Passive Crowd-based Monitoring of World Wide Web Infrastructure and its Performance

Martin Arlitt
HP Labs and University of Calgary
martin.arlitt@hp.com

Niklas Carlsson
Linköping University
niklas.carlsson@liu.se

Carey Williamson
University of Calgary
carey@cpsc.ucalgary.ca

Jerry Rolia
HP Labs
jerry.rolia@hp.com

Abstract—**The World Wide Web and the services it provides are continually evolving. Even for a single time instant, it is a complex task to methodologically determine the infrastructure over which these services are provided and the corresponding effect on user perceived performance. For such tasks, researchers typically rely on *active* measurements or large numbers of volunteer users. In this paper, we consider an alternative approach, which we refer to as *passive crowd-based monitoring*. More specifically, we use passively collected proxy logs from a global enterprise to observe differences in the quality of service (QoS) experienced by users on different continents. We also show how this technique can measure properties of the underlying infrastructures of different Web content providers. While some of these properties have been observed using active measurements, we are the first to show that many of these properties (such as location of servers) can be obtained using passive measurements of actual user activity. Passive crowd-based monitoring has the advantages that it does not add any overhead on Web infrastructure, it does not require any specific software on the clients, but still captures the performance and infrastructure observed by actual Web usage.**

I. Introduction

The popularity of the Web coupled with user expectations for free and swiftly delivered content strain the economics for content providers.[1] This may motivate some content providers to change their infrastructure; for example, Google has built larger data centers [20] and deployed its own wide area network (WAN) [10], which is now one of the busiest in the world [17]. Growing global concern over climate change is another reason why content providers may alter their infrastructures. Since data centers are significant contributors of carbon emissions [7], and carefully designed delivery architectures can improve the performance of the Web, further changes in the design, location, and operation of data centers are expected.

While it is important to be able to capture the evolution of the World Wide Web and the services it provides, it is a complex task to methodologically determine the infrastructure over which these services are provided, even for a single time instant. A typical approach for exploring infrastructure changes is to use active measurements (e.g., `traceroute`, as used in [10]). One attractive feature of active measurements is they can be conducted by anyone. However, two shortcomings of active measurements are (1) they add overhead to the infrastructure, potentially degrading user experience, and (2) they are not measurements of actual user experience [8].

---

[1]While companies such as Google provide much more than static content, we use the generic term "content providers" for lack of a better one.

Passive measurements can avoid these shortcomings, but they may be more difficult to obtain. This often leads researchers to prematurely dismiss this option.

However, a recent study by Choffnes *et al.* [4] reveals that with a little creativity, large-scale passive measurements can be acquired. In particular, they used a peer-to-peer system to demonstrate the power of crowdsourcing (event) monitoring for detecting network events such as service outages. A key observation in their work is that obtaining passive measurements of actual usage from a diverse set of clients can be done by leveraging existing infrastructure; in their case, computers running peer-to-peer client software. Rather than extending the client application, as done by Choffnes *et al.* [4], we show how crowd-based monitoring can be applied using system logs, with no modifications to the systems. This is important, as many organizations already collect network measurements (e.g., proxy or intrusion detection system logs) for auditing or security purposes, but are not likely to make changes to these systems for research purposes.

Our work examines four billion Web transactions, collected over a week-long period from 95 proxies located in 38 cities and 27 countries. We demonstrate how this passive approach enables actual user and infrastructure level attributes of the World Wide Web to be examined. Our results reveal differences in Web user experience by geography, distinguish content providers by their quality of service, determine differences in network throughputs, and identify services that were provided by a common infrastructure. While some of these properties have been observed using active measurements, we are the first to show that these properties (such as location of servers) can be obtained using passive measurements. A key advantage of this passive approach is that it provides insights on the infrastructure used to support *actual* Web usage, without placing any additional load on the measured infrastructure or on the end users; it is completely transparent to them. Our dataset also provides new insights into user experience and infrastructure differences between popular content providers.

