Transport Area Working Group
Internet-Draft
Intended status: Informational
Expires: May 15, 2008

B. Briscoe
BT & UCL
T. Moncaster
L. Burness
BT

November 12, 2007

Problem Statement: We Don't Have To Do Fairness Ourselves
draft-briscoe-tsvwg-relax-fairness-00

Status of this Memo

Copyright Notice

Abstract

Nowadays resource sharing on the Internet is largely a result of what applications, users and operators do at run-time, rather than what the IETF designs into transport protocols at design-time. The IETF now needs to recognise this trend and consider how to allow resource sharing to be properly controlled at run-time.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

Table of Contents

1. Introduction 3
2. What Problem? 4
  2.1. Two Incompatible Partial Worldviews 4
    2.1.1. Overlooked Degrees of Freedom 7
  2.2. Average Rates are a Run-Time Issue 8
  2.3. Protocol Dynamics is the Design-Time Issue 9
3. Concrete Consequences of Unfairness 10
  3.1. Higher Investment Risk 11
  3.2. Losing Voluntarism 12
  3.3. Networks using DPI to make Choices for Users 13
  3.4. Starvation during Anomalies and Emergencies 14
4. Security Considerations 15
5. Conclusions 15
6. Acknowledgements 15
7. Comments Solicited 15
8. References 15
  8.1. Normative References 15
  8.2. Informative References 16
Editorial Comments . . . .
Appendix A. Example Scenario 19
  A.1. Base Scenario 19
  A.2. Compounding Overlooked Degrees of Freedom 20
  A.3. Hybrid Users 21
  A.4. Upgrading Makes Most Users Worse Off 21
Authors' Addresses 23
Intellectual Property and Copyright Statements 25

1. Introduction

The strength of the Internet is that any of the thousand million or so hosts may use nearly any network resource on the whole public Internet without asking, whether in access or core networks, wireless or fixed, local or remote. The question of how each resource is shared is generally delegated to the congestion control algorithms available on each endpoint, most often TCP.

We (the IETF) aim to ensure reasonably fair sharing of the congested resources of the Internet [RFC2914]. Specifically, the TCP algorithm aims to ensure every flow gets a roughly equal share of each bottleneck, measured in packets per round trip time [RFC2581]. But our efforts have become distorted by unfair use of protocols we intended to be fair, and further by the attempts of operators to correct the situation. The problem is we aim to control fairness at protocol design-time, but resource shares are now primarily determined at run-time--as the outcome of a tussle between users, app developers and operators.

For instance, about 35% of total traffic currently seen (Sep'07) at a core node on the public wireline Internet is p2p file-sharing {ToDo: Add ref}. Even though file-sharing generally uses TCP, it uses the well-known trick of opening multiple connections--currently around 100 actively transferring over different paths is not uncommon. A competing Web application might open a couple of flows at a time, but perhaps only actively transfer data 1-10% of the time (its activity factor). Combining 50x less flows and 10-100x lower activity factor means the traffic intensity from the Web app can be 500-5,000x less. However, despite being so much lighter on the network, it gets 50x less bit rate through the bottleneck.

The design-time approach worked well enough during the early days of the Internet, because most users' activity factors and numbers of flows were in proportion to their access link rate. But, now the Internet has to support a jostling mix of different attitudes to resource sharing: carelessness, unwitting self-interest, active self-interest, malice and sometimes even a little consideration for others. So although TCP sets an important baseline, it is no longer the main determinant of how resources are shared between users at run-time.

Just because we can no longer control resource sharing at design time, we aren't saying it isn't important. In Section 3, we show that badly skewed resource sharing has serious concrete knock-on effects that are of great concern to the health of the Internet.

And we are not saying the IETF is powerless to do anything to help.

However, our role must now be to create the run-time _policy framework_ within which users and operators can control relative resource shares. So the debate is not about the IETF choosing between TCP-friendliness, max-min fairness, cost fairness or any other sort of fairness, because whatever we decide at design-time won't be strong enough to change what happens at run-time. We need to focus on giving principled and enforceable control to users and operators, so they can agree between themselves which fair use policy they want locally [Rate_fair_Dis].

The requirements for this resource sharing framework will be the subject of a future document, but the most important role of the IETF is to promote _understanding_ of the sorts of resource sharing that users and operators will want to use at run-time and to resolve the misconceptions and differences between them (Section 2.1).

We are in an era where new congestion control requirements often involve starting more aggressively than TCP or going faster than TCP, or not responding to congestion as quickly as TCP. By shifting control of fairness from design to run-time, we will free up all our new congestion control design work, so that it can first and foremost meet the objectives of these more demanding applications. But we can still quantify, minimise and constrain the effect on others due to faster average rate and different dynamics (Section 2.3). We can say now that the framework will have to encompass and endorse the practice of opening multiple flows, for instance. But alongside recognition of such freedoms will come constraints, in order to balance the side-effects on other users over time.

