Market Managed Multi-service Internet (M3I): Economics driving Network Design

Bob Briscoe[1] David Songhurst[1]
Martin Karsten[2]*
[1]BTexact Technologies Research,
B54/130, Adastral Park, Martelsham Heath, Ipswich, IP5 3RE, UK
<{rbriscoe|dsonghurst}@jungle.bt.co.uk>
[2]School of Computer Science, University of Waterloo,
200 University Ave W, Waterloo, ON, N2L 3G1, Canada
<kalli@bbcr.uwaterloo.ca>

25 Oct 2002

Abstract

The fundamental economics of packet networking has led us to a mechanism to *guarantee* quality of service (QoS) with no flow handling in the core Internet, giving far better scalability, robustness and simplicity than previously. The same packet congestion mechanism is then generalised to encourage customers to manage *all* classes of traffic responsibly and with optimal utilisation. A vision of the future is given, driven by the inexorable logic of economics. The economic concepts behind the engineering are briefly explained.

Introduction

Controlling quality of service (QoS) between any two points on the Internet has been an ongoing research endeavour since the early 1990s. The problem of guaranteeing QoS for the duration of a session has been particularly difficult to reconcile with the connectionless design of the underlying packet network. To guarantee some flows it is necessary to suppress other traffic (admission control). Traditional approaches create a connection-oriented overlay over the packet network for a fixed proportion of total traffic. However, in order to maximise the value of the network, economics dictates that the proportion of guaranteed traffic should vary dependent on the demand for *all* traffic types determined separately along *each* path through the whole Internet. One would think this would require an extremely complicated mechanism, but our engineering design described in this paper is actually far simpler, more scalable and more inherently robust than all other current solutions, and it doesn't require radically new pricing models for retail customers.

This solution to the guaranteed QoS problem was one of the main results from the Market Managed Multi-service Internet (M3I) project. This article first focuses on that solution, then expands the scope to cover more general control of QoS using pricing, encompassing the relative quality of *all* traffic not just guaranteed sessions. Having outlined the concrete engineering, we explain the fundamental economics of packet networking that lies behind these designs. Although the engineering results stand on their own merits, understanding of the economic concepts behind them is necessary, both to appreciate why the engineering is maximally efficient and to prevent these subtle insights being lost in the process of incorporating the design into operational networks and their associated tariff structures.

Our aim is to make economically efficient use of network resources through correct pricing, while still allowing flexibility in the tariffs that each provider can offer. Network pricing that derives from fundamental economics can form the basis of specific tariff plans, or in the future it may be used in its raw form, at least for certain types of customer. We end this article with a brief vision of what is likely to happen into the distant future, by following the march of research on network economics to its logical conclusions.

An earlier article in this journal [21] outlined the approach of the M3I project. More recently, M3I's approach has been extended to wireless networks in the M4I project (Mobile M3I) the main results of which are also briefly included within.

*Dr Karsten conducted this work while in the Multimedia Communications Lab, Darmstadt University of Technology.

Guaranteed QoS over packet networks

Recent history

In the early 1990s the Integrated Services Architecture (Intserv [4]) proposed that flow requirements should be signalled (the control path) to every node on a reserved path, so that every node could then police every subsequent packet in the data path to check compliance with admitted reservations. By the end of the 1990s, it was realised that there was no need to police the data path any more often than on entry to each 'domain of trust' (administrative domain) [30]. This approach paralleled naturally with multi-protocol label switching (MPLS [24]), in the process of development around the same time, where the initial label for a switched path could be introduced at the ingress to a domain, effectively confirming that tagged traffic had already been policed within a reservation.

The differentiated services architecture (Diffserv [7, 2]) introduced in 1997 pragmatically avoided the need to police each flow separately except at the first trust boundary. In Diffserv, traffic is policed on entry to one of a small number of separate classes. Traffic entering a guaranteed class [20] must signal its request and, if admitted, the flow is only policed at the first network edge. At subsequent trust boundaries only the total amount of traffic in the guaranteed class need be policed, not each flow. The assumption is that, if the first domain allows too much traffic into the guaranteed class, the second domain will have to randomly choose which packets to demote from the class, damaging all flows a little, rather than focusing reduced service on a few. Therefore, bulk policing per class downstream provides sufficient incentive for correct per-flow policing upstream.

The efforts described so far correctly focus on efficiency in the more critical data path. In the control path, each reservation request is signalled using a reservation protocol, such as RSVP [31, 5], which also returns an admission control decision. The question of whether there is sufficient capacity on a path can either be determined by checking available capacity with all nodes on the path, as in Intserv, or by redirecting requests to a centralised 'oracle' per network domain. Such a 'bandwidth broker' [20], to which all reservation requests are made, must hold an up to date capacity map of the network so it can reliably accept or deny any request.

Prior to the invention of our scheme described below, difficult problems remained with all the above solutions:

- Per flow control path processing scales badly, increasing linearly with total load of flows across the core (whether handled by backbone routers or bandwidth brokers). Aggregation of reservations does not appear feasible, because there would be no way to disaggregate them on the other side — unlike the telephone network, reservations may all be of different sizes.
- Routing adaptation for each reservation is divorced from underlying packet routing (e.g. using route pinning), requiring extra complexity to re-route around failures or congestion.
- Management configuration is required to set the proportion of capacity assigned to guaranteed traffic for every path across the Internet. If certain paths have greater relative demand for guarantees than other paths, the resultant complexity of this management task becomes immense, so usually a compromise is settled on for all nodes and therefore all paths, considerably reducing overall network efficiency.

Some backbone operators consider it is more economic to over-provision their capacity than invest in any of the above QoS mechanisms. Guarantees can be maintained most of the time with this approach, but abnormal traffic patterns can converge on (or diverge from) popular locations (e.g. televotes, emergencies). Negative customer experience during these episodes negates much of the value of such excess investment.

