Scalable QoS Approach in a Core Internet Network

Cláudia J. Barenco Abbas[1] and L. Javier García Villalba[2]

[1] Dept. Electric Engineering, University of Brasilia, Brazil
barenco@redes.unb.br
[2] Dept. Computer Systems and Programming, Complutense University of Madrid, Spain
javiergv@sip.ucm.es

Abstract. A special attention about scalability has to be paid to QoS solutions for Core Internet Networks as they deal with a lot of flows and demand many resources. This paper analyses and proposes integrated solutions from the physical layer (SDH/SONET and DTM) to the IP layer (IntServ and DiffServ), concentrating not only on scalability, but also QoS guarantees.

1 Introduction

The QoS (*Quality of Service*) concept can be interpreted as a method to give a preferential treatment to some traffic, according to their QoS requirements as opposed to the Best-Effort treatment. The QoS solutions in a Core Network are different from those used in the Access Network, as the former has a higher cost of bandwidth and deals with a large number of flows. Consequently, scalability is an important aspect in this case. It motivated us to analyse the Core Internet environment from the transport solutions (SDH and DTM) to the IP QoS models (DiffServ and IntServ), highlighting the scalability view and proposing scalable QoS integrated architectures.

The paper is organised as follows. Section 2 analyses the SDH/SONET and DTM as transport solutions, in relation to scalability aspect. Section 3 describes a comparative study of IP/ATM and IP/DTM in aspects of multicast support, throughput, delay, jitters and losses. It also presents a study of maximum throughput of IP/SDH. Section 4 proposes architectures to integrating DiffServ/MPLS/ATM and DiffServ/DTM. Section 5 describes a study of the bandwidth demand of aggregated RSVP reservations, reaffirming that the IntServ model is not appropriate in a Core environment. Finally in section 6 we have the conclusion.

2 Network Transport Solutions – A Scalability Study

As the Internet is dynamic and grows rapidly, the increase of the number of access networks connected to a Core Network can bring some scalability problems principally in a full-mesh scenario. This section analyses the well-known SDH/SONET and the new DTM in this kind of topology. The SDH/SONET has been used as a transport solution and it showed to be very simple and with low overhead.

J.M. Cueva Lovelle et al. (Eds.): ICWE 2003, LNCS 2722, pp. 364–373, 2003.

Another transport solution, DTM (*Dynamic Synchronous Transfer Mode*) [1], is fundamentally similar to SONET/SDH in terms of low complexity and low overhead. But, to increase the flexibility and avoid the hierarchical structure of SONET/SDH, DTM includes signalling and switching.

SDH. This technology doesn't support multichannel interface. So if we want a full-mesh environment, we need to provide a complete combination of physical connections, which make it not scalable. To solve this is necessary to build a topology where we have to pass through a number of routers from one side of the network to the other. This introduces delay and jitter, a problem for QoS sensitive traffic.

DTM. It is an architecture based on high-speed circuit switching architecture, with dynamic resource reallocations which don't appear in the traditional solutions of circuit switching. It provides multicast services, channels with varied bit-rates (from 512 Kbps until the specific media capacity) and low circuit configuration time (few milliseconds). It can be used as an end-to-end solution or as a transport protocol like ATM and IP, which is our focus here. The folded bus and ring topologies are unsuitable for large geographical distances. If a receiving node is located on the "wrong" side, the delay propagation can be large. Two rings can be established to avoid it, one in each direction. So the information is transmitted on both rings, but this scheme requires more bandwidth. With a dual-bus topology the average distance between nodes is minimized. The bus can support full duplex communication. Moreover, two-dimensional mesh is significantly higher than the capacity of a linear topology with the same number of nodes [2]. Here we will study only one direction bus, as we can have the same conclusions about the other direction. We assume a bus topology of 'M' x 'N' where not every router (node) is active (figure 1).

