LUND UNIVERSITY

QoS in Today's Internet

Andersson, Jens A; Hossein, Zaher

2004

[Link to publication](#)

Citation for published version (APA):
Andersson, J. A., & Hossein, Z. (2004). *QoS in Today's Internet*. Department of Communication Systems, Lund University.

Total number of authors:
2

General rights
Unless other specific re-use rights are stated the following general rights apply:
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.
• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal

Read more about Creative commons licenses: https://creativecommons.org/licenses/

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

LUND UNIVERSITY

PO Box 117
221 00 Lund
+46 46-222 00 00

QOS IN TODAY'S INTERNET (MAY 2004)

Jens A. Andersson, Zaher Hossein, LTH
jason@telecom.lth.se, d01zn@efd.lth.se

***Abstract*—To be able to guarantee service quality end-to-end Quality of Service has to be deployed. This thesis addresses the problems with applying QoS end-to-end over today's Internet. A rather pessimistic conclusion states that QoS over Internet is hard (impossible?) to realize without introducing virtual circuits or similar. The concept of flows and label switching is introduced. Some QoS techniques are presented.**

I. INTRODUCTION

INTERNET has evolved to become one of the most important global communication infrastructures. In some senses it is more important than normal telecommunication. It is the basis for many major companies, and education and research rely totally upon it. From carrying only a few but vital applications it is now possible to do anything from advanced decentralized computing (GRID) to network games via radio and TV broadcasting as well as plain telephony.

The greatest growth is found in the field of real time applications. Real time applications depend on a network that can transfer messages, or part thereof, and signalling end-to-end without delay and loss of information. But today's Internet protocols deliver best-effort service. Any constrain has up till now been solved by adding more capacity at bottlenecks. This has been possible both due to technical developments within the field and the fact that an increase in link capacity has been made available at a reduced price. This condition is now probably changing, and we therefore must search for other methods and techniques to deliver the service quality the users expect.

Internet is a packet switching network. Generically there is no flow concept implemented. Delay and other characteristics are hard to predict as well as the exact traffic path end-to-end. Any signalling needed to manage the session has to use the same channel as the data (in-band signalling). Any constrain that affects the data also affect the signalling. This is quite unlike the telephone system, where each data channel has its own circuit and signalling is done outside these data channels (out-of-band signalling).

Real time applications demand low delay and jitter. Delay is the amount of time it takes for at data unit to be transferred from the sender to the receiver. Jitter is the delay variation. An application can often handle some amount of delay but suffers severely from jitter. Delay and jitter increase if the available capacity in some part of the path is decreasing. When utilization is near the available capacity the session can experience packet loss, which in turn causes more delay and utilization since a lost packet must be resent.

Different applications have different needs. Therefore some sort of classification of end-to-end flows can be seen as a way to overcome some of the above-related problems. Different levels of best effort delivery are possible [Ferguson 1998]. But best effort is not enough for many applications. Network service providers must be able to deliver predicted or guaranteed quality of service. Since Internet is a network of networks, and each network can be built with many different types of links, Internet Protocol is the only common denominator or factor [Ferguson 1998].

This paper will discuss the introduction of Quality of Service, QoS, on Internet and some proposals to solve the QoS problem.

II. DEFINITION

The term QoS refers to a broad collection of networking technologies and techniques. The goal of QoS is to provide guarantees on the ability of a network to deliver predictable results. Elements of network performance within the scope of QoS often include availability (uptime), bandwidth (throughput), latency (delay), and error rate [Mitchell 2004].

III. SERVICE QUALITY AND QUALITY OF SERVICE

Ferguson and Huston [Ferguson 1998] define the difference between service quality and quality of service. They define service quality as "delivering consistently predictable service". Quality of Service, or QoS for short, is defined as a "method to provide preferential treatment to some arbitrary amount of traffic". The service quality is something the end user experiences and as the case might be expects, take for granted or even pays for. The QoS is the means to within the network provide the customer with service quality. Networks that are built according to best effort delivery might not be possible to adapt to a QoS scenario. It is also true that bad network design cannot be covered up by applying QoS to it [Ferguson 1998].