Guaranteed QoS Synthesis

Below we describe how to guarantee QoS for flows across all the domains of the Internet but with only bulk treatment of traffic and absolutely no per-flow signalling except in access networks. Gateways between access and core synthesise fully guaranteed QoS from simple, bulk congestion signalling, effectively engineering the original theoretical idea published in [11] (also conceived within BT in parallel). The combination of strong not just statistical guarantees and an extremely simple core, even at interconnect points, is unique, giving the scheme the potential to become an Internet-wide solution, in preference to Diffserv, MPLS with traffic engineering (TE), bandwidth brokers, or over-provisioning.

Our solution uses three standard Internet QoS *protocols*, but all in a simpler *arrangement* to that in which they were originally designed. Thus, the reservation protocol (RSVP [31, 5]) is used, but not the Integrated Services Architecture [4]; differentiated services code points [19] are used, but not the Diffserv architecture; and explicit congestion notification (ECN [8, 23] — see box on p3) is used, but

Explicit congestion notification (ECN)

Before ECN, the only way a router could signal its congestion was by dropping packets. ECN was designed to allow a router to signal that it was approaching congestion by marking packets, allowing early avoidance of both congestion and retransmission delays. ECN involved redefinition of the IP packet header itself (specifically the last two bits of the differentiated services byte in both IPv4 and IPv6).

Each ECN-capable router probabilistically marks packets in proportion to the severity of the prevailing congestion as they enter its egress queue. The random early detection (RED) algorithm [10, 3] is used to determine the likelihood of marking each packet, dependent on the moving average of the recent (exponentially weighted) queue length. The simple RED algorithm is applied equally to all the packets arriving at an egress router interface, with no regard to flows.

In summer 2001, ECN was accepted by the Internet Engineering Task Force (IETF) as a Proposed Standard, although it had already been implemented for some time by the major router manufacturers. Standardisation of ECN was a significant event in the history of the Internet, given it is a far more robust way to achieve closed-loop control at the packet level than loss-detection and given designs using closed-loop control tend to be far simpler than open-loop.

not for its original role of end to end congestion control.

Fig 1 shows a 'ring-fence' of guaranteed QoS gateways surrounding a number of core network clouds. A few flows are shown entering or leaving each gateway, representing its attached access network. For clarity these flows are not shown crossing the core, except for one, which is highlighted along its length. On the outer, access network side of each gateway, our solution appears to be traditional, using the reservation protocol (RSVP) in the control path. However routers within the ring-fence of gateways around the core of the network do not have reservation recognition enabled, so reservations are silently treated as data packets in the interior. The mode of RSVP usage is also traditional, with the data sender preparing routers on each access network path by announcing the flow specification it intends to send (not shown). Data receivers may then respond with a reservation request back along the same set of routers (represented by the dashed arrow). The various data path processing steps applied to this flow are represented by circled numbers, (1) being traditional policing of the reservation in access network equipment.

The QoS gateways keep guarantees by only allowing traffic matching an accepted reservation to be tagged with a class of service chosen to represent 'guaranteed', denoted CoS_g in the figure. Any traffic not under a reservation, including traffic with a bit rate in excess of that reserved for it, is remarked to another class of service, denoted by CoS_b, before being allowed into the ring-fence (2). Guaranteed class traffic is given priority over other classes on all interior nodes within the ring-fence. Differentiated services code-points are set in data packet headers to denote these classes of service.[1] If any interior router experiences congestion it will mark a proportion of packets it forwards with ECN (3). On reaching the egress guaranteed QoS gateway on the far side of the ECN cloud, the fraction of ECN in arriving traffic is metered and stored (4). A value is stored for the aggregate traffic from each upstream guaranteed QoS gateway. Each value is inflated by the ratio between total reserved and total measured traffic for each path[2].

A new reservation request will be denied, if the inflated ECN fraction of traffic from the relevant upstream gateway exceeds a threshold. Reservations can be guaranteed because they will not be admitted unless congestion is below the threshold, and no guaranteed QoS gateway will allow guaranteed traffic to be added to any path beyond the point where the threshold is exceeded on any router. Traffic in other classes is not policed by the gateways, so it could continue to increase the level of congestion beyond the threshold on any path. However, because the guaranteed class is prioritised over others, any excess over the threshold can only consist of lower priority traffic and not harm the guaranteed traffic. Starvation of the lower priority class is not a problem because total load irrespective of class triggers admission control to the higher priority class.

Note that there is no need to decide what propor-

[1] Diffserv traffic conditioning agreements are not used.

[2] Smarter overbooking algorithms could be used for statistical guarantees.

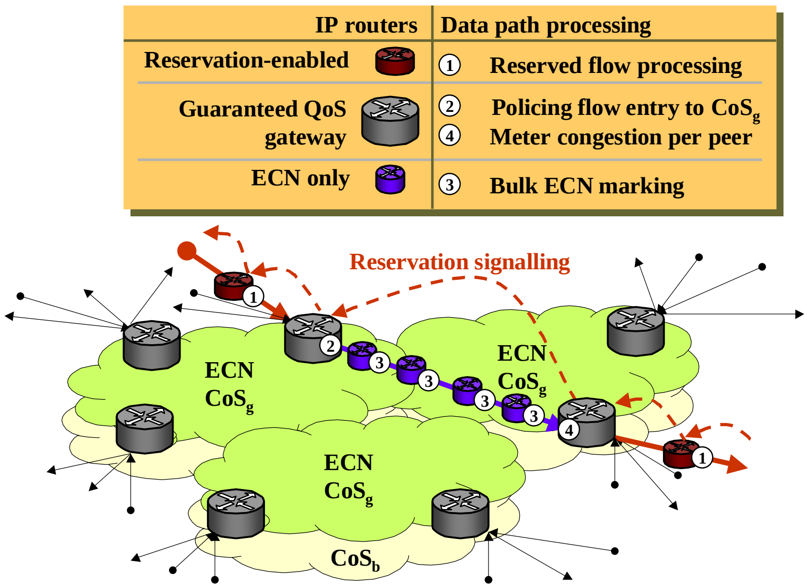

Figure 1: A ring of guaranteed QoS gateways

tion of capacity to set aside for guaranteed service, because congestion marking is arranged to represent congestion of *total* load across all classes. For instance, let us assume the threshold chosen for the onset of admission control is when the *total* load across all classes reaches 90% of capacity, giving a 10% safety margin. One day the ratio between demand for guaranteed and for other classes on one path may be 1:8, so when guaranteed traffic reaches 10% of load, the total will have reached 90% triggering admission control to prevent further reservations on that path. On another path, or on another day, the ratio may be 2:1. So guaranteed traffic may rise to 60% before admission control kicks in on that path, the total load again having reached 90%.

