QoS's Downfall: At the bottom, or not at all!

Jon Crowcroft
University of Cambridge
15 JJ Thomson Avenue
Cambridge CB3 0FD, UK
jon.crowcroft@cl.cam.ac.uk

Steven Hand
University of Cambridge
15 JJ Thomson Avenue
Cambridge CB3 0FD, UK
steven.hand@cl.cam.ac.uk

Richard Mortier
Microsoft Research
7 JJ Thomson Avenue
Cambridge CB3 0FB, UK
mort@microsoft.com

Timothy Roscoe
Intel Research
2150 Shattuck Ave
Berkeley, CA 94704, USA
troscoe@intel-research.net

Andrew Warfield
University of Cambridge
15 JJ Thomson Avenue
Cambridge CB3 0FD, UK
andrew.warfield@cl.cam.ac.uk

ABSTRACT

Quality of Service (QoS) has been touted as a technological requirement for many different networks at many different times. However, very few (if any) schemes for providing it have ever been successful, despite a huge amount of research in the area of QoS provision. In this position paper we analyze some of the reasons why so many QoS mechanisms have failed to be widely deployed. We suggest two factors in this failure: the timeliness of QoS mechanisms (they rarely arrive when they are needed), and the inherent contradiction of layering QoS mechanisms over a best-effort network. We also give some thoughts on how future QoS research might increase its chances of successful deployment by better positioning itself relative to other developments in networking.

1. INTRODUCTION

A network that supports *quality of service* (QoS) is a network that presents its capabilities to the user and allows them to make *choices* as to the service they receive. Choices can be made in a number of dimensions: bandwidth (a little or a lot), availability (how guarantees of service are expressed in terms of unit and likelihood of delivery), latency (a lot or a little, and its variability or jitter), and loss (both the absolute amount and the quantum). A huge amount of QoS research over the years[1] has attempted to provide mechanisms to support such choices in a variety of networks, from the telephone network to the Internet. Unfortunately, almost none of this research has had impact in any way proportional to the expended time and effort. Why this apparent waste?

Our position is that this is due to the overly *reactive* nature of most QoS research. Only when the ratio of resources at the edges of a network to those available in the core of a network becomes high is the problem of service differentiation interesting, and only then do networking researchers begin to attack the problem in practical ways. When this ratio is low, any QoS mechanism appears redundant as most users receive the service they require anyway, and so the cost introduced by a QoS scheme appears unjustified, and research into QoS mechanisms appears unnecessary.

Networks are inherently dynamic systems and so this ratio changes over time. We observe that large networks like the Internet go through a cycle characterized alternately by periods where core bandwidth is deemed 'infinite,' and others where the network is deemed near 'congestion collapse.' The former often correspond to the introduction of new core network technologies, and the latter to the introduction of new access network technologies; both provide increases in bandwidth of at least an order of magnitude in their respective domains. Unfortunately, by the time a network is congested enough to require QoS, it is typically far too late to provide it: existing research is often out-of-date, and if suitable research does exist, engineering and deployment timescales are too long to make it preferable to simply increasing capacity.

In the remainder of this position paper, we argue that reactive QoS research is futile for two fundamental reasons. Firstly, the time required to research, engineer, and retrofit a specific QoS solution for an existing networking technology is typically longer than the remaining life-span of the congested core. Secondly, the alternative approach of layering QoS facilities over a network with no inherent QoS support is impractical, if not impossible.

It follows that there are two preconditions for successful realization of a QoS scheme: it must be *timely*, and it must be *inherent* in the network. Mechanisms providing guarantees must be researched and engineered before a network is deployed, and they must be inherent in the network from day one, even though there may be no immediate need for them. We draw a parallel here with well-established principles for engineering secure systems, which emphasize the

[1] For example, CiteSeer (http://citeseer.org/) lists 7,800 papers in the past 10 years with 'QoS' in the title.

ACM SIGCOMM 2003 Workshops August 25&27, 2003, Karlsruhe, Germany

need to design security in from the start.

The structure of the paper is as follows: we provide a general, working definition of Quality of Service in Section 2, and continue with a partial history of grand failures in QoS in Section 3. We then turn our attention to recent attempts to provide QoS using overlay networks, and the inherent problems with this approach in Section 4. Finally, we give some constructive ideas about how to move forward in Section 5 and conclude in Section 6.

