Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Institut für
Technische Informatik und
Kommunikationsnetze

Searching for Periodic Flows in an Internet Backbone

Placi Flury [*]
flury@tik.ee.ethz.ch
Computer Engineering and Networks Laboratory,
Swiss Federal Institute of Technology (ETH) Zürich, Switzerland

TIK-Report No. 210

11th January 2005

Abstract

If questioned on a traffic pattern occurring every day at precisely the same time, one would intuitively attribute it to some timed application such as e.g. a mirror update. Nonetheless has research so far neglected to investigate the key properties of above, apparently obvious, phenomenon. This report presents a methodology for detecting traffic exchanges at periodical intervals between hosts. We applied the methodology on NetFlow statistics of an Internet backbone. We discovered about 15% of the TCP traffic on a border router to be periodic; moreover, comparably few flows contribute to this phenomenon. With the prospect of traffic engineering we examined the traffic's property of time persistence.

1 Introduction

The dynamics of the Internet are today rather tamed by experience values of network operators, as by traffic models and theoretical analyses of protocols. The power of rules of the thumb and experience values however, is expiring as networks become more complex. Moreover, assuring a smooth interplay of novel and legacy network applications depends increasingly on the well funded understanding of key properties of the Internet. The recent devotion to network measurement and traffic analyses seeks therefore to derive assumptions upon which, network operators and researchers can rely on.

The expressiveness of assumptions and models, and their usability in simulations, in theoretical analyses, and in the development of protocols and applications depend on the accuracy of how well they succeed in reflecting the peculiarities of their real life counterpart. The network traffic analysis on a backbone is our contribution in catching some of these very peculiarities.

We analyze long term traffic data collected at border routers of the Swiss Education and Research Network (SWITCH). Our prime interest in on periodic occurrences of traffic patterns, more precisely, on data exchanges among hosts, that take place repeatedly at identical times of

[*] This work was made possible by generous provision of NetFlow data by SWITCH.

day. The objective is to assess its potential usage for novel traffic engineering techniques. We will therefore single out and present the most prominent characteristics of periodic traffic.
Searching for traffic patterns in NetFlow [7] traces of several months asks for concepts, algorithms, and systems that can cope with large amounts of data in reasonable time. The report starts therefore with a section on the applied methodology. A brief introduction of the NetFlow capturing technology and the presentation of the characteristics of the measured network precede the report on the concepts and implementation of our processing architecture. In section four we present and analyze our results and findings. Section five positions our findings with current state of the art research. The report concludes with an outlook and mentions open issues.

2 Methodology

The elaboration of a methodology for real life traffic analysis is to a great extent predetermined by the properties and characteristics of the metered system. The general settings of the topology and the dimension of a network, and of the profile of the user base, are usually beyond the sphere of influence, such as, in our case, are the choice of the measurement technology and the choice at what locations to measure. To us SWITCH is the network, NetFlow version 5 the measurement technology, and four border routers of SWITCH the interception points. Upon these three pillars shall we elaborate our methodology.

NetFlow

NetFlow is a traffic capturing technology widely supported by Cisco routers. It collects and aggregates traffic statistics on the level of single flows. In NetFlow terminology, a flow is defined as a unidirectional sequence of packets described by the identical tuple of source (IP, port, interface), destination (IP, port, interface), protocol and the type of service (TOS) field. The statistics on flows are cached in a flow table on the router. A flow gets flushed out of the table if any of the following events occurs:

- the flow has been idle for a specific time (30 seconds in our setting).
- the flow exceeded a maximal living time in the cache, (15 minutes in our setting).
- the cache becomes full. The policy is to apply heuristics in order to flush out entries aggressively.
- if, for a TCP flow, a FIN or RST was captured.

The statistics of flows, called flow records, are grouped in UDP datagrams and exported to external collecting devices. Note, NetFlow statistics are not exhaustive. In particular, under heavy load do routers obey their prime duty of routing and defer from the continuous collection of statistics.
NetFlow's capturing strategy asks for various techniques and heuristics when it comes to reconstructing the close to thorough picture of the reality at capturing time. Sommer et al. present in [12] a heuristic for reconstructing TCP connections from NetFlow records. As will be explained later, we are using similar heuristics for the reconstruction of flows.

Swiss Education and Research Network (SWITCH)

The Swiss Education and Research Network (SWITCH [13]) interconnects all national universities and educational institutions, including the European Organization for Nuclear Research (CERN). It maintains peering agreements to international educational networks (GÉANT, Internet2) and to the commercial Internet by national and international carriers; the later explicitly for global transit at high bandwidth [14].
A peculiarity of SWITCH is its negligible amount of transit traffic, i.e. traffic of external autonomous systems (AS) that traverses SWITCH . The monthly traffic volume leaving SWITCH's Gigabit/Ten Gigabit Ethernet network (upstream) resides currently at about 200 Terabyte (as for summer 2004).

Interception Points and Data Volume

A close and generous collaboration with SWITCH gives us access to NetFlow data of SWITCH's border routers at Geneva, Basel and Zurich, which permits us to trace all traffic leaving and entering the SWITCH backbone. An objective therefore is to treasure the raw NetFlow data in a continuous and complete archive lasting for several months, thus also keeping a foundation for future research questions on network traffic. Since the data generated by a single router ranges from about 9 million NetFlow records at low activity to an average of about 20 million NetFlow records per hour at peak activity (as for summer 2004), above objective asks for a prudent design.

2.1 Concepts and Design Issues

One obvious challenge in analyzing real network traffic at a large scale (in terms of time and space) lies in managing and processing the enormous amount of accumulated data. Depending on the capturing technology, not only can the amount of data but also the information value of the data vary considerably. For a comparison of capturing methodologies and the impact on available information see Sommer et al. in [12].

