Characterizing Organizational Use of Web-based Services: Methodology, Challenges, Observations, and Insights

PHILLIPA GILL
University of Toronto, Toronto, ON, Canada
MARTIN ARLITT
HP Labs, Palo Alto, CA, USA
NIKLAS CARLSSON
Linköping University, Linköping, Sweden
ANIRBAN MAHANTI
NICTA, Eveleigh, NSW, Australia
CAREY WILLIAMSON
University of Calgary, Calgary, AB, Canada

Today's Web provides many different functionalities, including communication, entertainment, social networking, and information retrieval. In this paper, we analyze traces of HTTP activity from a large enterprise and from a large university to identify and characterize Web-based service usage. Our work provides an initial methodology for the analysis of Web-based services. While it is non-trivial to identify the classes, instances, and providers for each transaction, our results show that most of the traffic comes from a small subset of providers, which can be classified manually. Furthermore, we assess both qualitatively and quantitatively how the Web has evolved over the past decade, and discuss the implications of these changes.

Categories and Subject Descriptors: C.2.0 [**Computer-Communications Networks**]: General

General Terms: Measurement

Additional Key Words and Phrases: Workload characterization, Web-based services, Organizational use

1. INTRODUCTION

The World Wide Web was initially created as a means for conveniently sharing information among distributed research teams [Berners-Lee et al. 1994]. Over the past 20 years, the Web has been broadly adopted, due to its transparency and ease of use. Today, in addition to information sharing, the Web supports a myriad of functionalities and "services" (i.e., software functionalities), such as email, instant messaging, file transfer, multimedia streaming, electronic commerce, social networking, advertising, and online document processing.

Emerging trends such as cloud computing and software-as-a-service may result in even more business-oriented functions offered as services. These may be offered as "traditional" Web services (i.e., machine-to-machine interaction[1]), or as "Web-based" services (e.g., a person accesses the service via a Web browser). Many organizations are exploring how Web-based services could reduce the cost of providing important business functionalities, allow the organization to adapt nimbly to changes, or seize new business opportunities (e.g., it may be simpler to use existing Web-based services than to develop new services internally).

Information on Web-based service usage is valuable to several different types of individuals. Researchers may use such information to build workload models, or to learn about open problems. Network and system administrators can use the information to proactively plan IT infrastructure upgrades. Business managers may need information for risk assessment. We analyze current Web workloads of an enterprise and a university as a first step towards fulfilling these needs.

Our paper makes three primary contributions:

—We examine an initial methodology for analyzing Web-based services and their usage within organizations, and assess its strengths and weaknesses. Our analysis allows us to determine which services are dominant, and who provides them. As part of this investigation, we discover and identify many gaps that exist in composing useful information in a scalable, automated and verifiable manner.

—We characterize today's Web-based service usage, providing a comparison point for future studies on the evolution of the Web. Our results provide insights on what popular functionalities the Web provides to organizations today. Our results are relatively consistent between our traces collected from an academic and an enterprise environment.

—We study how Web workloads have evolved over the past decade. As previous studies have not considered service usage, we first re-examine service usage as observed in a historical trace. We then examine *traditional* characteristics and see how these have been affected by the evolution of the Web (e.g., the introduction of "Web 2.0" services). For this comparison, we leverage earlier Web characterization studies [Breslau et al. 1999; Crovella and Bestavros 1997; Cunha et al. 1995; Duska et al. 1997; Glassman 1994; Mahanti et al. 2000; Wolman et al. 1999]. In particular, we show that while the Web has undergone significant transformation in the scope of services and service providers, many underlying object access properties have not fundamentally changed.

---

[1]http://www.w3.org/TR/ws-gloss

The remainder of the paper is organized as follows. Section 2 introduces related work. Section 3 describes our data sets. Section 4 explores relevant new characteristics of Web traffic. It also examines techniques to identify service instances and providers, and to classify the functionality of popular Web-based services. Section 5 compares current access characteristics to those observed a decade ago, and discusses similarities and differences. Section 6 concludes the paper with a summary of our work and possible future directions.

2. RELATED WORK

During the first decade of its existence, the Web's primary function was information sharing. Characterization of organizational Web usage at that time focused primarily on objects and their access characteristics [Cunha et al. 1995; Duska et al. 1997; Kelly and Mogul 2002; Mahanti et al. 2000; Wolman et al. 1999]. These studies have influenced the development of technologies such as Web caching, prefetching, and Content Distribution Networks (CDNs) to improve the user experience (i.e., by reducing retrieval latency) and reduce distribution costs (e.g., by eliminating redundant object transfers). We use these earlier studies to evaluate how the Web has evolved over the past decade.

Today, the Web supports a multitude of services, and some recent studies have begun to examine this shift. For example, a recent poster paper by Li *et al.* [2008] considers "What kinds of purposes is HTTP used for?". Our work extends well beyond theirs. In particular, we examine two different data sets representing larger user populations and longer periods of time, provide more detailed results, and elaborate on the challenges of identifying the functionality of a service. Schneider *et al.* [2008] consider characteristics of Web-based services of a specific type (e.g., AJAX). Krishnamurthy and Wills examine changes in the distribution of Web content, and its effects on performance [Krishnamurthy and Wills 2006a] and privacy [Krishnamurthy and Wills 2006b; 2009]. Our work focuses on identifying Web-based services, who provide these services, and who consumes them. Our work is complementary to these studies.