Given this scheme is so simple, why not use congestion onset to manage guarantees in the access network too? Unlike in core networks, the additional load of one extra flow can consume a significant proportion of the capacity of an access link. Therefore, inspecting congestion levels *before* adding a flow would not be a good test for remaining access capacity. Further, most access equipment is already designed to accept classical reservations — e.g. DSL, cable and wireless (but see later) including 3G and WLAN. Therefore guaranteed QoS gateways are best placed at the earliest point between access and core where statistical multiplexing can be exploited.

A complete prototype version of this guaranteed QoS gateway has been designed, implemented and validated [16, 15] by colleagues at Darmstadt University of Technology in collaboration with the partners of the M3I consortium, initiated and led by BTexact Technologies from Jan 2000 to Mar 2002. The experiments show that the dynamics of these gateways under a rapidly rising load are identical to those of an equivalent system consisting solely of traditional, reservation-capable routers. We may conclude that services guaranteed by this arrangement would be robust to surges in guaranteed demand, such as those experienced during televotes. However, validation on a production system would be necessary to fully support such claims.

It is even possible to ensure that the guaranteed service synthesised by our scheme has no congestion delay. The marking algorithm on each interior node can be arranged to start indicating congestion before there is a likelihood that the queue will ever be greater than one packet. But this requires software upgrades to add an alternative to the RED marking algorithm on all interior routers. Through experiment it was found that RED is anyway not well suited to indication of a long term load trend, being designed for fast reaction to queue build up. Therefore deployment would need a load-based [27], rather than queue-based algorithm, which could also provide delay avoidance through early marking. Each marking algorithm could be tailored very differently for each non-guaranteed class, although still driven by the underlying load from all classes. For instance, the best effort class could use the standard RED algorithm for ECN marking or dropping, but still driven by the length of the queue of all classes.

Our arrangement described above cannot absolutely promise to maintain guarantees during rerouting or failures within the ECN cloud, but re-routed guaranteed traffic[3] will simply push lower priority classes out of the way, causing increased dropping in the lowest, best effort class and TCP flows rapidly adapting to the new situation. The combination of re-routed and existing guaranteed traffic would have to exceed the total capacity (for all classes) of the alternate path to cause problems. But even then, reservations would fail gracefully, rather than catastrophically, degrading quality not tearing down reservations.

To summarise so far, the above scheme guarantees QoS sessions between any two points on the Internet. Scalability problems across core networks apparent in such schemes as MPLS-TE or Diffserv bandwidth brokers have been eliminated using just congestion marking across the core, ring-fenced by guaranteed QoS gateways converting from and to reservation signalling in access networks. Both core and border routers and links within the ring-fence have the simplicity of an over-provisioned network but the low capacity cost and guarantees of a managed solution. To support guarantees, the cost of over-provisioning is unlikely to undercut our scheme, if it must be sufficient to maintain QoS during surges of demand into unpredictably located hot-spots, for example during televotes or responses to emergencies. Gateways to the access network are required, which have slightly higher complexity than a Diffserv edge router, but unlike for Diffsev, require zero management.

Interconnect

Multiple providers may interconnect within the ring-fence of guaranteed QoS gateways. No special border routers are required between them. Only ECN marking is carried in packets across interconnect interfaces. So, while the commodities sold on the retail market are reservations, these do not exist on the wholesale market, because they are synthesised at the gateways. This not only raises the question of *how* to price, but also the more fundamental question of *what* should be priced, in order to encourage economic use of network resources.

First, in the access network we can assume, for brevity, that economic pricing of reservations to retail customers can be similar to telephony pricing. That is, a variable element sufficient to discourage unnecessary hogging of resources and a fixed capacity subscription element to recover the balance of the sunk cost of the infrastructure. The variable part would simply involve metering reservations by duration using time-of-day pricing (the price would also rise with reserved bandwidth, which is not a variable in telephony).

On the interconnect market, there will still be a need to cover the cost of infrastructure using fixed, capacity-related charges. A cursory analysis would conclude that a variable element is not needed for interconnect. Access providers seem to need no further incentive to prevent their users hogging the resources of an interconnect link unnecessarily. They already have every incentive to smooth their customers' use over the day, in order to minimise their own need for interconnect capacity.

However, a fixed charge for a local link gives no incentive to balance use of remote links further upstream in relation to their congestion. To give this incentive, interconnect pricing should include a variable element related to congestion along the *whole* length of each path. All that is necessary to achieve this is to set a fixed price per byte of congestion marked traffic — wholesale 'congestion pricing'. Simple bulk measurement of ECN marks is all that is needed, requiring just a simple counter per class of service at an interconnect point, totalling to just one item for the whole guaranteed class of service on the monthly interconnect bill. This simple pricing scheme gives the downstream access provider the incentive to introduce admission control on any one path when congestion is high — to cut off traffic that would cost more on the wholesale market than it would raise in revenue on the retail market, given the prevailing time-of-day price. So the guaranteed QoS gateway can be thought of as synthesising flat rate, fixed QoS sessions from congestion priced, bulk, connectionless traffic.

Because congestion markings are additive, if every interconnect contract is congestion priced, revenues will effectively flow upstream to the providers that added to the marking rate as traffic flowed downstream. In fact, revenues effectively flow to congested router interfaces in exact proportion to the marginal cost of upgrading them to remove the congestion. It may seem that a provider could undersize its network or over-declare congestion in order to profit from congestion pricing. However, in a competitive interconnect market, alternative routes will be chosen that use providers with better sized networks or more honest congestion marking.