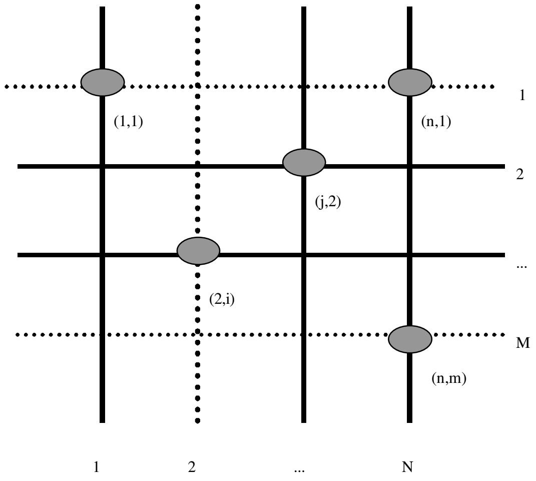


Fig. 1. Full-mesh Topology.

The inactive buses are on dotted lines and the buses necessaries to the full interconnection between the active nodes (painted nodes) are on highlight black lines.

Suppose each router has a geometric address (x_j , y_i) with $1 \le j \le M$ and $1 \le i \le N$. Being NB(R) the set of buses seen by router 'R' and B the total set of buses, a sufficient condition for the best connectivity is NB(R) > B/2.

As we can see, the scalability doesn't depend on the number of nodes but on the number of buses and its distribution. In figure 1, each active node (red node) can see at least 3 buses, so they have the best full interconnection.

3 IP/Link Layer QoS Solutions

In figure 2 we can see some solutions for the integration of IP with the link and physical layers. It is not necessary to say that we are walking to have the IP level directly over a dynamic physical layer, like IP over SDH. In this section we compare some features of IP/ATM, IP/DTM and IP/SDH like throughput, delay and so on. The MPLS does not introduce any QoS service, but it provides a better forwarding of IP packets and is a good "tunnelling" solution. It will be our discussion on next section.

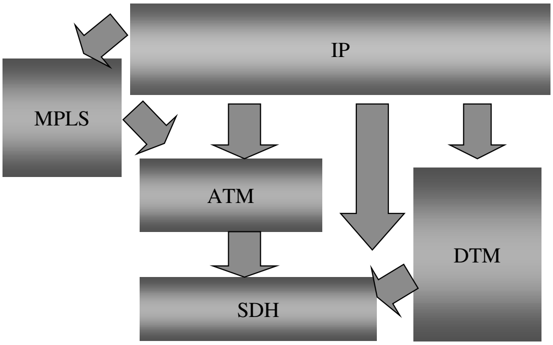


Fig. 2. IP/Link Layers Solutions.

Maximum Throughput. A study [3] of the IP flows characteristics in an Internet Backbone showed that the TCP is the most significant traffic (95% or more of the bytes, 85-95% of the packets and 75-85% of the flows), so the following study is based on TCP protocol and not on UDP protocol, even though the latter is being used more frequently by multimedia applications.

A TCP maximum segment size (MSS) used is 512 bytes. The TCP data is encapsulated over ATM as follows: a set of headers and trailers are added to every TCP segment. We have 20 bytes of TCP header, 20 bytes of IP header, 8 bytes for the RFC1577 LLC/SNAP encapsulation and 8 bytes of AAL5 (*ATM Adaptation Layer*), that is, a total of 56 bytes (figure 3). Hence, every 512 bytes become 568 bytes. This payload with padding requires 12 ATM cells of 48 data bytes each. The maximum throughput of TCP over ATM is (512 bytes/(12 cells x 53 bytes/cell))=80.5%.

DTM has several proposals for the frame format to be used to multiplex packets on top of DTM channels. But the usual implemented environment uses a 64 bits trailer after each TCP/IP packet. With a 512 bytes MSS 520 bytes will be transported on the medium (65 slots), where 472 bytes are data (59 slots) and 6 slots are overhead (figure 4). So we have 90% of maximum throughput. Moreover in DTM there's a overhead of control messages (e.g., having 40 nodes attached to a bus of 622 Mbps, with one control slot each, the overhead is 3.3% and a final maximum throughput of 86.7%). Obviously, the overhead of control messages depends on the number of nodes per bus, the capacity of the bus and the number of control slots per node.

20 bytes TCP Header	20 bytes IP Header	8 bytes LLC/SNAP	8 bytes AAL5

Fig. 3. TCP/IP Encapsulation over AAL5 ATM.

20 bytes TCP Header	20 bytes IP Header	472 bytes DATA	8 bytes DTM Trailer

Fig. 4. TCP/IP Encapsulation over DTM.