The remainder of the paper is organized as follows. Section II discusses related work. Section III describes our dataset and explains our methodology. Section IV examines the user experience. Section V discusses insights on Internet infrastructure gleaned from our dataset. Section VI considers the global Web usage across services and regions. Section VII summarizes our contributions and lists future work.

II. Related Work

Passive measurements of actual application, system or network usage are highly desired. Often, such measurements are difficult to obtain. Recently, Choffnes *et al.* proposed a crowdsourcing approach to network monitoring [4]. As mentioned in Section I, our work complements their work in several ways. Gerber *et al.* [8] demonstrate how to use netflow logs to accurately assess network download speeds. Section IV-C offers several complementary results.

Our study uses logs from a set of globally deployed Web proxies. Web proxy logs are a rich source of information. They have been used for purposes such as workload characterization [3], [18] evaluating the benefits of Web caching [16], [19], and improving cache management [13], [24]. In this work we seek to extract additional useful information from Web proxy logs. In particular, we leverage different file transfer measures observed from different geographic locations.

Numerous works have examined aspects of the Internet topology, often via *active* measurements. For example, Gill *et al.* [10] used `traceroute` to reveal the size of content provider wide-area networks. Katz-Bassett *et al.* developed `reverse traceroute`, to improve the accuracy of path discovery [12]. Other works consider how to quickly identify which mirrored server a client should use [6], [11]. Stemm *et al.* [23] considered how to achieve this using *passive* measurements. Eriksson *et al.* [5] also used a passive technique, extracting network structure information from packet-level traces. Our work differs in that we investigate what information about the infrastructure can be extracted from passively collected, application-level data in existing system logs.

Lastly, Padmanabhan *et al.* used passively collected server-based measurements to glean insights into client and infrastructure performance [21]. In contrast, passive crowd-based monitoring provides similar capabilities across multiple content providers (each with many servers).

III. Methodology

A. Dataset

To facilitate this study, we obtained transaction logs from a set of globally distributed Web proxy caches used by an enterprise with roughly 300,000 employees world-wide. The logs were gathered for the same one week-long period across 95 proxies. These proxies are located in 38 cities in 27 countries on six continents; 47 proxies are from the western hemisphere, and 48 from the eastern hemisphere. Although the activity reflects business rather than residential use of the Web, the data enables us to passively monitor actual Web traffic from a "crowd" of users around the globe.

The logs we obtained were in numerous different formats. We extracted the common fields from each log to create a single aggregated log in a consistent format. The aggregated log contains one record for each completed HTTP or HTTPS transaction. Each record contains the start time, the duration, the number of request bytes (including HTTP headers), the number of response bytes (including HTTP headers), the action taken by the cache, the content type, the response status, the request method, the `Host` for the transaction, and an identifier to indicate which proxy handled the transaction.

We collected the Web proxy logs from midnight (UTC) on Sunday, September 20, 2009 until midnight on Sunday, September 27, 2009, across all 95 proxies. During this period, a total of 4.015 billion transactions were served by the proxies, and 68.83 TB of response data were delivered to the clients.

Overall, the North American clients accounted for just over half of the HTTP transactions and data transferred. Asian and European clients each accounted for about 20% of the activity. Clients in South America, Middle East/Africa, and Australia together account for 7% of activity. This roughly corresponds to the distribution of employees in the enterprise.
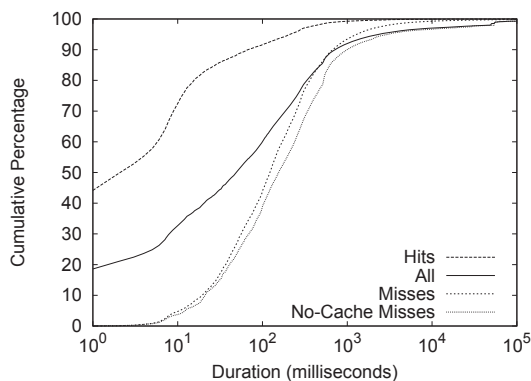
B. Information Extraction

