otros dos flujos.

Los demás parámetros son idénticos a los de la prueba anterior.

### 4.2. ESCENARIO DOS (CANAL SATURADO)

Para este escenario cada emisor envía una taza de bits de 2Mbps, mientras que la red troncal apenas garantiza una capacidad



de 3Mbps. Los enlaces de los hosts deben entonces soportar una capacidad de 2Mbps y la topología es mostrada en la figura 2.



Figura 2. Topología de la red sin garantía de recursos para el tráfico total de datos

### A. Red sin calidad de servicio (QoS)

De acuerdo a lo explicado en el literal A de la sección 4.1, se deben iniciar y configurar los enlaces de acuerdo ahora a la topología mostrada en la figura 3. La configuración para el enlace entre el nodo 6 y el nodo 7, sería:

```
set routero [$ns node]
set router1 [$ns node]
$ns duplex-link $routero $router1 3Mb
10ms DropTail
```

El resto de parámetros se mantienen igual de acuerdo a lo indicado en el literal mencionado.

### B. Red con calidad de servicios diferenciados con prioridad igual para los 3 flujos

De acuerdo a lo indicado en el literal B de la sección 4.1. se deben configurar los enlaces, establecer el marcado de los paquetes y la implementación de las políticas (para routers edge en el primer caso y routers core para el segundo).

En este caso el CIR es 1000000 pues esta cifra es el equivalente a un tercio de la capacidad del canal troncal, por lo que se

selecciona por pretender establecer igual prioridad a todos los flujos y que no se exceda la capacidad del canal. El resto de configuraciones no varían en absoluto de acuerdo al literal mencionado.

### C. Red con calidad de servicios diferenciados con prioridad para uno de sus flujos

El flujo para el cual se dará prioridad, será al identificado como el 1, esto es, el tráfico entre los nodos pco y pc3. Esta prioridad se indica mediante los parámetros CIR y CBS en cada uno de los flujos. Una cantidad mayor en el parámetro CIR aumentara la prioridad al flujo de datos al cual se le asigne este parámetro y viceversa. Entonces para dar prioridad al flujo 1 sobre los flujos 2 y 3, se aumenta el CIR del flujo 1 y se disminuye el CIR de los otros dos flujos. Los demás parámetros son idénticos a los de la prueba anterior.

## 5. ANÁLISIS DE RESULTADOS

Se muestran los resultados obtenidos en cuanto a ancho de banda, delay, throughput, jitter y paquetes enviados/recibidos se refiere. Para el caso del ancho de banda los resultados se muestran a través de la herramienta XGRAPH, mientras que para los otros parámetros de medición de calidad de servicio (obtenidos a través de varios scripts) se tabulan y muestran en diferentes tablas. Se muestran y analizan entonces los resultados para los dos escenarios propuestos, en el que no se excede la capacidad del canal, y en la que se excede su capacidad. Los resultados para cada caso, de acuerdo a la arquitectura implementada son:



## 5.1. ESCENARIO UNO (ANCHOS DE BANDA GARANTIZADOS)

Por cada prueba se obtuvieron entonces los siguientes resultados.

### A. Sin implementar calidad de servicio

Los resultados del ancho de banda, delay, throughput, jitter y paquetes perdidos/recibidos, se muestran en la figura 3, y las tablas 1 y 2, para cada uno de los flujos sin implementar calidad de servicio.

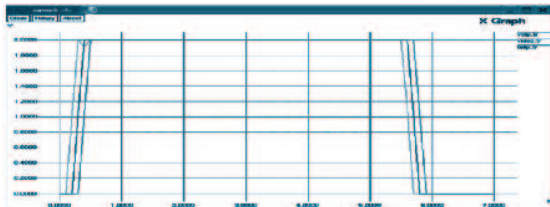


Figura 3. Ancho de Banda sin implementar calidad de servicio.

Flujo	Delay (s)	Troughput (kbps)	Jitter		
			Maximo	Promedio	paquetes
1	0,012852	2001,06	0,001143	0,000001	307
2	0,0125715	2001,48	0	0	234
3	0,0131269	2000,63	0,001143	0,000002	369

Tabla 1. Delay, Troughput y Jitter para Literal A, escenario 1

Flujo	Paquetes Enviados	Paquetes Perdidos	Recepcion (%)
1	1351	0	100
2	1351	0	100
3	1351	0	100

Tabla 2. Paquetes perdidos/recibidos según Literal A, escenario 1

### B. Diffserv con igual prioridad a los flujos

Los resultados del ancho de banda, delay, throughput, jitter y paquetes perdidos/recibidos, se muestran en la figura 4, y las tablas 3 y 4 para cada uno de los flujos con igual prioridad de calidad de servicio.

recibidos, se muestran en la figura 4, y las tablas 3 y 4 para cada uno de los flujos con igual prioridad de calidad de servicio.