The SONET framing includes a 7 bytes per packet PPP (*Point-to-Point*) header and additional SONET framing overhead of 90 bytes per 2340 bytes of payload. So for IP over SONET, the bit-efficiency is 94% [3]. It has however a static hierarchical structure, if it comes with some flexible protocol, such as ATM

Delay and Jitter. The recommendation of ITU-T I.356 [4] gives the worst guess case for a 27500 km connection, passing 29 ATM systems. Here we are only interested in ITU-T QoS Class 1, which is appropriate for real time services and demands strict QoS performance.

Table 1. ATM QoS Classes according to I.356 [4].

QoS Class	CTD	CDV (2 points)	CLR all cells
Class 1	400 ms	3 ms	3.0 E -7

In DTM the most representative delay is the access delay, since the transfer delay is constant and depends only on the number of hops along the path.

Access delay depends mainly on the load on the control slots, and on how many control messages that need to be sent to establish a channel. The access delay is typically a summation of several delays (and typical values): (a) node controller processing delay (5 µs); (b) delay in finding and allocating free tokens, which depends on the kind of slot control (on average 100 µs. Distributed control requires also time for sending and receiving slot requests. Central control requires 10 µs before the message hits the medium, physical distance 10 µs, 10 µs before the replay hits the fiber and 10 µs more to the physical distance); (c) waiting for the first available control slot to pass (50 µs); (d) waiting for the first allocated data slot to be filled with user data (62.5 µs) and finally, (e) waiting in the queue at the input to node controllers, that for high priority traffic is zero. If one message is generated at each cycle time, the total average access delay is 0.217 ms. Average delay on each hop is around 125 µs, but the fundamental delay on each hop is the time between an incoming slot and the scheduled outgoing slot, which is on average 62.5 µs, since there is not processing nor queuing on DTM switches.

Having 29 DTM hops along the path, as on the table 1, we have around 3.625 ms of transfer delay, considering that we are using a fiber medium with propagation delay irrelevant. We have a total of slot transfer delay around 3.842 ms. Here we are not considering the worst case, but the average in case of a normal load in the network. So total slot delay is extremely lower than it is an ATM environment with 29 systems (400 ms). The jitter is almost non-existent because there isn't congestion on the hops, and DTM isolates one channel from another.

Losses. In DTM looses are controlled on the queues in the transmitters, so it is only necessary to implement the correct buffer management algorithm. For high priority it is not expected to have losses, if the controller node has capacity to efficiently handle traffic flows and the channel capacity is adequate. In ATM CBR (*Constant Bit Rate*) and rt-VBR (*Real-Time Variable Bit-Rate*) the losses are guaranteed and low as can be seen on the table 1.

Multicast. As DTM uses a shared medium multicast transmission is easily supported. The number of multicast channels for a full-mesh scenario does not depend on the number of nodes of the multicast group, since various senders can share the same channel. So it is only necessary to establish a unique multicast tree per multicast group. Considering the multicast service on ATM, the number of possible multicast trees in a full-mesh of VCs (*Virtual Circuit*) is

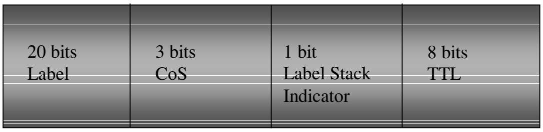
$$\sum_{m=2}^{n} \frac{n!}{(n-m)!\,m!} \qquad n \geq 2$$

being n the number of border routers and m the number of members of a multicast group (unicast is a special case of multicast (m=2)).

4 Interaction of IETF DiffServ Model with Link Layers QoS

The IP over ATM has limited scalability due to the "n-squared" problem when a full mesh of VCs is provided. This section suggests an IP/MPLS/ATM architecture as it is more appropriate for large networks, like Core Internet, in terms of flexibility, scalability and manageability. Another architecture discussed is IP/DTM. As DTM automatically performs the rapid forwarding of packets by the time-switching slots, the MPLS does not have any functionality on this architecture. The Traffic Engineering feature provided by the MPLS may be performed by IP QoS protocols, since routing on this architecture depends on the IP routing decision. The DiffServ (*Differentiated Service*) [6] recently proposed by IETF provides simple guarantees of QoS that, in majority, are `qualitative´ because it is based on that some applications don't need an explicit guarantee of QoS. So adequate traffic engineering and classification of flows by priority suffice for the necessary functionality. Due to these characteristics DiffServ model seems more scalable than IntServ model.