Other aspects of the Web have also been studied. Fetterly *et al.* [2003] studied the evolution of Web pages over time. Baeza-Yates *et al.* [2007] examined Web page characteristics of various national domains, looking at topics such as cultural differences. These studies used data from crawling Web sites, which does not provide insights into how an organization uses or is dependent on Web-based services. Bent *et al.* [2006] measured usage of multiple Web sites hosted within a single ISP. This approach gives the service provider insights about organizations that use their services, which is a different perspective than the one in which we are interested. Williams *et al.* [2005] examined how the workloads of several Web sites had changed over a ten-year period. While our study shares the longitudinal perspective, we consider proxy rather than ("Web 1.0") server workloads.

The main reason for the lack of recent studies of organizational use of the Web is the difficulty in obtaining data. Trestian *et al.* [2008] use search engines as a means of circumventing this problem. While intriguing, it does not provide the same information as is available in actual usage traces.

The desire of network operators to identify Peer-to-Peer (P2P) traffic on their

networks motivated research on traffic classification (e.g., [Ma et al. 2006]). Such techniques tend to group applications by network-level similarities such as the application protocol (e.g., HTTP) they use, rather than by the functionality they provide. Cormode and Krishnamurthy [Cormode and Krishnamurthy 2008] identify tangible differences between "Web 1.0" and "Web 2.0" sites. Their technical comparison of the two considers functional aspects. We focus on the functionality of services, rather than labeling them as "Web 1.0" or "Web 2.0". Similar to us, they rely on manual labeling.

3. DATA SETS

This section describes our data collection methodology and provides a statistical summary of our two new data sets and one historical data set.

3.1 Enterprise Data Set

Our enterprise data set comprises the logs from 28 caching proxies (belonging to a single enterprise) that are geographically distributed around North America. The enterprise has approximately 60,000 employees in North America. The HTTP traffic (regardless of port) of these employees, destined to servers on the Internet, must use one of these 28 proxies. These logs contain a record on each transaction that either traversed a proxy to reach a server on the Internet, or that was served from cache. Each record contains fields such as the protocol, method, user agent, status code, bytes transferred, and cache action (e.g., hit or miss). The logs do not contain "personal" information such as client IP addresses or cookies.

3.2 University Data Set

Our second data set is a trace of HTTP transactions at the University of Calgary, which has approximately 35,000 faculty, staff and students. The traces were collected during the Fall 2008 semester, for the same week as the enterprise traces.

A `Bro` [Bro Intrusion Detection System 2008] script was developed to summarize HTTP transactions on port 80 on the university's Internet link in real time. This methodology is advantageous in that it limits the amount of data stored (compared to full packet traces) and offers better protection of user privacy, as sensitive information like client IP addresses or cookies is not written to disk. The `Bro` script captures application-layer statistics (e.g., HTTP method, status code, `Host:` header, etc.), and transport-layer statistics (e.g., transfer duration, bytes transferred, etc.).

3.3 Historical Data Set

To facilitate a comparison of how Web-based services have changed over time, we obtained a historical proxy data set. This data set represents the Web usage of residential cable modem users in 1997. A detailed characterization of this data set is available in [Arlitt et al. 1999].

3.4 Overview of Traces

Table I provides summary statistics for the two new data sets and the one historical data set. Each new data set is one week long, and spans the same 168 hour period. The week-long duration facilitates direct comparison between these two data sets,

Table I.　Summary of data sets used.

Property	Enterprise	University	Historical
Start Date	Sep. 14, 2008	Sep. 14, 2008	Jan. 1, 1997
Start Time	0:00:00 GMT	0:00:00 GMT	01:30:05 EST
End Date	Sep. 20, 2008	Sep. 20, 2008	Jun. 1, 1997
End Time	23:59:59 GMT	23:59:59 GMT	01:30:05 EDT
Total Transactions	1,442,848,763	121,686,977	117,652,652
Data Uploaded	1.4 TB	0.22 TB	0.20 TB
Data Downloaded	22.6 TB	3.5 TB	1.3 TB
Unique HTTP Objects	232,780,505	25,986,309	25,276,253
Unique Servers	1,685,388	732,287	318,894
Unique User Agents	645,656	19,445	8,622

as well as a comparison with those collected by Wolman *et al.* in 1999 [Wolman et al. 1999].[2] The historical data set spans a five month period in 1997.

Comparing our two new data sets, we see that the enterprise data set contains an order of magnitude more transactions (1.4 billion) than the university data set (122 million), and an order of magnitude more unique URLs (233 million versus 26 million). The enterprise data set includes about 6.5 times as much downloaded data, from twice as many unique servers. While these differences are in part due to roughly twice as many users, it appears that the Web is used more extensively at the enterprise than at the university.