Of course, providers may bilaterally agree to use other common pricing schemes for a variable element, e.g. volume charging. But private metering of the 'congestion price' will allow them to understand the 'true' price beneath these less economically efficient (and no less costly) schemes. As another example, it would be possible to understand

[3] Packets caught up a failing equipment and lines would be lost, but remember that no guaranteed packets will be sitting in queues.

whose interest a mutual peering agreement was really serving. The later section on economic principles expands on why a congestion price represents a 'true' price.

Retail market diversity

Clearly, the retail market has more complex requirements than just reservation of fixed sessions. The goal of a multi-service network recognises that a much larger and faster growing proportion of the market is elastic data, but the two must co-exist. An elastic application is one that is able to vary its rate over a large range in response to a changing constraint like congestion or pricing. File transfer and Web browsing are typical elastic applications, but streaming technologies using layered codings such as RealPlayer can also be elastic to a degree. In fact, the early research on congestion pricing that led up to the M3I project did focus primarily on elastic traffic.

Later (under Retail Flexibility) we discuss several traditional retail pricing models that could still be used by Internet retailers around a core network that uses congestion pricing. But first we examine the implications of a retail congestion charging model that has the capability to efficiently handle any mix of traffic types (elastic and inelastic) by allowing end-systems complete freedom in their use of network resources subject only to appropriate pricing.

Retail congestion charging

Today elastic applications use the TCP algorithm running on their terminal devices. TCP continues to increase bit-rate until it finds the limiting bottleneck on each user's path through the network. Congestion is signalled by a dropped packet, but in the near future ECN will provide advance warning of congestion. Elastic applications can and do react quickly to such congestion signals.

Currently the size of this reaction is standardised to a halving of the rate. However, it is possible for a non-co-operative sender to not react, or react less severely, by either not using TCP or altering the TCP software. Between congestion events it is also possible to increase the bit rate faster than the TCP standard. Any of these methods results in improved performance relative to others who are keeping to the rules.

Much effort has been put in to techniques for network operators to police users (whether end-user or service provider) to penalise those not complying with TCP's co-operative rules [9]. An alternative approach is to set a fixed price for congestion marks, causing the price on each path through the network to exactly track congestion on that path. Because the bottleneck is marking a set proportion of everyone's traffic, at any one level of congestion the charge seen by everyone sharing that bottleneck will be proportional to the average rate they choose.

It is then possible to allow end-system applications the freedom to react how they choose. Cheating is no longer an issue, because you pay for what you get at a price determined by what is available. Effectively, this creates a very simple but all encompassing QoS mechanism — a Market Managed Multi-service Internet (M3I).

Clearly users will need to have appropriate software support to remove the decision-making load of reacting to varying prices. Such support takes the form of preset QoS buying policies associated with each application task. These policies are simple information objects amenable to be exchanged between users, for example at session initiation when one user is agreeing how much it is willing to pay for another. For instance, they have been described using XML within the new generation of the session initiation protocol (SIP) used in Internet Telephony. The section on Economic Principles illustrates how several such buying policies can be represented as demand functions for different application types. Elsewhere we report on a Dynamic Price Handler agent which acts on behalf of the user to control transmission rate against congestion pricing driven by such policies [14].

User acceptability is a major issue facing congestion pricing as a retail service. Our user experiments and many reported elsewhere have shown that users like to know what they will be paying, and (as with insurance) they will usually be prepared to pay extra on average to have this predictability. A corollary borne out by our user experiments [13] is that users become far less interested in price stability when the price is high. Some examples can be identified where retail congestion pricing may be acceptable:

- Unattended applications where data can be transferred at zero or minimal charge because waiting for cheap periods isn't detrimental. Examples might include uploading to web or video servers (as opposed to downloading to clients), downloads where only the service provider pays the variable price, peer-to-peer file swapping, overnight data backups, slow-scan security video.

- Cellular networks where the high cost makes price-sensitive customers willing to forego some price predictability.
- Large customers (corporate sites, academic institutions) may accept congestion charging if it is clearly likely to reduce their overall charges. With large usage the variability of congestion charges will average out. To clarify, our corporate user experiments [13] have clearly shown the unacceptability of congestion pricing applied *within* a corporation. But applying dynamic pricing externally gives large users the flexibility to implement internal policies to control costs.

Retail flexibility

So far we have talked as if we are trying to find one pricing model for all Internet providers, but this could not be further from the truth — our aim is to encourage tariff flexibility for Internet retailers. In order to move directly to a retail congestion pricing model, providers would need to overcome several potential problems. These include the likelihood of strong user resistance (see above), the need for end-system software as discussed above, and natural concerns about the impact of introducing completely new revenue and service models.

The congestion pricing approach described above simply gives us the base case, which a provider undercuts at its peril. For instance, a provider who offers a flat-rate subscription encourages users to use their full capacity at less cost to themselves than if they were charged a congestion price. The provider only has itself to blame if users invent applications to exploit this invitation (peer-to-peer file-swapping services are a classic example). In other words, flat charging is not 'incentive compatible'. Such users will reduce the value of the network for all other users, because that is what the congestion price represents - the external effect on others.

So how would a provider take advantage of tariff flexibility?

Volume charging. Again, one should take congestion pricing as the base case, which is like volume charging, but with only bytes in marked packets paid for instead of all bytes. A provider that offers volume charging is charging more than the natural (congestion) price during uncongested periods, and thus runs the risk of attracting traffic only during busy periods. To counter this the provider might use several different volume prices at different times of day based on historic congestion information (as with the present PSTN) — an approximation to congestion pricing. The ideal tariff design depends on what tariffs are offered by competitors, and what the customers will accept (e.g. does the predictability and familiarity of a volume-charged tariff outweigh the possibility of getting a cheaper service elsewhere?).

Flat charging. It is difficult to make flat charging incentive compatible, but nonetheless its popularity with customers cannot be disputed, as it allows freedom to experiment without fear of variable charges. One solution is to sell just a limited amount of flat rate access capacity to each user at a high enough price to ensure that anyone wanting to completely fill this capacity all the time would have a cheaper alternative tariff. By limiting the capacity sold at flat rate, the network service to other users is still protected if all flat rate users fill this capacity at the same time.