Another, less obvious challenge consists in reconstructing and interpreting the setting at capturing time. Here again does the requested granularity determine the amount of efforts to deploy. Reconstructing traffic volumes and shares of protocols at precise points in time needs less effort and fewer information than a full reconstruction of single flows or connections for example.
The difficulty and discrepancy is caused by an asymmetry in the available information, which increases with the demand for more precision and granularity. It is the classical situation, where anticipating the cause in a causality chain by judging on the effects is often not univocal. The information asymmetry results from displaced points of views on cause and on effect.
In network research such situations are frequent. They emerge from the diametrical setting, where the cause occurs either in the network or the hosts and the effect on the opposite is intercepted either at hosts or the network. A prominent example is TCP's flow control mechanisms, which attempts to cope with network congestion. Another will be our effort to reconstruct a TCP connection from NetFlow data. (Flow fragments shall be used to anticipate the full characteristics of the flow, possibly to the extent of knowing the state of the protocols (e.g. TCP's state protocol machine)).

Two approaches exist for dealing with information asymmetry. One targets at extracting most of the information from the effects. In our case this means to enhance the precision and granularity of the traffic interception mechanisms. This approach however results in a tradeoff between low information asymmetry and large data volume.
The other approach suggests to use heuristics. Finding good heuristics is however not a trivial issue. Nevertheless are we attempting to use both approaches, although the highest granularity is delimited by NetFlow's interception precision.
Concepts we apply in order to handle and process NetFlow data with above background are:

- fragmentation, i.e. splitting up the continuous stream of NetFlow statistics into files of fixed duration, e.g. store statistics on an hourly basis.
- use of a unique scalable naming scheme for both files of raw NetFlow data and preprocessed files.
- compression of data. We envisage two kinds of data compression, the first targets at an optimization of storage and memory usage. It applies in particular to raw NetFlow data. The implementation of tools for processing NetFlow data ought consequently to support the decompression and compression of raw NetFlow files on the fly.
  The second kind of compression, we call reduction, targets at an optimization of the processing speed. The idea is to extract only relevant information from the large pool of NetFlow data in pre-processing steps, and to run the algorithms doing the real job on this filtered data set. The reduction factor of filtering is far higher than for loss-free file compression algorithms (as e.g. bzip2). Filtering and still keeping a significant and expressive sample of the whole, however, is a challenge, which needs to be carefully assessed.
- speeding up of processing time by deployment of parallel processing techniques, distributed computing, and pipelining. For parallel processing and load distribution we use Scylla [9], a cluster of 22 Linux nodes.
- usage of heuristics, in particular for the reconstruction of settings at capturing time.
- identification and implementation of intermediate processing steps. Different research questions may have common intermediate pre-processing requirements. The intermediate results may be archived with the raw data, in order to save processing time for future research.
- visualization of results in order to simplify the recognition of patterns.
- preservation of privacy and confidentiality by contracts and a strict usage policy. (e.g. cluster Scylla decoupled from production network.)

3 Searching for Periodic Flows

We define a flow as a unidirectional sequence of packets that belong to the same four–tuple of <`src_IP, dst_IP, src_port, dst_port` >. A flow is furthermore fully delimited by a starting and an ending time. A TCP connection consists therefore, given this definition, out of two unidirectional flows.

We define two or more flows to be periodic if *a.)* their starting times are identical to some additive period of time (typically relative to multiples of 24 hours) and *b.)* the source IP addresses and destination IP addresses of the flows are identical. We call the point in time and the IP addresses of a periodic flow its signature. Ports are not included in the signature, because hosts rarely choose twice the same pair of ports for the same type of communication. The definition does moreover not imply that two periodic flows need to be identical at (application layer) protocol level [1].

Periodic flows may presumably result from daily mirror updates, adjustments of news group content, online journalism, possibly customer habits etc.. We expect the point in time when periodic flows occur not to be under control of the ISP, in particular if the ISP is a backbone provider. The target of our research is to first prove the existence of periodic flows, and second to collect the characteristics on periodic flows that seem to be suitable for traffic engineering (TE). Prime characteristics is their share on overall bandwidth consumption, their bandwidth variation over daily time, and of particular interest for TE, their persistence in time, i.e. the ability to identify periodic flows on consecutive time intervals based on a fix set of signatures.

3.1 Approach

A decent and expressive search on periodic flows requires at least observation samples in the range of several hours (for inter–hour periodicity), but preferably in the range of days. Our setting targets at diurnal periodicity and focuses on TCP data. The extension to UDP data is kept for future work.

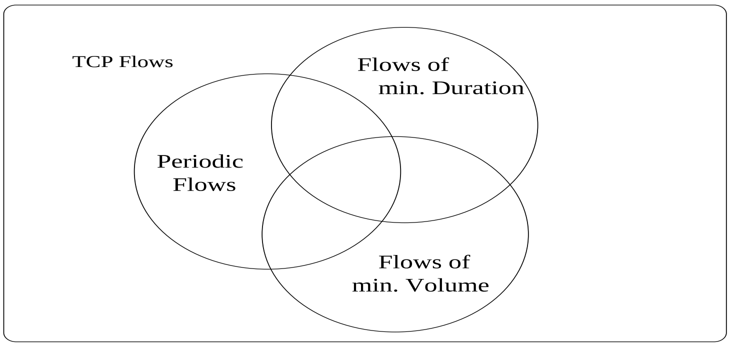

Figure 1: *Filtering criteria and search space for periodic flows.*

Our approach consists out of three steps:

1. Reconstruct the unidirectional flows of the TCP connection. We use an almost identical methodology as proposed by Sommer et al. in [12]. The reasons for the deviation are, *a.)* we are interested in flows and not in bidirectional connections, and *b.)* Our NetFlow data lacks of the accumulated information on TCP flags.

[1] Although we expect the application layer protocols almost always to be the same. For our objective of assessing traffic characteristics for novel traffic engineering mechanisms, above definition of periodic flows is appropriate enough.