The historical data set has a similar number of transactions and unique HTTP objects requested as the university data set. However, the historical data set spans a much longer period of time (five months rather than one week), and downloaded about 1/3 as much data from roughly half as many servers.

Our data sets indicate that there has been significant growth in the number of Web servers used to deliver content to users over the last decade. For example, Wolman *et al.* [1999] observed 360,586 distinct servers in their enterprise trace, and 244,211 unique servers in their academic trace. In contrast, we observed 1,685,388 unique servers in our enterprise trace and 732,287 unique servers (by `Host:` name) in our university trace. While direct comparison with previous studies [Cunha et al. 1995; Mahanti et al. 2000; Wolman et al. 1999; Kelly and Mogul 2002] is difficult, this growth is consistent with observations made by Krishnamurthy and Wills [Krishnamurthy and Wills 2006a; 2006b; 2009].

Because obtaining concurrent traces from multiple large organizations is onerous, we limited our data collection to a one-week period. This is sufficiently long to gain insights into interesting properties of current Web use, as well as to understand what the challenges are for examining the properties on an ongoing basis.

4. EXAMINING WEB-BASED SERVICE USAGE

Over the past decade, Web-based services have played an increasingly important role in the day-to-day operations of many organizations. We expect this trend to

---

[2]Wolman *et al.* [1999] examined week-long traces of activity from the University of Washington (23,000 users) and a different enterprise, also with 60,000 users.

increase in the future, and believe that *information rich* data sources such as proxy logs can assist organizations with a variety of needs, from operational aspects such as security issues, to business decisions such as quantifying costs associated with specific services or providers.

In this section we explore how to identify unique Web-based services, who provides the services, what classes of services are used, and who (or what) uses them. We also describe the challenges that exist for each of these tasks.

4.1   Definitions

A *Web-based service* offers specific functionality (e.g., email, search) to end-users via the HTTP protocol. The service is made available by a *service provider*, and is *consumed* by a client such as a Web browser. There are different *service classes*, which are distinguished from other classes by the offered functionality. A service provider offers an *instance* of a service belonging to a given class. Other service providers may offer their own (competing) service instances. For example, Google, Inc. is a service provider. In the email service class, they offer the `Gmail` service instance. Microsoft is another service provider, which provides the `Hotmail` service instance in the email service class.

4.2   Identifying Service Instances

A necessary first step towards characterizing Web-based services is to identify the unique service instances in our traces. As there is no established method to do this, we leverage data available in the `Host:` header of each transaction. We use this header to identify unique host names, domains, brands, and service instances.

Unique Hosts: The `Host:` header contains the name of the host contacted to fulfill the given request. This value is typically in domain name format (e.g., `www.google.com`) rather than IP address format, as it provides an additional layer of abstraction that the service provider can use for purposes such as load balancing. Table II indicates that the enterprise data set contains transactions with almost 1.7 million unique host names, the university data set involved about 732,000 unique host names, and the historical data set contains 319,000 unique host names. With the above host name specification, there can be at most one unique service instance per host name, and the number of host names is an upper bound on the number of unique service instances in each trace.

Unique domains: We divide the domain names into three parts: functional labels (e.g., `www`), the brand (e.g., `google`), and the Top Level Domain (TLD; e.g., `com`). We also consider secondary domains, such as state codes for country TLDs (e.g., `tx` for `us` TLD). A domain name may contain zero or more functional labels, and will contain one brand and at least one TLD.[3] An example with more than one TLD is `google.co.uk`.

We consider a unique "domain" to be the combination of the brand and the TLD (and any secondary-level domains). For example, `google.com` is the "domain" for `www.google.com`. Table II indicates that by ignoring the functional terms, we reduce the number of unique items substantially (e.g., from 1.7 million unique hosts to 904,000 unique domains for the enterprise data set). The primary cause of this

---

[3]Domain hacks such as `del.icio.us` are exceptions [Wikipedia Article 2009].

Table II. Breakdown of Host values.

Property	Enterprise	University	Historical
Unique Host Names	1,685,388	732,287	318,894
Unique Domains	904,129	205,263	205,355
Unique Brands	831,243	194,348	185,031
Unique Service Instances	932,620	229,299	203,349

Table III. Functional terms in Host values.

Number of Terms	Enterprise	University	Historical
	% of host names	% of host names	% of host names
0	10.6	5.9	9.2
1	69.3	39.1	73.3
2	18.4	54.2	12.5
> 2	1.7	0.8	4.9

is the sophisticated and extensive content distribution infrastructures, particularly for popular Web-based services. For example, the domain with (by far) the most unique host names in both data sets was `facebook.com`, with over 125,000 distinct host names in the enterprise trace, and more than 340,000 distinct host names in the university trace. The top 10 domains in each trace accounted for 15.8% and 50.8% of the unique `Host` values, for the enterprise and university data sets, respectively. In contrast, in the historical data set from 1997 the top 10 domains only account for 2.2% of the unique `Host` values.

Unique brands: Another method of consolidation is to consider only the brand, and ignore the TLD term(s) and any functional terms. This approach amalgamates regional variations of the same service instances (e.g., `google.com`, `google.de` and `google.co.uk`), though at the risk of incorrectly merging some domains (e.g., if brand.TLD1 and brand.TLD2 are owned by different service providers). Table II shows that another 5–10% reduction occurs by consolidating domains to brands.