In their study, Choffnes *et al.* instrumented a P2P client to collect the exact information of interest to them [4]. In this study we are using logs that were intended for a different purpose. To obtain the information we want, we combine (in various ways) three types of log data: the *action* taken by the cache for each transaction, the amount of data transferred, and the transfer duration. In the remainder of this section, we describe these fields and how we used them.
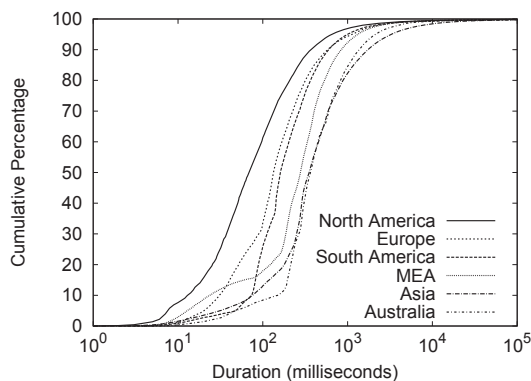
Since the proxies that collected the logs cache content, it is important to discern the different *actions* taken by the proxy. Across the geographies, on average 41% of all transactions resulted in a *hit*; i.e., were served from the proxy's cache. 39% of transactions result in a *no-cache miss*. This means that the requested object was labeled *uncacheable* by the content provider (via an HTTP response header). 11% of transactions resulted in a *miss*; this means a cacheable object was not available in the cache when requested by a user. Each type of action reveals some information that we can use. We elaborate on several specific ways below.

Web transfers tend to be for small objects [2]. This characteristic holds in our dataset. The median overall transfer size is only 758 bytes. This is influenced by the large number of transactions that return only HTTP headers (e.g., response status 304) or small objects like graphical icons. We exploit this characteristic to use the transfer duration of small cache misses as a first order approximation of the network latency between the cache and the content provider's server. Similarly, we use cache misses for large objects (about 1% of responses are larger than 10^5 bytes) to evaluate differences in the throughput with different providers.

The transfer *duration* value in each transaction record is the elapsed time from when the client's request is received by the proxy until the proxy has sent the complete response to the client. Figure 1 illustrates this. If the requested object is found in the proxy's cache (a *hit*), then the total duration $D = d_1 + d_2$; otherwise the proxy must retrieve the object from the content provider. In this case, the total duration $D \leq d_1 + d_2 + d_3 + d_4$. D is not strictly equal to the sum of d_1, d_2, d_3 and d_4, since the proxy attempts to hide latency when possible, e.g., by prefetching embedded images on HTML pages.

(a) By Cache Action (Global)

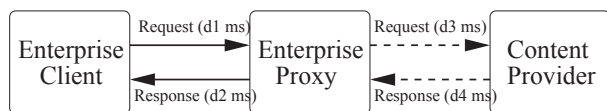(b) Cache Misses (by Region)

Fig. 2. Transfer Durations.

Fig. 1.   Explanation of Transfer Durations.

IV. User Experience

A general topic of interest for the Web is user experience. For example, projects such as Google's SPDY are devoted to making the Web faster.[2] In this section, we leverage the proxy logs to analyze the performance observed by a large number of clients. We first consider the aggregate performance over all services, then we explore the service provided by a few popular providers, and examine their infrastructure and performance differences by geography.

A. High-level Transfer Duration Differences

The primary indicator of user perceived performance in our dataset is transfer duration. Figure 2(a) shows the CDF of the transfer durations for all transactions in the dataset. Cache hits tend to have very short durations, with a median of 3ms. In general, caching improves the experience for all users. This can be seen from the wide gap between hits and misses in Figure 2(a). This is one indication of the importance of content providers marking content as "cacheable".

The overall transfer duration distribution ("All") has a median value of 50ms. This is significantly (and positively) influenced by the high cache hit rate (around 40% across geographies). Cache misses take considerably longer than cache hits, as each such transaction must be fulfilled by an external server. The median duration jumps to 123ms. The median value for no-cache misses is even higher, at 154ms, even though no-cache transactions tend to transfer less data. This indicates the degree to which the content provider's server(s) rather than the (wide area) network are the cause of the degraded user experience for these transactions.