2. What Problem?

2.1. Two Incompatible Partial Worldviews

When looking at the current Internet, some people see a massive fairness problem, while others think there's hardly a problem at all. This is because two divergent ways of reasoning about resource sharing have developed in the industry:

- o IETF guidelines on fair sharing of congested resources [RFC2357],[RFC2309],[RFC2914] have recommended that flows experiencing the same congested path should aim to achieve broadly equal window sizes, as TCP does [RFC2581]. We will characterise this as the "flow rate equality" worldview, shared by the IETF and large parts of the networking research community.[Note_Window]

- o Network operators and Internet users tend to reason about the problem of resources sharing very differently. Nowadays they do

not generally concern themselves with the rates of individual flows. Instead they think in terms of the volume of data that different users transfer over a period [Res_p2p]. We will term this the "volume accounting" worldview. They do not believe volume over a period (traffic intensity) is a measure of unfairness in itself, but they believe it should be _taken into account_ when deciding whether relative bit rates are fair.

The most obvious distinction between the two worldviews is that flow rate equality is between _flows_, whereas volume accounting shares resources between _users_. The IETF understands well that fairness is actually between users, but generally considers flow fairness to be a reasonable approximation as long as users aren't opening too many flows.

However, there is a second much more subtle distinction. The flow rate equality worldview discusses fair resource sharing in terms of bit rates, but operators and users reason about fair bit rates in the context of byte volume over a period. Given bit rate is an instantaneous metric, it may aid understanding to convert 'volume over a period' into an instantaneous metric too. The relevant metric is traffic intensity, which like traffic rate is an instantaneous metric, but it takes account of likely activity _over time_. The traffic intensity from one user is the product of two metrics: i) the user's desired bit rate when active and ii) how often they are active over a period (their activity factor).

Operators have to provision capacity based on the aggregate traffic intensity from all users over the busy period. And many users think in terms of how much volume they can transfer over a period. So, because traffic intensity is equivalent to 'volume over a period', both operators and users often effectively share the same worldview.

To further aid understanding, Appendix A presents an example scenario where heavy users compete for a bottleneck with light users. It has enough similarities to the current Internet to be relevant, but it has been stripped to its bare essentials to allow the main issues to be grasped.

The base scenario in Appendix A.1 starts with the light users having TCP connections open for less of the time than heavy users (a lower activity factor). But, when they are active, they open as many connections as heavy users. It shows that users with a lower activity factor transfer less volume of traffic through the bottleneck over a period because, even though TCP gives roughly equal rate to each flow, the heavy users' flows are present more of the time.

The volume accounting view is _not_ that it is unfair for some users to transfer more volume than others--afterall the lighter users have less traffic that they want to send. However, they believe it is reasonable for users who put a heavier load on the system to be given less bottleneck bit rate than lighter users.

Appendix A.2 continues the example, giving the heavy users the added advantage of using 50x multiple flows, just as they do on the current Internet. When multiple flows are compounded with their higher activity factors, they can get 500-2,000x greater traffic intensity through the bottleneck.

Certainly, the flow rate equality worldview recognises that opening 50x more flows than other users starts to become a serious fairness problem, because some users get 50x more bit rate through a bottleneck than others. But the volume accounting worldview sees this as a much bigger problem. They first see 2,000x heavier use of the bottleneck over time, then they judge that _also_ getting 50x greater bit rate seems seriously unfair.

But are these numbers realistic? Attended use of something like the Web might typically have an activity factor of 1-10%, while unattended apps approach 100%. A Web browser might typically open two TCPs when active [RFC2616], while a p2p file-sharing app on a 512kbps upstream DSL line actively uses anything from 40-500 connections [az-calc]. Heavy users generally compound the two factors together (10-100x greater activity factor and 20-250x more connections), achieving anything from 200x to 25,000x greater traffic intensity through a bottleneck than light users.

The above question of what size the different worldviews think resource shares _should_ be is separate from the question of whether to _enforce_ them and how to (see Section 3.2). Within the volume accounting worldview, many operators (particularly in Europe) already limit the bit rate of their heaviest users at peak times in order to protect the experience of the majority of their customers.[Note_Neutral] But, enforcement is a separate question. Although prevalent use of TCP seems to be continuing without any enforcement, even the flow rate equality worldview generally accepts that opening excessive multiple connections can't be solved voluntarily. Quoting RFC2914, "...instead of a spiral of increasingly aggressive transport protocols, we ... have a spiral of increasingly ... aggressive applications").

To summarise so far, one industry worldview aims for equal flow rates, while the other prefers an outcome with very unequal flow rates. Even though they both share the same intentions of fairer resource sharing, the two worldviews have developed subgoals that are

fundamentally at odds.

2.1.1. Overlooked Degrees of Freedom

So which worldview is correct? Actually, our reason for pointing out these divergent worldviews is to show that both contain valuable insights, but that each also highlights weaknesses in the other. Given our audience is the IETF, we have tried to explain the volume accounting worldview in terms of flow rate equality, but volume accounting is by no means rigorous or complete itself. Table 1 identifies the three degrees of freedom of resource sharing that are missing in one or the other of the two worldviews.

Degree of Freedom	Flow Rate Equality	Volume Accounting
Activity factor	X	Y
Multiple flows	X	Y
Congestion variation	Y	X

Table 1: Resource Sharing Degrees of Freedom Encompassed by Different Worldviews; Y = yes and X = no.