2. reduce the search space for periodic flow by filtering out most promising data. We use the heuristic as depicted in figure 1. Only TCP flows, which have either a certain volume or a certain duration, are considered.

3. Search for periodic flows in the filtered data. The result (i.e, the intersection of both the volume and duration sets, with the set of all periodic flows as shown in figure 1) are flows of two consecutive days (if daily periodicity is our target) that have either a certain volume or a certain duration and that start at the same relative time (e.g. with some small tolerance of e.g. $+/-$ 2 minutes).

3.2 Limitations and Caveats

Above approach has several limitations and caveats with, depending on the research questions one intends to answer, diverse impacts. The most important are:

- lack of completeness. Short lived flows, which do not exceed a certain volume threshold (e.g.10MB) are not considered candidates for periodic flows. Such a decision accommodates the tradeoff between processing speed, feasibility, and accuracy.

- liberal definition of periodicity of flows. Omitting information on application layer protocols classifies any communication between two hosts starting at the same relative point in time to be periodic. Intuitively uncorrelated flows may thus be classified as periodic flows.
  A strict definition of periodic flows, including the protocol would be more appropriate. Estimating the application layer protocol from NetFlow data, however, requires good heuristics and a strong faith in the correct and consequent deployment of standards. Though even with such efforts would we by far not be able to classify all flows (e.g. almost every flow not using well known ports).

- good natured routing. For the reconstruction of TCP flows NetFlow data captured on a single router is used. Besides of losing NetFlow records while the router is under heavy load, records may also not show up again due to route changes. The later case, in fact infrequent, should nevertheless be kept in mind while searching for periodic flows at consecutive days or months, on data collected at a single router. The consolidation of data from multiple routers may be used in future as a remedy for lost NetFlow information.

- designed for TCP, i.e. the reconstruction of the setting at measuring time applies for TCP. UDP traffic will need other types of heuristics.

- potential problems with long-lived flows. A flow (as e.g. in P2P traffic) may live for several days. Since we reconstruct such a flow say for day A and the consecutive day B independently, it will show up as a periodic flow pair. Another problem is that long-lived flows may be bursty and have long time spans without activity. If the time of inactivity is larger than NetFlow's maximal caching time, the reconstruction of such flows is not anymore applicable with the heuristics we currently use. Yet another difficulty and imprecision with long lived-flows spanning several hours is the distribution of their cumulative transfered volume to hourly shares. We currently use the simple and computationally inexpensive assumption of a constant transfer rate, i.e. a uniform distribution.

- tolerance values. When searching for periodicity we use a tolerance value that may range up to several minutes.

3.3 Assessment of Approach

Scope of Eligibility

How well does the sample of filtered data reflect the properties of the underlying full data set? What is the range of research questions that can be answered by such a sample, consequently? In an attempt to answer both questions we identify the relevant properties of the full data set, and upon those are we to estimate the impact of our methodology and approach.
An unfiltered 48 hour measurement at SWITCH's border router in Zurich provides us with statistics on $552'978'572$ TCP flows. Figure 2 shows the distribution of the flows, and figure 3 the corresponding accumulated volume. For the plotting of those figures we varied the sampling rate, i.e. the size of the bins, for aggregation at discrete steps of $10^{floor(\log 10(x))}$ where x denotes either the volume or the duration[2]. A value of 80ms on the duration–axis comprises therefore all flows within the time-range of 80ms to 90 ms, and a value of 3000Bytes on the volume–axis flows of a size ranging between 3000Bytes and 4000Bytes.
Table 1 lists the peaks seen in figure 2 that include more than 2 percent of the total number of TCP flows. An interesting property of the peaks is shown in figure 4. The immediate neighborhood of the peaks is sharply delineated. Apparently the largest fraction of TCP flows consists of one or two packets of a size between 40 and 50 Bytes. TCP packets of such sizes carry

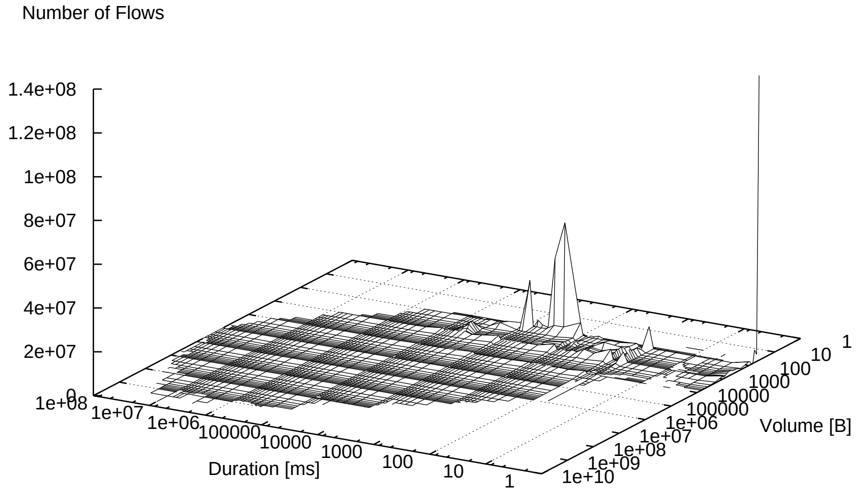

Figure 2: *Volume – Duration landscape of TCP flows from an unfiltered data set (Tue 7.9.04 at 00:00 – Wed 8.9.04 at 24:00).*

[2] Note, since our NetFlow configuration has a minimal resolution of 1ms, flows below 1ms duration have been rounded up to 1 ms (instead to 0 ms as our variable sampling rate would imply).