Considering only the domain or the brand portion of a `Host` value underestimates the number of unique service instances (e.g., `mail.google.com` and `www.google.com` are indistinguishable). To better estimate the number of service instances, we next attempt to extract useful information from the functional terms.

Unique service instances: Our data sets contain a wide range of functional terms that complicate the analysis of service instances. We first consider the *number* of functional terms in each `Host` value. Table III indicates that 5–11% of `Host` values have no functional terms. In the enterprise data set, 70% of `Host` values had a single functional term; two-thirds of these were the term `www`. In the university data set, the term `www` accounts for the majority of the single functional `Host` values as well. However, the university data set has a much larger percentage of `Host` values with two functional terms. This skew is created by `facebook.com` servers, which account for almost half of all unique `Host` values in that data set. The majority of the observed `facebook.com` servers had two functional terms of the form `X.channelY`, where `X` and `Y` are numeric identifiers.

Table IV. Composition of terms.

Type	Enterprise % of Terms	University % of Terms	Historical % of Terms
alphabetic-only	56.7	18.0	85.5
numeric-only	13.4	37.9	0.1
alphanumeric	23.1	41.6	10.3
others	6.9	2.4	4.0

We next examined the *syntactic* composition of functional terms. Table IV shows that over half of the functional terms in the enterprise data set are entirely composed of alphabetic characters. About 23% of the remaining terms have alphabetic components that could be extracted. The remaining terms (primarily numeric-only terms) are likely labels used to systematically manage servers in large-scale data centers; such terms are unlikely to help identify unique service instances. The fraction of alphanumeric terms in the university data set are skewed by the large number of `facebook.com` host names. The historical data set has mostly alphabetic-only terms (94% of which are `www`). This suggests that there has been a significant change in the composition of terms, which may be largely due to the growth of the Web. For example, with the increased scale of the infrastructure underlying the Web, organizations may have found it easier to use names such as "mail1.orgname.com", "mail2.orgname.com", ..., "mail$N$.orgname.com", rather than coming up with distinct human readable names for each additional server. The higher number of numeric-only terms observed today suggests that more and more `Host` names are not intended for direct use by Web users.

For further insights on the *semantic* functionality of each server, we compare each alphabetic functional term against an English word list [Atkinson 2008].[4] As an initial approximation of the number of service instances, we consider a match for an English word (for the first functional term) to indicate that we have identified a unique service instance. Using this technique, we determined there were about 933,000 unique service instances in the enterprise data set, 229,000 in the university data set, and 203,000 in the historical data set (see Table II). This represents a slight increase over the number of unique domains, and 10–17% increase over the number of unique brands. The increase is relatively small for several reasons: some `Host` values have no functional terms; many `Host` values have functional terms that offer no insight on the service (e.g., `www`); and many domains or brands were seldom observed, and thus considering the functional terms did not result in the identification of additional service instances.

4.3 Concentration of Activity

Although the domain, brand, and service instance approaches significantly reduce the number of items to consider (compared to the number of unique `Host` names or URLs), they are still too numerous to label, inspect, or investigate manually. Fortunately, the highly non-uniform popularity of Web-based services allows much of the activity to be understood by focusing on only a few of the most popular

---

[4]A possible improvement on this approach would be to match only against domain-specific terms.

items. For example, in the enterprise data set, 511 servers accounted for 50% of the transactions, and 770 accounted for 50% of the response bytes. These represent less than 0.1% of the total observed servers in that data set.

As we consolidate from servers to brands, or service instances, the concentration increases. In the enterprise data set, the 121 most popular brands received 50% of all transactions. This concentration property means that a reasonable understanding of the traffic can be obtained by examining only a few popular services.

The data traffic is even more concentrated than the transaction traffic, due to the heavy-tailed transfer size distribution. The 36 most popular brands in the enterprise data set account for half of the downloaded data (only 15 brands in the university data set). This is a 20-fold reduction in the number of entities to observe, if tracking was done based on the entire domain name. This is an important property. It means that a network provider could mirror content locally, and reduce the bandwidth demands on their external network connection to the Internet. To do this effectively, network operators will want to focus on the companies responsible for the most traffic, rather than on individual servers. For example, an organization with a substantial fraction of traffic for Google's services could consider a peering arrangement via the Google Global Cache service.[5] Google would then install servers on the company's internal network, to serve HTTP requests internally.