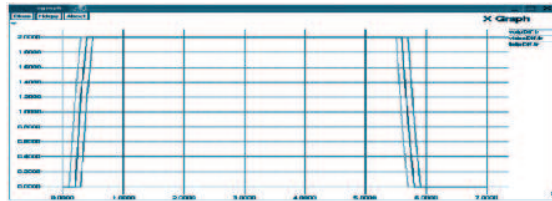


Figura 4. Ancho de Banda con diffserv con igual prioridad a cada uno de sus flujos.

Flujo	Delay (s)	Troughput (kbps)	Jitter		
			Maximo	Promedio	paquetes
1	0,010352	2001,06	0,001143	0,000001	118
2	0,0100715	2001,48	0	0	75
3	0,0106269	2000,63	0,001143	0,000002	314

Tabla 3. Delay, Troughput y Jitter para Literal B, escenario 1

Flujo	Paquetes Enviados	Paquetes Perdidos	Recepcion (%)
1	1351	0	100
2	1351	0	100
3	1351	0	100

Tabla 4. Paquetes perdidos/recibidos según Literal B, escenario 1

### C. Diffserv con mayor prioridad al flujo 1

Los resultados del ancho de banda, delay, throughput, jitter y paquetes perdidos/recibidos, se muestran en la figura 5, y las tablas 5 y 6 para cada uno de los flujos con igual prioridad de calidad de servicio.

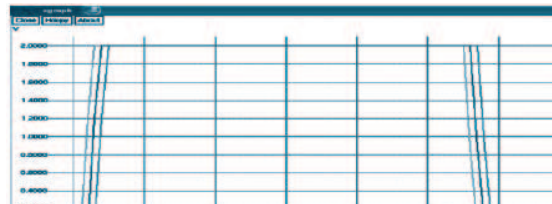


Figura 5. Ancho de Banda con diffserv con mayor prioridad al flujo 1.

Flujo	Delay (s)	Troughput (kbps)	Jitter		
			Maximo	Promedio	paquetes
1	0,010352	2001,06	0,001143	0,000001	118
2	0,0100715	2001,48	0	0	75
3	0,0106269	2000,63	0,001143	0,000002	314

Tabla 5. Delay, Troughput y Jitter para Literal C, escenario 1

Flujo	Paquetes Enviados	Paquetes Perdidos	Recepcion (%)
1	1351	0	100
2	1351	0	100
3	1351	0	100

Tabla 6. Paquetes perdidos/recibidos según Literal C, escenario 1

## 5.2. ESCENARIO DOS (CANAL SATURADO)

Por cada prueba se obtuvieron entonces los siguientes resultados.

### A. Sin implementar calidad de servicio

Los resultados del ancho de banda, delay, throughput, jitter y paquetes perdidos/recibidos, se muestran en la figura 6, y las tablas 7 y 8, para cada uno de los flujos sin implementar calidad de servicio.



Figura 6. Ancho de Banda sin implementar calidad de servicio.

Flujo	Delay (s)	Troughput (kbps)	Jitter		
			Maximo	Promedio	paquetes
1	0,0385798	1010,62	0,004	0,000188	80
2	0,0454447	1979	0,004	0,000071	422
3	0,0155414	951,374	0,004	0,001742	56

Tabla 7. Delay, Troughput y Jitter para Literal A, escenario 2

Flujo	Paquetes Enviados	Paquetes Perdidos	Recepcion (%)
1	1351	653	51,665433
2	1351	0	100
3	1351	1276	5,5514434

Tabla 8. Paquetes perdidos/recibidos según Literal A, escenario 2

### B. Diffserv con igual prioridad a los flujos

Los resultados del ancho de banda, delay, throughput, jitter y paquetes perdidos/recibidos, se muestran en la figura 7, y las tablas 9 y 10 para cada uno de los flujos con igual prioridad de calidad de servicio.