Figure 2(b) explores how user experience differs by location. This figure shows the CDF for the transfer durations of cache misses by continent. This reveals what the experiences

of the enterprise's users would be across geographies if no caching were in place. Users in North America experience the best service, with a median duration of 69ms and the lowest durations overall. Users in Europe and South America have similar transfer durations for cache misses, with medians of 134ms and 162ms, respectively. Users in Middle East/Africa, Asia, and Australia have the poorest service, with median durations for cache misses of 281ms, 354ms, and 371ms, respectively. However, from this graph alone one cannot determine if the differences are due to the enterprise's network infrastructure or more general Internet infrastructure. We investigate this further in the following sections.

B. Content Provider QoS

We next consider the *quality of service* (QoS) experienced by users from different providers. For this analysis, we use the transfer durations of small cache misses (responses less than 1,460 bytes, including headers, which will typically fit in one TCP packet) as a first QoS measure. Such transactions represent the smallest exchange that we could observe between a proxy and an external server.

One of the most popular providers in our dataset is MSN. Figure 3(a) shows the distribution of small cache miss transfer durations across selected landmarks. This figure reveals that the shortest durations occur in the US, but even within the transactions at a given landmark, there is noticeable variability.

Figure 3(b) shows the results for Google. An obvious difference in this figure is that the durations at each landmark are remarkably consistent. This can be seen from the almost vertical lines through the initial 80% of the distributions. One hypothesis is that Google's use of its own WAN [10] enables them to provide more consistent response times.

Figure 3(b) also shows that 10–20% of the small miss transactions (the upper tail of the distribution) have much longer durations at each landmark. In a study of Google's CDN, Krishnan *et al.* observed that a sizeable fraction of clients experience higher latencies than other clients in the same geographic region, due to routing inefficiencies and packet queueing [15]. This is a case where the service level event monitoring described by Choffnes *et al.* could identify
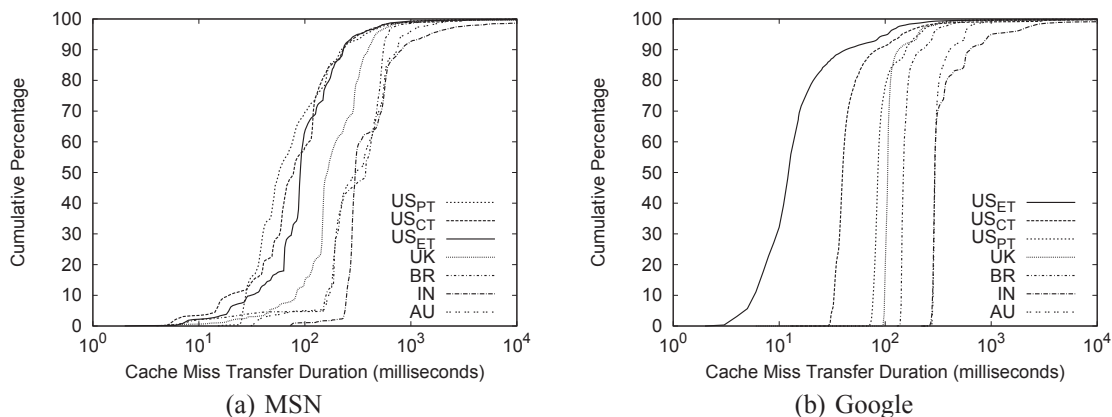
(a) MSN   (b) Google

Fig. 3. Small miss transfer duration distributions from landmarks to MSN and Google.

the incidents as they occur, and help find the root causes.

Another opportunity for content providers to improve the QoS they provide is to take advantage of Web caching. Table I provides information on the cacheability of content from different providers as well as their observed (global) cache hit rates. Several of the globally popular providers (MSN, BBC) have cache hit rates above the overall average. Google, the most popular provider in the dataset, saw only about half of its cacheable content result in hits. All of these providers labeled a significant fraction of their content as uncacheable.

Advertising is an important aspect of the Web from an economic perspective. Table I shows the caching results on the three most common advertisers seen in our workload.[3] There are dramatic differences in the fraction of transactions that resulted in cache hits (80% for 2MDN) and the fraction of transactions labeled uncacheable by others (91% for DoubleClick). The latter approach could degrade user experience (and thus perception) of on-line advertising.

