P2P, the Gorilla in the Cable

Alexandre Gerber, Joseph Houle, Han Nguyen, Matthew Roughan, Subhabrata Sen
AT&T Labs - Research

Abstract

There is considerable interest in Peer-to-peer (P2P) traffic because of its remarkable increase over the last few years. By analyzing flow measurements at the regional aggregation points of several cable operators, we are able to study its properties. It has become a large part of broadband traffic and its characteristics are different from older applications, such as the Web. It is a stable balanced traffic: the peak to valley ratio during a day is around 2 and the Inbound/Outbound traffic balance is close to one. Although P2P protocols are based on a distributed architecture, they don't show strong signs of geographical locality. A cable subscriber is not much more likely to download a file from a close region than from a far region.

It is clear that most of the traffic is generated by heavy hitters who abuse P2P (and other) applications, whereas most of the subscribers only use their broadband connections to browse the web, exchange e-mails or chat. However it is not easy to directly block or limit P2P traffic, because theses applications adapt themselves to their environment: the users develop ways of eluding the traffic blocks. The traffic that could be once identified with five port numbers is now spread over thousands of TCP ports, pushing port based identification to its limits. More complex methods to identify P2P traffic are not a long-term solution, the cable industry should opt for a a "pay for what you use" model like the other utilities.

INTRODUCTION

File Sharing Applications

KaZaA, Gnutella and DirectConnect are all decentralized, self-organizing file sharing systems with data and index information (metadata for searching) distributed over a set of end-peers or peers, each of which can be both a client and a server of content. Peers can join and leave frequently, and organize in a distributed fashion into an application-level overlay via point-to-point application-level connections between a peer and a set of other peers (its neighbors). By default, all the communications occur over well known ports.

The process of obtaining a file can be broadly divided into two phases – a search followed by a object retrieval. First, a peer uses the P2P protocol to search for the existence of a certain file in the P2P system, receives one or more responses, and if the search is successful, identifies one or more target peers from which to download that file. The search queries as well as the responses are transmitted via the overlay connections using protocol-specific application level routing. The details of how the signaling is propagated through the overlay is protocol-dependent. In earlier P2P protocols exemplified by Gnutella version 4.0, a peer initiates a query by flooding it to all its neighbors in the overlay. The neighboring peers in turn, flood to their neighbors, using a scoping mechanism to control the query flood. In contrast, for both KazaA and DirectConnect as well as newer versions of Gnutella, queries are forwarded to and handled by only a subset of special peers (called SuperNodes in KazaA, Hubs in DirectConnect, and UltraPeers in Gnutella). A peer transmits an index of its content to the ``special peer'' to which it is connected. The special peer then uses the corresponding P2P protocol to forward the query to other such peers in the system.

Once search results are in, the requesting peer directly contacts the target peer, typically using HTTP (the target peer runs has a HTTP server listening by default on a known,protocol-specific port), to get the requested resource. Some newer systems, such as KazaA and Gnutella, use "file swarming" -- a file download is executed by retrieving different chunks from multiple peers.

Although the earlier P2P systems mostly used their default network ports for communication, there is substantial evidence to suggest that substantial P2P traffic nowadays is transmitted over a large number of non-standard ports. This seems to be primariliy motivated by the desirwe to circumvent firewall restrictions as well as rate–limiting actions by ISPs targeted at such applications - we shall discuss this more later in the paper.

Another recent development has been the development of tools for allowing an end-user to explicitly select the SuperNode it connects to. This appears to be an attempt to improve the quality of the best-effort search process in the P2P system, for files that may exhibit locality in storage. For instance, connecting to a SuperNode in Brazil may increase the chances of locating Samba-related content.

Data Collection

We have access to "flow-level" data at the regional aggregation points for several broadband ISPs. Flow-level data is considerably more detailed than data sets such as SNMP, and at least this level of detail if needed to perform application classification. The regional aggregation points provide the MSOs with access to the backbone for traffic between regions and to the rest of the Internet, where a region typically ranges from an extended metropolitan area to a state.

By flow, we mean a sequence of packets exchanged by two applications. More precisely we define a flow to be a series of uni-directional packets with the same IP protocol, source and destination address, and source and destination ports (in the case of TCP and UDP traffic). The flow measurements used here are called Cisco Netflow; they are implemented in many of Cisco's routers. The data collected about a flow (apart from the information above) are the duration, the number of packets, and bytes transmitted, and which header flags (SYN, ACK, …) were used in the flow. Measured flows are also constrained in time (Cisco Netflow collection sends flows from the router at 15 minute intervals), so there is a need to reconstruct the actual traffic from a single "connection". After reconstruction there will be one flow per connection – a potentially enormous volume of information.

In order to minimize any performance impact on the routers collecting the flow measurements the measurements are based on sampled packets collected on the routers, which then export the flows to aggregators. To reduce the huge data volume the aggregator further samples the flows using the smart sampling algorithm [SAMP] that is better suited for heavy tailed distribution, such as typically found in Internet flows. In addition to that there is also an uncontrolled sampling due to measurement packet losses. These three types of sampling can be estimated and corrected and don't affect our results that are based on the weekly or monthly average traffic generated by hundreds of thousands of cable subscribers.

More precisely, we used data ranging from May 2002 to February 2003 from five different MSOs. When we were not collecting all the traffic coming from a region, we were using SNMP data to extrapolate the actual traffic. However, when we analysed the behaviour per broadband user, we selected only regional aggregation points for which we were collecting all the flow level measurements.