DiffServ/MPLS/ATM. MPLS [7] provides an overhead of 4 bytes (figure 5) but the packet path is completely determined by the label. In terms of header overhead, it is more efficient than other tunnelling solutions (e.g. RSVP TE extensions [8]).



20 bits Label	3 bits CoS	1 bit Label Stack Indicator	8 bits TTL

Fig. 5. MPLS Overhead.

Using MPLS to have a full-mesh of LSPs (*Label Switched Paths*) between `n´ border routers, we can establish a maximum of ‘n’ “sink trees” in DiffServ. So this architecture is scalable. Figure 6 shows a proposed architecture for the interaction of DiffServ, MPLS and ATM technologies.

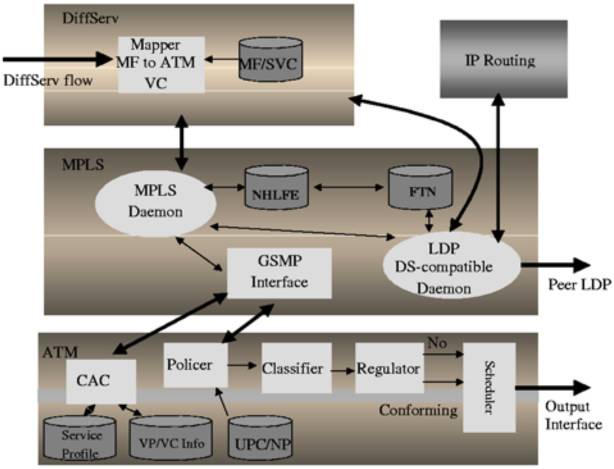


Fig. 6. DiffServ/MPLS/ATM Architecture.

The functional elements are:

(i) Mapper MF to ATM SVC**.** Responsible for the mapping of a DiffServ MF (Multi Field) filter. For example: DSCP bits, source/destination address, ingress/egress routers, etc., to an ATM SVC. The capacity of the SVC can be defined by static provisioning or by signalling (e.g. RSVP protocol). The QoS class of the SVC inside the ATM Switch can be mapped from the LDP CoS TLV.

For the EF (*Expedited Forwarding*) mapping of the DSCP bits to the 3-bit LDP CoS TLV may be CoS TLV = `111´and BE (*Best-Effort*) = `000´.

In ATM we only have two CLPs (*Cell Loss Priority*). As [9] show that more than 2 priorities is not better than 2 priorities, we only define two classes of Drop Priority inside each EF Class (table 3).

New MPLS FEC (DSCP/Border Addresses TLV) is proposed here, which is compatible with the MF Filter (table 2).

Table 2. DSCP/Border Addresses TLV.

FEC Element Name	Type	Value
DSCP/Border Addresses	0x04	MF Field

The "Value" field could be for example: (a) DSCP bits (6 bits), (b) Ingress Address Length (8 bits), (c) Ingress Address (32 bits), (d) Egress Address Length (8 bits), (e) Egress Address (32 bits), (f) MPLS Daemon (the Core of the MPLS protocol), (g) LDP Daemon (Responsible for the label binding. Here it has extensions to transport the MF attributes on its LDP PDUs as the FEC DSCP/Border Addresses TLV described before), (h) GSMP Interface [10] (Before asking for the establishment of a new SVC, it is necessary to define the QoS capacities of the ATM Switch ports and group the VCs on QoS classes, allocating resources for QoS classes. The DiffServ QoS classes can be mapped directly to the QoS classes inside the switch. To do this we use the "Quality of Service" message, which is sent to the ATM switch. "Scheduler Establishment" messages configure the scheduler on each output port. Configuration of QoS Classes is made by sending "QoS Class Establishment" message. The SVCs can be established or turned-down using "QoS Connection Management". The resources can be allocated to the connections through "QoS Connection Management". The "QoS Configuration" message permits to discover the QoS capacities of each port of the ATM Switch). We will not detail the Traffic Control functions of an ATM Switch since it is widely deployed.

