**Appears in the Passive and Active Measurement Conference (PAM 2017)**

# Application Bandwidth and Flow Rates from 3 Trillion Flows Across 45 Carrier Networks

David Pariag[1] and Tim Brecht[2]

[1] Sandvine Incorporated
[2] Cheriton School of Computer Science, University of Waterloo

**Abstract.** Geographically broad, application-aware studies of large subscriber networks are rarely undertaken because of the challenges of accessing secured network premises, protecting subscriber privacy, and deploying scalable measurement devices. We present a study examining bandwidth consumption and the rate at which new flows are created in 45 cable, DSL, cellular and WiFi subscriber networks across 26 countries on six continents. Using deep packet inspection, we find that one or two applications can strongly influence the magnitude and duration of daily bandwidth peaks. We analyze bandwidth over 7 days to better understand the potential for network optimization using virtual network functions. We find that on average cellular and non-cellular networks operate at 61% and 57% of peak bandwidth respectively. Since most networks are over provisioned, there is considerable room for optimization.

Our study of flow creation reveals that DNS is the top producer of new flows in 22 of the 45 networks (accounting for 20 – 61% of new flows in those networks). We find that peak flow rates (measured in thousands of flows per Gigabit) can vary by several orders of magnitude across applications. Networks whose application mix includes large proportions of DNS, PeerToPeer, and social networking traffic can expect to experience higher overall peak flow rates. Conversely, networks which are dominated by video can expect lower peak flow rates. We believe that these findings will prove valuable in understanding how traffic characteristics can impact the design, evaluation, and deployment of modern networking devices, including virtual network functions.

## 1 Introduction

The Internet continues to grow in geographic reach and data volume. This growth is facilitated by considerable investment from fixed and cellular service providers into network infrastructure. This infrastructure includes numerous devices such as switches, routers, caches, middle boxes and other devices to supply provider branded services (e.g., streaming video). Traffic incurs higher cost and increased latency as it moves from the network's edge towards its core. This offers a natural incentive for providers to invest in infrastructure that reduces traffic to Internet Exchange Points (IXPs) and backbone networks by placing devices closer to subscribers. However, there are relatively few studies which provide an *application-aware* view of broad scale traffic across multiple ISP networks. This is primarily

due to the difficulty of building and deploying scalable measurement devices in independent, geographically distributed networks. We believe that a large scale, application-aware study of Internet traffic can provide valuable insights into how application protocols drive consumption of bytes and flows in network devices.

In this paper, we conduct a detailed analysis of data that has been gathered as part of Sandvine's series of Internet Phenomena reports [27]. The collection of this data is facilitated by an ongoing partnership between Sandvine and participating ISPs. Our key contributions in this paper are:

- We find that there is a wide variety in the bandwidth consumed and rate at which new flows are created by the same application or service across networks. This makes it very difficult to describe a "typical network".
- We analyze bandwidth over time with respect to peak bandwidth and show that, on average, non-cellular and cellular networks operate at 57% and 61% of peak bandwidth, respectively. Since most networks are over provisioned, this suggests that the use of virtual network functions to offer elastic bandwidth may offer significant reductions in operating costs.
- We find that DNS is the top producer of flows in 22 of 45 networks, accounting for 20% to 61% of flows in those networks. We believe this places a significant load on flow-aware network devices including SDN routers, security middle boxes, and subscriber billing systems.

## 2 Methodology

Table 1 details the size and scope of the data used in our study. The dataset covers 22 3G and 4G subscriber networks (which we refer to as either cellular or mobile) and 23 cable, DSL and WiFi (fixed or non-cellular) subscriber networks across 26 countries on six continents. These networks range from a cellular network with peak bandwidth of 240 Mbps to a fixed network with peak bandwidth of nearly 600 Gbps. In total, our dataset covers 7 days in each network for a total of 62.8 petabytes and over 3 trillion flows of anonymized traffic.

This data was obtained from networks which have deployed Sandvine's Policy Traffic Switch (PTS). The PTS is a family of programmable network appliances which can be configured for applications including traffic inspection, subscriber billing, and network attack mitigation. It is a high performance device, capable of inspecting traffic in real time at network line rates. As such, our methodology does not rely on flow or packet sampling. We are able to inspect every packet of every flow. The PTS is usually deployed at the *edge* of the network, typically connected to termination points for cable and/or digital subscriber lines. The PTS is often used for subscriber billing purposes, which guarantees visibility of all subscriber traffic. Other deployments are possible, but we believe them to be uncommon in our dataset.