Identifying Applications

There are a number of ways one could go about identifying individual applications within IP traffic. However, as noted, Netflow only keeps data on some aspects of flows. The most useful of these for application breakdowns are the source and destination port numbers, and the IP protocol number. The protocol numbers used are well documented [IANA1], with TCP being protocol 6, and UDP being 17. TCP, and UDP traffic also define (16 bit) source and destination port numbers intented (in part) to for use by different applications. The port numbers are divided into three ranges: the Well Known Ports (0-1023), the Registered Ports (1024-49,151), and the Dynamic and/or Private ports (49,152-65,535).

A typical TCP connection starts with a SYN/ACK handshake from a client to a server. The client addresses its initial SYN packet to the server port for a particular application, and uses a dynamic port as the source port for the SYN. The server listens on its port for connection. UDP uses ports similarly though without connections. All future packets in the TCP/UDP flow use the same pair of ports at the client and server ends. Therefore, in principle the server port number can be used to identify the higher layer application using TCP or UDP, by simply identifying which port is the server port (the one from the well-known, or registered port range) and mapping this to an application using the IANA list of registered port [IANA2].

However there are many barriers to determining applications from port numbers:

1. many implementations of TCP seem to use registered port ranges as dynamic ports ,

2. priveledged applications may use dynamic port numbers inside the well-known port range (for instance some old versions of bind use source and destination port 53).,

3. well known and registered ports are not defined for all applications (and this is typical of P2P applications).

4. an application may use ports other than its well-known port because these can only be used with special priveledges, e.g. WWW servers often run on ports other than port 80, for instance ports 8080, and 8888.

5. an application may run on different ports to avoid blocking by firewalls. (e.g. non-WWW servers are sometimes run on port 80 to avoid firewalls, and P2P applications are often run on alternate ports for the same reason).

6. There are some ambiguities in port registrations, e.g. port 888 which is used for CDDBP (CD Database Protocol) and accessbuilder .

7. in some cases server ports are dynamically allocated as needed (for instance, one might have a control

connection on which a data port is negotiated).

8. trojans and other security attacks (e.g. DoS) will break the port mapping.

Note that the use of firewalls to block unauthorized, and/or unknown applications from using a network has spawned work arounds that have made the mapping from port number to application ambiguous.

Despite this a great deal can be said about the mapping of port to application, though obviously there will still be some ambiguity, and chance for errors. Note that both ports must be considered as possible candidates for the server port, unless other data is available to rule out one port.

The algorithm that we have adopted here chooses the server port by (1) looking for a well known port, (2) a registered port, or (3) an unregistered port which is known (from reverse engineering of protocols) to be used by a particular (unregistered) application. If both source and destination port could be the server, then we choose the most likely one through ranking applications by how prevalent they are in detailed (packet level) traffic studies – for instance, WWW is considered a high ranking application, as are email, and P2P applications.

The result is a mapping from flows to applications, that while not perfect, has been shown to be reasonably effective. The biggest problem is that there are still a substantial number of flows which cannot be mapped to an application. We further classify these unknown flows by the size of the flows: the category of most interest here is "TCP-big", which consists of unknown flows that transmit more than 100kB in less than 30 minutes.

We shall argue in this paper that the TCP-big traffic is primarily P2P traffic that is using unregistered ports unknown to us. P2P applications already use unregistered ports, and the struture of P2P protocols (with separate control and data traffic) allows data traffic to be assigned to arbitrary ports. In the past the major applications have typically used default ports (for instance 1214 for KaZaa) but in the recent past many efforts have been made to constrain P2P traffic through rate limiting single ports or by blocking some ports at firewalls, with the result that P2P users commonly use work-arounds. Where-ever we refer to P2P traffic we are using the traffic on the ports known to be directly associated with P2P applications: we shall keep this separate from TCP-big except where explicitly noted. Also note that some P2P traffic may be misclassified into other application classes (for instance WWW), and so our estimates of the total volumes of P2P traffic are conservative.

We should note that we are not collecting any information about URL's, or individual subscribers usage: IP addresses measured are not related to individual subscribers, and we only view the bulk properties of the traffic, such as its distributions.

APPLICATION COMPOSITION

Overview

Table 1 shows the application traffic composition for 2 MSOs in May 2002 and January 2003. For each MSO, we examine both the traffic coming from outside the MSO to some IP address within the MSO (referred to as IN) and the traffic sourced within the MSO and destined for outside the MSO (OUT). For each time period, MSO, we display the per-application traffic volume in each direction as a percentage of the total traffic in that direction. For a given application we also show the traffic normalized by dividing by its IN traffic volume for May 2002, in order to show the IN/Out ratio, and the growth between the two periods.

We note that in either direction, for both MSOs, the P2P traffic forms a much smaller percentage of the overall traffic in January 2003 than in May 2002. TCP-big registered dramatic increases in traffic contribution in both directions (10.5 times for Outgoing and 6.02 times for Incoming) over the same period. The normalized figures show that the P2P incoming and outgoing traffic are very similar for either of the 2 months considered. For example for MSO X, the ratio between incoming and outgoing TCP-big traffic volumes changes from 1.94:1 in May 2002 to a more balanced 1.12:1 in January 2003.