IV. ELASTIC AND NON-ELASTIC APPLICATIONS

Applications and protocols that are able to adapt in a well-behaved way to network constrains are called elastic. The TCP protocol with its built in congestion control is an example of an elastic protocol. If an application uses TCP as transport protocol the application becomes elastic. Real-time applications are by nature more non-elastic; instead of reducing the transfer rate or increase delay it is better if an audio stream reduces the quality of the application and thereby decreases bandwidth demand to reduce the contribution to the utilization and also the effects of network constrains on itself.

Many non-elastic applications and protocols are greedy and grabs all available resources. They thereby choke the well-behaved, elastic applications and protocols. UDP is a very simple transport protocol in the IP protocol suite. It lacks completely any kind of congestion control. It is up to the application to handle the congestion control as well as error control and error handling. An aggressive application that uses UDP may starve all TCP session sharing the same path to complete standstill. However, there are proposals how to implement TCP's congestion control into applications based on UDP [Mahdavi 1997].

V. THE MULTISERVICE NETWORK

The evolution within the Internet has up to now been driven by "IP over anything". The waste amount of different links available has made it necessary to adapt IP to run on any link type. The applications have been relatively few and of a more elastic appearance (web browsing, file transfer, mail transfer). Recently we have seen an increase in real time applications such as ip telephony and video conferencing but also broadcasting of audio and video streams. The evolution is now driven by "anything over IP". The Internet is becoming a multiservice network. However delivery is still done according to best effort. Toady's applications have different and diversified demands and the network must adapt to this fact. The problem is the common channel.

Another problem is the fact that the end-to-end users do not see the multitude of networks that are involved in transferring a session's data. The user expects the same service quality independent of who he/she is exchanging data with. The problem with providing service level agreements over domain boarders is discussed by Blefari-Melazzi, Sorte and Reali [Blefari-Melazzi 2003]. Due to the fact that flat rate is not fair in an overloaded network differentiated taxes must be implemented. This calls for Service Level Agreements that span provider domains, which in turn call for support in the networks for methods and techniques that realize SLA.

Blefari-Melazzi et al notes the difference between provider of service and provider of network infrastructure and capacity. In most cases the Internet providers also provided different customer services. Today services are often provided by companies not involved in networking. Instead they are customers of the network providers and as such they are demanding service level agreements not only by the network provider that supply the service provider with Internet access but also with the network providers collaborate partners.

A problem is that no flow or classification method scales well [Ferguson 1998]. Classification methods may work well in the limited domain, though.

VI. QOS-AWARE MULTICAST ROUTING

One of the main problems of the current Internet infrastructure is its inability to provide services at consistent quality of service levels. At the same time, many emerging Internet applications, such as tele-education, and teleconferencing, require multicast protocols that will provide the necessary QoS.

QoSMIC

QoS Multicast Internet protoCol is a multicast routing protocol for the Internet, that provides QoS-sensitive paths in a scalable, resource-efficient, and flexible way, which requires to support: 1) support QoS; 2) be scalable; and 3) use resources efficiently [Yan 2002].

QoSMIC is based on the notion of multiple path and QoS-aware routing. The multicast participant joining a group is offered multiple paths and selects the one that best satisfies its

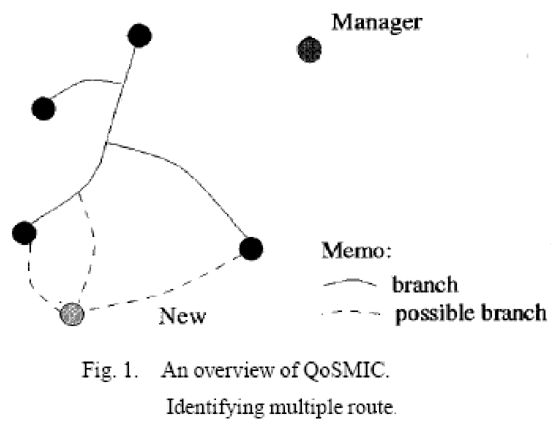

Figure 1. An overview of QoSMIC [Yan 2002]

QoS requirements.