2. QOS GOALS

QoS is about allowing the *user* to select between quantitative performance guarantees. By this definition, various existing mechanisms are useful but not sufficient to provide QoS; for example, *circuits* lack the flexibility to meet the needs of different users, while *weighted fair queuing* (WFQ) may provide protection between service classes, but does not offer quantitative or end-to-end guarantees. Something more is required.

To this end, many QoS architectures have been proposed, discussed, and even implemented. A successfully submitted QoS architecture generally includes components for monitoring resource usage, signalling QoS requirements, and performing admission control, policing, and scheduling. It may also include facilities such as traffic shaping and grooming, buffer management, authorization, access control, accounting, and billing.

Parameters of interest when providing QoS to customers include throughput, delay, jitter, loss, and availability. We note that a situation in which levels of service are strictly 'better' than one another requires an incentive for users to choose lower levels or the QoS scheme will be rendered meaningless. Of course, no such requirement exists if service levels are orthogonal — for example, allowing a user to select between low-latency and high throughput.

Even a packet switched network such as the Internet that does not advertise itself as providing QoS is still a *goal seeking system*: providers engineer it to achieve certain goals. Routers, paths, and cables are installed to ensure reliability, guaranteeing at least some level of connectivity and throughput in the face of equipment failure; and core networks are provisioned with sufficient capacity and FIFO buffering to achieve a target loss ratio assuming a particular traffic matrix.

Routes are found using a shortest path algorithm of some form, where 'shortest' is with respect to a particular metric such as throughput or delay. Since delay depends on load, routing decisions based on short term delay measurements will lead to unstable routes and so are not used. Route metrics are thus based on longer timescale measurements and sit within the domain of traffic engineering, which takes place at network provisioning timescales [11]. At the same time users are goal seeking: hence protocols such as TCP which greedily try to use all the available capacity subject to certain fairness criteria.

From an abstract point of view, we can observe that the goal is to divide a shared resource, belonging to one or more providers, between users who do not cooperate. For a QoS offering or *service level agreement* (SLA) to be meaningful — that is, to be more than a surface level agreement — it must implement certain functions including predictability/repeatability, isolation/protection, and auditability. Almost by definition, these functions cannot be offered through an overlay[2].

3. NET BALANCE: A MOVING TARGET

There have been a number of high-profile adventures in QoS, all of which have failed to have the expected impact. In this section we examine some of them and attempt to explain these unfortunate results. In essence, all failed (or are failing) due to the pace of change in networking: by the time the QoS mechanisms mature enough for widespread deployment, they are technologically less relevant and cannot deliver sufficient benefit given their overhead.

Figure 1 depicts the ratio of core to access bandwidth as it has changed with time. The peaks are times of plentiful core bandwidth, and include the early 1980s when the core was composed of 2Mb/s frame relay and access was mostly via 56kb/s or 64kb/s lines, and the present (the early 2000s), where the core makes use of various multigigabit technologies such as OC-192 and DWDM, but access is typically restricted to 100Mb/s Ethernet and lower. The troughs are droughts, where core bandwidth becomes close to being matched or exceeded by access bandwidth. Examples include the late 1970s, when access networks were often 3Mb/s Ethernets or 10Mb/s Cambridge Rings and the core was mostly 56kb/s leased lines; and the mid-1990s, where access had already reached today's levels, but the core had only deployed OC-12 technology such as 622Mb/s ATM.

In reality, the picture is more complex than this: the number of edge users, and the fraction of generated traffic that goes across the core are but two additional factors in the balance between core and access network capabilities. However, the point still stands: a crucial statistic is the amount of core traffic that can be generated by access links versus the capacity of the core itself. Furthermore, as the network scales with the number of edge users, even linear scaling of the core capacity will result in reduced delay and delay variance [6]. The law of large numbers pushes us to a regime where core QoS is fairly fruitless if the economics and technology can keep pace with the deployment of access networks. This last point is important, and led us to the discussion in this section.