Traffic conditioning agreements. A more sophisticated version of flat rate is where the user is contractually bound not to complain if service is poor when a certain traffic profile is exceeded. Such contracts are only appropriate for certain corporate customers of sufficient size for aggregate traffic to be predictable and of sufficient expertise to understand the implications of the contract. Using TCAs for residential traffic would require too much costly over-provisioning in the access network to reduce the risk of coincidental usage causing severe service impairment.

To summarise so far, congestion pricing has been shown to be a useful test case which should not be undercut by any other tariff scheme. Incidentally it also has potential as a tariff in its own right.

In the interests of brevity, we have omitted mention of some other important issues in the retail market that have been fully addressed in the research behind this paper.

Market structure & edge pricing [1]. With congestion pricing, as with other pricing schemes, charges may need to be paid by either sender or receiver or apportioned between them. Price signals carried through ECN marking are seen by receivers, and can be reflected back to senders via acknowledgement packets. It is natural to assume an *edge-pricing* model whereby the edge providers set the retail prices for marks and core network providers are paid by edge providers. This structure provides the opportunity for a separate clearing-house business role with the ability to collect appropriate payments from both senders and receivers and to apportion the revenue across edge network providers.

Price-setting and revenue. Congestion pricing solves the micro-economic problem of relating pricing to resource usage and demand in a fair way leading to efficient network usage, but it does not solve the macro-economic problem of setting prices to achieve revenue targets in a competitive environment. It is likely that providers will introduce congestion pricing alongside subscription or fixed usage charges (perhaps reducing these to compensate), with the aim of maintaining a stable revenue in the long term and providing appropriate incentives for service differentiation and efficient network operation in the short term.

Wireless is different

In wireless sub-networks, a significant and variable part of the spectrum resource is consumed by interference, which unlike legitimate customers' traffic is outside the operator's control. Also capacity between two points varies due to shadowing etc. whether the nodes or the shadow are moving. So all guarantees in wireless are largely statistical (no-one expects PSTN-like guaranteed calls over wireless).

The complexity of adding reservation mechanisms in the access network and a QoS gateway into the core just to give guarantees which are nullified by the radio environment makes little economic sense. One would think that removing the complexity of a guarantee mechanism from the network would require additional complexity on the terminal device to adapt to the unguaranteed environment. But most wireless applications are already designed with the ability to adapt to conditions (variable rate codecs etc.), because of the unpredictable radio environment.

Therefore, although congestion pricing is unlikely to be considered seriously as a retail wireless tariff, it will be highly beneficial to use congestion prices as an underlying parameter to control radio channel allocations etc. as well as to determine base-station and access point placement on longer timescales. For example, a method of service differentiation by assigning wireless resources in proportion to user-declared weights is proposed in [26].

The shadow-pricing approach has the added benefit that it could seamlessly integrate with resource control mechanisms on the fixed network and across different wireless technologies (wireless LAN, 3G etc.). So if congestion invariably occurs in wireless links, but occasionally a flash crowd causes congestion in the fixed network, the simplicity of explicit congestion notification signalling covers both scenarios, and the reaction can be the same irrespective of how and where congestion arose.

Economic Principles

Congestion pricing is favoured by economic theorists because it conveys correct incentives to customers leading to fair sharing of network resources. In this section we discuss some of the economics issues of congestion pricing for communications, and in particular the response of end-system applications, for a pure retail congestion pricing model.

How should a fair congestion price be assessed? In an important early paper [18] MacKie-Mason and Varian proposed a *smart market* for packet-switched capacity. In their proposal (which proved to be influential even though impractical to implement) users submit bids for their packets to be carried in a given time-slot. The network accepts these bids from the highest downwards, rejecting the lowest bids if capacity is insufficient. Crucially, the accepted packets are not charged what they bid but instead are charged the price corresponding to the highest rejected bid (if all bids can be accepted within a time-slot then there are no charges). Users are therefore paying a fair congestion price — the cost to others of the congestion that they are causing. Prices defined in this way are known as *shadow prices*, and they have the desirable property of *incentive compatibility* — users have no incentive to bid anything other than their true valuation.

This time-slotted model was not practical to implement, but Kelly *et al* [12, 17] investigated instead the use of packet marking to represent price feedback (ECN — see box on p3). If end-systems are *elastic* with respect to price, they will vary their rates up or down as the marking rate varies. Kelly *et al* showed that such a network would have a stable equilibrium which is optimal in the economic sense that total welfare is maximised.

Congestion pricing requires end-system applications to use buying policies in the form of demand curves (demand for rate as a function of price). A demand curve encapsulates knowledge of the range of QoS acceptable for an application, and also gives the elasticity with price — QoS requirements are not absolute. Some examples of demand curves are as follows:

(a) An elastic application that is able to vary its rate arbitrarily in response to price (Figure 2).

(b) An application such as real-time video streaming that is able to vary its rate within certain limits (Figure 3). Real-time applications are likely to have both a maximum rate beyond which no further real improvement in quality is gained, and a minimum rate below which the quality is so bad that it is never worth using.

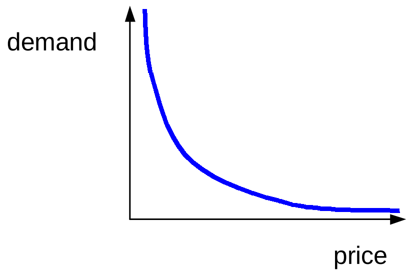

Figure 2: Demand function for an elastic application

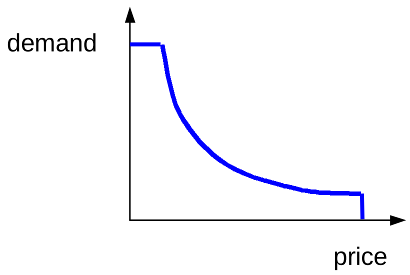

Figure 3: Demand function for a semi-elastic real-time application

(c) A real-time application with only one possible rate (Figure 4).