Activity factor: We have already pointed out how flow rate equality does not take different activity factors into account. On the other hand, volume accounting naturally takes the on-off activity of flows into account, because in the process of counting volume over time, the off periods are naturally excluded.

Multiple flows: Similarly, it is well-known [RFC2309] [RFC2914] that flow rate equality does not make allowance for multiple flows, whereas counting volume naturally includes all flows from a user, whether they terminate at the same remote endpoint or many different ones.

Congestion variation: Flow rate equality, of course, takes full account of how congested different bottlenecks are at different times, ensuring that the same volume must be squeezed out over a longer duration, the more flows it competes with. However, volume accounting doesn't recognise that congestion can vary by orders of magnitude, making it fairly useless for encouraging congestion control. The best it can do is only count volume during a 'peak period', effectively considering congestion as either 1 everywhere during this time or 0 everywhere otherwise.

These clearly aren't just problems of detail. Having each overlooked whole dimensions of the problem, both worldviews seem to require a

fundamental rethink. In a future document defining the requirements for a new resource sharing framework, we plan to unify both worldviews. But, in the present problem statement, it is sufficient to register that we need to reconcile the fundamentally contradictory worldviews that the industry has developed about resource sharing.

2.2. Average Rates are a Run-Time Issue

A less obvious difference between the two worldviews is that flow rate equality tries to control resource shares at design-time, while volume accounting controls resource shares once the run-time situation is known. Also the volume accounting worldview actually involves two separate functions: passive monitoring and active intervention. So, importantly, the run-time questions of whether to and how to intervene can depend on policy.

The "spiral of increasingly aggressive applications" [RFC2914] has shifted the resource sharing problem out of the IETF's design-time space, making flow rate equality insufficient (or perhaps even inappropriate) in technical and in policy terms:

Technical: At design time, it is impossible to know whether a congestion control will be fair at run-time without knowing more about the run-time situation it will be used in--how long flow durations will be and whether users will open multiple flows.

Policy: At design time, we cannot (and should not) prejudge the 'fair use' policy that has been agreed between users and their network operators.

A transport protocol can no longer be made 'fair' at design time--it all now depends how 'unfairly' it is used at run-time, and what has been agreed as 'unfair'.

However, we are not saying that volume accounting is the answer. It just gives us the insight that resource sharing has to be controlled at run-time by policy, not at design-time by the IETF. Volume accounting would be more useful if it took a more precise approach to congestion than either 'everything is congested' or 'nothing is congested'.

What operators and users need from the IETF is a framework to judge and to control resource sharing at run-time. It needs to work across all a user's flows (like volume accounting). It needs to take account of idle periods over time (like volume accounting). And it needs to take account of congestion variation (like flow rate equality).

2.3. Protocol Dynamics is the Design-Time Issue

Although fairness is a run-time issue, at protocol design-time it requires more from the IETF than just a policy control framework. Policy can control the _average_ amount of congestion that a particular application causes, but the Internet also needs the collective expertise of the IETF and the IRTF to standardise best practice in the _dynamics_ of transport protocols. The IETF has a duty to provide standard transports with a response to congestion that is always safe and robust. But the hard part is to keep the network safe while still meeting the needs of more demanding applications (e.g. high speed transfer of data objects or media streaming that can adapt its rate but not too abruptly).

If we assume for a moment that we will have a framework to judge and control _average_ rates, we will still need a framework to assess which proposed congestion controls make the trade-off between achieving the task effectively and minimising congestion caused to others, during _dynamics_:

o The faster a new flow accelerates the more packets it will have in flight when it detects its first loss, potentially leading many other flows to experience a long burst of losses as queues overrun. When is a fast start fast enough? Or too fast [RFC3742]?

o One way for a small number of high speed flows to better utilise a high speed link is to respond more smoothly to congestion events than TCP's rate-halving saw-tooth does [proprietary fast TCPs] [FAST],[RFC3649]. But then new flows will take much longer to 'push-in' and reach a high rate themselves.

o Transports like TCP-friendly rate control [proprietary media players], [RFC3448], [RFC4828] are designed to respond more smoothly to congestion than TCP. But even if a TFRC flow has the same average bit rate as a TCP flow, the more sluggish it is, the more congestion it will cause [Rate_fair_Dis]. How do we decide how much smoother we should go? How large a proportion of Internet traffic could we allow to be unresponsive to congestion over long durations, before we were at risk of causing growing periods of congestion collapse [RFC2914]?[Note_Collapse]

o TFRC has been proposed as a possible way for aggregates of flows crossing the public Internet to respond to congestion (pseudo-wire emulations may contain flows that cannot, or do not want to respond quickly to congestion themselves) [I-D.rosen-pwe3-congestion], [I-D.ietf-capwap-protocol-specification], [TSV_CAPWAP_issues].

But it doesn't make any sense to insist that, wherever flows are aggregated together into one identifiable bundle, the whole bundle of perhaps hundreds of flows must keep to the same mean rate as a single TCP flow.