	MSO X								MSO Y							
	Applicationx Mix (percentage)				Normalized Consumption				Applicationx Mix (percentage)				Normalized Consumption			
	May 2002		January 2003		May 2002		January 2003		May 2002		January 2003		May 2002		January 2003	
	OUT	IN	OUT	IN	OUT	IN	OUT	IN	OUT	IN	OUT	IN	OUT	IN	OUT	IN
All	100.0%	100.0%	100.0%	100.0%	1	1.65	1.97	3.2	100.0%	100.0%	100.0%	100.0%	1	2.19	1.83	4.08
ESP/GRE	0.4%	0.5%	0.6%	0.5%	1	1.98	3.12	4.3	0.4%	0.5%	0.3%	0.4%	1	2.71	1.7	4.67
OTHER	4.4%	3.7%	5.7%	4.5%	1	1.37	2.54	3.23	4.6%	3.2%	5.4%	3.4%	1	1.53	2.16	2.97
TCP-BIG	8.9%	10.5%	47.5%	32.5%	1	1.94	10.5	11.68	9.5%	11.8%	45.3%	32.1%	1	2.71	8.71	13.72
AUDIO/VIDEO	0.2%	1.6%	0.2%	1.6%	1	16.61	2.77	32.64	0.1%	1.5%	0.2%	1.5%	1	23.71	3.1	44.29
CHAT	0.7%	1.3%	1.0%	1.7%	1	3.08	2.93	7.93	0.7%	1.2%	0.7%	1.4%	1	3.81	2.02	8.67
FTP	1.0%	1.3%	1.0%	0.7%	1	2.22	1.91	2.4	1.4%	1.4%	0.4%	0.9%	1	2.24	0.56	2.64
GAMES	1.6%	1.2%	3.6%	2.5%	1	1.29	4.54	5.15	1.3%	1.2%	3.4%	2.4%	1	1.92	4.73	7.43
MAIL	1.7%	0.6%	1.1%	0.7%	1	0.6	1.26	1.28	1.0%	0.5%	0.9%	0.5%	1	1.13	1.71	1.88
NEWS	0.3%	7.3%	0.2%	5.3%	1	38.52	1.51	54.55	0.7%	17.5%	0.7%	14.6%	1	54.99	1.76	85.33
P2P	75.2%	45.6%	32.9%	20.6%	1	1	0.86	0.87	75.1%	38.5%	36.7%	19.5%	1	1.12	0.9	1.06
WEB	5.6%	26.4%	6.2%	29.4%	1	7.8	2.2	16.88	5.2%	22.8%	5.9%	23.5%	1	9.53	2.06	18.27

Table 1: Application Composition of two MSOs in May 2002 and January 2003.

Time of Day Pattern

We next examine the diurnal behavior of P2P traffic. Figure 1 plots the time series of the incoming and outgoing traffic volumes (P2P, web and TCP-big) for a given MSO across a week in February 2003. For each application, all the data values are normalized by the mean per-hour incoming data volume for that application, averaged across that week.

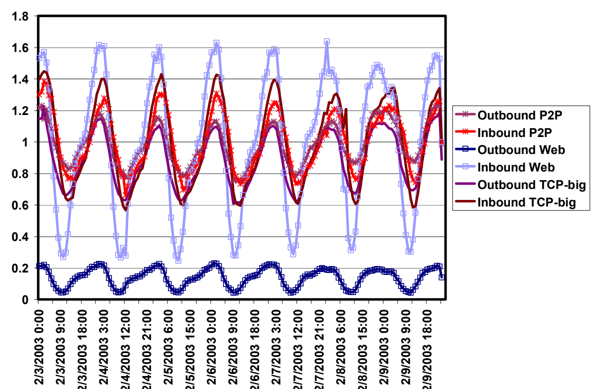

Figure 1: Time od day pattern of P2P and Web traffic.

All three applications exhibit similar diurnal behaviors with peak loads (in either direction) around 2.00 AM GMT (10.00 PM EST, 7.00 PM PST). The P2P traffic exhibits less variability across a day than Web traffic. The peak load is about 2 times the minimum as opposed to 5 times for Web traffic. The smaller variance in P2P traffic across a day may be a function of the programmed download feature in P2P applications that allow users to specify multiple files in advance, that can be downloaded asynchronously by the P2P application.

For Web, the outgoing traffic is significantly smaller than (atmost 20% of) the incoming traffic, suggesting that the MSOs clients are mostly consumers of web data. In contrast, for P2P, the traffic in the 2 directions track each other much more closely, across a day and across the week. Another notable here is that the TCP-big traffic distribution across time is very similar to the P2P traffic. Also, just like P2P, the TCP-big traffic in the 2 directions are similar. These behavorial similarities are another indicatior that the TCP-big traffic includes some P2P applications. Finally for all 3 applications, we do not see significant variations across days and beween weekdays and weekends.

P2P LOCALITY

One of the potential advantages of P2P applications is that by distributing content, they provide the ability to download this content from locations closer to a user. It is therefore interesting to consider whether this really happens, and moreover to consider the question of locality in P2P traffic in general.

We approach this question by considering the simplest possible counter examples to localized traffic: the simple gravity model [?]. In this model, a packet entering the network at S, makes its decision about its destination D independent of the arrival point. That is, the packet is drawn (as if by gravity) to destinations in proportion to the volume of traffic departing at those locations.