The PTS software stack examines packet headers at Ethernet, IP, and TCP/UDP layers as well as packet payloads at higher network layers. In concert with information gathered from lower layers, packet payloads are matched

against an extensive collection of known signatures. Strong signatures can identify an application after a single packet. Other signatures may require multiple consecutive packets before a match is returned. Categorization of encrypted traffic relies on heuristic methods instead of payload inspection. For example, encrypted HTTPS traffic is usually immediately preceded by a DNS request from the client endpoint. The IP address in the DNS response primes the PTS to *expect* a new flow from the client to the resolved hostname in the near future. Similarly, SSL handshakes are sent as clear text and include information that identifies the server endpoint. These methods allow the PTS to identify services being delivered over HTTPS with a high degree of confidence. The accuracy of the Sandvine recognition engine is verified using a regression suite consisting of several thousand flows generated from live application testing (i.e., ground truth data). The recognition engine is updated on a monthly basis to account for new or changed application signatures.

| Region | Abbrev. | Countries | Sites | Traffic (PB) | New Flows (billions) |
|---|---|---|---|---|---|
| Asia Pacific | APAC | 3 | 3 | 1.0 | 152.9 |
| Caribbean and Central America | CCA | 5 | 7 | 1.7 | 162.1 |
| Europe | ERP | 8 | 13 | 22.7 | 949.1 |
| Middle East and Africa | MEA | 5 | 5 | 25.3 | 1,276.5 |
| North America | NA | 1 | 11 | 11.3 | 327.6 |
| South America | SA | 4 | 6 | 0.8 | 152.9 |
| Total | | 26 | 45 | 62.8 | 3,021.0 |

**Table 1.** Scope of data collected

The aforementioned recognition engine is run against every new flow in the network. A new flow is one whose 5-tuple (source IP, source port, destination IP, destination port, transport protocol) has no entry in the device flow table. Each new 5-tuple adds an entry to this flow table. For UDP flows, the flow entry is expired after 10 seconds without a packet transmission. Any subsequent packets transmitted with that 5-tuple are treated as a new flow. The 10 second UDP flow timeout is a default PTS configuration which conserves flow-related memory while accurately representing flow lifetimes. As a result, all UDP flows in our data are defined by this flow timeout. For TCP connections, the flow entry is terminated after a proper connection termination (i.e., after a TCP four-way handshake) or after the TCP TIME-WAIT timeout has expired without a packet transmission (i.e., at least 2 Maximum Segment Lifetimes, which is 240 seconds). The PTS classifies each new flow into one of nearly 2,000 application protocols. Once a flow is categorized, the PTS attributes all bytes and packets of that flow towards the identified application protocol. These application protocols include well-specified protocols such as DNS, FTP, SIP and MGCP. However, they also include traffic generated by well-known applications or services such as Skype, Windows Update, and WhatsApp. The popularity of HTTP as a transport protocol has led to several *refinements* of HTTP being classified as separate applications. For example, YouTube, Facebook, and Hulu are recognized as separate application protocols even though each is delivered over HTTP. In

the interest of succinct analysis, we have created 21 application categories. Most categories are self-explanatory; those that require explanation are discussed when they are introduced. However, it is worth mentioning that the Misc category includes traffic that does not fit in any of the other 21 categories, as well as traffic that the PTS could not recognize.

Each PTS logs time-series data including byte, packet, and flow counts per application protocol to a centralized data store every 15 minutes. For the purposes of our study, we have retrieved the aforementioned time-series data from each network site for 7-day periods from June 2014 to September 2015. Note that packet payloads are not captured, only metadata gathered from payload inspection. Data collection and retention policies vary across operators, and as a result the 7-day periods vary across networks. Ownership of the data remains with the network operator and access to the data must be granted by each operator. Our analysis is based on post-processing data extracted from data stores.

## 3 Understanding Bandwidth Consumption

This section seeks to identify the applications which drive byte consumption and peak bandwidth in the networks under study. Figure 1 plots byte consumption by application category for all 45 networks. The x-axis lists each of the 21 application categories, and the y-axis shows the percentage of bytes consumed by each category. Fixed (non-cellular) networks are plotted to the left of the grid line using a green square, while mobile (cellular) networks are plotted to the right of the grid line using a red circle. Recall that the 7 day periods may differ across networks.