In view of the continual demand for alternate congestion controls, the IETF has recently agreed a new process for standardising them [ion-tsv-alt-cc]. The IETF will use the expertise of the IRTF Internet congestion control research group, governed by agreed general guidelines for the design of new congestion controls [RFC5033]. However, in writing those guidelines it proved very difficult to give any specific guidance on where a line could be drawn between fair and unfair protocols. The best we could do were phrases like, "Alternate congestion controllers that have a significantly negative impact on traffic using standard congestion control may be suspect..." and "In environments with multiple competing flows all using the same alternate congestion control algorithm, the proposal should explore how bandwidth is shared among the competing flows."

Once we have agreed that average behaviour should be a policy issue, we can focus on the dynamic behaviour of congestion controls, which is where the important standards issues lie, such as preventing congestion collapse or preventing new flows causing bursts of congestion by unnecessarily overrunning as they seek out the operating point of their path.

As always, the IETF will not want to standardise aspects where implementers can gain an edge over their competitors, but we must set standards to prevent serious harm to the stability and usefulness of the Internet, and to make transports available that avoid causing _unnecessary_ congestion in the course of achieving any particular application objective.

3. Concrete Consequences of Unfairness

People have different levels of tolerance for unfairness. Even when we agree how to measure fairness, there are a range of views on how unfair the situation needs to get before the IETF should do anything about it. Nonetheless, lack of fairness can lead to more concretely pathological knock-on effects. Even if we don't particularly care if some users get more than their fair share and others less, we should care about the more concrete knock-on effects below.

3.1. Higher Investment Risk

Some users want more Internet capacity to transfer large volumes of data, while others want more capacity to be able to interact more quickly with other sites and other users. We have called these heavy and light users, although of course, many users are mix of the two in differing proportions.

We have shown that heavy users can use applications that open multiple connections, so that TCP gives the light users very little of a bottleneck. But unfortunately, upgrading capacity does little for the light users unless the heavy users run out of data to send (which doesn't tend to happen often). In the reasonably realistic example in Appendix A.4, the light users start off only being able to use 10kbps of their 2Mbps line because heavy users are skewing the sharing of the bottleneck by using multiple flows. But a 4x upgrade to the bottleneck, which should add 500kbps per user if shared equally, only gives the light users 30kbps extra.

But, the upgrade has to be paid for. A commercial ISP will generally pass on the cost equally to all its customers through its monthly fees. So, to rub salt in the wound, the light users end up paying the cost of this 500kbps upgrade but we have seen they only get 30kbps. Ultimately, extreme unfairness in the sharing of capacity tends to drive operators to stop investing in capacity. Because all the light users, who experience so little of the benefit, won't be prepared to pay an equal share to recover the costs--the ISP risks losing them to a 'fairer' competitor.

But there seems to be plenty of evidence that operators around the world are still investing in capacity growth despite the prevalence of TCP. How can this be, if flow rate equality makes investment so risky? One explanation, particularly in parts of Asia, is that some investments are Governernment subsidised. In the US, the explanation is probably more down to weak competition. In Europe, the main explanation is that many commercial operators haven't allowed their networks to become as unfair as the above example--they have made resource sharing fairer by _overriding_ TCP's flow rate equality.

Competitive operators in many countries limit the volume transferred by heavy users, particularly at peak times. They have effectively overriden flow rate equality to achieve a different allocation of resources that they believe is better for the majority of their customers (and consequently better for their competitive position). Typically these operators use a combination of tiered pricing of volume caps and throttling of the heaviest so-called 'unlimited' users at peak times. In this way they have removed some of the investment risk that would otherwise have resulted if flow rate

equality had been relied on to share congested resources.

3.2. Losing Voluntarism

Throughout the early years of the Internet, flow rate equality resulted in approximate fairness that most people considered sufficient. This was because most users' traffic during peak hours tended to correlate with their access rate. Those who bought high capacity access also generally sent more traffic at peak times (e.g. heavy users or server farms).

As higher access rates have become more affordable, this happy coincidence has been eroded. Some people only require their higher access rate occasionally, while others require it more continuously. But once they all have more access capacity, even those who don't really require it all the time often fill it anyway--as long as there's nothing to dissuade them. People tend to use what they desire, not just what they require.

Of course, more access traffic requires more shared capacity at relevant network bottlenecks. But if we rely on TCP to share out these bottlenecks, we have seen how those who just desire more can swamp those who require more (Section 3.1).

Some operators have continued to provision sufficiently excessive shared capacity and just passed the cost on to all their customers. But many operators have found that those customers who don't actually require all that shared infrastructure would rather not have to pay towards its cost. So, to avoid losing customers, they have introduced tiered volume limits (this hasn't happened in the US yet though). It is well known that many users are averse to unpredictable charges [PMP] (S.5), so many now choose ISPs who limit their volume (with suitable override facilities) rather than charge more when they use more.

Thus, we are seeing a move away from voluntary restraint (within peak access rates) towards a preference for enforced fairness, as long as the user stays in overall control. This has implications on the Internet infrastructure that the IETF needs to recognise and address. Effectively, parts of the best effort Internet are becoming like the other Diffserv classes, with traffic policers and traffic conditioning agreements (TCAs [RFC2475]), albeit volume-based rather than rate and burst-based TCAs. (In fact, the addition of congestion accounting or policing need not be confined to just the best effort class.)