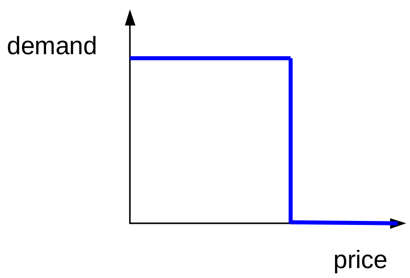

Figure 4: Demand function for an inelastic real-time application

Elastic applications like (a) are best served by rate-sharing — TCP encapsulates one such demand function. However applications like (b) and (c) need admission control — they do not benefit from rate-sharing when the price is too high. This is the principle applied by the guaranteed QoS gateway — it converts congestion pricing within the network into an admission-controlled access interface. The admission control policy of the gateway corresponds to the policy that would be applied by inelastic applications (this is discussed further in the next section) and is thus economically efficient.

The M3I project [28, 29] devoted much effort to analysis of economic and performance issues of congestion pricing, including:

- economic interaction between providers using different pricing schemes
- performance of networks using congestion pricing
- design of marking algorithms for routers, and of rate control algorithms for end-system applications

Future vision

Self-admission control

We started this paper with a description of a gateway that synthesises guaranteed QoS from a congestion priced wholesale network market. However, such a gateway is not strictly necessary. If a network operator does offer service priced by congestion, and users do run dynamic price handling agents as described above, surprisingly they can synthesise their own admission control.

If the application they are running becomes unusable below a certain bandwidth, this will be reflected in the use of an inelastic demand curve to control it (Figs 3 or 4). With a few probe packets, the agent can determine the current congestion price on the path to its corresponding terminal device. If this price is higher than the cut-off on its demand curve, the application will simply decide it is not worth starting — self admission control. Exactly the same admission control logic is applied here as in the the guaranteed QoS gateway, but without the involvement of a network operator.

However, self-admission control does not give guarantees equivalent to those from our guaranteed QoS gateways, as there is no isolation between guaranteed and non-guaranteed traffic. Users who have decided to admit themselves can find the price may rise during the session. As long as there is sufficient statistical multiplexing at bottlenecks, self-admission control can allow for this risk by setting its entry price slightly higher than the price it would be willing to sustain longer term.

Nonetheless, the fundamental cost economics of networks dictates there will always be more congestion in access networks where fan-out into more numerous, smaller links increases all resource costs relative to capacity. Hence congestion will generally arise at points of low statistical multiplexing in access networks, making self-admission control less workable in the more common cases. So although self-admission control will be possible and perhaps common in the longer term future, there will still be a role for an admission control *service* like the guaranteed QoS gateway, giving true guarantees rather than statistical ones. Unfortunately, the option of self-admission control will diminish the demand for this service, making it expensive to provide for a small audience.

Openness and new service evolution

The research reported here provides a simple mechanism for the quality of a network to be directly controlled by its customers, and only indirectly by the operator through congestion prices, which in turn are really controlled by the market.

This possibility has important commercial consequences for the communications industry. It further commoditises the role of a network operator. More importantly, QoS is the last remaining major item of added value inherently under a network operator's control[4]. Already there are problems with revenues from communications applications not feeding down to network operators, tending to suppress infrastructure investment below that appropriate to the market's potential.

However, one can view this as an opportunity rather than a problem. Customer-controlled QoS is only possible if the network operator chooses to offer raw congestion pricing to the retail market. We can expect operators to hold back from giving away control for many years, using the resulting revenues to invest in infrastructure to the benefit of all. However, one day QoS will no longer be a market differentiator, with all competing providers offering the same undifferentiated products. At this point, QoS control can be gradually opened up to customers and intermediaries, simply by offering a congestion priced tariff. This will open up the possibilities for application programmers and service providers to create innovative new products and services synthesised from the fundamental building blocks of packet QoS, rather than less flexible session QoS offered hitherto. Much the same process was witnessed late in the life of the PSTN, when its management functions were opened up to third parties through the publication of APIs by the Parlay Group[5].

[4]Of course, many customers will still pay network operators to offer QoS, security, session control, location awareness etc. even if they do now have the capability to do it themselves, but the issue is one of erosion of margins, not spectacular collapse of markets.

[5]www.parlay.org/

The 'end to end design principle' behind this tendency was first articulated in the early 1980s [25], to explain why a connectionless network with connections synthesised by end systems had greater potential than the connection-oriented equivalent. One of its original authors has since co-authored a paper [6] questioning its continued relevance, pointing out a flaw in end to end design because it requires end systems to be trusted to co-operate with the goals of the whole community. The research reported in this paper and elsewhere [12] shows there is no flaw in the end to end design principle because pricing can maintain incentives to co-operate, giving a new lease of life to end to end design and the potential it offers for future evolution of new services.

Concluding remarks

Further work: Stability

In the present Internet, stability of traffic behaviour depends on the use by end-systems of the standard rate control protocols in TCP. With congestion pricing there would be no requirement to use such standards — pricing would provide the incentive to constrain rates appropriately. However there would be no constraint (other than round-trip times) on the speed with which end-systems react to changing prices, and there is a theoretical risk of oscillatory traffic behaviour if end-systems react rapidly with large rate changes.

In fact, file-transfer applications may indeed have an incentive to generate rapid rate changes [28]. On the other hand, M3I user trials have suggested that users of real-time video services have a positive preference for stable rates [13]. It is therefore not yet clear whether this would be a problem in practice, but it may not be difficult in principle to avoid instability — by constraining the speed of end-system reaction, or by damping the variation in network price signals — at some cost to overall efficiency.

Further information

See the M3I Web site[6] for information on the M3I project, including the public deliverables referred to in this article. Publications from the M4I project are collected via the home page of ICS FORTH's NetLab[7].

[6]www.m3i.org/

[7]www.ics.forth.gr/netlab/wireless.html

For BT staff, the internal M3I site[8] provides access to material yet to be publicly released, and information on seamless resource control across wireless and other networks can be found on the internal M4I home page[9].