The gravity model can be used to make predictions of the traffic volumes between two regions based purely on the volumes entering and exiting at those two regions, by the formula

$$T^{S,D} = \frac{T_{in}^{S} T_{out}^{D}}{T}$$

where T is the total volume of traffic across the network, T_{in}^{S} is the traffic entering the network at region S, and T_{out}^{D} is the traffic exiting the network at region D. Figure 2 below shows a comparison of the gravity model predictions for inter-regional traffic on one cable company. The plot is based on netflow traffic collected (from the May time interval where we have data across a wider spread of regions and MSOs) above the regional aggregation routers, and therefore shows traffic traversing the backbone between regions. The figure shows a scatter plot of the real inter-regional traffic versus the gravity model prediction, for both P2P traffic, and the total traffic to the cable company. On can see that in both cases the gravity model predicts the true traffic within about ±20%.

What does that tell us? Well the main point is that the gravity model above explicitly excludes any notion of geographic, or topological distance. Therefore, as the measured traffic fits this model to some extent, we may believe that neither P2P traffic nor the traffic overall exhibit strong locality at the regional level. A further, somewhat subjective conlusion one might drawn from the graph is that P2P traffic actually seems to fit the gravity model slightly worse, and so we may hypothesize that P2P traffic shows more locality than other traffic sources.

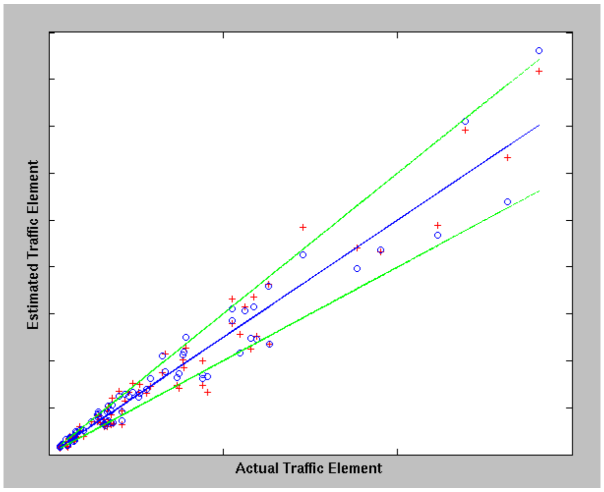

Figure 2: Comparison of the real matrix elements to the estimated traffic matrix elements for one MSO. The circles represent purely P2P traffic and crosses represents the total traffic. The blue solid diagonal line shows equality and the green dashed lines show ± 20%.

To examine these hypothesis in more details we present Table 2, which shows the normalized traffic volumes between regions for the P2P traffic. The table shows the normalized probability that traffic originating from a particular region in one cable company, will depart from each region in the same cable company (given it stays on the same cable companies network). Table 2 can be seen to have a number of almost identical rows (for instance the group of regions R1, R2, and R5 are very similar, as is the group R6, R7 and R8) indicating a complete lack of locality of traffic with reference to these regions. Other regions (specifically R3 and R4) are not dramatically far away, but rather fall somewhere in between the other two groups.

However the table also shows some disparity between the groups of rows. This disparity is at its height when comparing the regions in the Eastern Standard Timezone (EST), with those in the Pacific Timezone (PST). This is an indication of some degree of weak locality in P2P traffic, at the "super-regional" level.

From/To	R1 (PST)	R2 (PST)	R3 (MST)	R4 (MST)	R5 (CST)	R6 (CST)	R7 (EST)	R8 (EST)
R1 (PST)	-	0.18	0.14	0.126	0.174	0.128	0.124	0.127
R2 (PST)	0.172	-	0.141	0.126	0.19	0.132	0.118	0.12
R3 (MST)	0.132	0.12	-	0.189	0.135	0.145	0.139	0.14
R4 (MST)	0.107	0.111	0.182	-	0.124	0.163	0.155	0.158
R5 (CST)	0.161	0.18	0.136	0.132	-	0.135	0.127	0.129
R6 (CST)	0.107	0.108	0.145	0.155	0.125	-	0.187	0.173
R7 (EST)	0.107	0.106	0.137	0.157	0.127	0.182	-	0.184
R8 (EST)	0.109	0.111	0.127	0.161	0.128	0.178	0.185	-

Table 2: Normalized inter-regional traffic matrix of MSO X weighted by P2P+TCP-big traffic (Longitude defined by the Timezone).

This super-regional locality could arise for a couple of reasons (other than P2P applications explicity taking advantage of content locality to improve performance). Firstly, because of usage patterns (specifically the times at which a user is connected to the P2P network), there is a slight increase in the likelihood that a search will find content in a local time zone. Secondly, there may be a group of people within a super-region with content that is slightly more relevant to the local super-region. However, the data so far suggests that both of these effects are not dominant, and certainly there is no strong locality influence such as might be seen if the main P2P applications exploited locality information.

In both of the above examples the monitoring location (above the regional aggregation router) limits our data to seeing only inter-regional traffic. Thus, one might argue, we are missing the key component in any study of traffic locality: the intra-regional traffic.

While the data limitations prevent us from seeing the intra-regional traffic on a single cable company, we can gain a good view of this data by considering the traffic between cable companies. If locality were being exploited in P2P applications, then one would expect traffic from company Y, region R to prefer going to company X, region R, rather than the alternative regions.