We distinguish between two types of Content Distribution Networks (CDNs). Some large providers have deployed their own infrastructure to deliver static content. Table I includes three examples. These brand-specific CDNs provide mostly static images, resulting in rather high hit rates in the enterprise's caching infrastructure. The second type of CDNs are brand-neutral. These are third-party companies that deliver content for any provider for a fee. The only noticeable difference seen in Table I for this group is they provide slightly more uncacheable content than do the brand-specific CDNs.

C. Network Throughput

Next, we consider the throughput achieved on cache misses. To calculate the throughput, we consider only the largest 1% of transfers (i.e., cache misses for objects larger than 10^5 bytes). For these connections, we divide the total size by the transfer duration to determine the average throughput for the transfer.

Figure 4(a) shows the results for large transfers from MSN servers. The results clearly differ across the different locations. At the US_{PT} proxy, 90% of the large transfers saw an average throughput over 1 Mb/s (10^6 bps), while the IN proxy only

<hr>

[3]We identified advertisers using the labels from [14].

TABLE I
CACHE PERFORMANCE BY PROVIDER.

Class	Brand	Cache Hits (%)	Cache Misses (%)	No-Cache Misses (%)
Providers	Google	21	21	58
	MSN	45	2	53
	Facebook	12	3	85
	BBC	50	15	35
Advertisers	DoubleClick	7	2	91
	2MDN	80	19	1
	ATDMT	19	18	63
Brand-Specific CDNs	fbcdn	64	36	0
	yimg	89	10	1
	ytimg	78	25	0
Brand-Neutral CDNs	Limelight	80	10	10
	Edgesuite	49	44	7
	Akamai	73	19	8

saw 10% of large transfers at or above this rate. Figure 4(b) shows similar results for large transfers from Google's servers. Figure 4(b) also shows that roughly 5% of large transfers at the US_{CT} and US_{ET} proxies exceeded 10 Mb/s (10^7 bps). These estimates represent lower bounds, as the proxy infrastructure may limit the speed at which a provider serves the objects.

Figure 4(c) provides the average throughput results observed for YouTube servers. A notable observation from this graph is that the distributions look remarkably similar across geographies. Since YouTube is streaming video over HTTP, this seems to be an artifact of their delivery configuration, rather than a measurement artifact. These may be one type of the rate-limited flows discussed by Gerber *et al.* [8].

The median throughputs for the YouTube transfers shown in Figure 4(c) range from 436 Kb/s to 692 Kb/s. These are all above the median bit rate of YouTube content (328 Kb/s) reported by Gill *et al.* in 2007 [9].

V. INFRASTRUCTURE CHARACTERIZATION

This section examines infrastructure differences by geography, and investigates the underlying causes.