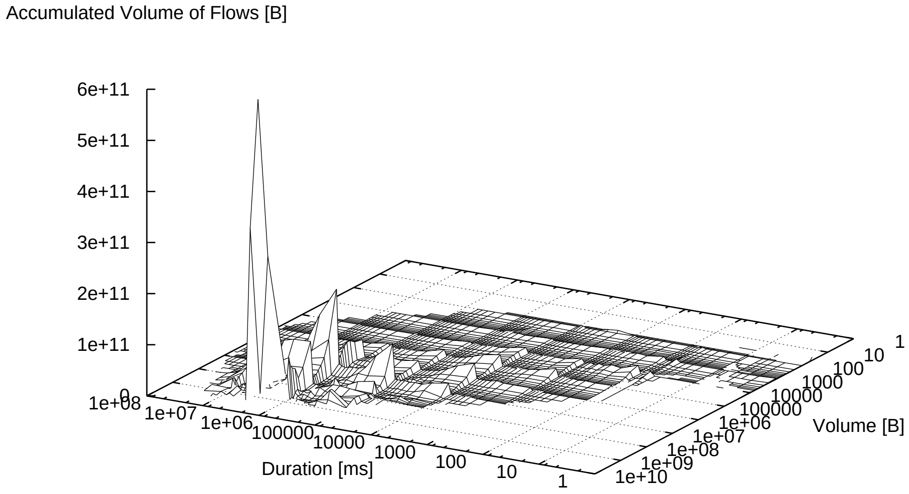

Figure 3: *Volume – Duration landscape of TCP flows from an unfiltered data set (Tue 7.9.04 at 00:00 – Wed 8.9.04 at 24:00).*

no payload but may still contain options though. Typical suspects may be *syn* packets, or *ack* packets of asymmetrically routed TCP connections. The *ack* packet thesis, however, suggests that the inter-arrival time of the *ack*s of the same TCP connection needs to range beyond 15 minutes, as this is the expiration time of NetFlow[3]. We did not yet search for any explanation for this phenomenon.

Figure 3 states that few though heavy flows contribute to the overall bandwidth consumption.

Duration/Volume	Percentage
1 ms/40 Bytes	23.51 %
2000 ms/90 Bytes	9.08 %
3000 ms/90 Bytes	6.02 %
8000 ms/100 Bytes	3.89 %
9000 ms/100 Bytes	2.56 %

Table 1: *Peaks of Number of flows distribution.*

This observation is not new and accords with findings of Fang et al. in [3] as well as Brownlee et al. in [2]. The decision to filter flows exceeding a duration of 910 secs OR a volume of 10MB seems, when comparing the plots of figures 2 and 3, to be reasonable iff we are to catch the flows responsible for the largest fraction of bandwidth use in the backbone.

A yet not answered question is what protocols are most affected by our filtering policy. We classify protocols in interactive and system protocols. Interactive protocols reflect user behavior, whereas system protocols those that get triggered by events such as timers. Given that

[3] With our reconstruction heuristics the time interval of *ack*-absence would be 15 minutes plus a TCP expiration value of circa 3 minutes.

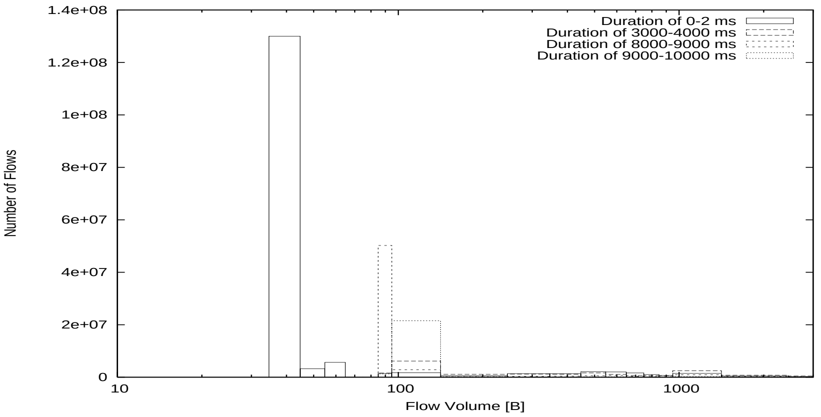

Figure 4: *Close-up of flow peaks of table 1.*

taxonomy we postulate that interactive protocols will be affected most by our filtering policy (as long as TCP traffic is concerned). A more detailed analysis on that matter is kept for future work.

Benefits of Approach

For the estimation of the gain of filtering, we look at the ratio of total number of TCP flows and the number of TCP flows, which get filtered. We call this ratio the reduction factor. Figure 5 shows the variation of the reduction factor for a representative day on an hourly basis. The mean value of the reduction factor given the hourly sampling rate ranges at factor 557. The sampling rate, however, introduces a not negligible error, since it cuts flows that span the borders of an hour into two pieces, so they show up twice in the statistics. Although both filtered and unfiltered flows are affected, the effect on the result is not proportional (i.e. not linear); the

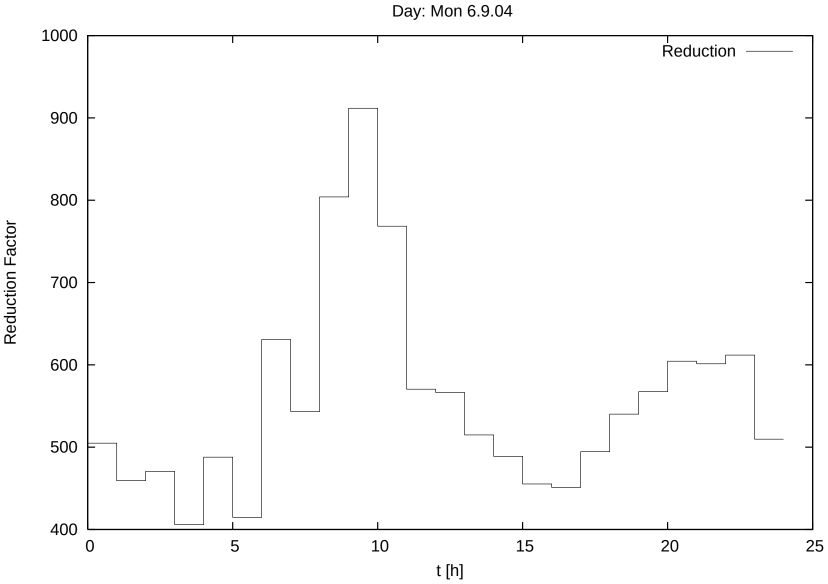

Figure 5: *Reduction factor of TCP flows obtained by filtering flows of minimal duration or of minimal volume. Set to 910 seconds and 10MB respectively.*