Table 3 shows an example, giving the normalized probabilies that traffic from company Y to X will go from regions M to R. Although the regions for the two companies are slightly different,. Regions M3 and R7 are very closely matched as are M4 and R8. However, we see only very minor bias towards traffic from M3 to R7 (compared to other EST regions), and similarly from M4 to R8.

From / To	R1 (PST)	R2 (PST)	R3 (MST)	R4 (MST)	R5 (CST)	R6 (CST)	R7 (EST)	R8 (EST)
M1 (MST)	0.133	0.121	0.157	0.125	0.118	0.111	0.089	0.146
M2 (CST)	0.121	0.095	0.114	0.158	0.117	0.145	0.094	0.156
M3 (EST)	0.12	0.114	0.12	0.138	0.119	0.128	0.14	0.122
M4 (EST)	0.11	0.115	0.109	0.137	0.135	0.119	0.133	0.142
M5 (EST)	0.117	0.115	0.133	0.135	0.129	0.12	0.121	0.129

Table 3: Normalized traffic matrix from MSO Y to MSO X weighted by P2P+TCP-big traffic.

Our conclusion is that, although there is some evidence for weak locality at a large spatial scale, P2P applications do not yet exploit such information on a large scale, and consequently, P2P traffic does not show strong signs of geographic locality. More recent developments of Kazaa provide methods for selected the super-node to which one connects, and so more locality may be introduced in the future. **(Subho needs to fix this line)**

HEAVY HITTERS AND P2P

It is well known in the cable industry that some heavy hitters consume most of the bandwidth. We shall divide subscribers into classes by their total usage, and analyze their consumption characteristics such as the application composition and the traffic balance per class. We define three groups of users: the heavy users who consume more than 1 Gbytes/day in average over a week, the medium users who consume between 50 Mbytes/Day and 1 Gbytes/Day and the light users who consume less than 50 Mbytes/Day.

User Distribution

We first compare the distribution of traffic per subscriber. In order to see if there are consistent patterns we compare two regions of one MSO with a region from another MSO, all at two different points in time: during the week ending June 26th 2002 and during the week ending February 9th 2003. In order not to bias the results, we choose two MSOs that are not multi-homed and regions that have a decent size, i.e. between 25,000 subscribers and 140,000 subscribers. By subscriber, we mean an active IP address. Even though the IP address is not statically assigned (the user obtains an IP automatically via DHCP), in the networks we examined it is “sticky”. That is, over a week a subscriber maintains the same IP address in practice, because the DHCP lease expires only after 4 days and it is reassigned to him if it is still available. However, the IP address distribution doesn’t reflect exactly the subscriber distribution since it misses the inactive subscribers and the subscribers with a very low usage that may not be sampled. For instance, for a given region, we identified 107,000 unique IP addresses whereas the MSO was claiming that there were 115,000 subscribers, i.e. a 7.5% difference.

The six distributions in Figure 3 and 4 are quite consistent; the two most different distributions being the ones belonging to different MSOs. In each case, the top 1% of the IP addresses account for 18.6—24.4% of the total traffic and the top 20% of the active IP addresses account for slightly more than 80% of the traffic. For one MSO the average total consumption – the sum of IN and OUT traffic – went from 12.5 kbps per IP address in June to 13.3 kbps in February in one region, and 12.2 kbps to 13.5 kbps for the other region. The total consumption of the second MSO remainded stable at 14 kbps per unique IP address. For all these regions, the median consumption was only between 2 and 3 kbps, showing that the distribution was strongly skewed.

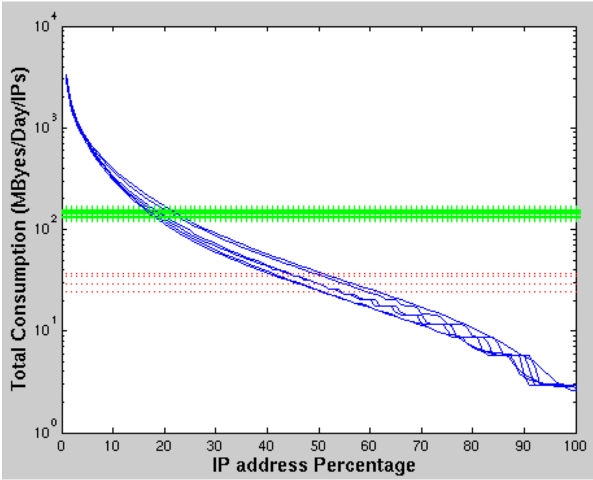

Figure 3: Consumption per percentile of IP addresses of two regions of MSO X and one region of MSO Y during a week in June 2002 and a week in Februray 2003. The mean consumptions are around 140 Mbytes/Day/IP and the medians are roughly 30 Mbytes/Day/IP.

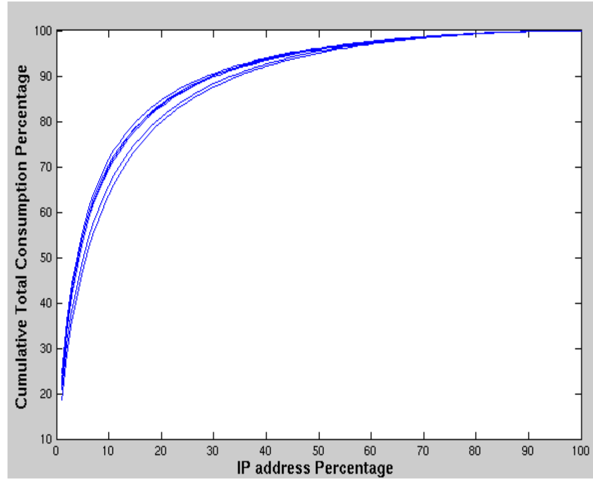

Figure 4: Cumulative Consumption of two regions of MSO X and one region of MSO Y during a week in June 2002 and a week in Februray 2003.