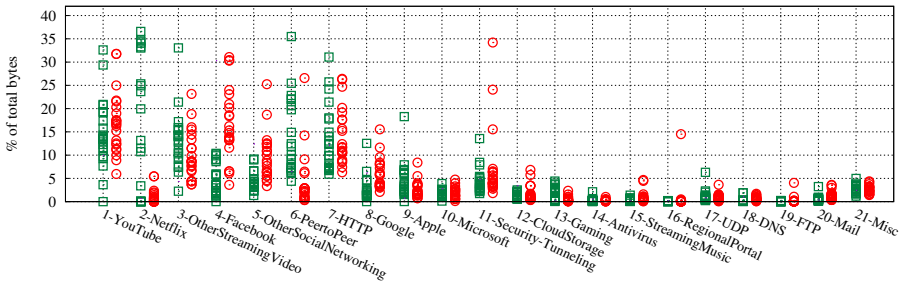


**Fig. 1.** Percentage of bytes by application

As might be expected, video traffic (YouTube, Netflix and OtherStreamingVideo) is a significant consumer of bytes on many networks. The byte consumption ranges for YouTube are similarly large on both fixed and mobile networks, while Netflix consumption is noticeably lighter on mobile networks. Facebook byte consumption on many mobile networks is higher than seen on fixed networks. However, the most striking feature of Figure 1 is that most application categories exhibit a large spread in byte consumption across many networks. For example, PeerToPeer traffic ranges from 0.35% of bytes in a Central American mobile network, to 35.49% of bytes in an Asian fixed network. Netflix, YouTube, Facebook, and other traffic categories exhibit similarly large spreads in either

fixed or mobile networks. Figure 1 shows that there is wide diversity in the popularity of different application categories in different networks. We have examined traffic by region, and except for noting that several non-cellular North American networks are dominated by Netflix traffic, we find few similarities across different networks, even within the same region.

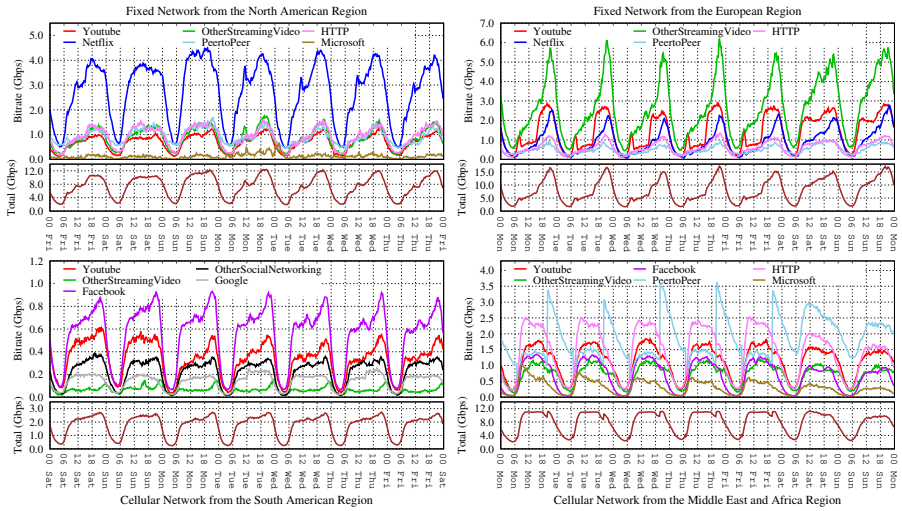


**Fig. 2.** Bandwidth in fixed North American (top-left), fixed European (top-right), mobile South American (bottom-left) and mobile African (bottom-right) networks

Figure 2 presents 7-day bandwidth versus time plots for four different networks. data points are plotted every 15 minutes, with each point representing the average bandwidth over the previous 15 minutes. Each graph consists of two panels, with the top panel showing bandwidth for individual applications which consume more than 10% of bandwidth at any point during the week. The bottom panel plots total bandwidth for *all* applications over the week.