We are not saying that the Internet _requires_ fairness enforcement, merely that it has become prevalent. We need to acknowledge the

trend towards enforcement to ensure that it does not introduce unnecessary complexity into the basic functioning of the Internet, and that our current approach to fairness (embedded in endpoint congestion control) remains compatible with this changing world. For instance, when a rate policer introduces drops, are they equivalent to drops due to congestion? are they equivalent to drops when you exceed your own access rate? do we need to tell the difference?

3.3. Networks using DPI to make Choices for Users

We have seen how network operators might well believe it is in their customers' interests to override the resource sharing decisions of TCP. They seem to have sound reasons for throttling their heaviest users at peak times. But this is leading to a far more controversial side-effect: network operators have started making performance choices between _applications_ on behalf of their customers.

Once operators start throttling heavy users, they hit a problem. Most heavy volume users are actually a mix of the two types characterised in our example scenario (Appendix A). Some of their traffic is attended and some is unattended. If the operator throttles all traffic from a heavy user indiscriminately, it will severely degrade the customer's attended applications, but it actually only needs to throttle the unattended applications to protect the traffic of others.

Ideally, the threat of heavy throttling of all a user's traffic would encourage the user to self-throttle the traffic she least valued, in order to avoid the operator's indiscriminate throttling. But many users these days have neither the expertise nor the software to do this. Instead, operators have generally decided to infer what they think the user would do, using readily available deep packet inspection (DPI) equipment.

An operator may infer customer priorities with honourable intentions, but such activity is easily confusible with attempts to discriminate against certain applications that the operator happens not to like. Also customers get understandably upset every time the operator guesses their priorities wrongly.

It is well documented (but less well-known) that user priorities are task-specific, not application-specific [AppVsTask]. P2p filesharing can be used for downloading music with some vague intent to listen to it some day soon, or to download a critical security patch. User intent cannot be inferred at the network layer just by working out what the application is. The end-to-end design principle [RFC1958] warns that a function should only be implemented at a lower layer after trying really hard to implement it at a higher layer.

Otherwise, the network layer gradually becomes specialised around the functions and priorities of the moment--the middlebox problem [RFC3234].

To address this problem of feature creep into the network layer, we need to understand whether there are valid reasons why this DPI is being deployed to override TCP's decisions. We shouldn't deny the existence of a problem just because one solution to it breaks a fundamental Internet design principle. We should instead find a better solution.

3.4. Starvation during Anomalies and Emergencies

The problems due to unfairness that we have outlined so far all arise when the Internet is working normally. However, fairness concerns become far more acute when a part of the Internet infrastructure becomes extremely stressed, either because there's much more traffic than expected (e.g. flash crowds), or much less capacity than expected (e.g. physical attack, accident, disaster).

Under non-disaster conditions, we have already said that fair sharing of congested resources is a matter that should be decided between users and their providers at run-time. Often that will mean "you get what you've paid for" becomes the rule, at least in commercial parts of the Internet. But during really acute emergencies many people would expect such commercial concerns to be set aside [I-D.floyd-tsvwg-besteffort].

We agree that users shouldn't be able to squeeze out others during emergencies. But the mechanisms we have in place at the moment don't allow anyone to control whether this happens or not, because they can be overriden at run-time by using the extra degress of freedom available to get round TCP. It could equally be argued that each user (not each flow) should get an equal share of remaining capacity in an emergency. Indeed, it would seem wrong for one user to expect 100 continuously running flows downloading music & videos to take 100 times more capacity than other users sending brief flows containing messages trying to contact loved ones or the emergency services [Hengchun_quake].[Note_Earthquake]

We argue that fairness during emergencies is, more than anything else, a policy matter to be decided at run-time (either before or during an anomaly) by users, operators, regulators and governments--not at design time by the IETF. The IETF should however provide the framework within which typical policies can be enforced. And the IETF should ensure that the Internet is still likely to utilise resources _efficiently_ under extreme stress, assuming a reasonable mix of likely policies, including none.

The main take-away point from this section is that the IETF should not, and need not, make such life-and-death decisions. It should provide protocols that allow any of these policy options to be chosen at the time of need or by making contingencies beforehand. The congestion accountability framework in {ToDo: ref sister doc} provides such control, while also allowing different controls (including no control at all) in normal circumstances. For instance an ISP might normally allow its customers to pay to override any usage limits. But during a disaster it might suspend this right. Then users would get only the shares they had established before the disaster broke out (the ISP would thus also avoid accusations of profiteering from people's misery). Whatever, it is not for the IETF to embed answers to questions like these in our protocols.

4. Security Considerations

{ToDo:}

5. Conclusions

{ToDo:}

6. Acknowledgements

Arnaud Jacquet, Phil Eardley.

7. Comments Solicited

Comments and questions are encouraged and very welcome. They can be addressed to the IETF Transport Area working group mailing list <tsvwg@ietf.org>, and/or to the authors.