Figure 1(a) compares the concentration (based on transactions) between the enterprise, university and historical data sets. Although there are slight differences between the results for the two new data sets, there are many similarities, due (not surprisingly) to the "global" popularity of certain brands. For example, seven of the ten most popular brands by transaction count or data downloaded were the same between the two data sets (e.g., `google` and `facebook`). Figure 1(a) also reveals that user demand has become more concentrated between the collection of the historical and new data sets, despite the fact that there are now a lot more services to choose from. Figure 1(b) shows that the concentration property is even more pronounced for data downloads.
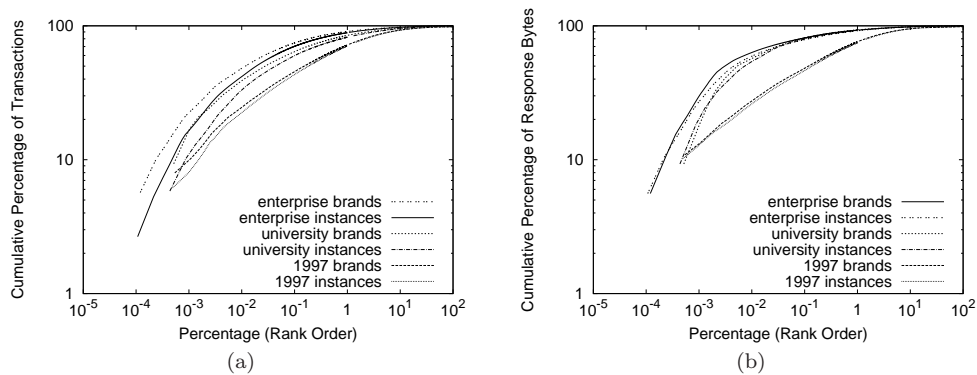

Fig. 1. Concentration properties comparison: (a) transactions; (b) response bytes.

---

[5]`http://ggcadmin.google.com/ggc`

Table V.   Service provider information.

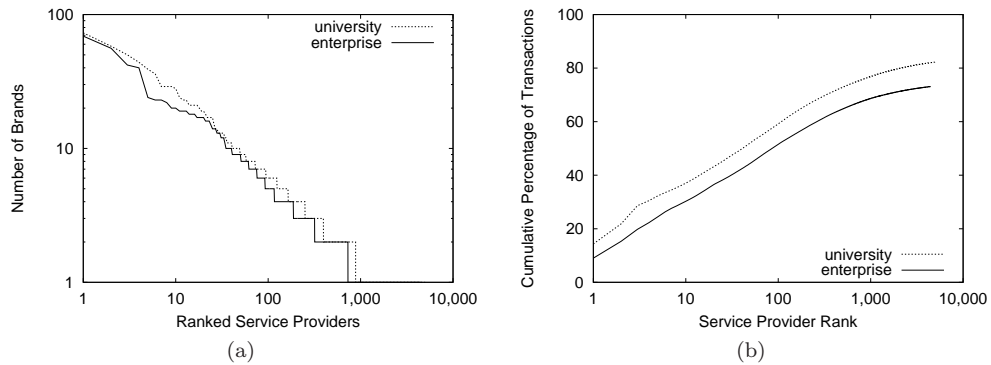|	Enterprise	University
initial brands	10,000	10,000
brands with DNS & RIR data	6,518	7,580
estimated service providers	4,458	4,995
provide own name service	1,334	1,468

4.4 Identifying Service Providers

Fig. 2.   Concentration properties in Service Providers: (a) brands; (b) transactions.

We now consider who provides each service. While determining the "brands" and "service instances" is important for understanding the types of services used within an organization, it is not sufficient for establishing which service providers are involved, as many service providers have multiple brands in their portfolio.

Identification methodology: To systematically identify unique service providers, we use data from three sources. First, we extract the brands, corresponding domains, and a `Host` name from the original traces, as previously described. Second, we query a Regional Internet Registry (RIR) such as LACNIC[6] for registration information on each domain. Third, we perform a DNS query to determine the authoritative name servers for each domain. If no information is available, we issue a query for the specific host name.

We analyze each transaction in three steps. First, from the DNS query results, we determine the brand for each of the name servers affiliated with the brand of a Web-based service. We also identify which brands share the same name servers. From the RIR data, we identify the Organization Identifier (OrgID) associated with each brand. Second, we create service provider groups, composed of brands that have common name server brands and common OrgIDs, in an attempt to group together brands owned by a single entity. This step provides an initial hint as to whether the infrastructure used by the Web-based service is operated by the same company, or provided by a third party (e.g., as an infrastructure service), and shared by many

---

[6]http://www.lacnic.org/

different service providers. Our third step extracts other information on the service providers, such as how long they have been in the registry.

As an example, consider the two `Host` names `www.youtube.com` and `www.google.com`, which have the brands `youtube` and `google`. First, we use `dig` to obtain the name servers for the `Host` names. In this case, both `Hosts` returned the same list (`ns1.google.com, ns2.google.com, ns3.google.com, ns4.google.com`). Also using `dig`, the IP addresses resolved to `74.125.113.100` and `66.249.91.104`, respectively.[7] Using `whois`, the OrgID for these IP addresses resolved to GOGL. Since both the OrgID and the brand of the name servers are the same for these two "brands" (`youtube` and `google`), we determine a single service provider administers them. Furthermore, since the brand of the name servers (`google`) is the same as one of the member brands, we say `google` is the dominant brand (or potentially the parent company).