Table 3. Mapping DSCP bits to LDP CoS TLV.

DSCP bits	LDP Cos TLV	DSCP bits	LDP Cos TLV
EF 101110	111	AF22 010100	010
AF11 001010	110	AF31 011010	011
AF12 001100	100	AF32 011100	001
AF21 010010	101	BE 000000	000

(ii) The MF/SVC database stores the relation of the DiffServ MF filter used to group the flows to the output configured SVC ATM. For example

Table 4. Mapping FEC/SVC.

FEC			SVC		
DSCP bits	Ingress Router	Egress Router	Port	VPI	VCI

(iii) Like the MF/SVC, the FTN and NHLFE databases have also to be extended to support the MF DiffServ filter.

DiffServ/DTM. In this architecture (figure 7), the DiffServ MF filter defines the flows aggregation to an specific DTM channel. As DTM supports the dynamic reallocation of bandwidth, it supports dynamic signalling of the amount of bandwidth (e.g. RSVP protocol) to be allocated on the channel. But it is out of scope of this paper.

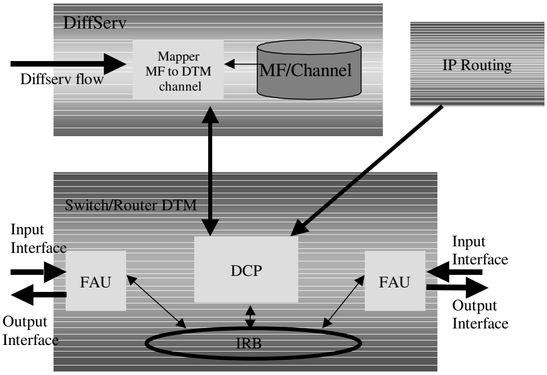


Fig. 7. DiffServ/DTM Architecture.

5 IETF IntServ Model Aggregated Reservations

The IETF IntServ model [11] demands a lot of processing and memory on the routers along the path (source-destination), because it is a per flow policing, scheduling, classification and reservation method. The aggregation of reservations [12] simplifies the classification and scheduling meanwhile it maintains less reservation states. It diminishes the bandwidth reservation demand too.

The guaranteed service (GS) [13] specified by this model guarantees bandwidth and delay but wastes a lot of resources, especially for low bandwidth (characteristic of the actual Internet traffics [3]) and short delay (necessity of real-time applications), even though the reservations are aggregated. We confirm this by calculating the bandwidth aggregated reservation needs for GS service in some links of the UUNET Europe Internet Backbone (figure 8) with arbitrary number of flows [3] [13] [14].

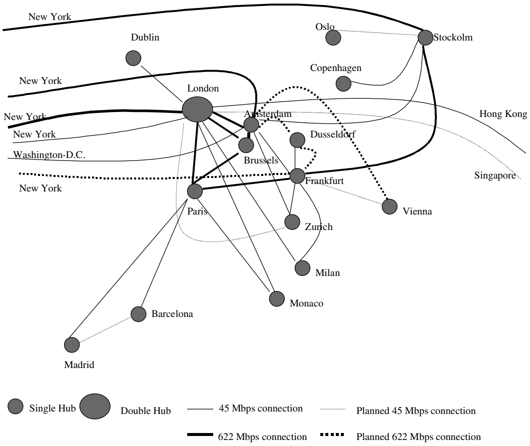


Fig. 8. European Internet Backbone (UUNET).

In table 5 we can see that on a link of 45 Mbps (Madrid-Paris) the maximum number of aggregated flows is around 300, with almost 100% of bandwidth use. In a real case, we must limit the resource for GS service, so in this example we may consider 200 flows agregated reservation. Without any QoS guarantees, these 200 flows, based on the Tspec (450,900) used in this study, need 90,000 Bps, which is 41,77 lower than with the GS service. So there is a high price to pay for the GS guarantees.

Table 5. Bandwidth reservation in function of the number of flows (Link Madrid-Paris).

Number of flows	Reservation (Bps)	% Total Bandwidth
200	3,759,375	6.6%
250	4,696,875	82.2%
300	5,634,375	100%
350	6,571,875	
400	7,509,375	
500	9,384,375	

In the link London-New York (622 Mbps), obviously we can support more aggregated reservation. For example, (table 6) to have a limit of 23.63% of the total bandwidth used by GS service, it can support 1000 RSVP aggregated flows. With this study we reaffirm that the IntServ/RSVP model is not adequate for a large network, because apart from the scalability problem, it wastes a lot of resources.