A new node joining a multicast tree can select a path that suits best its QoS needs with respect to the collected information. QoSMIC implements a greedy routing heuristic, which attempts to find routers in the tree near the new router, as shown in Figure 1(a).

Another important novelty of QoSMIC is that it does not use a core router. This way, the tree is always "near" the active group members and, as a result, QoSMIC is more efficient than previous single-path core-based protocols; QoSMIC provides better end-to-end performance and accommodates more users.

VII. THE CONCEPTS OF FLOWS

We have earlier discussed the difference between Internets shared channel and the dedicated channel offered by the telephone system. A way to introduce a pseudo dedicated channel in the Internet is to introduce the flow concept. A flow is made up of all packets belonging to one session, for example the transfer of one file from one server to an end users computer. This concept has no support in generic Internet Protocol version 4, Ipv4.

When a flow is recognized it is also possible to mark it so that it can be allotted a service class. Nodes which the flow traverses can then handle different flows in different ways according to their service class. Methods for identifying and classifying flows and marking packets belonging to this flow have to be introduced [Bhatti 2000]. Both router hardware and software have to be implemented to realize new queue handling techniques and scheduling [Bhatti 2000, Ferguson 1998].

Even if we introduce the flow concept in the network layer this is not a guarantee for elastic applications and protocols to work well. The individual packets in a flow might have different paths. The packets going to the remote host might follow one path while the packet returning from the remote host might follow a completely different path. The paths might have different characteristics in respect of delay, jitter etcetera. TCP needs symmetrical paths to work well [Ferguson 1998]. Either the flow concept needs to deploy symmetrical paths or methods and procedures for QoS must be able to adapt to asymmetrical paths. A relation with ATM VC is close [Xiao 1999].

It is important that methods for QoS rely as little as possible to explicit signalling between nodes. This is due to the fact that signalling share the same channel as the data flows and any constrain that hits the data flow also affects signalling.

One method to preserve QoS is to allocate recourses for each flow according to each flow's demands. Methods for this are developed, but as said above, they scale badly or are problematic to implement in the ISP domain [Xiao 1999].

VIII. CONSTRAIN BASED ROUTING

Natural approaches to add Quality of Service to the Internet is of course to add constrain parameters to existing routing protocols [Xiao 1999]. Parameters or criteria can be reliability, delay, jitter, and available bandwidth. But adding information to the routing protocols increases the amount of data exchanged and handled by the network nodes.

A. Reliability

Reliability is a characteristic that a flow needs. Lack of reliability means substantial risk of losing a packet or acknowledgment, which entails retransmission.

B. Delay

Source-to-destination delay is another flow characteristic. Different application can tolerate delay in different degrees. Telephony, audio/video conferencing and remote log-in need minimum delay, while delay in file transfer or email is less important.

C. Jitter

Jitter is the variation in delay for packets belonging to the same flow. Real-time audio and video cannot tolerate high jitter. For example a real-time video broadcast is useless if there is a 2-ms delay for the first and second packets and 60-ms delay for the third and fourth.

D. Bandwidth

Different applications need different bandwidths. In video conferencing we need to send millions of bits per second to refresh a colour screen while the total number of bits in an email may not reach even a million.

IX. QOS: IMPROVING TECHNIQUES

Traffic Shaping

Shaping is used to create a traffic flow that limits the full bandwidth potential of the flow(s). Two techniques can shape traffic: leaky bucket and token bucket.

Leaky Bucket

The leaky bucket is a flow control mechanism that is designed to reduce the effect of the inevitable variability in the input stream into a node of a communication network. Leaky bucket allows input rate vary, when output rate remains constant, which can smooth out bursty traffic.