number of small flows cut increases faster than does the number of long-lived flows that get cut. For the assessment of the diurnal reduction factor we therefore use a bin of 24 hours. The error of cutting flows becomes then negligible, resulting in an average reduction factor of 420 (with filtering criteria of 910 seconds for the duration and 10 MB for the volume).

Given a computational complexity of approximately $\mathcal{O}(n)$ for most of our algorithms we expect the speedup by application of above filtering policy to be within the range of the reduction factor. There is an exception though, the algorithm for searching and detecting identical flow signatures has a complexity of $\mathcal{O}(n'^2)$, where n' denotes the size of the search space. The search space of this algorithm enfolds a very small subset of a trace, typically all signatures within the same relative time plus some tolerance value of circa +/- 2 minutes.

4 Findings and Discussion

The results presented in the following section refer to TCP data only, which has been captured at SWITCH's border router in Zurich. The filtering parameters were set to 910 secs and 10 MB. In order to minimize side–effects at the beginning and end of diurnal statistics, the input data has a safety margin of 15 minutes to the previous, and 60 minutes to the subsequent day.

We were able to prove the existence of periodic flows leaving and entering the SWITCH network. The following section presents the most outstanding characteristics of the located flows. A subsequent discussion appraises their aptitude for traffic engineering.

4.1 Periodicity

A characteristic of prime interest is the amount of traffic periodic flows account for. This is shown in figure 6. The figure depicts the total amount of TCP traffic, the fraction of filtered traffic, which is used for the quest on periodic flows, and the fraction of periodic traffic found, as seen for week 33[4]. Both, the filtered and the periodic fractions are given relative to the total amount of TCP traffic. The absolute values are shown in figure 7.

In week 33, TCP accounts for 96% of the traffic volume with a deviation of $\sigma = 0.26\%$. Measurements on other weeks resulted in comparable findings on total amounts and fractions. Periodic flows contribute approximately to 15% of the total amount of traffic if using periodicity intervals of 24 hours. Table 2 holds the values for periodicity intervals of a week and of a month with week 33 as reference. The weekly periodicity has been computed to the days of week 32, i.e. Sunday 8 has been compared with Sunday 1^{st} etc.. As for statistics on monthly periodicity we used calendar months, i.e. August's Sunday 8 is compared with September's Wednesday 8.

In addition to percental fractions the table lists the mean duration and mean volume of the flows. We observe that the number of flows as seen with diurnal periodicity drops dramatically for weekly and monthly periodicity intervals. The associated cumulated traffic amounts, however, do not change considerably for the various periodicity intervals. In an attempt to interpret this observation, we postulate that a large portion of flows identified on diurnal periodicity still reflects interactive behavior. We further suggest that interactive flows are rather sporadic, i.e. not very persistent in time (due to the variability of user behavior). The two assumptions would explain the drop in number of flows.

[4] Week of Sunday 8 to Saturday 14 of August 2004

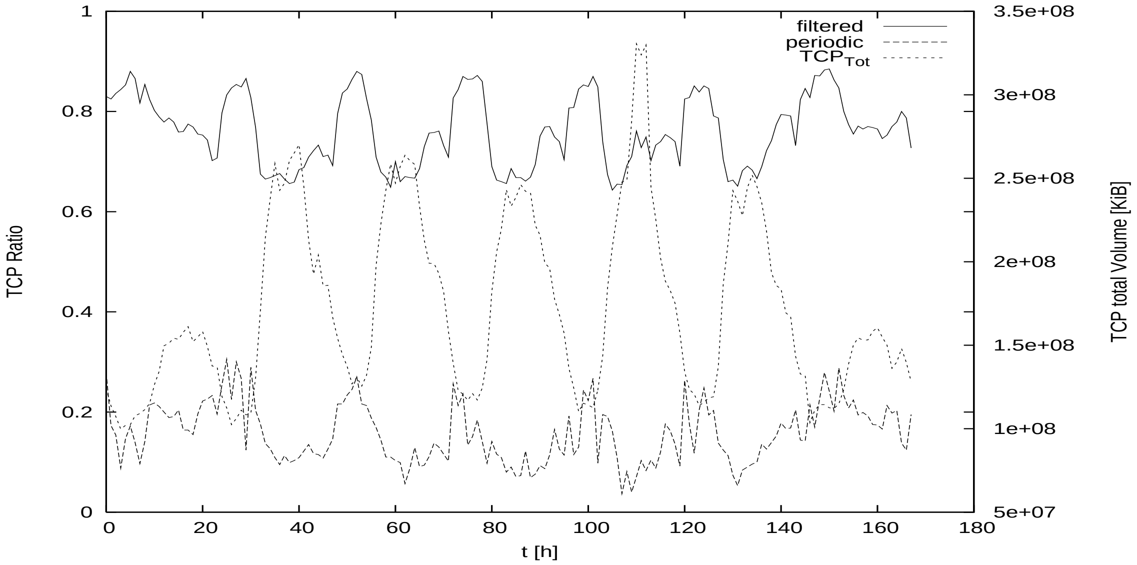

Figure 6: *Percentage of filtered and periodic data volume, as seen for week 33 on a single router with a periodicity interval of 24 hours.*