	Week ending June 26th 2002								
User Type	Heavy		Medium		Light		Heavy	Medium	Light
Direction	OUT	IN	OUT	IN	OUT	IN	IN/OUT	IN/OUT	IN/OUT
Normalized Traffic per Sub	266.8	445.5	27.0	48.9	1.0	4.8	1.7	1.8	4.8
AUDIO/VIDEO	0.1%	0.3%	0.1%	1.9%	0.4%	2.7%	3.2	26.4	29.8
CHAT	0.2%	0.4%	0.6%	0.8%	2.9%	2.0%	3.2	2.4	3.4
NEWS	1.1%	34.9%	0.5%	13.5%	0.2%	2.1%	53.6	54.1	55.1
MAIL	0.4%	0.1%	1.5%	0.4%	8.3%	2.3%	0.5	0.5	1.4
FTP	0.7%	0.9%	0.6%	1.1%	0.8%	0.3%	2.2	3.5	1.7
GAMES	0.4%	0.5%	1.5%	1.5%	2.8%	1.0%	2.0	1.7	1.7
ESP/GRE	0.0%	0.2%	0.7%	1.1%	5.3%	2.8%	6.9	3.0	2.6
P2P	87.4%	44.0%	82.3%	43.2%	18.5%	6.8%	0.8	1.0	1.8
TCP-BIG	6.9%	8.4%	3.3%	6.3%	2.4%	2.5%	2.0	3.4	5.1
WEB	0.9%	5.3%	5.1%	26.6%	46.2%	71.6%	10.1	9.5	7.5
OTHER	2.0%	5.1%	4.0%	3.7%	12.2%	5.7%	4.3	1.7	2.3

	Week ending February 9th 2003								
User Type	Heavy		Medium		Light		Heavy	Medium	Light
Direction	OUT	IN	OUT	IN	OUT	IN	IN/OUT	IN/OUT	IN/OUT
Normalized Traffic per Sub	288.3	415.1	26.1	47.8	1.1	5.2	1.4	1.8	4.8
AUDIO/VIDEO	0.1%	0.5%	0.2%	2.2%	0.4%	2.6%	4.9	17.3	28.4
CHAT	0.3%	0.6%	0.7%	1.2%	2.6%	2.3%	3.0	3.0	4.1
NEWS	1.0%	32.8%	0.4%	10.5%	0.1%	1.4%	49.6	46.6	46.2
MAIL	0.1%	0.3%	1.3%	0.7%	8.1%	2.7%	2.7	0.9	1.6
FTP	0.8%	0.7%	0.5%	0.8%	0.6%	0.2%	1.4	2.8	1.9
GAMES	3.3%	1.9%	4.1%	2.7%	2.9%	1.0%	0.8	1.2	1.7
ESP/GRE	0.1%	0.3%	1.0%	1.4%	6.0%	3.1%	5.6	2.5	2.5
P2P	37.7%	22.9%	29.5%	14.0%	7.0%	2.3%	0.9	0.9	1.6
TCP-BIG	51.2%	30.5%	47.6%	29.3%	13.1%	6.8%	0.9	1.1	2.5
WEB	1.6%	6.5%	6.4%	31.5%	46.7%	72.3%	5.7	9.0	7.5
OTHER	3.9%	3.1%	8.2%	5.8%	12.5%	5.3%	1.1	1.3	2.1

Table 4: Comparison of the application composition of the heavy, medium and light users of a region having more than 100 000 subscribers.

Consumption Characteristics

Since the median consumption is 4 to 5 times smaller than the average consumption, it is clear that the average consumption doesn't reflect the behaviour of most of the subscribers. This still holds if we compare the application composition of each group of users, as defined earlier, with the average application composition that were studied earlier in this paper. Indeed, in a close look at one of these regions Table 4 shows that the light users (67% of the IP addresses) are still mainly browsing the web, exchanging e-mail and chatting online. Their traffic balance – the IN/OUT ratio – is 4.8, which is far from the that of the heavy and medium users at 1.4-1.7 and 1.8, respectively. Table 5 makes it clear that they are not familiar with P2P or News since only 12.6 % of these light users are lightly using one of these applications and they generate less than 2% of the total traffic of these applications.

	Week ending June 26th 2002					
Direction	Outbound			Inbound		
User Class	Heavy	Medium	Light	Heavy	Medium	Light
IP address Percentage	2.9%	30.1%	67.0%	2.9%	30.1%	67.0%
Traffic Percentage	46.6%	49.4%	4.1%	41.6%	47.9%	10.5%
NEWS	68.6%	30.4%	1.0%	68.4%	30.5%	1.1%
P2P	49.6%	49.5%	0.9%	46.2%	52.1%	1.8%
TCP-BIG	64.9%	33.1%	2.0%	51.5%	44.5%	4.0%
WEB	8.5%	52.2%	39.3%	9.8%	56.6%	33.6%
P2P Users in that Class	83.6%	63.4%	10.1%	83.6%	63.4%	10.1%
News Users in that Class	25.8%	12.4%	2.6%	25.8%	12.4%	2.6%
News or P2P Users	96.7%	71.6%	12.6%	96.7%	71.6%	12.6%

Table 5: P2P and News Users in a region having more than 100 000 subscribers.