8. References

8.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2309] Braden, B., Clark, D., Crowcroft, J., Davie, B., Deering, S., Estrin, D., Floyd, S., Jacobson, V., Minshall, G., Partridge, C., Peterson, L., Ramakrishnan, K., Shenker, S., Wroclawski, J., and L. Zhang, "Recommendations on

Queue Management and Congestion Avoidance in the Internet", RFC 2309, April 1998.

[RFC2581] Allman, M., Paxson, V., and W. Stevens, "TCP Congestion Control", RFC 2581, April 1999.

[RFC2914] Floyd, S., "Congestion Control Principles", BCP 41, RFC 2914, September 2000.

8.2. Informative References

[AppVsTask]
Bouch, A., Sasse, M., and H. DeMeer, "Of packets and people: A user-centred approach to Quality of Service", Proc. IEEE/IFIP Proc. International Workshop on QoS (IWQoS'00) , May 2000, <http://www.cs.ucl.ac.uk/staff/A.Bouch/42-171796908.ps>.

[FAST] Jin, C., Wei, D., and S. Low, "FAST TCP: Motivation, Architecture, Algorithms, and Performance", Proc. IEEE Conference on Computer Communications (Infocom'04) , March 2004, <http://www.ieee-infocom.org/2004/Papers/52_2.PDF>.

[Hengchun_quake]
Wikipedia, "2006 Hengchun earthquake", Wikipedia Web page (accessed Oct'07) , 2006, <http://en.wikipedia.org/wiki/2006_Hengchun_earthquake>.

[I-D.floyd-tsvwg-besteffort]
Floyd, S. and M. Allman, "Comments on the Usefulness of Simple Best-Effort Traffic", draft-floyd-tsvwg-besteffort-01 (work in progress), August 2007.

[I-D.ietf-capwap-protocol-specification]
Calhoun, P., "CAPWAP Protocol Specification", draft-ietf-capwap-protocol-specification-07 (work in progress), June 2007.

[I-D.rosen-pwe3-congestion]
Rosen, E., "Pseudowire Congestion Control Framework", draft-rosen-pwe3-congestion-04 (work in progress), October 2006.

[PMP] Odlyzko, A., "A modest proposal for preventing Internet congestion", AT&T technical report TR 97.35.1, September 1997,

<http://www.dtc.umn.edu/~odlyzko/doc/modest.proposal.pdf>.

[RFC1958] Carpenter, B., "Architectural Principles of the Internet", RFC 1958, June 1996.

[RFC2357] Mankin, A., Romanov, A., Bradner, S., and V. Paxson, "IETF Criteria for Evaluating Reliable Multicast Transport and Application Protocols", RFC 2357, June 1998.

[RFC2475] Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z., and W. Weiss, "An Architecture for Differentiated Services", RFC 2475, December 1998.

[RFC2616] Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., and T. Berners-Lee, "Hypertext Transfer Protocol -- HTTP/1.1", RFC 2616, June 1999.

[RFC3234] Carpenter, B. and S. Brim, "Middleboxes: Taxonomy and Issues", RFC 3234, February 2002.

[RFC3448] Handley, M., Floyd, S., Padhye, J., and J. Widmer, "TCP Friendly Rate Control (TFRC): Protocol Specification", RFC 3448, January 2003.

[RFC3649] Floyd, S., "HighSpeed TCP for Large Congestion Windows", RFC 3649, December 2003.

[RFC3742] Floyd, S., "Limited Slow-Start for TCP with Large Congestion Windows", RFC 3742, March 2004.

[RFC4828] Floyd, S. and E. Kohler, "TCP Friendly Rate Control (TFRC): The Small-Packet (SP) Variant", RFC 4828, April 2007.

[RFC5033] Floyd, S. and M. Allman, "Specifying New Congestion Control Algorithms", BCP 133, RFC 5033, August 2007.

[Rate_fair_Dis]
Briscoe, B., "Flow Rate Fairness: Dismantling a Religion", ACM CCR 37(2)63--74, April 2007, <http://portal.acm.org/citation.cfm?id=1232926>.

[Res_p2p] Cho, K., Fukuda, K., Esaki, H., and A. Kato, "The Impact and Implications of the Growth in Residential User-to-User Traffic", ACM SIGCOMM CCR 36(4)207--218, October 2006, <http://doi.acm.org/10.1145/1151659.1159938>.

[TSV_CAPWAP_issues]

Borman, D. and IESG, "Transport Issues in CAPWAP", In Proc. IETF-69 CAPWAP w-g, July 2007, <http://www3.ietf.org/proceedings/07jul/slides/capwap-1.pdf>.

[az-calc] Infinite-Source, "Azureus U/L settings calculator", Web page (accessed Oct'07) , 2007, <http://infinite-source.de/az/az-calc.html>.

[ion-tsv-alt-cc] "Experimental Specification of New Congestion Control Algorithms", July 2007, <http://www.ietf.org/IESG/content/ions/ion-tsv-alt-cc.txt>.

Editorial Comments

[Note_Collapse] Some would say that it is not a congestion collapse if congestion control automatically recovers the situation after a while. However, even though lack of autorecovery would be truly devastating, it isn't part of the definition [RFC2914].