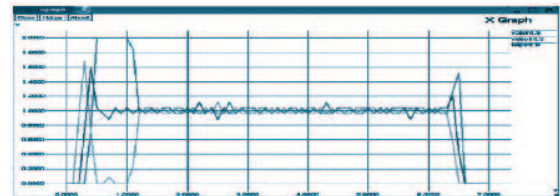


Figura 7. Ancho de Banda con diffserv con igual prioridad a cada uno de sus flujos.

Flujo	Delay (s)	Troughput (kbps)	Jitter		
			Maximo	Promedio	paquetes
1	0,0287637	1015,7	0,004	0,00086	296
2	0,0297511	1110,56	0,004	0,000762	290
3	0,0272811	932,067	0,004	0,000891	277

Tabla 9. Delay, Troughput y Jitter para Literal B, escenario 2

Flujo	Paquetes Enviados	Paquetes Perdidos	Recepcion (%)
1	1351	661	51,073279
2	1351	606	55,144338
3	1351	713	47,224278

Tabla 10. Paquetes perdidos/recibidos según Literal B, escenario 2

### C. Diffserv con mayor prioridad al flujo 1

Los resultados del ancho de banda, delay, throughput, jitter y paquetes perdidos/

recibidos, se muestran en la figura 8, y las tablas 11 y 12 para cada uno de los flujos con igual prioridad de calidad de servicio. Figura 8. Ancho de Banda con diffserv con mayor prioridad al flujo 1.

Flujo	Delay (s)	Troughput (kbps)	Jitter		
			Maximo	Promedio	paquetes
1	0,0329732	1639,82	0,004	0,00066	384
2	0,0278355	798,315	0,004	0,0001034	263
3	0,0237899	616,589	0,004	0,00001279	257

Tabla 11. Delay, Troughput y Jitter para Literal C, escenario 2

Flujo	Paquetes Enviados	Paquetes Perdidos	Recepcion (%)
1	1351	231	82,901554
2	1351	814	39,748335
3	1351	931	31,088083

Tabla 12. Paquetes perdidos/recibidos según Literal C, escenario 2

### 5.3. COMPARACIONES

Observando las tablas anteriores, se puede apreciar que los cambios más relevantes o significativos suceden cuando se implementa calidad de servicio sobre un flujo, por lo tanto, solamente se compararan los resultados de los literales C y C de las secciones 5.1 y 5.2.

Los resultados para cuando hay suficientes recursos para transportar los flujos de datos, se comparan en las graficas 9, 10, 11 y 12. En el eje X el valor correspondiente a "1" indica la arquitectura diffserv, mientras que el valor "2" corresponde a la arquitectura intserv. Los resultados son entonces:

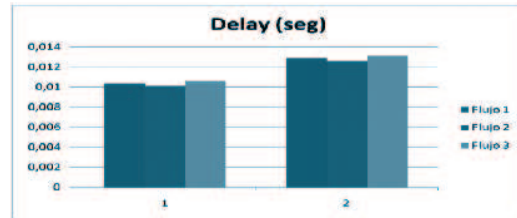


Figura 9. Delay obtenido para el escenario 1 con prioridades al flujo 1. Diffserv se identifica mediante el valor 1 mientras Intserv con el valor 2

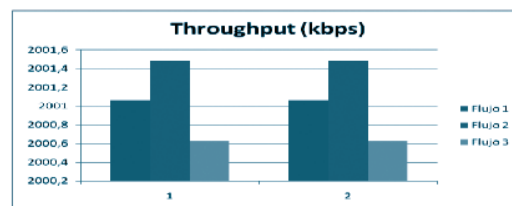


Figura 10. Throughput obtenido para el escenario 1 con prioridades al flujo 1. Diffserv se identifica mediante el valor 1 mientras Intserv con el valor 2

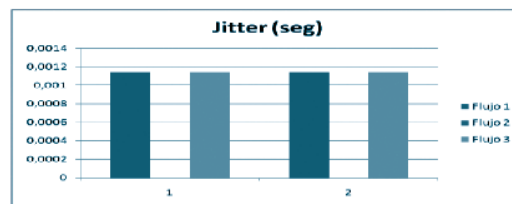


Figura 11. Jitter obtenido para el escenario 1 con prioridades al flujo 1. Diffserv se identifica mediante el valor 1 mientras Intserv con el valor 2