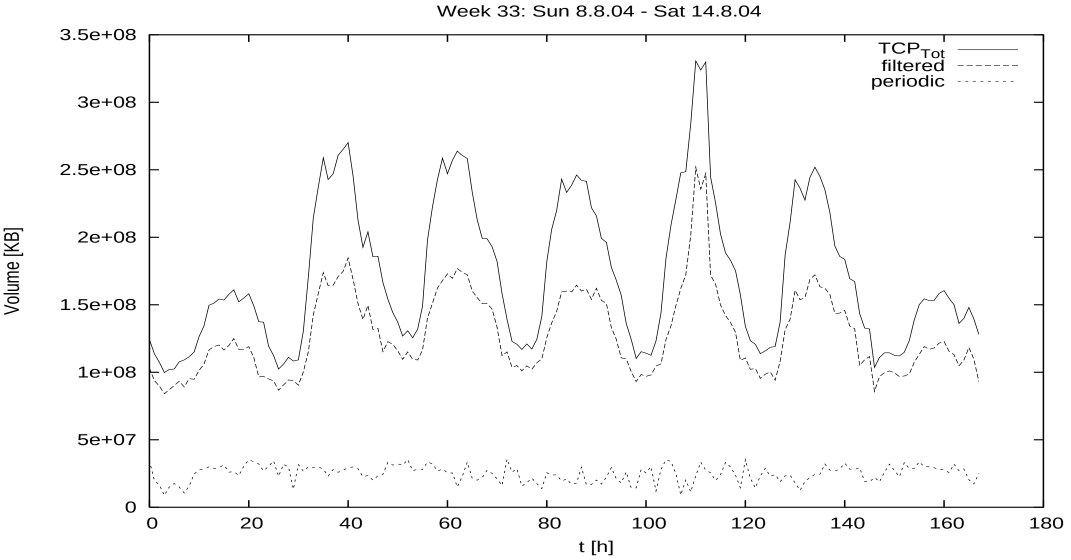

Figure 7: *Volume of filtered and of periodic data as seen during week 33 on a single router.*

The docile variation of the accumulated traffic, on the other hand, implies that the traffic amount generated by the interactive flows (or by the flows that do not show up anymore), is relatively modest. A look at the mean values for the volumes confirms the last statement, although not only do the mean values but also the variations increase with larger periodicity intervals.
We refrain from a thorough validation of above argumentation chain, in particular whether flows are of interactive nature or not, since the result is rather insignificant (for traffic engineering), whereas the effort would not be. We settle however for an analysis of involved protocols. Since estimating the protocols by means of port numbers is, as mentioned, not a trivial task, besides being imprecise and error prone, we refer to port numbers without direct associations to protocols.

Periodicity Interval	Mean	Variance	Number of Flows	Total Volume
Diurnal (Duration) (Volume)	15.74 % (of overall TCP Volume) $\mu_D = 4865$[sec] $\mu_V = 56'029$[KiB]	0.35% $\sigma_D = 9653$ [sec] $\sigma_V = 250'902$ [KiB]	75714	4046 GiB
Weekly (Duration) (Volume)	15.4% (of overall TCP Volume) $\mu_D = 4212$ [sec] $\mu_V = 92'211$[KiB]	0.4% $\sigma_D = 8687$ [sec] $\sigma_V = 322'428$ [KiB]	44370	3902 GiB
Monthly (Duration) (Volume)	14.34% (of overall TCP Volume) $\mu_D = 3512$ [sec] $\mu_V = 122'038$[KiB]	0.46% $\sigma_D = 7156$ [sec] $\sigma_V = 359'129$ [KiB]	30522	3552 GiB

Table 2: *Characteristics of periodic flows for various periodicity intervals. Numbers apply to cumulated data of week 33.*

Diurnal Periodicity Interval

Port Range	# of Flows	Σ Volume
0 – 1023	16298	175.9 GiB
1024 – 49151	44531	2013.2 GiB
49152 – 65535	14885	1856.5 GiB

Monthly Periodicity Interval

Port Range	# of Flows	Σ Volume
0 – 1023	10664	104.8 GiB
1024 – 49151	13745	1809.0 GiB
49152 – 65535	6113	1638.5 GiB

Table 3: *Port grouping, source ports considered.*

We cluster traffic statistics into the groups of *well–known*, *registered*, and *dynamic or private* ports as suggested by IANA in [5]. The resulting statistics for week 33 are presented in table 3, if source ports, and in table 4, if destination ports are considered. Note, the notion of source and destination port is different if referred to connections or to flows. Figure 8 depicts the difference.

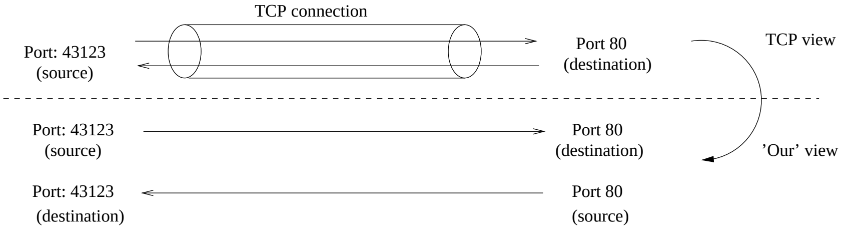

Figure 8: *Different perception on source and destination ports.*

Our clustering reveals the largest share of traffic volume to be *push*ed to well–known ports. This is surprising, since the most popular protocols are known to behave as *pull*ers, i.e. send small requests and get large(r) replies. Periodic flows are apparently different. The question is, ”who pushes here?” Is it the backbone with its connected research networks or is it an ’outsider’? A closer look on the volume of periodic traffic leaving and entering SWITCH (cf. table 5) unveils that about six times more traffic is leaving as is entering SWITCH on that