[Note_Earthquake] On 26 Dec 2006, the Hengchun earthquake caused faults on 12 of the 18 undersea cables passing between Taiwan and the Philippines. The Internet was virtually unusable for those trying to make their emergency arrangements over these cables (as well as for much of Asia generally). Each of these flows was still having to compete with the multiple flows of video downloads for remote users who were presumably oblivious to the fact they were consuming much of the surviving capacity. When the Singaporean ISP, SingNet, announced restoration of service before the cables were repaired, it revealed that it had achieved this at the expense of video downloads and gaming traffic .

[Note_Neutral] Enforcement of /overall/ traffic limits within an agreed acceptable use policy is a completely different question to that of whether operators should discriminate against /specific/ applications or service providers (but they are confusible—see the section on DPI.

[Note_Window] Within the flow rate equality worldview, there are differences in views over whether window sizes

should be compared in packets or bytes, and whether a longer round trip time (RTT) should reduce the target rate or merely slow down how quickly the rate changes in order to reach a target rate that is independent of RTT [FAST]. However, although these details are important, they are merely minor internal differences within the flow rate equality worldview when compared against the differences with volume accounting.

Appendix A. Example Scenario

A.1. Base Scenario

We will consider 100 users all sharing a link from the Internet with 2Mbps downstream access capacity. Eighty bought their line for occasional flurries of activity like browsing the Web, booking their travel arrangements or reading their email. The other twenty bought it mainly for unattended volume transfer of large files. We will call these two types of use attended (or light) and unattended (or heavy). Ignoring the odd UDP packet, we will assume all these applications use TCP congestion control, and that all flows have approximately equal round trip times.

Imagine the network operator has provisioned the shared link for a contention ratio of 20:1, ie 100x2Mbps/20 = 10Mbps. For simplicity, we assume a 16hr 'day' and that the attended use is only in the 'day', while unattended use is always present, having the night to itself.

During the 'day', flows from the sixty attended users come and go with about 1 in 10 actively downloading flows at any one time (a downstream activity factor of 10%). To start with, we will further assume that, when active, every user has approximately the same number of flows open, whether attended or unattended. So, once all flows have stabilised, at any instant TCP will ensure every user (when active) gets about 10Mbps/(80*10% + 20*100%) = 357kbps of the bottleneck.

Table 2 tabulates the salient features of this scenario. Also the rightmost column shows the volume transferred per user during the day, and for completeness the bottom row shows the aggregate.

Type of use	No. of users	Activ- ity factor	Day rate /user (16hr)	Day volume /user (16hr)
Attended	80	10%	357kbps	257MB
Unattended	20	100%	357kbps	2570MB
Aggregate	100		10Mbps	72GB

Table 2: Base Scenario assuming 100% utilisation of 10Mbps bottleneck and each user runs approx. equal numbers of flows with equal RTTs.

This scenario is not meant to be an accurate model of the current Internet, for instance:

- Utilisation is never 100%.

- Upstream not downstream constrains most p2p apps on DSL (but not all fixed & wireless access technologies).

- The activity factor of 10% in our base example scenario is perhaps an optimistic estimate for attended use over a 16hr peak period. 1% is just as likely for many users (before file-sharing became popular, DSL networks were provisioned for a contention ratio of about 25:1, aiming to handle a peak average activity factor of 4% across all user types).

- And rather than falling into two neat categories, users sit on a wide spectrum that extends to far more extreme types in both directions, while in between there are users who mix both types in different proportions [Res_p2p].

But the scenario has merely been chosen because it makes it simple to grasp the main issues while still retaining some similarity to the real Internet. We will also develop the scenario as we go, to add more realism (e.g. adding mixed user types).

A.2. Compounding Overlooked Degrees of Freedom

Table 3 extends the base scenario of Appendix A to compound differences in average activity factor with differences in average numbers of active flows.

During the 'day' at any instant we assume on average that attended use results in 2 flows per user (which are still only open 10% of the time), while unattended use results in 100 flows per user open continuously. So at any one time 2016 flows are active, 16 from

attended use (10%*80=8 users at any one time * 2 flows) and 2000 from unattended use (20 users * 100 flows). TCP will ensure each of the 8 users who are active at any one time gets about 2*10Mbps/2016 = 9.9kbps of the bottleneck, while each of the 20 unattended users gets about 100*10Mbps/2016 = 496kbps. This ignores flow start up effects, which will tend to make matters even worse for attended use, given briefer flows start more often.

Type of use	No. of users	Activ-ity factor	Ave simultaneous flows /user	Day rate /user (16hr)	Day volume /user (16hr)
Attended	80	10%	2	9.9kbps	7.1MB
Unattended	20	100%	100	496kbps	3.6GB
Aggregate	100		2016	10Mbps	72GB

Table 3: Compounded scenario with attentive users less frequently active and running less flows than unattentive users, assuming 100% utilisation of 10Mbps bottleneck and all equal RTTs.