3.1 X.25 Load-Sensitive Routing: too little, too early

In the late 1970s, in parallel with the early ARPANET development, the Europeans, especially in France and England, were working on early packet switched networking technologies, and gradually evolving the set of protocols that would become the CCITT X.25 standard. These protocols are characterized by a virtual circuit model, and use hop-by-hop flow control as well as hop-by-hop reliability and

[2] There are many other problems for QoS, such as deployment end-to-end, pricing, and the games with incentives one has to play — we leave these for others to dissect, possibly in this very workshop!

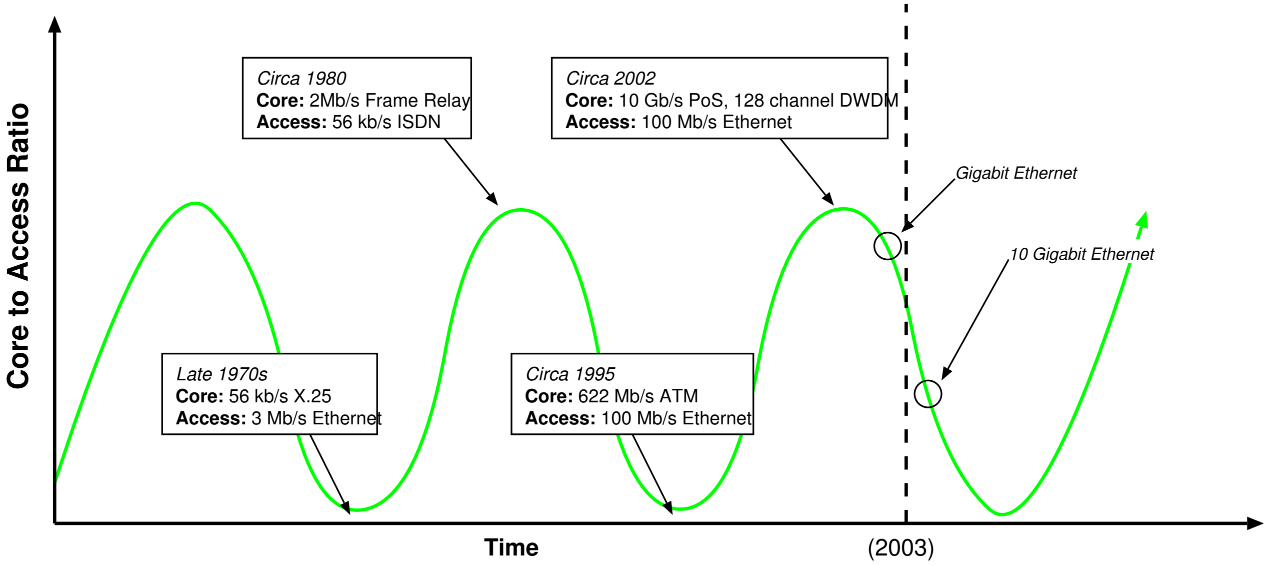

Figure 1: Relative core to access bandwidth over time.

end-to-end flow control. A perceived need for QoS in the France Transpac network and UK SRCNET projects led to attempts to provide admission control and load sensitive routing. It is relatively hard to find documentary evidence, but problems with oscillatory behavior were frequently reported anecdotally. Later, in the UK JANET network of the late 1980s, X.25 switches built by Netcomm were capable of a then impressive 2Mbps per linecard – and the company proudly announced that their systems operated internally using *datagram* mode, to achieve lower overheads. The need for QoS had been obviated by increased capacity.

3.2 ATM: too much, too late

Later still, the recurrent theme of QoS emerged in another connection oriented architecture. The Broadband ISDN standards developed by both the ITU (as the CCITT became) and the ATM Forum, entailed a very high degree of control. A stated goal was to unify the transmission networks for voice, data and broadcast services. However, by the time ATM was widely deployed, Packet over SONET was taking off reducing control requirements. Shortly after this, the speed of networks reached the point where the per-cell header and processing overheads were perceived as prohibitive at the high end. Its mismatch in satellite and other wireless transmission systems was also becoming apparent, although the notion of scaling and modularity in the architecture were very well thought through. It is also fairly clear that among signalling and routing researchers, the PNNI work was highly respected as a very well designed piece of work; however it was too much, and too late.