Service provider identification results: As an initial experiment, we applied our methodology on the 10,000 most popular brands (by transaction count) in the two new data sets. As shown in Table V, we were able to obtain relevant DNS and RIR data on 65–75% of these. Using this subset of the brands, we identified approximately 4,500 service providers in the enterprise data set, and roughly 5,000 in the university data set. About 30% of these use their own DNS servers to support the service. For example, we found 24 (service) brands that mapped to Google, Inc.'s OrgID (GOGL) and used Google to provide DNS service.

In contrast to service providers such as Google, which maintains its own DNS servers for its many brands and services, we observed that many organizations rely on "third-party" DNS providers, which serve brands and services registered to other organizations. Close to 71% of the identified service providers use third-party DNS providers. Among these service providers, roughly 24% used one of the ten most popular DNS providers. The most prevalent DNS providers observed were UltraDNS and Domain Control, which were used by about 5% and 3% of the service providers using third party DNS, respectively. Unlike the high concentration of brands to service providers, the mapping to DNS providers is much more balanced. This flatter distribution may be an effect of DNS services being distributed among many ISPs.

Figure 2(a) shows the number of brands associated with each identified service provider. A Zipf-like distribution is observed, indicating that some service providers offer many more "brands" than do other providers. Figure 2(b) shows that a small set of service providers receive a significant proportion of the transactions. The top provider in both data sets is Google, Inc., handling about 9% of all transactions. The top 10 service providers account for 30% of transactions in the enterprise data set, and 37% in the university data set. The skew is even greater in terms of the response data, with the top 10 service providers accounting for 35% of the response data in the enterprise data set, and 47% in the university data set. As more critical business functions are moved to the Web, an organization may wish to understand if they are inadvertently "putting all of their eggs in one basket."

Age of service providers: Figure 3 shows how long each of the identified service providers has been registered with an Internet registry. Curves are shown

---

[7]The name server list and IP addresses assigned to these hosts may change over time.

for the registration dates of the brands observed in the enterprise trace and the university trace, respectively. Fewer than 10% of the service providers are from the early years of the Web. There is a noticeable surge during the late 1990s (the "dot com era"). Since 2000, the "birth rate" of popular service providers appears to be relatively stable. One-half of the service provider organizations were registered in 2003 or later.
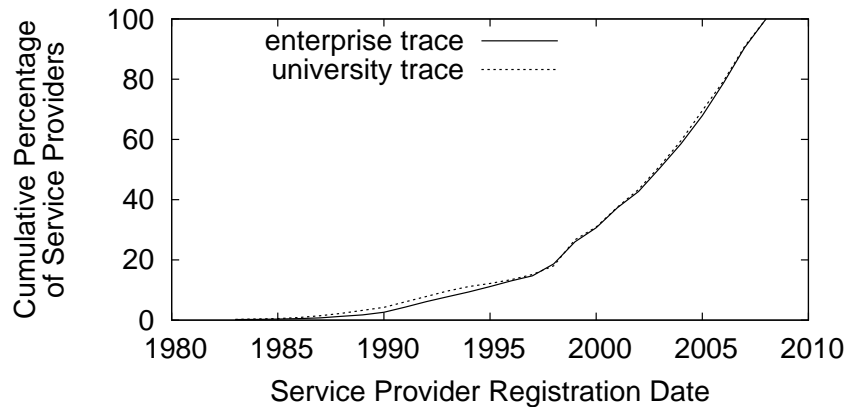

Fig. 3.   Registration of Service Providers.

4.5   Service Classes

Another question of interest is what classes of Web-based services are used by an organization? Once individual service instances have been identified, they can be grouped together under common classes to provide a high-level understanding of how Web-based services are used within an organization. This task requires service classes to be defined, and service instances to be mapped to different classes. In this section, we describe the approaches we have tried, evaluate their success, and discuss the lessons we have learned.

4.5.1   *Manual Classification.* To illustrate the classification of service instances, we leverage the observation made in Figure 1 that a relatively small number of instances are responsible for a majority of the transactions and response bytes. We created two lists of service instances: one containing the 259 instances that collectively provide > 50% of the transactions across both new data sets, and the 67 instances that generate > 50% of the response bytes (29 instances appear in both lists). We then manually labeled each instance as belonging to one of the service classes listed in Table VI. Each selected service instance was manually visited to determine its class. Labels were based on the primary functionality of the service. For example, we considered a file download from `flickr.com` to be "photo sharing", as that is the primary purpose of the service. We considered all transactions for `gmail.com` to be (Web-based) "email", even though it could be used to exchange photos (as email attachments).

Table VI.   Manual classification results of new data sets.