A. Server Location and Replication

One topic of interest is understanding where popular content providers locate their servers (or data centers). A common
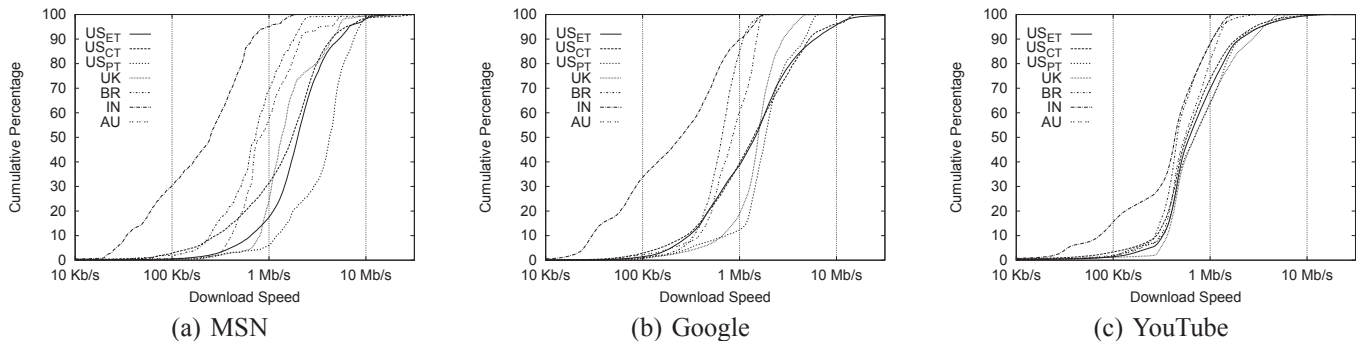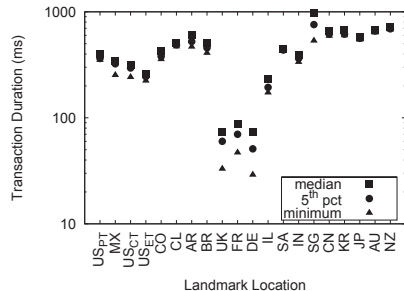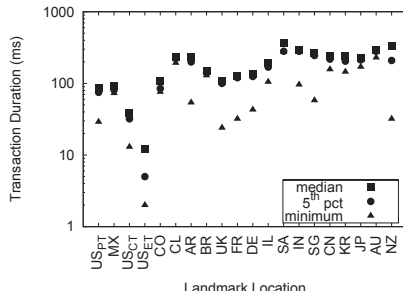
(a) MSN   (b) Google   (c) YouTube

Fig. 4. Average throughput of large transfers from MSN, Google, and YouTube.

(a) RapidShare   (b) Google

Fig. 5. Small miss transfer durations from landmarks to RapidShare and Google.
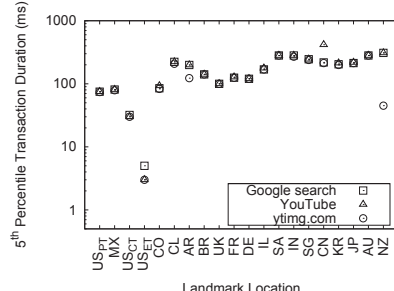
Fig. 6. Small miss transfer durations from landmarks to Google services.

method to assess this is to perform active measurements from a set of *landmarks*, i.e., a set of computers at known locations. For example, this approach was used by Antoniades *et al.* [1] to show that the *RapidShare* "one click" file sharing service deploys all of its server infrastructure in Germany.

In this paper we are concerned with the actual infrastructure serving our users, and how well it can be identified using passively collected proxy logs. In contrast to Antoniades *et al.*'s active measurements, our logs do not contain the round trip times (RTTs) from a proxy to an external server, so we must improvise. Similar to Section IV-B, we use the transfer duration of small cache misses (less than 1,460 bytes).

To demonstrate that the information obtained by Antoni-ades *et al.* [1] can be gleaned from passive proxy logs, we also study RapidShare. Figure 5(a) shows the minimum, 5th percentile and median durations for small no-cache misses to RapidShare, from 20 landmark locations (proxies) across the globe.[4] Each landmark location is denoted by its country code; the US landmarks are further divided by their timezone. Fig-ure 5(a) shows that the shortest durations occur in Germany, matching the conclusion in [1].

Figure 5(a) reveals that the same conclusion would be reached by considering other values, such as the 5th percentile or the median. Using a value other than the minimum may be desired, as it could be more robust to data quality issues.

Figure 5(b) shows the minimum transfer durations for small cache misses from the 20 landmarks to Google. It reveals that some of the minimum transfer times are quite low;

e.g., 2ms for the US_{ET} landmark. Also, there appear to be servers in multiple locations (i.e., decentralized placement, unlike RapidShare). Lastly, the transfer times for Google are generally lower than for RapidShare. This is to be expected, due to the use of decentralized infrastructure. While such observations for Google were expected, the results show that our passive technique is capable of revealing them.

In this case study, our ability to pinpoint the location of a provider's servers is limited by the number of landmark locations available to us. However, with a more diverse set of landmark locations, the crowd-based monitoring technique should be able to provide more precise location information.