Diurnal Periodicity Interval

Port Range	# of Flows	Σ Volume
0 – 1023	19’513	3809.5 GiB
1024 – 49151	41390	135.1 GiB
49152 – 65535	14811	101.1 GiB

Monthly Periodicity Interval

Port Range	# of Flows	Σ Volume
0 – 1023	12939	3439.6 GiB
1024 – 49151	10143	62.4 GiB
49152 – 65535	7440	50.3 GiB

Table 4: *Port grouping, destination ports considered.*

	Entering Traffic	Leaving Traffic
Diurnal	584.74 GiB	3460.94 GiB
Monthly	514.76 GiB	3037.52 GiB

Table 5: *Volume of periodic traffic entering and leaving SWITCH on a single border router.*

particular router. Since the amount of leaving traffic clearly outperforms the entering traffic, and since the amount of pushed traffic to destinations (on well–known destination ports) is of about the same size, we state the backbone and the attached research networks to be the pushers. Note however, that we still consider the property of how the traffic is distributed among ports and the property how much traffic is leaving or entering a network to be orthogonal (at least we do not recognize clear patterns of dependency yet).
Table 6 ranks the ports according to their number and according to the accumulated volume as seen for week 33. The first four ranked ports on traffic volume are all well-known ports (although port 435 seams, given the purpose of well–known ports, to be misused for a different purpose). Remarkably, port 25 shows up with only 9 flows and a volume of 52 MiB in the monthly periodicity interval, while it ranks at place four and seven on daily periodicity. We keep the closer investigation on that matter.

	Diurnal Periodicity Interval				Monthly Periodicity Interval			
Rank	Port	# of Flows	Port	Σ Volume	Port	# of Flows	Port	Σ Volume
1	119	9172	435	2271 GiB	119	8712	435	2139 GiB
2	6667	5951	119	1434 GiB	435	2224	119	1208 GiB
3	80	3932	433	73.82 GiB	6667	1250	433	77.95 GiB
4	6881	2456	25	17.86 GiB	433	590	22	13.95 GiB
5	435	240	22	9.92 GiB	80	497	62404	2.6 GiB (1)
6	33434	1808	62061	7.4 GiB (27)	22	290	49954	1.7 GiB (2)
7	25	1781	62681	4.2 GiB (1)	9050	208	62061	1.23 GiB (4)
8	6882	736	64463	4.08 GiB (1)	7777	115	10022	1.13 GiB (1)
9	433	580	36480	2.96 GiB (1)	143	111	45213	796 MiB (2)
10	7000	570	63307	2.79 GiB (6)	110	93	63307	751 MiB (1)

Table 6: *Ranking of top 10 Ports.*

4.2 Aptitude for Traffic Engineering

Special treatment of 15% of the traffic alone will not satisfy most TE objectives. Incorporating periodicity attributes, however, can be a beneficial complement of existing TE practices. We focus on a topic in TE, which enjoys some attention in current network research , namely the persistence in time of the classification of heavy hitter flows. Persistence, in this context, refers to a property of time stability, more formally, it characterizes the classification performance of an unaltered reference set applied to a data set that varies over time. Two reasons make the issue of time persistent classification of heavy hitters difficult, first flow sizes and the durations are independent dimensions [2], and second most flows experience large bandwidth fluctuations during their lifetime [8].

Figure 9 depicts the persistence property of periodic flows. For the first curve we created a reference classification set by comparing Monday and Tuesday and extracting the flow signatures, which belong to Monday. We used the classification set unaltered for the search on periodic flows on all subsequent days, starting from, and including Monday. The reason why the auto-comparison on Monday results in more true positives is shown in figure 10. Note, signature a matches with a' and a*.