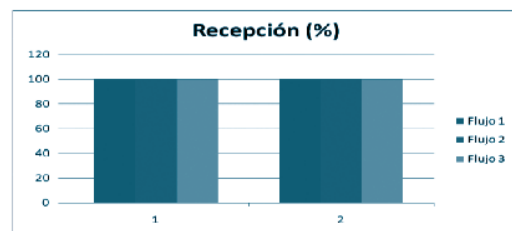


Figura 12. Recepción obtenida para el escenario 1 con prioridades al flujo 1. Diffserv se identifica mediante el valor 1 mientras Intserv con el valor 2

Los resultados para cuando no hay suficientes recursos para transportar los flujos de datos, se comparan en las graficas 13, 14 15 y 16 . En el eje X el valor correspondiente a "1" indica la arquitectura diffserv, mientras que el valor "2" corresponde a la arquitectura intserv. Los resultados son entonces:

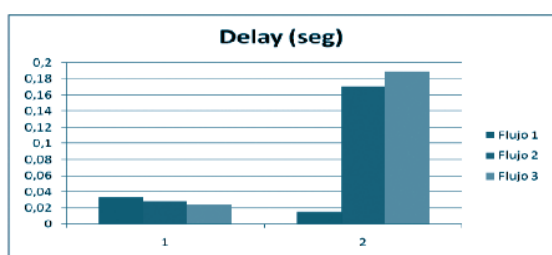


Figura 13. Delay obtenido para el escenario 2 con prioridades al flujo 1. Diffserv se identifica mediante el valor 1 mientras Intserv con el valor 2

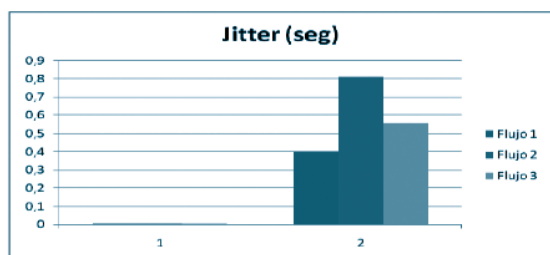


Figura 14. Jitter obtenido para el escenario 2 con prioridades al flujo 1. Diffserv se identifica mediante el valor 1 mientras Intserv con el valor 2

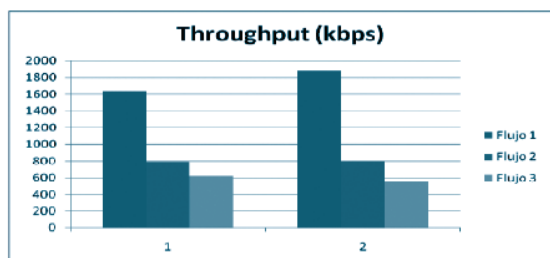


Figura 15. Throughput obtenido para el escenario 2 con prioridades al flujo 1. Diffserv se identifica mediante el valor 1 mientras Intserv con el valor 2

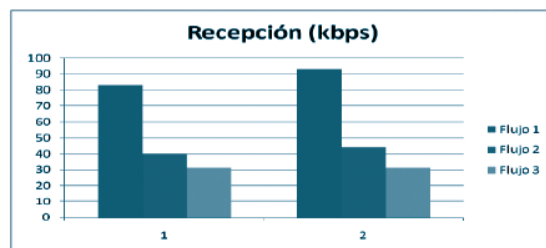


Figura 16. Recepción obtenida para el escenario 2 con prioridades al flujo 1. Diffserv se identifica mediante el valor 1 mientras Intserv con el valor 2

Los resultados obtenidos para el escenario 1, nos muestran de acuerdo a las figuras 9-12, que la implementación de una u otra calidad de servicio, cuando hay suficientes recursos para atender la demanda de tráfico, no presenta mejoría considerable y su comportamiento es similar en ambos casos, apenas notándose una mejoría en el delay para el caso de servicios diferenciados.

Las figuras 13-16 nos muestran que la incorporación de la arquitectura diffserv ante intserv trae consigo una menor cantidad de retardos, bastante notorias en cuanto a delay y jitter se refiere. Para los otros ítems de comparación, el comportamiento entre ambas arquitecturas es semejante, no pudiendo definir que la una supera ampliamente a la otra en el caso de throughput y pérdida de paquetes.

## 6. CONCLUSIONES

Servicios integrados y servicios diferenciados cumplen con el mismo objetivo de asignar prioridades de anchos de banda a diferentes flujos de datos según lo desee el administrador de la red.