3.3 Intserv: too much, too early

Integrated Services was the IETF's attempt to do ATM. To all intents and purposes, RSVP with route pinning replaces Q.2931 and virtual circuits as the control plane, and the service classes replace the bearer services. The small difference was the alleged support for variable length packets, although admission control and scheduling were really only figured out for the MTU.

As with ATM, the overheads are high. Worse, router vendors had no experience with the hardware design necessary to do queuing and buffer management in hardware: after all, routers are simple so why should they?

Ironically, nowadays, routers are designed by switch hardware people, but it is too late to add complex queuing and scheduling strategies, and in the core, where it might matter, just chasing pure forwarding speed is the goal. The complexity of Intserv seems ill-suited in this case.

3.4 Diffserv: too little, too late

Differentiated services was an attempt to scale down the complexity of Integrated services — but chasing an exponential curve with a linear optimization is bound to fail. Diffserv was too little, too late. Among other problems, it never addressed issues of provider cooperation; many tier-1 ISPs asked "who needs it anyway in the current bandwidth glut?".

3.5 IP QoS: none at all?

This view of a bandwidth glut continues to this day, with the effect that the current deployed strategy for IP 'QoS' is essentially overprovisioning to achieve low loss. Anecdotal evidence suggests that many backbone networks operate at a load of 5-15%, with core Internet providers provisioning their networks to meet capacity approximately 3–6 months ahead of demand. Further anecdotes suggest that some also control the rate at which they sell significant bandwidth to customers to ensure that demand maintains this lag with respect to supply.

3.6 MPLS: layer 2 QoS?

What of MPLS? Since it is typically an intra-domain technique used for traffic engineering, and since networks today are global and multi-provider, we would argue that MPLS is not relevant to the scope of this discussion. In particular, MPLS is not a method to empower users — it is a network operator's tool — and so we can rule it out for the same reasons that we argued circuit networks do not provide meaningful QoS.

4. THE CASE AGAINST OVERLAYS: PUTTING THE RON IN WRONG

Overlay networks are the current networking incarnation of the philosopher's stone. They are being proposed as solutions for everything from availability [1] to mobility to multicast [4] to zero-configuration as well as (of course!) for QoS. We contend that, although overlays are without peer for building distributed algorithm testbeds and for deploying novel *qualitative* services, they will never be useful for deploying novel *quantitative* services. For example, where performance advantage can be gained, it is limited to the delay metric [13].

Essentially, an attempt to overlay QoS onto a network such as the Internet is an attempt to beat the two goal seeking systems of Section 2: the network seeking to achieve goals such as 'no loss,' and users seeking to maximize their own throughput. *Such attempts will fail* in anything other than the short term or the local region:

- Capacity on network links is purposely provisioned; if a routing overlay offers routes making use of currently 'redundant' capacity, then those routes will fail when that capacity is sooner or later used by non-overlay users [8] or users of other routing overlays. If the 'redundant' capacity is indeed redundant, then the provider has made a provisioning, traffic engineering or shortest-path weight-setting mistake allowing other network providers to punish them in the marketplace.
- An overlay attempting to provide a subset of users with guaranteed capacity in a lossy network cannot be TCP-friendly [5]. *All* users will therefore have an incentive to use the overlay, causing it to suffer congestive collapse. For a concrete example, consider an overlay providing guaranteed capacity by using Forward Error Correction (FEC) [15]: since the overhead introduced by using FEC to mask packet loss rate p must be at least p, the total offered traffic will rise proportionally from 1 to $1+p$. But both the loss rate and round trip time (RTT) are functions of the offered traffic: $throughput = \frac{1}{\sqrt{p} * RTT}$. Hence as the offered traffic increases, so do both the RTT and the loss incurred by a flow, while at the same time the capacity goes down. The result is that the system collapses.

The fundamental problem with any overlay QoS scheme is that it introduces another control system into the network, without clear understanding of the interactions between even existing control systems. This leads to increased instability unless damped to the point where the control loop in the QoS overlay is so slow as to make any guarantees provided useless. At the same time QoS overlays introduce yet more overhead to the network in terms of both per-packet encapsulation and additional routing overhead. All these problems are only exacerbated when multiple QoS overlays are introduced.