Conclusions

This article has shown how driving Internet design from its fundamental economics can lead to elegant engineering solutions to some of the most challenging problems of Internet Architecture.

At the most general level, we have shown that there is no 'crisis of trust' inherent in the Internet's end to end design principle. Not only have we shown that there is no inherent trust problem in giving users control of TCP's congestion algorithm, but also that in the future congestion pricing can allow network operators to give even more control to customers and application developers if they choose. This would commoditise the market, but enable a new phase in the evolution of arbitrary new patterns of QoS usage on a per packet basis.

More specifically, we have presented the engineering design of our scheme to synthesise strong QoS guarantees from a purely connectionless network, avoiding per-flow processing or signalling across the whole of the core of the Internet within a ring-fence of guaranteed QoS gateways. The result is far simpler, more robust and more scalable than the best previous work. It works best with bulk congestion pricing on the wholesale market which is extremely cheap to meter. The gateways insulate retail customers from dynamic pricing, allowing traditional telephony-like charging models to be used.

All this has been enabled by the use of per packet explicit congestion notification as the 'common currency' to communicate the cost of one users packets on others. This brings the solutions under full closed-loop control with network resources automatically shared on each path in the most efficient possible ratio, for instance between guaranteed and non-guaranteed demand. In contrast, were an open-loop control solution like Diffserv to be used, either excessive management intervention would be required as bulk demand for each class changed on each path, or a compromise configuration would have to be resorted to, resulting in serious economic inefficiency.

[8] www.jungle.bt.co.uk/projects/m3i/home.html
[9] www.jungle.bt.co.uk/projects/bt-m4i/

Acknowledgements

Some of the work reported here was funded in part by the EU Fifth Framework Project M3I (Market-Managed Multiservice Internet, IST-1999-11429). The M3I Consortium comprised BTexact Technologies, Hewlett Packard Ltd, Telenor, Athens University of Economics and Business, Darmstadt University of Technology, ETH Zürich and FTW Vienna. In addition, work on seamless resource control across wireless and wired networks is a collaboration between BTexact Technologies and ICS FORTH in Greece.

Glossary

CoS Class of Service

Diffserv Differentiated Services

DSL Digital Subscriber Line

ECN Explicit Congestion Notification

IETF Internet Engineering Task Force

IP Internet Protocol

Intserv Integrated Services

LAN Local Area Network

M3I Market-Managed Multi-service Internet

M4I Market-Managed Mobile Multi-service Internet

MPLS-TE Multi-Protocol Label Switching - Traffic Engineering

PSTN Public Switched Telephone Network

QoS Quality of Service

RED Random Early Detection

RSVP Resource Reservation Protocol

SIP Session Initiation Protocol

TCA Traffic Conditioning Agreement

TCP Transport Control Protocol

XML Extensible Markup Language

Authors' biographies

Bob Briscoe joined BT in 1980 and now leads the Edge Lab, one of the Research Labs of BTexact Technologies. In the late-1980s he managed the transition to IP of many of BT's R&D networks and systems. In the mid-1990s he represented BT on the HTTP working group of the IETF and in the ANSA distributed systems research consortium, which led to the creation of the OMG and CORBA. In 2000 he initiated and was technical director of the Market Managed Multi-service Internet (M3I) consortium. His published research, standards contributions and patent filings are in the fields of loosely coupled distributed systems, scalable network charging and security solutions (esp. multicast), managing fixed and wireless network loading using pricing and on the structure of communications markets. He is also studying for a part-time PhD at University College London.

Dave Songhurst is a research consultant at BTexact. Until 1995 he led the performance engineering section at BT Laboratories, working on traffic studies and network modelling and performance. In 1995 he left BT and worked for three years with Lyndewode Research, Cambridge, on the CA$hMAN project — a collaborative project in the European ACTS programme dealing with multi-service network charging schemes. He then joined BTexact to work on the M3I project — Market Managed Multi-Service Internet. He is the editor of the book "Charging Communication Networks: from Theory to Practice".

Dr. Martin Karsten studied joint computer science and business economics at the University of Mannheim, Germany and received his diploma degree in 1996. In December 1996, he joined the Multimedia Communications Lab at Darmstadt University of Technology. He received his PhD in computer science from Darmstadt University of Technology in July 2000 and has been a lecturer and group leader responsible for various research projects afterwards. Since September 2002, he has been an assistant professor in the School of Computer Science at the University of Waterloo, Canada. His current research interests are network layer technologies for IP networks with a strong emphasis on prototype development and experiments.

References

[1] Ragnar Andreassen (Ed.). M3I; Requirements specifications; reference model. Deliverable 1, M3I Eu Vth Framework Project IST-1999-11429, URL: http://www.m3i.org/, July 2000.

[2] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss. An architecture for differentiated services. Request for comments 2475, Internet Engineering Task Force, URL: rfc2475.txt, December 1998.

[3] B. Braden, D. Clark, J. Crowcroft, B. Davie, S. Deering, D. Estrin, S. Floyd, V. Jacobson, G. Minshall, C. Partridge, L. Peterson, K. Ramakrishnan, S. Shenker, J. Wroclawski, and L. Zhang. Recommendations on queue management and congestion avoidance in the Internet. Request for comments 2309, Internet Engineering Task Force, URL: rfc2309.txt, April 1998.

[4] R. Braden, D. Clark, and S. Shenker. Integrated services in the Internet architecture: an overview. Request for comments 1633, Internet Engineering Task Force, URL: rfc1633.txt, June 1994.

[5] R. Braden (Ed.), L. Zhang, S. Berson, S. Herzog, and S. Jamin. Resource ReSerVation Protocol (RSVP) — version 1 functional specification. Request for comments 2205, Internet Engineering Task Force, URL: rfc2205.txt, September 1997.

[6] David Clark and Marjory Blumenthal. Rethinking the design of the Internet: The end-to-end arguments vs. the brave new world. In *Proc. Telecommunications Policy Research Conference (TPRC'00)*, URL: http://www.tprc.org/abstracts00/rethinking.pdf, September 2000.