A.3. Hybrid Users

{ToDo:}

A.4. Upgrading Makes Most Users Worse Off

Now that the light users are only getting 9.9kbps from their 2Mbps lines, the operator needs to consider upgrading their bottleneck (and all the other access bottlenecks for its other customers), so it does a market survey. The operator finds that fifty of the eighty light users and ten of the twenty heavy users are willing to pay more to get an extra 500kbps each at the bottleneck. (Note that by making a smaller proportion of the heavy users willing to pay more we haven't weighted the argument in our favour--in fact our argument would have been even stronger the other way round.)

To satisfy the sixty users who are willing to pay for a 500kbps upgrade will require a 60*500kbps = 30Mbps upgrade to the bottleneck and proportionate upgrades deeper into the network, which will cost the ISP an extra $120 per month (say). The outcome is shown in Table 4. Because the bottleneck has grown from 10Mbps to 40Mbps, the bit rates in the whole scenario essentially scale up by 4x. However, also notice that the total volume sent by the light users has not grown by 4x. Although they can send at 4x the bit rate, which means

they get more done and therefore transfer more volume, they don't have 4x more volume to transfer--they let their machines idle for longer between transfers reflected in their activity factor having reduced from 10% to 4%. More bit rate was what they wanted, not more volume particularly.

Let's assume the operator increases the monthly fee of all 100 customers by $1.20 to pay for the $120 upgrade. The light users had a 9.9kbps share of the bottleneck. They've all paid their share of the upgrade, but they've only got 30kbps more than they had--nothing like the 500kbps upgrade most of them wanted and thought they were paying for. TCP has caused each heavy user to increase the bit rate of its flows by 4x too, and each has 50x more flows for 25x more of the time, so they use up most of the newly provisioned capacity even though only half of them were willing to pay for it.

But the operator knew from its marketing that 30 of the light users and 10 of the heavy ones didn't want to pay any more anyway. Over time, the extra $1.20/month is likely to make them drift away to a competitor who runs a similar network but who decided not to upgrade its 10Mbps bottlenecks. Then the cost of the upgrade on our example network will have to be shared over 60 not 100 customers, requiring each to pay $2/month extra, rather than $1.20.

Type of use	No. of users	Activ- ity factor	Ave simultaneous flows /user	Day rate /user (16hr)	Day volume /user (16hr)
Attended	80	4%	2	40kbps	11MB
Unattended	20	100%	100	2.0Mbps	14GB
Aggregate	100		2006.4	40Mbps	288GB

Table 4: Scenario with bottleneck upgraded to 40Mbps, but otherwise unchanged from compounded scenario.

But perhaps losing a greater proportion of the heavy users will help? Table 5 shows the resulting shares of the bottleneck once all the cost sensitive customers have drifted away. Bit rates have increased by another 2x, mainly because there are 2x fewer heavy users. But that still only gives the light users 80kbps when they wanted 500kbps--and, to rub salt in their wounds, their monthly fees have increased by $2 in all. The remaining 10 heavy users are probably happy enough though. For the extra $2/month they get to transfer 8x more volume each (and they still have the night to themselves).

We have shown how the operator might lose those customers who didn't want to pay. But it also risks losing all fifty of those valuable light customers who were willing to pay, and who did pay, but who hardly got any benefit. In this situation, a rational operator will eventually have no choice but to stop investing in capacity, otherwise it will only be left with ten customers.

Type of use	No. of users	Activ-ity factor	Ave simultaneous flows /user	Day rate /user (16hr)	Day volume /user (16hr)
Attended	50	2.5%	2	80kbps	14MB
Unattended	10	100%	100	4.0Mbps	29GB
Aggregate	60		1002.5	40Mbps	288GB

Table 5: Scenario with bottleneck upgraded to 40Mbps, but having lost customers due to extra cost; otherwise unchanged from compounded scenario.

We hope the above examples have clearly illustrated two main points:

o Rate equality at design time doesn't prevent extreme unfairness at run time;

o If extreme unfairness is not corrected, capacity investment tends to stop--a concrete consequence of unfairness that affects everyone.

Finally, note that configuration guidelines for typical p2p applications (e.g. BitTorrent calculator [az-calc]), advise a maximum number of open connections that increases roughly linearly with upstream capacity.

Authors' Addresses

Bob Briscoe
BT & UCL
B54/77, Adastral Park
Martlesham Heath
Ipswich IP5 3RE
UK

Phone: +44 1473 645196
Email: bob.briscoe@bt.com
URI: http://www.cs.ucl.ac.uk/staff/B.Briscoe/

Toby Moncaster
BT
B54/70, Adastral Park
Martlesham Heath, Ipswich IP5 3RE
UK

Phone: +44 1473 645196
Email: toby.moncaster@bt.com
URI: http://research.bt.com/networks/TobyMoncaster.html

Louise Burness
BT
B54/77, Adastral Park
Martlesham Heath
Ipswich IP5 3RE
UK

Phone: +44 1473 646504
Email: Louise.Burness@bt.com
URI: http://research.bt.com/networks/LouiseBurness.html

Full Copyright Statement

Intellectual Property

Acknowledgments

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA). This document was produced using xml2rfc v1.32 (of http://xml.resource.org/) from a source in RFC-2629 XML format.