Service Class	Example Brand	Enterprise		University	
		% Trans.	% Bytes	% Trans.	% Bytes
content distribution	`llnwd`	28.4	20.6	18.5	22.3
information retrieval	`cnn`	27.3	7.6	20.6	1.3
advertisements	`doubleclick`	18.8	3.4	18.6	1.1
search	`bing`	6.9	3.7	11.5	3.3
email	`gmail`	5.1	2.4	6.6	0.7
social networks	`facebook`	4.3	0.7	14.1	2.9
updates	`windowsupdate`	1.9	14.9	2.1	6.3
video	`youtube`	1.0	24.8	2.3	48.3
music	`pandora`	0.8	4.4	0.3	0.1
photo sharing	`flickr`	0.5	1.4	0.5	0.9
repositories	`rapidshare`	0.0	16.1	0.0	12.9
other	`travelocity`	5.1	0.0	5.3	0.0

The results in Table VI are sorted by the percentage of transactions for the enterprise data set. The top three classes are related to Web browsing. The *information retrieval* class includes popular news sites or other information sharing sites (e.g., `cnn.com`). Many of these sites are globally popular, and thus use *content distribution* mechanisms (e.g., servers located on edge networks that are dedicated to serving static content like images) to handle the workload. Much of the content distribution is likely related to information retrieval; however, the content distribution is achieved via providers of infrastructure services (e.g., CDNs such as Limelight Networks - `llnwd.net`) or via dedicated servers (with unique names) in the server provider's own domain. The third class is *advertising*. While advertising (e.g., `doubleclick.net`) may not be a service selected directly by the user, online advertising is now an integral part of the business models of many Web-based services, and this is reflected in the percentage of transactions. The main differences between our two data sets are that (online) *social networks* (e.g., `facebook.com`) and *search* (e.g., `bing.com`) account for a larger percentage of transactions at the university.

In terms of the bytes downloaded, the largest component is the *video* service class (e.g., `youtube.com`), which accounts for 24% of the data in the enterprise trace, and 48% in the university trace. Other classes responsible for significant data transfer volume are *data repositories* (e.g., `rapidshare.com`) and *updates* (e.g., OS patches from `windowsupdate.com`). Music services like `pandora.com` (Internet radio) also account for a larger fraction of data traffic than request traffic in the enterprise data set. However, the same characteristic was not seen in the university data set, making this another difference between the data sets.

Sites that did not belong to one of the 11 service classes were included in the "other" class. For example, `travelocity.com` enables customers to make travel reservations. The "other" class accounted for approximately 5% of transactions in both new data sets, but created negligible traffic volumes.

While this experiment illustrates how the large volume of data in each trace can be transformed into meaningful information (e.g., helpful to a network operator for planning, or a business manager for tracking risk), there are challenges. It was both

difficult and time-consuming to label $O(10^2)$ service instances, while the data sets contain $O(10^6)$. A related challenge is developing a useful taxonomy; while ours is sufficient for illustrative purposes, a more formal process is needed in practice.

4.5.2 *Automated Classification.* As an initial step towards an automated solution, we apply a Web page classification technique to the problem of classifying previously unidentified service instances. Web page classification methods can use many sources of information including on-page features such as text content and `html` tags, or visual features such as images and layout information [Qi and Davison 2007]. We consider the Centroid method [Han and Karypis 2000], which uses text information taken from `html` tags to classify Web pages. As our data sets do not contain `html` tags or similar information, we actively retrieved the home pages of popular `Hosts` (as determined from our two new data sets).

The Centroid method works by taking in a labeled set of Web pages. Term vectors are then generated for each labeled page using `html` tags, and "centroid" term vectors are produced for each category, by combining the term vectors of the labeled Web pages within each category [Han and Karypis 2000]. Using these centroid vectors, unlabeled Web pages are then classified by generating their individual term vector (using `html` tags) and determining the centroid that is nearest to them. Distance is calculated using the cosine measure [Han and Karypis 2000].

Specific terms that we use in our analysis are taken from the `html keyword` and `description` meta-tags as well as the title tag. These tags are selected because they work well for classifying Web pages [Kwan and Lee 2003]. We consider two different methods for generating the term vectors; the first is to always use terms from the title tag (*title*), and the second uses terms from the title tag if no other terms are found in the keyword and description meta-tags (*selective title*).

Of the labeled Hosts from Section 4.5.1, we used the 221 `Hosts` for which we were able to obtain `html` titles or tags. We split these into two disjoint sets. Our experiments use each set once as the training set, with the remaining set used as testing data. The performance of the classifier is then averaged across these trials.

We evaluate the performance using the *F-measure* metric, which is the weighted harmonic mean of *precision* (the number of true positives divided by the sum of true and false positives) and *recall* (true positives divided by the sum of true positives and false negatives) [Manning et al. 2009]. The F-measure ranges from 0 to 1; higher values are better.

Figure 4 shows the F-measure for the different categories. The median F-measure across the categories is 0.70, with typical values between 0.60 and 0.80.

The Centroid method is promising for categories such as classified ads and e-commerce, but it does not perform well for categories with an ill-defined set of terms. For example, the terms used by CDNs are often similar to those used by instances in the updates category. Similarly, social networks are often mislabeled as instances of blog or photo sharing services.