[7] David D. Clark. A model for cost allocation and pricing in the Internet. In *Proc. MIT Workshop on Internet Economics*, URL: http://www.press.umich.edu/jep/works/ClarkModel.html, March 1995.

[8] Sally Floyd. TCP and explicit congestion notification. *ACM SIGCOMM Computer Communication Review*, 24(5):10–23, October 1994. (This issue of CCR incorrectly has '1995' on the cover).

[9] Sally Floyd and Kevin Fall. Promoting the use of end-to-end congestion control in the Internet. *IEEE/ACM Transactions on Networking*, 7(4):458–472, August 1999.

[10] Sally Floyd and Van Jacobson. Random early detection gateways for congestion avoidance. *IEEE/ACM Transactions on Networking*, 1(4):397–413, August 1993.

[11] Richard J. Gibbens and Frank P. Kelly. Distributed connection acceptance control for a connectionless network. In *Proc. International Teletraffic Congress (ITC16), Edinburgh*, pages 941–952, URL: http://www.statslab.cam.ac.uk/~frank/dcac.html, 1999.

[12] Richard J. Gibbens and Frank P. Kelly. Resource pricing and the evolution of congestion control. *Automatica*, 35(12):1969–1985, December 1999.

[13] David Hands (Ed.). M3I user experiment results. Deliverable 15 Pt2, M3I Eu Vth Framework Project IST-1999-11429, URL: http://www.m3i.org/private/, February 2002.

[14] Arnaud Jacquet (Ed.). M3I; price reaction: Detailed design. Deliverable 12.2, M3I Eu Vth Framework Project IST-1999-11429, URL: http://www.m3i.org/private/, February 2002.

[15] Martin Karsten and Jens Schmitt. Admission control based on packet marking and feedback signalling – mechanisms, implementation and experiments. Technical Report TR-KOM-2002-03, TU-Darmstadt, URL: http://www.kom.e-technik.tu-darmstadt.de/publications/abstracts/KS02-5.html, May 2002.

[16] Martin Karsten (Ed.). GSP/ECN technology & experiments. Deliverable 15.3 PtIII, M3I Eu Vth Framework Project IST-1999-11429, URL: http://www.m3i.org/, February 2002. (updated by [15]).

[17] Frank P. Kelly, Aman K. Maulloo, and David K. H. Tan. Rate control for communication networks: shadow prices, proportional fairness and stability. *Journal of the Operational Research Society*, 49(3):237–252, 1998.

[18] Jeffrey K. MacKie-Mason and Hal R. Varian. Pricing the Internet. In B. Kahin and J. Keller, editors, *Public Access to the Internet*. Prentice-Hall, Englewood Cliffs, New Jersey or URL: http://www.sims.berkeley.edu/~hal/people/hal/papers.html, 1994.

[19] K. Nichols, S. Blake, F.Baker, and D. Black. Definition of the differentiated services field (DS field) in the IPv4 and IPv6 headers. Request for comments 2474, Internet Engineering Task Force, URL: rfc2474.txt, December 1998.

[20] Kathleen Nichols, Van Jacobson, and Lixia Zhang. A two-bit differentiated services architecture for the Internet. Request for comments, Internet Engineering Task Force, URL: rfc2638.txt or URL: http://diffserv.lcs.mit.edu/Drafts/draft-nichols-diff-svc-arch-00.pdf, July 1999. Based on draft-nichols-diff-svc-arch-02 (April, 1999, originally Nov 1997).

[21] Huw Oliver and David J. Songhurst. Market managed multiservice Internet. *Journal of the IBTE*, 2(1), January–March 2000. (This article first appeared as [22]).

[22] Huw Oliver and David J. Songhurst. Market managed multiservice Internet. *Telektronikk*, 96(2):38–44, 2000.

[23] K. K. Ramakrishnan, Sally Floyd, and David Black. The addition of explicit congestion notification (ECN) to IP. Request for comments 3168, Internet Engineering Task Force, URL: rfc3168.txt, September 2001.

[24] E. Rosen, A. Viswanathan, and R. Callon. Multiprotocol label switching architecture. Request for comments 3031, Internet Engineering Task Force, URL: rfc3031.txt, January 2001.

[25] Jerome H. Saltzer, David P. Reed, and David D. Clark. End-to-end arguments in system design. *ACM Transactions on Computer Systems*, 2(4):277–288, November 1984. An earlier version appeared in the Second International Conference on Distributed Computing Systems (April, 1981) pages 509–512.

[26] Vasilios A. Siris. Resource control for elastic traffic in CDMA networks. In *Proc. MobiCom 2002*, URL: http://www.ics.forth.gr/netlab/wireless.html, September 2002. ACM.

[27] Vasilios A. Siris, Costas Courcoubetis, and George Margetis. Service differentiation and performance of weighted window-based congestion control and packet marking algorithms in ECN networks. *Computer Communications*, 2002.

[28] David J. Songhurst (Ed.). M3I global interaction models. Deliverable 11, M3I Eu Vth Framework Project IST-1999-11429, URL: http://www.m3i.org/, June 2001.

[29] David J. Songhurst (Ed.). M3I global interaction models — Further results. Deliverable 11 PtII, M3I Eu Vth Framework Project IST-1999-11429, URL: http://www.m3i.org/, March 2002.

[30] Ion Stoica and Hui Zhang. Providing guaranteed services without per flow management. *Proc. ACM SIGCOMM'99, Computer Communication Review*, 29(4):81–94, October 1999.

[31] Lixia Zhang, Stephen Deering, Deborah Estrin, Scott Shenker, and Daniel Zappala. RSVP: A new resource ReSerVation protocol. *IEEE Network*, September 1993.

Document history

Version	Date	Author	Details of change
A	10 Oct 2002	Bob Briscoe	First draft started 26 Sep 2002.
B	11 Oct 2002	Bob Briscoe	Merged with Dave S version and demarcated.
C	21 Oct 2002	Bob Briscoe	Dave's sections finished. Bob still going.
D	22 Oct 2002	Bob Briscoe	First complete draft.
1	25 Oct 2002	Bob Briscoe	Added co-author MKarsten & minor corrections.