B. Mapping Services to Providers

In Section V-A, we showed how the transfer durations from a set of landmarks could locate a provider's infrastructure. These measurements could also serve another purpose: map-ping a service to its infrastructure provider (and possibly its owner). These measurements form a *delay vector* [22], which is essentially a fingerprint of the infrastructure provider.

As an example of how this works, we consider three *services*: Google search, YouTube, and ytimg.com, which serves images for YouTube. As these services are all provided by Google, we use them to demonstrate the potential of the approach. Figure 6 reveals how nearly identical "fingerprints" can be extracted for these services, even though Google search has several orders of magnitude more transactions. In other words, even with relatively few transactions at a landmark, a reasonable fingerprint can still be obtained.

Given the large number of unique "services" in Web traffic, a systematic method for grouping services into a smaller num-

---

[4]No-cache misses were used since there were few cache misses for RapidShare. Also, please note the y-axis in Figure 5 is in log-scale.

ber of clusters (e.g., by infrastructure provider) is of interest. Our investigation suggests delay vectors may be applicable for this purpose. Using a more diverse "crowd" would increase the number of landmarks, and improve the uniqueness of provider fingerprints. This type of fingerprints could also be used to classify the infrastructure used by different services and providers. Tracking how these fingerprints (or the number of fingerprints of each type) change with time could help characterize the evolution of the World Wide Web.

VI. Usage Access Patterns

Lastly, we consider the global popularity of external hosts visited by users. Host names (e.g., `www.google.com`) in HTTP headers often refer to a service rather than a specific (physical) server, so this allows us to estimate the global popularity of services. For each unique Host name in the aggregate log, we determined how many transactions and how much data were transferred from it to each continent. In total, 5.4 million unique Host names were seen. Figure 7 shows that 83% of these were accessed from only one continent (note the y-axis is in log-scale). The popularity of "services" decreases roughly exponentially as the number of continents increases. Less than 0.4% of the unique Host names were observed in the activity of all continents. However, this small set of "services" accounted for 56.4% of all transactions and 47.3% of response data. The remaining half of the activity is split roughly equally among "services" that were seen on one to five continents.
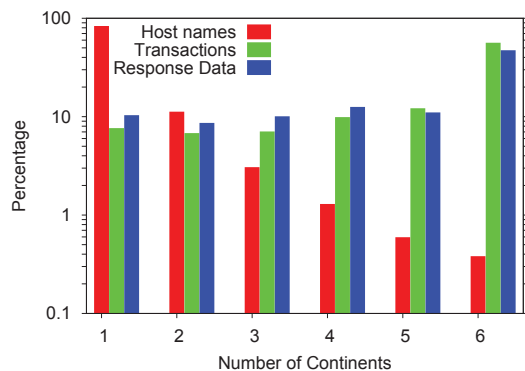

Fig. 7. Hosts across geographies.

VII. Conclusions

This paper examined the use of passive crowd-based monitoring of the World Wide Web infrastructure and its performance. The proxy logs of a large, global enterprise were used to demonstrate how the approach allows us to reveal geographic differences in user experience, distinguish content providers by their quality of service, determine differences in network throughputs, and identify services that were provided by a common (or similar) infrastructure. In contrast to active measurements, this passive approach allows us to monitor and provide insights on the infrastructure used to support *actual* Web usage, without placing any additional load on the measured infrastructure or its users.

There are numerous ways to enhance our work. For example, applying crowd-based monitoring to a more diverse audience (possibly via multiple edge networks) could find additional insights on the World Wide Web. Alternatively, examining passively collected traces that cover a longer period of time would enable a study of the evolution of the Web. These are left for future work.

References

[1] D. Antoniades, E. Markatos, and C. Dovrolis. One-click hosting services: A file-sharing hideout. In *Proc. IMC*, Chicago, IL, 2009.

[2] M. Arlitt and C. Williamson. Internet web servers: Workload characterization and performance implications. *IEEE/ACM Trans. on Networking*, 5(5):631–645, 1997.

[3] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker. Web caching and Zipf-like distributions: Evidence and implications. In *Proc. IEEE INFOCOM*, New York, NY, 1999.