If these overlays are controlled so that all users use only one, then the overlay has become an integral part of the network, and can no longer be considered an overlay [12]. This leads to an example of the 'layered multiplexing considered harmful' argument [16]: functionality has effectively been pushed back into the network underneath existing protocols and services. This increases the multiplexing points in the system with the associated problems, runs the risk of breaking existing assumptions and their reliant components, and in any case, is likely to have been much more easily and efficiently done from the network's inception.

5. A BIT OF QOS

Thus for a QoS scheme to be successful it must not only be timely in its arrival so that it is useful, but also be inherent to the network. If it is not built-in to the network from the start, it will never be supported everywhere allowing economies of scale to make it cheap. However, since research and development into such schemes must begin while the results are 'clearly' not necessary, the associated overheads must be minimal so that benefits that only exist *in potentia* are not outweighed by imagined costs. We claim that one suitable mechanism for packet networks is simply a provision for network stations to signal congestion by marking a single bit in any packet.

If a single bit per packet is provided then the network is able to easily signal to the end-systems that congestion is beginning, signified by build-up in router queues. End-systems can then implement algorithms that use this information to achieve fair shares in proportion to various factors such as the amount a user is willing to pay for bandwidth [3], or in response to some externally imposed weighting [9].

Although some consider providing even this single bit initially too expensive, there are two mitigating factors. First, mechanisms such as Weighted Fair Queueing (WFQ) used similarly to be considered too expensive; they were only deployed as providers discovered that access bandwidths were becoming capable of overwhelming the core. Nowadays, WFQ is considered a requirement in a router and has consequently been engineered to the point where the cost is negligible.

Second, the Internet Protocol is blessed with a Type of Service octet[3], useful for a variety of purposes: at least two ISPs make use of this field to deploy differentiated services[4]. It has been possible to co-opt one of the bits in this octet for the purposes of Explicit Congestion Notification.

Although necessary this is not sufficient for our purposes. As in the heady days of development of the Integrated Services over Specific Link Layers, *the bit* must map into a true service over the ether, in the switch, and on the wire. Once this parsimonious state is reached, many complex policies

[3] Whether by luck or sagacity...

[4] We contend that this is a local matter and thus does not allow the *user* true end-to-end control over QoS, as argued in Section 1.

and services are enabled, and the community is empowered to experiment with a plethora of techno-economic innovations. Just such an explosion of creativity has happened in London, England following the introduction of a congestion charge for the use of the roads: the users themselves have proposed schemes including plans to share vehicles (*compression*), to trade travel days (*deferred download*), and to vary parking versus driving costs (*caching*).

In a similar vein, a number of luminaries in the research community have seen how ECN together with a family of congestion pricing strategies can be used to build a wide variety of mechanisms, and implement various policies through them. For example: a rapidly varying price can put such back pressure on the user that it effectively implements an edge-based admission control algorithm (the user being the decision part of the process); a marking scheme can be triggered by active queue management systems so that low bandwidth, low delay services are achievable (e.g. VOIP), alongside higher latency, higher demand applications; competition between providers who *specialise* in particular services and tariffs is also feasible.

In information theoretic terms, it is hard to imagine either a simpler signalling service (a single bit per packet), or a more frequent signalling interval (at least one signalling opportunity per round trip time). We thus believe that this is a strong argument against accusations of 'wasting resources,' since committing to such a cheap proactive QoS approach in times of 'plenty,' enables so many valuable (and profitable!) services in times of 'scarcity.'

6. CONCLUSION

So, what should one conclude from the state of QoS? We believe that the key conclusion to draw is that QoS research *must* be more forward-looking. We mean this both in terms of the technologies to which it will be applied, and the need to begin when there is clearly *no need* for the results! Some examples of what we believe might be suitable topics — and this list is by no means exhaustive — are: all optical packet buffer management and scheduling; wideband CDMA modified to provide priorities, perhaps by manipulation of chipping; and efficient multicast for both all-optical and wireless CDMA networks.

In considering the practical commercial implications for developing QoS mechanisms at such a point in time, we realize that it may be hard for a network operator to justify such a large up-front cost in the short term, regardless of the cost saving over a longer period. Against this, we argue that QoS uniquely provides wide-area network operators with a means to extract value from their network by charging for different levels of service. The inability to extract value from the network is the prime factor in the non-profitability of most long-haul IP networks today.