On the other hand the heavy users are mainly generating file sharing traffic. Those who are using the popular P2P applications are now becoming a new type of content provider since their P2P traffic balance is below 1. Eventhough that subscriber group accounts for only 2.9% of the subscriber population, it generates almost half of the P2P traffic (table 5). What is more surprising is that not all of them are using these P2P applications to download files. Only 83.6 % of them installed one of these major P2P applications, while 13 % preferred to use only NetNews to get content. Finally the remaining 3 % chose other solutions that include FTP and downloads from the Web. It is interesting to notice that NetNews and the Web are only a means to download content but not to share it and so the traffic balance for these applications is very large: up to 50 bytes received for one byte sent.

Looking at the evolution of the traffic balance of Web traffic of the heavy users also leads to the conlusion that a more complex phenomenon is happening. Indeed in June 2002, the web traffic balance of the heavy users – 10.1 - was clearly higher than the web traffic balance of the light users whereas, in February 2003, that heavy hitter web traffic balance went down to 5.7, i.e. even lower than the one of the light users. This suggests that web traffic starts to be contaminated by a more balanced traffic, namely P2P applications. Furthermore, the traffic balance per application is another evidence that most of the traffic classified as TCP-big this year was actually what was classified as P2P last year. While the TCP-big traffic of the heavy hitters increased enormously, its traffic balance shifted from 2.0 to 0.9 and is now equal to the traffic balance of the P2P traffic that is still classified as P2P. It is now high time to understand why we are reaching the limits of port based identification of P2P traffic.

LIMITING P2P TRAFFIC

The ability to accurately identify P2P traffic is a crucial requirement for appropriately handling this traffic in the network - through either traffic engineering, provisioning, rate-limiting or pricing. However, P2P applications have evolved rapidly in a direction which makes accurate accounting of the traffic more difficult. In particular, previously the applications used default TCP ports, and it was possible to account for the bulk of the P2P traffic by monitoring a relatively small number of ports. However, the current widespread use port-hopping makes such mapping exceedingly impractical. We next present specific evidence of this trend and then discuss the implications for managing this traffic.

Kazaa Rate limiting Experiment

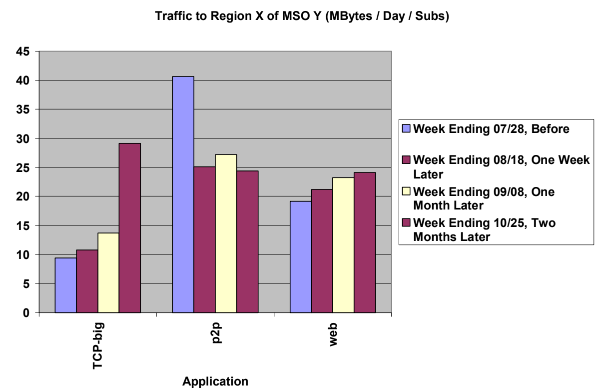

Figure 5: Mutation of P2P traffic into TCP-big traffic.

We first show an interesting case study which graphically illustrates how difficult it can be to limit P2P traffic. In Fall 2002, a particular MSO began rate limiting traffic on port 1214 (the default port for KaZaA). Fig. 5 shows the IN traffic for web, p2p and TCP-unknown for a particular region of that MSO before and after the rate limiting was initiated. Note that the P2P traffic decreases significantly after the rate-limitation was initiated. However, the TCP-unknown starts increasing and in 2 months has grown to **YYYY**% of its value just before rate–limiting began. The web traffic (port 80, 8000, 8080) also increases over the same period. A reasonable explanation for the jump in the TCP-unknown traffic coincident with the rate limiting action on the KaZaA port. **(Alex, do**

we know of the heavy users were the same for this TCP-unknown as for KaZaA ?) is that the traffic spurt was caused by KaZaA traffic migrating to other ports that were mapped to TCP-unknown.

This conclusion is supported by the previous findings of this paper, but we shall investigate in even more detail. Fig. 6 plots the per-port traffic distribution for July 2002 and 2003, for the heavy hitter ports for the 2 time periods. Note that in 2002, **x%** of the total traffic was contributed by only y ports. However, in Feb 2003, the traffic was much more uniformly distributed among a larger number of ports – the top **y** ports now account for only **z%** of the traffic. To get **x%** of the traffic we would need to monitor a larger number (**N**) of ports.

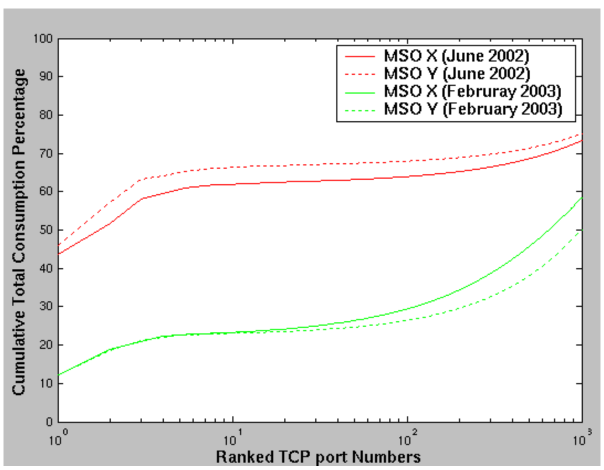

Figure 6: Distribution of traffic by TCP port numbers classified as P2P or TCP-big

Much more difficult is the task of mapping the traffic on these heavy-hitter ports to specific applications. Given the use of port-hopping by bandwidth-intensive applications like P2P, an important unanswered question is how much of the traffic on these ports can be attributed to the IANA-registered applications, and how much is P2P.