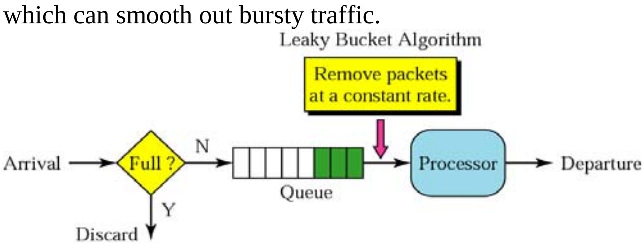

Figure 2. Leaky bucket [Forouzan 2004]

Token Bucket
Token bucket is another method to shape traffic. It allows bursty traffic at a regulated maximum rate. The token bucket algorithm allows idle hosts to accumulate credit for the future in the form of tokens.

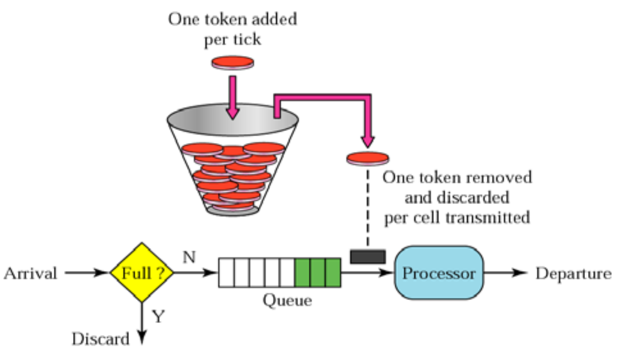

Figure 3. Token bucket [Forouzan 2004]

Sheduling
There are several scheduling techniques that are designed to improve quality of service. Here we represent three ways of queuing:

FIFO Queuing
In a First In First Out queuing, packets will waiting in a queue buffer (if the queue is not full), and will be processed when system is ready in order of their queue number as shown in the figure.

Priority Queuing
Each packet is assigned to a priority class. Packets are queued as their classes. Packets with higher priority are processed first. When the higher priority queues are empty, the packets on the lower priority queue will be processed. This will provide better quality of service than the FIFO.

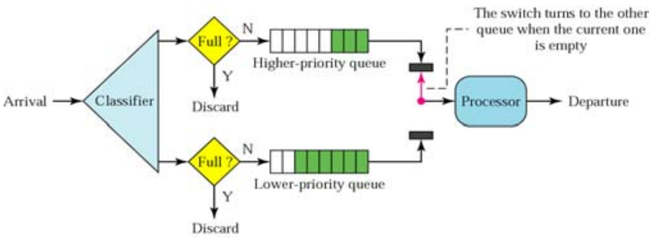

Figure 4. Priority Queuing [Forouzan 2004]

Weighted Fair Queuing
In the Weighted Fair Queuing the packets that arrives to the process is still divided in classes. The queues here are weighted based on the priority of the queues. Higher priority is similar to higher weight. The system processes packets from each queue depending on weight number of the queue. For example the system processes three packets from the queue with weight number 3, two packets from the queue with weight number 2 and one packet from the queue with weight number 1. (see Figure 5)

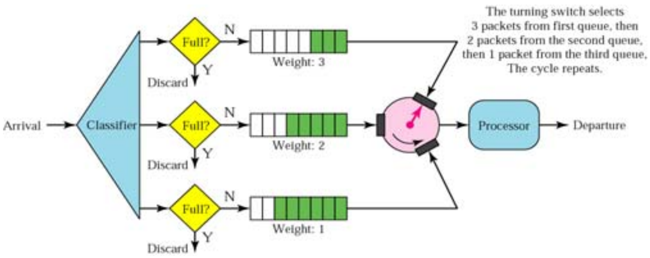

Figure 5. Weighted Fair Queuing [Forouzan 2004]

X. INTSERV

An initiative to implement QoS in Internet is Integrated Services, IntServ. Xiao et al describes the function [Xiao 1999]. Three service classes are proposed:

- Best-effort (default)
- Guaranteed service for application requiring fixed delay bounds
- Controlled-load service for applications requiring reliable and enhanced best-effort service

All involved routers must be able to reserve resources in order for them to be able to provide any guaranteed Quality of Service. It is therefore compulsory that they also can handle and manage states for each flow.