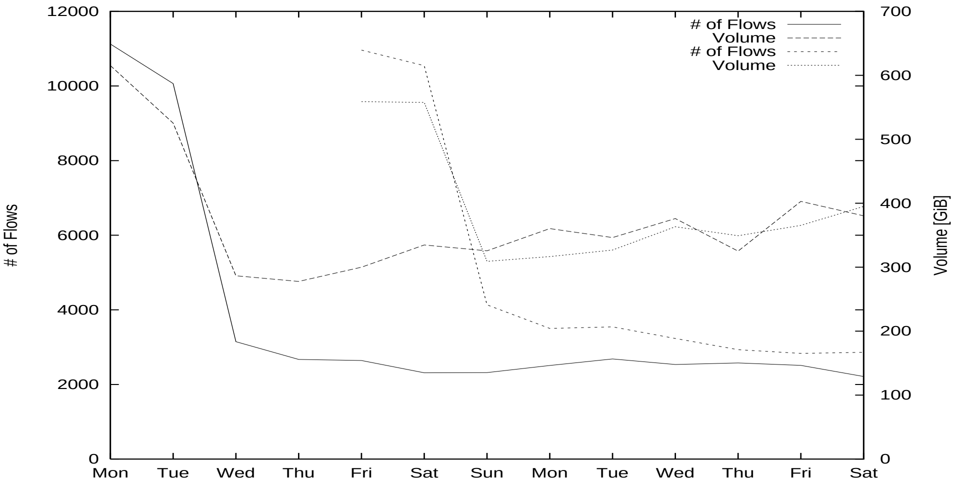

Figure 9: *Persistence of single set classifications.*

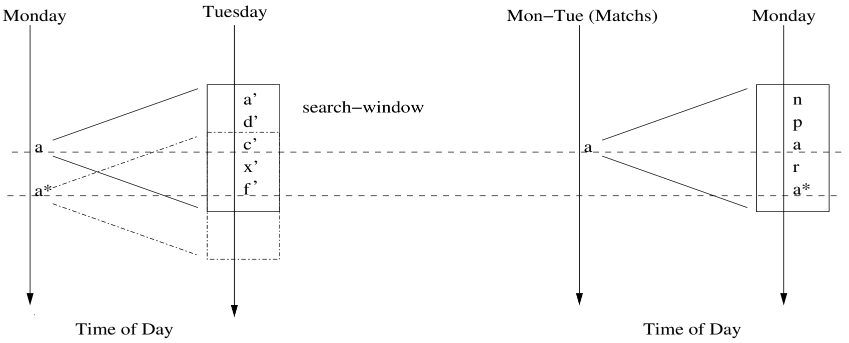

Figure 10: *Reason for higher auto-correlation.*

As figure 9 depicts, the number of flows as well as the volume immediately drop when the reference set is applied to days that have not been included in the computation of the reference set. While the number of flows stabilizes at around the number of 2500, the volume is steadily increasing. Interestingly, a re-classification on Friday (second curve) is not performing better on longterm basis. We may draw two tentative conclusions that need to be validated with further measurements. First, there are flows of considerable volume (about 2500 flows with a daily volume of about 380 GiB) that are quite persistent. Second, these persistent flows seem to be unaffected by temporal variations like usage fluctuations on week-ends; we conclude from the last observation that the responsible applications are time triggered, i.e. system applications.

5 State of the art

In [10] Shaikh et al. state that the granularity at which traffic needs to be classified for load–sensitive routing as a mean to improve utilization, is an important factor for the efficiency of traffic engineering. It primarily influences the route stability, i.e. the avoidance of route flapping, and minimizes the overhead of signalling. They proposed to route long-lived flows dynamically while proceeding with routing of short-lived flows on the pre-provisioned paths.

The ideas of using a finer granularity for traffic engineering were instantly incorporated, and the focus shifted towards the detection of so called heavy hitters and towards traffic measurement techniques for a better knowledge on traffic characteristics. Van Jacobson's volume motivated classification of heavy hitters into 'elephants' and 'mice' complemented by Brownlee et al.'s [2] duration motivated classification into 'dragonflies' and 'tortoises', reflect two sides of the same medal. Both views on heavy hitters impact practical problems of caching, router design and protocol performance [11]. Theoretical work on modeling (e.g. TCP fluid models) and simulations sensibly rely on assumptions on heavy hitters [6].

Currently the emphasis is set on the development of generic quantifier operators for the classification, and on the automation and optimization of the classification of heavy hitters [1, 8], in parallel more precise and granular metering tools [4] that can cope with large amounts of data are developed.

6 Future Work and Open Issues

The results we presented, apply to moderate time frames and few interception points. We clearly need to strengthen our findings with further measurements and analyses.

In order to be applicable for traffic engineering we need to evaluate the impact, a potential routing of few destination prefixes, has. We will focus on the granularity of the prefixes, since it determines the side-effects on non-periodic traffic.

A remaining and important issue is to estimate how representative the traffic of SWITCH with its 'research oriented' user base, is. Only in such a context may our findings be useful.

7 Conclusion

The conclusions we draw, still need to be validated with more data. We observed, however, that periodic flows contribute to a considerable fractions of the total amount of traffic. An interesting

property for traffic engineering is their persistence in time, which seems to be unaffected by fluctuations attributed to users, such as e.g. reduced activity on week-ends.
We believe a classification of heavy (periodic) hitters ought to include information on port numbers, if it targets at time persistence. The reason, port numbers permit to identify time-triggered flows, which are presumably the larger contributers. How far this applies to non-periodic flows will be part of future research.

8 Acknowledgements

We like to thank SWITCH for the generous provision of NetFlow data and the fruitful collaboration. Many thanks to Arno Wagner for providing libraries for analyzing NetFlow data, and to Martin May and Thomas Dübendorfer for the valuable comments and suggestions.

References

[1] A. Broido, Y. Hyun, R. Gao, and kc claffy. Their share: diversity and disparity in ip traffic. PAM 2004, Apr. 2004.

[2] N. Brownlee and kc claffy. Understanding Internet Traffic Streams: Dragonflies and Tortoises, 2002.

[3] W. Fang and L. Peterson. Inter-AS Traffic Patterns and Their Implications. IEEE Globecom, Dec. 1999.

[4] 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*, 2003.

[5] Internet Assigned Numbers Authority (IANA), Port Numbers. `http://www.iana.org/assignments/port-numbers`, Oct. 2004.

[6] Y. Joo, V. Ribeiro, A. Feldmann, A. Gilbert, and W. Willinger. On the impact of variability on the buffer dynamics in IP networks. In *Allerton Conference on Communication, Control and Computing*, September 1999.

[7] NetFlow Services and Applications, White paper. `http://www.cisco.com/warp/public/cc/pd/iosw/ioft/neflct/tech/napps_wp.h%tm`, Aug. 2004.

[8] K. Papagiannaki, N. Taft, and C. Diot. Impact of Flow Dynamics on Traffic Engineering Design Principles. In *IEEE Infocom*, Hong Kong, March 2004.

[9] TIK Experimental Cluster "Scylla". `http://www.tik.ee.ethz.ch/~ddosvax/cluster`, Aug. 2004.

[10] A. Shaikh, J. Rexford, and K. G. Shin. Load-sensitive routing of long-lived IP flows. In *Proceedings of the conference on Applications, technologies, architectures, and protocols for computer communication*, pages 215–226. ACM Press, 1999.

[11] S. Shakkottai, R. Srikant, N. Brownlee, A. Broido, and kc claffy. The RTT distribution of TCP flows on the Internet and its impact on TCP based flow control. Technical report, http://www.caida.org, 2004.

[12] R. Sommer and A. Feldmann. NetFlow: Information loss or win? Technical report, Saarland University, Saarbrücken, Germany, 2002.

[13] Swiss Education & Research Network (SWITCH). http://www.switch.ch, Aug. 2004.

[14] Connectivity map of the Swiss Education & Research Network (SWITCH). http://www.switch.ch/network/international.html, Aug. 2004.