The top two graphs in Figure 2 show video-dominated networks in North America (top-left) and Europe (top-right). The North American network is dominated by Netflix video. During peak bandwidth (which occurs at approximately 8 pm local time), Netflix consumes 40% of network bandwidth. Interestingly, the bandwidth plot also shows a local maxima just before noon on weekdays. As the bottom panel shows, the shape of the total bandwidth curve is shaped by Netflix usage patterns. More importantly, we have observed a similar degree of influence in nine other fixed North American networks where Netflix dominates bandwidth. The European network (top-right) is also video-dominated, with Netflix, YouTube and a regional provider (labelled OtherStreamingVideo, and intentionally anonymized) each consuming more than 10% of bandwidth. However, it is the regional service that exerts the greatest influence on peak bandwidth. Bandwidth for the regional service peaks between 8 pm and midnight on weekdays, and causes sharp but fairly short-lived peaks in total bandwidth.

The two lower graphs of Figure 2 show a South American mobile network (bottom-left), and a network from the Middle East and Africa (MEA) region (bottom-right) that are not video-dominated. Facebook is the leading consumer of bandwidth in the South American network, while PeerToPeer and HTTP are the top protocols in the MEA network. Our primary point in presenting these four graphs is to illustrate that there is no such thing as a *typical* network. In our dataset, networks differ significantly in terms of the applications which consume the most bandwidth, and the magnitude and duration of peak bandwidth. In addition, we have not seen any clear patterns emerge by region or network type, except in North American fixed networks where Netflix dominates the percentage of bytes consumed and peak traffic.

## 4 Peak Versus Off Peak Bandwidth

This section examines how daily patterns in bandwidth consumption can be used to identify opportunities for network function virtualization (NFV) to reduce resource consumption (including energy conservation). NFV presents network operators with the opportunity to replace dedicated physical appliances with virtual appliances built on commodity hardware. This can potentially reduce energy consumption during off peak periods by consolidating load onto a smaller pool of dynamically provisioned virtual appliances [17]. The use of commodity hardware permits operators to take advantage of the power management technology leveraged in data centers [26][10][24]. The combination of Software Defined Networking (SDN) and NFV allows network operators to establish tradeoffs between power consumption and network performance [6], ultimately leading to significantly more efficient network infrastructures [5]. For example, Bolla et al. [4] have examined the traffic profiles of a Greek research network and a Telecom Italia subscriber network, and they argue that energy-efficient techniques may offer energy savings in excess of 60%.

Intuitively, networks with low night time troughs and sharp peaks offer greater opportunity for energy savings than networks with higher troughs and broad daytime plateaus. Figure 3 shows the average bandwidth consumed relative to the peak over the 7-day period for each network. This chart shows that the average bandwidth consumed relative to the peak is slightly higher for cellular than non-cellular networks. This is because many non-cellular networks tend to have sharper, more short-lived daily peaks than cellular networks.

At first glance, the potential for savings may not seem very large. For example, for the two networks with the highest bandwidth to peak ratios (0.80 and 0.81), the potential for reduction would seem to be no more than 20%. However, most networks are provisioned with capacity that exceeds the 7-day observed peak. If we denote peak bandwidth by $p$, capacity by $c$, and define $r$ as the ratio of network capacity to the observed peak (i.e., $r = \frac{c}{p}$), then for networks which can dynamically adjust resources to meet demand, a bound on the possible bandwidth reduction is: $r + (1 -$ average to peak bandwidth ratio).

Across the networks studied on average these reduction bounds are: $r + (1 - 0.61) = r + 0.39$ for cellular networks and $r + (1 - 0.57) = r + 0.43$ for non-

cellular networks. Table 2 shows these reduction bounds for some values of $r$. If as one study has suggested, $r = 2$ [4], and more in some instances, then on average these networks provide significant opportunities for resource reductions by adjusting resources to efficiently meet demand.

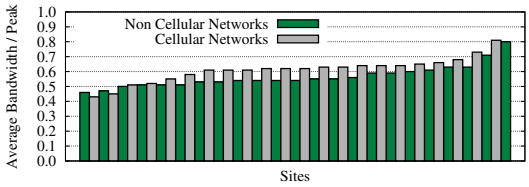


**Fig. 3.** Average bandwidth to peak ratio

| r | Cellular | Fixed |
|---|---|---|
| 0 | 0.39 | 0.43 |
| 1 | 1.39 | 1.43 |
| 2 | 2.39 | 2.43 |
| 3 | 3.39 | 3.43 |

**Table 2.** Reduction bounds

## 5 Peak Flow Rates

Networking devices often store per-flow state in memory, and perform a flow lookup to associate each packet with a new or existing flow. Flow state is useful for detecting network threats [3][23][21] such as address scans, port scans, and reflector attacks. In addition, per-flow state is required for usage-based subscriber billing, which is required by many network operators.