Cuando se asignan mal los anchos de banda para cada flujo, superando estos el ancho de banda total de la red, esta simplemente actúa como un sistema best effort.

Una diferencia entre servicios diferenciados e integrados radica en la forma en que cada uno configura la red para asignar prioridades de flujo, pues mientras diffserv establece unas políticas para el flujo de datos y las envía para su distribución entre routers frontera y de núcleo, intserv define un enlace virtual, al cual se le deben asignar los mismos recursos en toda la red.

Los recursos que se asignan en un flujo de datos en diffserv pueden variar dentro de la red dependiendo de las políticas que tenga cada router núcleo por los que pase dicho flujo de datos, mientras que en intserv, se asignan recursos a un flujo de datos por toda la ruta que transita, por lo que su implementación en redes de gran tamaño puede tener un costo de recursos muy altos.

Diffserv presenta un menor delay y jitter debido a que hace un mejor manejo de los datos de cola comparado con la arquitectura intserv.

Apesar de que la pérdida de paquetes totales cuando se implementa calidad de servicio es similar a cuando no se implementa, la diferencia radica en que el ancho de banda es mejor distribuido entre los flujos de datos según las políticas de calidad de servicio que se hayan implementado.

## 7. REFERENCIAS

- [1] Jorge Lázaro Laporta, Marcell Miralles Aguiñiga. Fundamentos de Telemática. Editorial Universidad Politécnica de Valencia, 2005, 408 páginas.
- [2] J. Romero, [en línea]. Consultado en Septiembre 5 de 2011. Disponible en: [http://personales.upv.es/aleon/pubs/URSI2004\\_QDSM.pdf](http://personales.upv.es/aleon/pubs/URSI2004_QDSM.pdf)
- [3] Srinivas Vegesna. IP quality of service. Editorial Cisco Press, CCIE, N° 139.
- [4] José Antonio Mañas. Mundo IP. Editorial nowtilus, Marzo 2004, 369 páginas.
- [6] Srinivas Vegesna. IP quality of service. Editorial Cisco Press, CCIE, N° 139.
- [7] R. Branden, D. Clark, S. Shenker. RFC 1633, Integrated Services in the Internet Architecture: an Overview. June 1994.
- [8] K. Nichols, S. Blake, F. Baker, D. Black. RFC 2474, Definition of the Differentiated Services Field in the IPv4 and IPv6. December 1998.
- [8] Marcos, Chait B. Evaluación de provisión de Calidad de Servicios mediante Servicios Diferenciados Departamento de Ingeniería de Sistemas y Computación. <http://es.scribd.com/doc/55070042/CIT027>
- [9] S. Terrasa, S. Sáez, J Vila. Comparing the utilization bounds of Intserv and Diffserv. Universidad Politécnica de Valencia.
- [10] Adrian Delfino, Sebastián Rivero, Marcelo San Martín. Ingeniería de tráfico en redes MPLS. Instituto de Ingeniería Eléctrica.

- [11] Sidnei De Oliveira Guerra. Tesis Doctoral Una propuesta de Arquitectura MPLS/diffserv para proveer mecanismos de calidad de servicio (QoS) en el transporte de la telefonía IP, 2004.
- [12] Carlos, García García. Tesis Doctoral Propuesta de Arquitectura en entorno inalámbrico 802.11e basado en Diffserv con ajuste dinámico de parámetros, Junio 2006.
- [13] Sebastián Andrés Álvarez, Agustín José González. Estudio y configuración de calidad de servicio para protocolos IPv4 e IPv6 en una red de fibra óptica WDM, 24 de Julio de 2005.
- [15] Werner Almesberger, Silvia Giordino, Robertto Mameli. A Prototype Implementation for the Intserv Operation over Diffser Network.
- [16] Heng-Chi Su, Ren-Hung Hwang. Provide Multicast Service in a combined Intserv and Diffserv Enviroment. July 11 de 2002.
- [17] María Carmen España Boquera, Servicios Avanzados de Telecomunicaciones. Ediciones Díaz Santos, 2003.
- [18] A. Mellouk, Consultado en Septiembre 10 de 2011.
- [19] . P. Loshin. . Loshin [en ñinea]. Consultado en Septiembre de 2011.
- [20] O. Helchman, [en linea]. Consultado en 10 de 2011.