Table 6. Bandwidth reservation in function of the number of flows (Link London-New York).

Number of flows	Reservation (Bps)	% Total Bandwidth
900	16,539793	21.22%
1000	18,376,530	23.63%
2000	36,743,877	47.10%
3000	55,111,224	70.73%
4000	73,478,571	94.37%
5000	91,845,918	

6 Conclusion

Scalability is a crucial aspect in a Core Network, so it was the aspect that was given more attention in the paper. The DTM technology is more scalable than SDH/SONET in a full-mesh topology, frequently used in Core Internet networks. The DTM is a fast circuit switching solution that guarantees latency, a little jitter, constant little transfer delay, traffic isolation, flexible resource reservation and has less overhead than ATM [15]. It does not provide a complete QoS model like ATM, but can be a simple and cost effective solution when it becomes a standard specification.

Meanwhile, an evolution of the ATM environment already existent can be provided with the MPLS technology, as a good tunnelling solution, which is scalable, as an improvement to forward IP packets. Based on the architectures presented, the scalable DiffServ model can be the aggregator and differentiator of QoS flows.

Finally, we conclude that the cost to provide a guaranteed QoS in the IntServ model can be high, as it wastes a lot of bandwidth, even though traffic is aggregated. So this is not the appropriate model in a Core Network where a great number of flows share the same resource. With this the use of the DiffServ model on the proposed architectures is ratified.

Acknowledgements. Javier García's work is supported by the Spanish Ministry of Science and Technology under Projects TIC2000-0735 and TIC2002-04516-C03-03. This author would like to express his appreciation to the Programa Complutense del Amo for providing him a grant to stay at IBM Research Division. Part of this work was done while he was with the Information Storage Group at the IBM Almaden Research Center (San Jose, California, USA).

References

1. C. Bohm et al. *Fast Circuit Switching for the Next Generation of High Performance Networks*. IEEE Journal on Selected Areas in Communications, Vol. 14 (2), February 1996.
2. A. Girard. *Routing and Dimensioning Circuit-Switched Networks*. Addison-Wesley Publishing Company. 1990
3. K. Thompson et al. *Wide Area Internet Traffic Patterns and Characteristics*. IEEE Network. November/December 1997.
4. ITU-T Recommendation I.356. *B-ISDN ATM Layer Cell Transfer Performance*. 1996.
5. L. H. Ramfelt. *Performance Analysis of Slot Management in DTM Networks*. Technical Report TRITA-T. Dept. of Teleinformatics, KTH, Stockholm. January 1996.
6. S. Blake et al. *An Architecture for Differentiated Services*. IETF RFC2475. December 1998.
7. E. C. Rosen. *Multiprotocol Label Switching Architecture*. Draft-ietf-mpls-arch-04.txt. February 1999.
8. D. Awduche et al. *Extensions to RSVP for Traffic Engineering*. Draft-ietf-mpls-rsvp-lsp-tunnel-02.txt. March 1999.
9. P. Goyal et al. *Effect of Number of Drop Precedences in Assured Forwarding*. Globecom´99. March 1999.
10. P. Newman et al. *Ipsilon´s General Switch Management Protocol Specification Version 2.0*. RFC2297. March 1998.
11. R. Brader, D. Clark and S. Shenker. *Integrated Services in the Internet Architecture: an Overview*. RFC1633. June 1994.
12. F. Baker et al. *Aggregation of RSVP for IPv4 and IPv6 Reservations*. Draft-baker-rsvp-aggregation-01.txt. February 1999.
13. S. Shenker, C. Partridge and R. Guerin. *Specification of Guaranteed Quality of Service*. RFC2212. September 1997.
14. J. Schmitt et al. *Aggregation of Guaranteed Service Flows*. IWQoS'99. May/June 1999.
15. C. J. Barenco, J. I. Moreno and A. Azcorra. *An architecture of QoS services for Core Internet Network over DTM*. IEEE ECUMN´2000. Colmar, France. October 2000.