In flow aware devices, which may include intrusion detections systems, carrier grade NATs, and some load balancers, the incoming packet rate determines the flow lookup rate, and the new flow rate determines the flow table *insertion rate*. The arrival of a new flow often triggers additional processing. For example, in an OpenFlow router, a new data flow may trigger a request to a controller node in order to complete the routing decision [12]. Similarly, the recognition engine of the Sandvine PTS executes on every new flow arrival. The new flow rate is thus an important determinant of performance for flow-aware systems as high new flow rates can lead to high processor load [13], and even flow exhaustion.

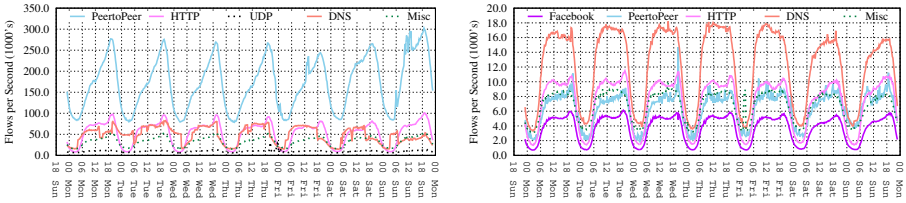


**Fig. 4.** New flow rates of fixed access (left) and mobile (right) European networks

In this section we study the applications that drive the creation of new flows. Figure 4 plots new flow rates over 7 days for fixed and mobile networks located in the same European country. As before, only applications which contribute more than 10% of total flows are plotted. In the fixed network (left graph), PeerToPeer applications are the chief creators of new flows, with peak new flow rates over 250,000 new flows/sec. At their peak, PeerToPeer flows constitute nearly 45% of all new flows. In the mobile network (right graph), it is DNS that drives flow creation with daily peak flow rates between 16,000 and 18,000 new flows/second. We offer these networks as examples of a broader trend: DNS and PeerToPeer

applications account for the majority of new flows across all networks. DNS accounts for the highest percentage of new flows in 22 of 45 networks and more than 50% of all new flows in 3 networks. PeerToPeer flows account for the largest percentage of new flows in 22 of the remaining 23 networks, and comprise more than 50% of new flows in 8 networks. Interestingly, while PeerToPeer protocols dominate new flows in many fixed networks, they also account for the largest percentage of flows in several cellular networks.

The proportion of PeerToPeer flows is not unexpected because 1) The BitTorrent protocol is often served over uTP, which is a transport protocol layered on top of UDP [1]. 2) Some P2P implementations will actively change source and destination ports in an attempt to evade detection. 3) Peers send control messages (e.g., keep-alives) to each connected peer every two minutes. Many of these will count as new flows if the same 5-tuple is not reused within 10 seconds. The proportion of DNS flows captured by our data may be initially surprising. Early studies (circa 1997) [28] report that DNS constitutes less than 18% of flows. More recent work [8] tracks the incidence of DNS flows in a longitudinal dataset, and reports 22.55% to 54.87% of flows being DNS. However, we have identified several factors which help to explain the large proportion of DNS flows in many networks. First, many application protocols utilize DNS. If the name being resolved is not found in the local host's cache, this will result in DNS request(s). Second, DNS is commonly served over UDP, which is not connection oriented, causing each transaction to generate a separate sequence of datagrams. Operating system implementations now randomize the source port used in successive DNS requests [9][18] resulting in new 5-tuples (and thus flows) being generated. Third, popular web browsers such as Chrome, Firefox, and Internet Explorer implement *DNS prefetching* in which the browser speculatively resolves hostnames for embedded page objects [19]. Modern web pages contain a median of 40 embedded objects, with 25% to 55% of pages requiring contact with *at least* 10 servers [7]. Lastly, many DNS responses use very short TTL values to better support load balancing and fault tolerance across multiple servers. As a result, even hostnames that are frequently referenced may require repeated DNS resolution.