IntServ is implemented by four components:

- A signalling protocol
- An admission control method
- A classifier
- A packet scheduler

The signalling protocol is RSVP, Resource Reservation Protocol. All involved routers must understand RSVP.

Xiao et al specifies some major problems with the IntServ approach:

- Since the state of each individual flow has to be managed by all intermediate routers in a path the amount of information held in each router is proportional to the number of flows.
- All routers must have RSVP, admission control, classification according to a multiple of fields in both network layer, transport layer and possibly also application layer headers. All routers also must have resources for packet scheduling.
- This puts high demands on hardware resources, read memory and cpu capacity, in the routers.
- For Guaranteed Service an implementation of IntServ must be performed on all routers in the IntServ domain simultaneously. This is possible in limited

domains, not in the whole of Internet. Controlled-load Service can be deployed incrementally by implementing RSVP on bottleneck routers and tunnelling of the RSVP messages between them.

XI. DIFFSERV

Xiao et al also describes another approach called Differentiated Services or DiffServ. This model tries to overcome the problems found in the IntServ model. It uses the Type of Service field in the Ip header and calls it the DS field. The contents of this field together with a base-set of packet forwarding treatments define the per-hop behaviour. Several differentiated service classes can be defined. At ingress to a DiffServ domain each packet is marked by the ingress router. All intermediate routers in the path from source to destination only have to handle the packet according to the contents of the DS field.

Compared with IntServ this method has major advantages. The number of states is now only dependent on the number of classes defined. Intermediate core routers needs only to act upon the DS field. The resource consuming classification and marking of the packets are handled by the ingress routers. Therefore the DiffServ model scales better than the IntServ model [Xiao 1999, Blefari-Melazzi 2003].

DiffServ is also easier to implement incrementally. Intermediate routers that are not DiffServ-aware simply ignore the DS field and forward packets according to best effort. In this case none DiffServ-aware routers suffering from constrain could break the delivery of QoS end-to-end.

It is also possible for customers to perform the marking already in the source host.

DiffServ does not specify different service classes. It only describes the DS field and the per-hop behaviour. It is up to the network service providers to define classes according to the different service level agreements offered to customers. For DiffServ to work over the whole of Internet all network service providers must agree upon a common classification scheme.

XII. LABEL SWITCHING

DiffServ only classifies packets according to some service level scheme. Forwarding is then done by combining the packets destination and the service level defined by the DS field. MultiProtocol Label Switching, MPLS, takes the flow concept one step further.

Xiao et al [Xiao 1999] and Nortel [Nortel 2001] describes MPLS. It has evolved from Cisco Systems propriety protocol Tag Switching and has many similarities with ATM Virtual Circuits. At ingress to an MPLS domain each packet or flow is classified and added a new label switch header by the ingress router. At egress the label switch header is removed. The label header consists of a Label field, a Class Of Service field, and a Time To Live field. MPLS aware routers, called Label-Switched Routers, only use the label switch header when forwarding packets along a specified path. Information of different paths are distributed within the MPLS domain by a Label Distribution Protocol, for example Constrain-based Routing LDP or RSVP. By extending the Border Gateway Protocol version 4 LDP information can be exchanged over inter domain borders.

Label Switching means quicker handling of packets in intermediate nodes [Fgee 2003], since labelled paths are predefined and the handling of the label is much easier than handling of IP address and subnet mask. Different Label Switched Paths are made up for the different flow classes effectively building virtual or logical networks in the physical network. It is thereby possible to provide better than best-effort service for prioritized classes. This is true both for the first choice and the backup path [Xiao 2000]. A drawback is of course the fact that adding the labels at ingress nodes increases the delay and strain in these nodes [Fgee 2003, Xiao 2000].