And the good news is that we only need to design something simple — recent advances such as Key and Kelly's [7] work on edge-only admission, and probe based admission, and Ostring et al's [2] work on price based differentiation, have shown that very simple mechanisms are necessary, and possibly sufficient; even if not sufficient, one can build on them. In his thesis [10], Peter Kim from HP showed that high utilization Integrated Services could be implemented on top of a 1-bit priority system that was available in the 100-VG-AnyLAN standard (in large bridged LANs).

Thus we would argue that for optical packet switching (which is arguably the way to go in the next 10 years or so for core devices) a 1-bit priority queue, possibly with ECN marking, ought to be considered — once systems are shipped in large numbers, we would expect the integration costs to mitigate the additional costs well. In WCDMA, we would argue that we need to consider price based transmit power and battery life management [2], and build on this toward some higher level, finer grain schemes. Again, integration (easier here) will make any such decision deliver efficiently.

Finally, whatever is done should be flexible. As Shenker argued so well [14], we do not want the mechanism to determine the policy.

7. REFERENCES

[1] Andersen, D. G., Balakrishnan, H., Kaashoek, M. F., and Morris, R. Resilient Overlay Networks. In *Proceedings of the 18th ACM Symposium on Operating System Principles (SOSP'01), Banff, Canada* (October 2001).

[2] Crowcroft, J., Gibbens, R., Kelly, F., and Ostring, S. Modelling incentives for collaboration in mobile ad hoc networks. In *Proceedings of WiOpt'03: Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks* (Mar. 2003).

[3] Gibbens, R., and Kelly, F. Resource pricing and the evolution of congestion control. *Automatica 35* (1999), 1969–1985.

[4] Jannotti, J., Gifford, D., Johnson, K., Kaashoek, M., and Jr, J. O. Overcast: Reliable multicasting with an overlay network. In *Proceedings of the Fourth Symposium on Operating System Design and Implementation (OSDI'00)* (Oct. 2000).

[5] Kelly, F. Charging and rate control for elastic traffic. *European Transactions on Telecommunications 8* (1997), 33–37.

[6] Kelly, F. Models for a self-managed Internet. *Philosophical Transactions of the Royal Society* (2000), 2335–2348.

[7] Kelly, F., Key, P., and Zachary, S. Distributed admission control. *IEEE Journal on Selected Areas In Communications 18*, 12 (2000), 2617–2628.

[8] Kelly, F., Maulloo, A., and Tan, D. Rate control in communication networks: Shadow prices, proportional fairness and stability. *Journal of the Operational Research Society 49* (1998), 237–252.

[9] Key, P., McAuley, D., Barham, P., and Laevens, K. Congestion pricing for congestion avoidance. Tech. Rep. MSR-TR-99-15, Microsoft Research, Feb. 1999. `http://www.research.microsoft.com/research/network/disgame.htm`.

[10] Kim, P. Resource reservation in shared and switched demand priority LANs. Available from `ftp://cs.ucl.ac.uk/darpa/pk_phd_thesis.ps.Z`, Sept. 1998.

[11] Mortier, R. Internet traffic engineering. Tech. Rep. UCAM-CL-TR-532, University of Cambridge, Computer Laboratory, Apr. 2002.

[12] Nakao, A., Peterson, L., and Bavier, A. A routing underly for overlay networks. In *Proceedings of ACM SIGCOMM, Karlsruhe, Germany* (August 2003).

[13] Qiu, L., Yang, Y. R., Zhang, Y., and Shenker, S. On selfish routing in Internet-like environments. In *Proceedings of ACM SIGCOMM, Karlsruhe, Germany* (August 2003).

[14] Shenker, S., Clark, D., Estrin, D., and Herzog, S. Pricing in Computer Networks: Reshaping the Research Agenda. In *The Internet and Telecommunications Policy*, G. Brock and G. Rosston, Eds. Lawrence Erlbaum Associates, 1996.

[15] Subramanian, L., Stoica, I., Balakrishnan, H., and Katz, R. OverQoS: Offering QoS using Overlays. In *First Workshop on Hot Topics in Networks (HotNets-I)* (Oct. 2002).

[16] Tennenhouse, D. L. Layered Multiplexing Considered Harmful. In *Proceedings of the 1st International Workshop on High-Speed Networks* (May 1989).