Given the limitation of port-based accounting, one might try to develop alternative techniques to accurately identify P2P applications. For example, additional information such as packet-level data, identification of SuperNodes etc. could help in developing signatures of P2P traffic. However, P2P applications have exhibited remarkable ability to rapidly evolve to evade detection and control.. For example, many P2P applications now encrypt their communications, making it more difficult to reverse-engineer and/or monitor such systems at the application-level.

The above trends have important implications for port-based traffic control of P2P applications. If the rate control is targeted to a few well-known P2P ports, a significant fraction of the P2P traffic will evade the limit., by hopping to other ports. The alternative is to track a larger number of ports that contribute significant traffic volumes and that are suspected to carry P2P traffic. The problem with this approach is that (i) it may not be feasible to track such a large and potentially dynamic set of ports), and (ii) such a widespread rate control may adversely affect the performance of many non-P2P users running valid applications on these other ports – this would be undesirable for the MSO.

SERVICE EVOLUTION TO TAME THE P2P GUERILLA

There are an assortment of approaches to address the "problem" of P2P traffic. Let's review a few.

Over the past few years may MSOs have incorporated "caps" into their service definition. These service caps tend to be implemented by controlling the rate at which data can flow into or out-of the network. The effect of these caps is to limit the instantaneous peaks of on-demand transactions. This has started us down the path of keeping bandwidth hogs in check.

Some MSOs are now adding "tiered caps". This allows the bandwidth hogs to identify themselves as such and pay a price for the enhanced service they are receiving. This tiering also allows self identification by modest users to get a price break.

Caps have been good to the industry and take us half way to where we want to go. However, P2P traffic is a relative "passive" phenomena. The requester can queue-up a set of requests for files then walk away. The file provider doesn't even need to be at the serving PC. In this situation rate capping will make the requests take longer, but will likely not change the behavior of the P2P participants. Figure **X** enforces this point with the lower correlation between of P2P traffic to the times users tend to be at their PCs.

Attempts to manage P2P traffic explicitly have met with little success. As illustrated in Figure **Y**, attempts to block standard ports of one P2P application only cause the user population to shift their behaviour so that the traffic reappears on other ports. Devices inside the network to block or significantly throttle specific port numbers have questionable economic return given the "slipperiness" of ports that P2P applications use and the risk of valid applications also using those ports.

Not that we should treat High Speed Data Services as a classic utility, but let's look at how other "utilities" handle the problem of consumption hogs. Water, power, landline phone utilities all have a "pay for what you use" model. There is no attempt in these industries to limit the usage, besides the economic consequence of paying for what is used. Cell phone providers put an additional twist on this model and provide usage bands. These bands allows a subscriber to sign-up for a usage band that best represents their need, but then gets charges for usage beyond what is included. With these revenue models consumption hogs are not "bad", they can pay for their usage and be good customers.

User response to these revenue models may not be as bad as we may fear. Users will be concerned that this will raise their rates. Surveys suggest that users on the average feel they themselves are heavy users. But Figure **Z** suggests only 5%-10% of the users are creating 50% of the traffic. With strategic selection of banding, the users will be pleasantly surprised to find that they can buy one of the lower bands. There will be a small percentage of users that truly will not be happy with their new rates and will balk to other broadband services, but those are the ones that the cable industry can afford to lose.

CONCLUSION

To summarize this paper, we have examined a large set of flow-based measurements of network traffic from several MSOs, over the the recent history of the Internet. The flow based measurements show several interesting features. Firstly, they illustrate that cable traffic is dominated by P2P applications. We further look into the properties of the various application classes, in particular the traffic patterns, and IN/OUT ratios, noting that P2P traffic has a much more balanced traffic pattern and IN/OUT ratio than applications such as the web. In addition we show that geographic locality is not yet a dominant feature of P2P traffic.

The paper then considers the traffic patterns of users, showing that the well known 80-20 rule (80% of the traffic is generated by 20% of the users) applies here, but moreover that the heavier users actually tend to use different applications: heavy-users tend to generate more P2P and Netnews traffic, while light users tend to use more web, email and chat applications.

Finally the paper considers how one might control the large volumes of P2P traffic that currently flood the cable networks. The more obvious controls, such as rate limiting traffic on particular ports are shown to be ineffective, because they simply push the traffic onto alternate ports. A more practical approach is to bill the customer for the resources they use directly.

REFERENCES

[IANA1] Internet Assigned Numbers Authority, protocol-numbers: http://www.iana.org/assignments/protocol-numbers

[IANA2] Internet Assigned Numbers Authority, port-numbers: http://www.iana.org/assignments/port-numbers

[SAMP] Properties and Prediction of Flow Statistics from Sampled Packet Streams.
Nick Duffield, Carsten Lund, Mikkel Thorup.