MPLS is not dedicated for one unique network protocol, emphasized by the name MultiProtocol Label Switching. Internet Protocol version 6 has built in support for label switching, the Flow Label and Traffic Class [Fgee 2003]. This means that all information needed to uniquely classify packets is available in the Ipv6 header.

XIII. CONCLUSION

The Internet as we se it today is not capable of fully providing assured quality to end-users. For Quality of service to work end-to-end not only the customers network provider has to deliver network service according to some Service Level Agreement. All intermediate networks between source and destination have to provide for the same or similar SLA.

The demands for service quality from end-users is increasing as new applications evolve. The simple mechanism of a shared medium, which effectively has contributed to the very fast expansion of the Internet, both physically and geographically as well as utilisation, has problems coping with these demands. Different methods to overcome these problems have been proposed, but all seem to suffer from the same problem, namely how to implement Quality of Service and delivery of Service Level Agreements throughout the entire Internet. This is true both for administrative aspects as well as for implementation of methods in working technique. It remains yet to prove that Internet technology (read packet switching technology) is really simpler compared to classical technology (read circuit switching technology) when QoS and security aspects are integrated [Schiller 2003]. The concept of circuit switching has great advantages over packet switching when it comes to delivery of assured quality. A strive towards Virtual Circuits in the Internet ad modum ATM VC can be recognized in the MultiProtocol Label Switching approach and the Ipv6 header fields Flow Label and Traffic Class.

REFERENCES

[Ferguson 1998] Paul Ferguson and Geoff Huston (1998). Quality of Service in the Internet: Fact, Fiction, or Compromise? *Paper submitted for presentation at INET'98, Geneva, Switzerland*

[Mahdavi 1997] Jamshid Mahdavi and Sally Floyd. (1997). TCP-Friendly Unicast Rate-Based Flow Control. *http://www.psc.edu/networking/papers/tcp_friendly.html*

[Fgee 2003] El-Bahlul Fgee, Williams J. Phillips, William Robertson, and Shyamala C. Sivakumar (2003). Implementing QoS capabilities in Ipv6 networks and comparison with MPLS and RSVP. *Canadian Conference on Electrical and Computer Engineering 2003,* pages 851 - 854

[Xiao 2000] Xipeng Xiao, Alan Hannan, Brook Bailey, and Lionel M. Ni (2000). Traffic Engineering with MPLS in the Internet. *IEEE Network*, March/April 2000: 28 - 33

[Xiao 1999] Xipeng Xiao and Lionel M. Ni (1999). Internet QoS: A Big Picture. *IEEE Network*, March April 1999: 8 - 18

[Nortel 2001] Nortel Networks (2001). MPLS- An introduction to Multiprotocol Label Switching. *Nortel Networks White Paper.*

[Bhatti 2000] Saleem N. Bhatti and Jon Crowcroft (2000). QoS-Sensitive Flows: Issues in IP Packet Handling. *IEEE Internet Computing*, July/August 2000: 48 - 57

[Blefari-Melazzi 2003] N. Blefari-Melazzi, D. Di Sorte, and G. Reali (2003). Accounting and pricing: a forecast of the scenario of the next generation Internet. *Computer Communications,* 26 (2003): 2037 - 2051

[Yan 2002] Shuqian Yan, Michalis Faloutsos, and Anindo Banerjea (2002). QoS-Aware Multicast Routing for the Internet: The Design and Evaluation of QoSMIC. *IEEE/ACM ransaction on Networking*, Vol. 10, no. 1, February 2002: pages 54 – 66

[Forouzan 2004] Behrouz A. Forouzan (2004). Data Communications and Networking, 3rd edition. *McGraw-Hill Companies.* ISBN 0-07-251584-8

[Mitchell 2004] Bradley Mitchell. http://compnetworking.about.com/library/glossary/bldef-qos.htm. *About.com*

[Schiller 2003] Jochen Schiller (2003). Mobile Communications, 2nd edition. *Addison-Wesley*, page 453. ISBN 0-321-12381-6.