### 5.1 Peak flow rates by application

Intuitively, one would expect streaming video services and bulk download protocols like FTP to transfer a large number of bytes over a small number of flows. At the other extreme, one would expect DNS to transfer relatively little data over each flow. However, the *flow profile* of other applications (e.g., Facebook) is more difficult to intuit.

Figure 5 plots bandwidth normalized peak flow rates by application for all 45 networks. We calculate these rates by first identifying the 15 minute window with the maximum flow rate (Flows/sec). This flow rate is broken down by application, and then normalized with respect to the bit rate (Gbps) over the 15 minute window. This results in a ratio with units of Flows/Gbit. We only include data points if either the number of flows or bytes accounts for more than 0.5%

of the application's flow or byte count, respectively. We normalize by bandwidth to compare networks of different sizes.

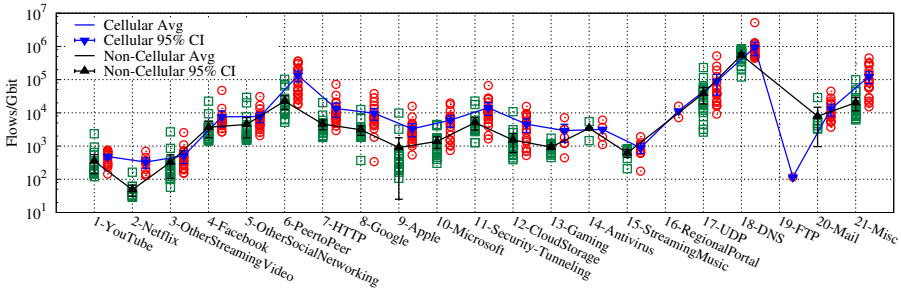


**Fig. 5.** Peak flow rates by application (log scale)

In Figure 5 the units on the y-axis are plotted using a log scale. Fixed networks are plotted to the left of the grid lines using a green square. Mobile networks are plotted in red and are offset to the right. Two trend lines are included to show the mean peak flow rates for each application along with 95% confidence intervals (CIs). Note that the CIs shown for Antivirus, RegionalPortal and FTP should be ignored because there were insufficient data points to compute CIs.

Figure 5 provides a number of key insights. First, we see that an individual application's peak new flow rates can vary by one or two orders of magnitude across different networks (e.g., YouTube ranges from 119 Flows/Gbit in one North American network to 2,347 Flows/Gbit in a MEA network). However, the variation across applications can be even larger, as can be seen by comparing the flow rates for Netflix and DNS in Figure 5. Lastly, for several application groups the cellular networks have significantly higher peak new flow rates.

As expected, video services are at the low end of the spectrum with YouTube, Netflix and OtherStreamingVideo averaging, 352, 49, and 326 Flows/Gbit in fixed networks, respectively. At the other extreme, DNS averages 549,651 new flows per gigabit (and 936,399 in cellular networks). In fixed networks, application groups like Facebook, OtherSocialNetworking, PeerToPeer, and HTTP average 3,679, 4,484, 22,531, and 4,593 Flows/Gbit peak, respectively.

Networks whose application mix includes large proportions of DNS, PeerToPeer, social networking, and other flow-intensive applications can expect to experience higher overall peak flow rates. Conversely, networks which are dominated by video can expect lower peak flow rates. The overall peak flow rate can impact processing load for flow aware devices including OpenFlow routers, security devices, and billing systems. The large range of flow rates observed across different networks poses challenges when building and deploying cost-effective devices. We believe our data will be useful in the design, sizing, and testing of future devices.

## 6 Related Work

Very early traffic studies focused on individual backbone networks [28] or research networks [14]. However, the constantly changing nature of Internet traffic [20] limits their value and necessitates new research. More recent reports from

Akamai [2] and Cisco [11] have included more geographically diverse data but are not application-aware. Sandvine's Internet Phenomena reports [27]) analyze regional traffic composition, often with a focus on identifying longitudinal changes or documenting the impact of special events. This paper focuses on identifying applications which drive bandwidth and flow creation over time and at peak.