[4] D. R. Choffnes, F. E. Bustamante, and Z. Ge. Crowdsourcing service-level network event monitoring. In *Proc. ACM SIGCOMM*, New Delhi, India, 2010.

[5] B. Eriksson, P. Barford, and R. Nowak. Network discovery from passive measurements. In *Proc. ACM SIGCOMM*, Seattle, WA, August 2008.

[6] P. Francis, S. Jamin, C. Jin, Y. Jin, D. Raz, Y. Shavitt, and L. Zhang. "IDMaps: A global internet host distance estimation service". *IEEE/ACM Trans. on Networking*, 9:525–540, 2001.

[7] Gartner. Gartner estimates ICT industry accounts for 2 percent of global CO2 emissions. Technical report, Gartner, 2007.

[8] A. Gerber, J. Pang, O. Spatscheck, and S. Venkataraman. Speed testing without speed tests: Estimating achievable download speed from passive measurements. In *Proc. IMC*, Melbourne, Australia, 2010.

[9] P. Gill, M. Arlitt, Z. Li, and A. Mahanti. Youtube traffic characterization: A view from the edge. In *Proc. IMC*, San Diego, CA, 2007.

[10] P. Gill, M. Arlitt, Z. Li, and A. Mahanti. The flattening internet topology: Natural evolution, unsightly barnacles or contrived collapse? In *Proc. PAM*, Cleveland, OH, 2008.

[11] J. Guyton and M. Schwartz. Locating nearby copies of replicated internet servers. In *Proc. ACM SIGCOMM*, Cambridge, MA, 1995.

[12] E. Katz-Bassett, H. Madhyastha, V. Adhikari, C. Scott, J. Sherry, P. van Weep, T. Anderson, and A. Krishnamurthy. Reverse traceroute. In *Proc. NSDI*, San Jose, CA, April 2010.

[13] T. Kelly and J. Mogul. Aliasing on the World Wide Web: Prevalence and performance implications. In *Proc. WWW*, Honolulu, HI, 2002.

[14] B. Krishnamurthy and C. Wills. Privacy diffusion on the Web: A longitudinal perspective. In *Proc. WWW*, Madrid, Spain, 2009.

[15] R. Krishnan, H. Madhyastha, S. Srinivasan, S. Jain, A. Krishnamurthy, T. Anderson, and J. Gao. Moving beyond end-to-end path information to optimize cdn performance. In *Proc. IMC*, Chicago, IL, 2009.

[16] T. Kroeger, D. Long, and J. Mogul. Exploring the bounds of web latency reduction from caching and prefetching. In *Proc. USITS*, Monterey, CA, 1997.

[17] C. Labovitz, S. Iekel-Johnson, D. McPherson, J. Oberheide, and F. Jahanian. Internet inter-domain routing. In *Proc. ACM SIGCOMM*, New Delhi, India, 2010.

[18] A. Mahanti, C. Williamson, and D. Eager. Traffic analysis of a Web proxy caching hierarchy. *IEEE Network*, 14(3):16–23, 2000.

[19] C. Maltzahn and K. Richardson. On bandwidth smoothing. In *Proc. WCW*, San Diego, CA, 1999.

[20] R. Miller. Google super-sizing its data centers? http://www.datacenterknowledge.com/archives/2007/04/23/google-super-sizing-its-data-centers/, 2007.

[21] V. Padmanabhan, L. Qiu, and H. Wang. "server-based inference of internet link lossiness". In *Proc. IEEE INFOCOM*, San Francisco, CA, 2003.

[22] V. Padmanabhan and L. Subramanian. An investigation of geographic mapping techniques for internet hosts. In *Proc. ACM SIGCOMM*, San Diego, CA, 2001.

[23] M. Stemm, R. Katz, and S. Seshan. A network measurement architecture for adaptive applications. In *Proc. IEEE INOFOCOM*, Tel Aviv, Israel, 2000.

[24] A. Wolman, G. Voelker, N. Sharma, N. Cardwell, A. Karlin, and H. Levy. On the scale and performance of cooperative Web proxy caching. In *Proc. ACM SOSP*, Kiawah Island, SC, 1999.