Maier et al. [22] study the characteristics of residential broadband traffic circa 2009. They report that HTTP dominated byte consumption (57% of bytes), and that peer to peer traffic may not be as high (14% of bytes) as previously reported [15]. This study [22] covers a single digital subscriber line (DSL) network, and their application analysis is based on two 24 hour packet captures and fourteen 90 minute captures. While their analysis of HTTP traffic reports HTTP content types, they do not differentiate *services* delivered over HTTP (e.g., Facebook).

Labovitz et al. [20] examine the evolution of inter-domain traffic from 2007 to 2009. They use deep packet inspection (DPI) to categorize traffic on five subscriber networks. They note the rise of legacy video protocols (e.g., RTSP), and the decline of peer to peer traffic over the study period. However, they do not separate video delivered over HTTP from other Web traffic. As a result, they attribute less than 3% of traffic to video.

Richter et al. [25] conduct an application-aware study of Internet traffic at one European IXP. Their methodology relies on random packet sampling (which may miss packets containing rich identifying information), and their application recognition examines just 74 bytes of TCP payload. They report that 57% of traffic is HTTP and 10% is HTTPS but offer no insight into the services that are delivered using those protocols.

As more services are delivered over HTTP, it is increasingly important to differentiate these services. As noted above, several earlier papers [22][25][16][20] have broadly classified 20% to 58% of bytes as HTTP, Web, or browsing. Our methodology can inspect entire packet payloads, and reliably identify HTTP-based services as well as proprietary protocols (e.g., Skype). This is important because, as we have demonstrated, peak utilization can vary by service and understanding such patterns can enable more efficient network management. Additionally, our data set is taken from the *network's edge* and spans 45 provider networks across 26 countries. As a result, we measure traffic that may not be routed to IXPs (e.g., PeerToPeer and content that is cached near the edge). Both IXP and edge perspectives are valuable, but we believe that edge measurements provide an important view that is under-represented in the literature.

## 7 Conclusions

This paper presents an application and service aware analysis of bytes and flows from 7 days of Internet traffic from 22 cellular and 23 non-cellular networks across 26 countries to better understand how application traffic patterns impact network resource consumption. The analysis covers 62.8 petabytes *of payload data* and over 3 trillion flows, which makes it one of the largest such studies that we are aware of. We find that flow rates and bandwidth patterns are highly

localized, with little similarity among networks or network types. In our analysis, we have not found factors which define a *typical* network.

We demonstrate that one or two applications can drive peak bandwidth and influence the shape of a network's bandwidth curve. This is important because the width and height of peak bandwidth and the depth of nightly troughs defines a *peak reduction bound* that can guide the deployment of NFV and SDN solutions which aim to reduce equipment and energy costs. We find that DNS traffic accounts for 25% of the three trillion flows examined and more than 50% of flows in several networks. We believe this is due to the large number of links embedded in modern web pages, aggressive DNS pre-fetching implemented in modern browsers, and short time-to-live settings for many DNS responses.

## 8 Acknowledgments

Tim Brecht's work was partially supported by a Natural Sciences and Engineering Research Council of Canada Discovery Grant. Thanks to Dan Deeth, Ian Wormsbecker, and Sau Cheng Lim at Sandvine for their assistance in gathering data, understanding network deployments, and for feedback on several drafts of this paper. We also thank Bernard Wong and S. Keshav from the University of Waterloo for their comments on an earlier version of this paper.

## References

1. http://www.bittorrent.org/beps/bep_0029.html.
2. Akamai Technologies Inc. Akamai's State of the Internet. Vol. 7, No. 4, Q4 2014.
3. P. Barford, J. Kline, D. Plonka, and A. Ron. A signal analysis of network traffic anomalies. In *2nd ACM SIGCOMM Workshop on Internet Measurement*, 2002.
4. R. Bolla, R. Bruschi, A. Carrega, F. Davoli, D. Suino, C. Vassilakis, and A. Zafeiropoulos. Cutting the energy bills of Internet service providers and telecoms through power management: an impact analysis. *Computer Networks*, 56(10):2320–2342, July 2012.
5. R. Bolla, R. Bruschi, C. Lombardo, and S. Mangialardi. Dropv2: energy efficiency through network function virtualization. *IEEE Network*, 28(2), March 2014.
6. R. Bolla, R. Bruschi, C. Lombardo, and D. Suino. Evaluating the energy-awareness of future Internet devices. In *IEEE Conference on High Performance Switching and Routing*, July 2011.
7. M. Butkiewicz, H. V. Madhyastha, and V. Sekar. Understanding website complexity: Measurements, metrics, and implications. In *ACM IMC*, 2011.
8. V. Carela-Español, P. Barlet-Ros, A. Bifet, and K. Fukuda. A streaming flow-based technique for traffic classification applied to 12 + 1 years of Internet traffic. *Telecommunication Systems*, pages 1–14, 2015.
9. S. Castro, M. Zhang, W. John, D. Wessels, and k. claffy. Understanding and preparing for DNS evolution . In *Traffic Monitoring and Analysis Workshop*, 2010.
10. K. Christensen, P. Reviriego, B. Nordman, M. Bennett, M. Mostowfi, and J. A. Maestro. IEEE 802.3 az: The road to energy efficient Ethernet. *IEEE Communications Magazine*, 48(11):50–56, 2010.
11. Cisco Systems Inc. The Zettabyte Era: Trends and Analysis, May 2015.

12. A. R. Curtis, J. C. Mogul, J. Tourrilhes, P. Yalagandula, P. Sharma, and S. Banerjee. Devoflow: Scaling flow management for high-performance networks. In *ACM SIGCOMM*, 2011.
13. C. Estan, K. Keys, D. Moore, and G. Varghese. Building a better NetFlow. In *ACM SIGCOMM*, 2004.
14. M. Fomenkov, K. Keys, D. Moore, and K. Claffy. Longitudinal study of Internet traffic in 1998-2003. In *Winter International Synposium on Information and Communication Technologies*, 2004.
15. C. Fraleigh, S. Moon, B. Lyles, C. Cotton, M. Khan, D. Moll, R. Rockell, T. Seely, and C. Diot. Packet-level traffic measurements from the Sprint IP backbone. *IEEE Network Magazine*, 17(6):6–16, 2003.
16. K. Fukuda, H. Asai, and K. Nagami. Tracking the evolution and diversity in network usage of smartphones. In *ACM IMC*, 2015.
17. B. Han, V. Gopalakrishnan, L. Ji, and S. Lee. Network function virtualization: Challenges and opportunities for innovations. *IEEE Communications Magazine*, 53(2):90–97, January 2015.
18. D. Kaminsky. Black ops 2008: It's the end of the cache as we know it, 2008. http://www.slideshare.net/dakami/dmk-bo2-k8.
19. S. Krishnan and F. Monrose. DNS prefetching and its privacy implications: When good things go bad. In *USENIX Conference on Large-scale Exploits and Emergent Threats*, 2010.
20. C. Labovitz, S. Iekel-Johnson, D. McPherson, J. Oberheide, and F. Jahanian. Internet inter-domain traffic. In *ACM SIGCOMM*, 2010.
21. J. Mai, C.-N. Chuah, A. Sridharan, T. Ye, and H. Zang. Is sampled data sufficient for anomaly detection? In *ACM IMC*, 2006.
22. G. Maier, A. Feldmann, V. Paxson, and M. Allman. On dominant characteristics of residential broadband Internet traffic. In *ACM IMC*, 2009.
23. D. Moore, C. Shannon, D. J. Brown, G. M. Voelker, and S. Savage. Inferring Internet denial-of-service activity. *ACM Transactions on Computer Systems*, 24(2):115–139, 2006.
24. S. Nedevschi, L. Popa, G. Iannaccone, S. Ratnasamy, and D. Wetherall. Reducing network energy consumption via sleeping and rate-adaptation. In *NSDI*, 2008.
25. P. Richter, N. Chatzis, G. Smaragdakis, A. Feldmann, and W. Willinger. Distilling the Internet's application mix from packet-sampled traffic. In *PAM*, 2015.
26. E. Rotem, A. Naveh, A. Ananthakrishnan, D. Rajwan, and E. Weissmann. Power-management architecture of the Intel microarchitecture code-named Sandy Bridge. *IEEE Micro*, 2(2):20–27, 2012.
27. Sandvine Inc. Global Internet Phenomena, December 2015. https://www.-sandvine.com/trends/global-internet-phenomena/.
28. K. Thompson, G. Miller, and R. Wilder. Wide-area Internet traffic patterns and characteristics. In *Network, IEEE Vol. 11, Issue 6*, Nov 1997.