4.6 Identifying Consumers

A third question we consider pertains to the consumers of Web-based services. Specifically, we use the `User-Agent` header to determine what generates the traffic. We divided the activity in each data set into four categories: *browsing*, *applications*,
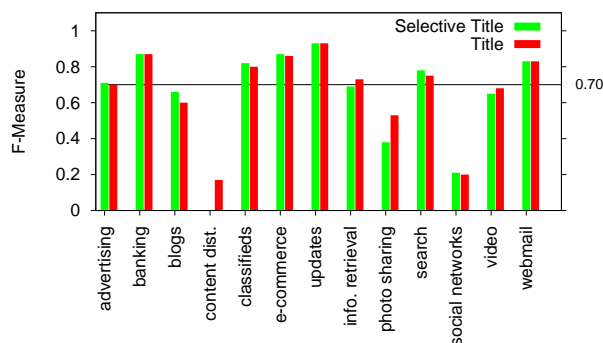
Fig. 4.   F-measure of Centroid classification method applied to Web pages.

updates, and *other*. For our data sets, the first three categories account for 99% of the observed transactions and response data.

Table VII shows the breakdown of activity in each data set, by the type of `User-Agent`. In the historical data set, browsing accounted for essentially all of the HTTP traffic. That proxy also supported FTP, which was seldom used but still accounted for over 20% of the data transferred. The results by class are quite similar between the enterprise and the university. Browsing is the largest class, accounting for over 90% of the transactions and 80% of the response data in each trace. Compared to the historical data set, browsing accounts for slightly less of the transactions, as HTTP is now being used in more ways. HTTP-enabled applications represent the next largest class, generating 5–8% of transactions and 15% of the bytes. These include "helper" applications (e.g., Adobe's Shockwave Flash player) that can be embedded in Web pages, as well as standalone applications (e.g., music or video players). The larger percentage of response bytes suggests some applications are responsible for the downloads of larger objects than might be seen in typical browsing. Updates for OS patches or revised definitions for anti-virus programs account for about 1% of transactions, and 1–3% of response bytes. We expect these are automatically generated transactions (i.e., no human involvement). Machine-initiated transactions may increase over time, as more system management tasks are automated, and as more such services are provided over the Web.

Table VII.   User-agent statistics.

Class	Enterprise		University		Historical	
	% Trans.	% Bytes	% Trans.	% Bytes	% Trans.	% Bytes
browsing	90.1	80.9	92.4	82.4	(HTTP) 99.3	87.7
application	7.9	14.4	5.1	15.2	(FTP) 0.3	12.1
updates	0.9	3.2	0.9	0.5	–	–
other	1.0	0.9	1.2	1.1	–	–

4.7   Challenges

Our work is motivated by the premise that as Web-based services become more business critical, organizations will want access to a variety of information about their own usage of such services. The previous subsections explored some possible questions of general interest; we now discuss some of the challenges that exist for systematically answering them, and list some potential solutions.

Identifying Service Instances.   In Section 4.2, we used information from the `Host` names to try and identify the distinct service instances. While we find this approach works reasonably well, it has its shortcomings. In particular, it would not properly identify service instances whose name appears in the URL (e.g., our approach would not distinguish between `www.provider.com/service1` and `www.provider.com/service2`). One solution would be a standardized naming convention.

Discerning User Actions.   Many HTTP transactions are invoked indirectly. For example, a user may request the home page of a selected Web site (termed a *first party* site), which may then trigger requests to *third party* servers of a content distribution network and an advertising service [Krishnamurthy and Wills 2009]. In some cases it may be helpful to know which transactions occurred as a direct result of a user action versus those generated to third party services. One method for doing this is to use the `Referer` and `Location` HTTP response headers to reconstruct the graph of user sessions. However, a proxy would have to be configured to collect this information to facilitate such an analysis.

Determining Service Providers.   In Section 4.4, we demonstrated a method for mapping service instances to service providers. While we had some success, there were many limitations. First, the RIR data is missing for many organizations (25–35%), and out-of-date for others (e.g., YouTube is now owned by Google, Inc., but still appeared as an independent organization in the RIR data). Second, some service providers do not have a single view of their own organization. For example, for reasons such as the acquisition of other companies, Google has duplicate OrgIDs [Krishnamurthy and Wills 2009], such as YOUTU. Similarly, Yahoo! has many different OrgIDs, some from acquisitions (e.g., INKT), and others from global operations (e.g., YAHOO-NET for Yahoo! Japan). Third, the scalability of the information services is quite limited. While we observed hundreds of thousands of distinct domains, we only resolved about 10,000, primarily due to the rate-limited query interface the information service provider offered. Different solutions to these issues are possible, but are clearly outside the scope of this work.

Classifying Service Instances.   In Section 4.5.1, we manually classified a small set of popular service instances. While this could be sufficient for an organization's information needs, it is time-consuming and clearly not scalable. In Section 4.5.2, we considered an existing automated classification technique. While this approach was more scalable than and about as accurate as our manual classification, it too has its disadvantages. In particular, the lack of descriptive meta-data is problematic for systematically classifying all service instances. The Centroid method may be coupled with other methods of classification to improve performance for some categories. For example, many hosts use words to describe their function (e.g., `news.yahoo.ca`, `finance.google.com`). By mining these domain keywords, the

accuracy of the classifier may improve. Another challenge is verifying the accuracy of automated classifications (at scale). One possible solution is "collective intelligence"; i.e., use a mechanism like Amazon's Mechanical Turk[8] to have automated classifications validated by humans.

Identifying Consumers. As more use of Web services (i.e., machine-to-machine) occurs, the interest in identifying the consumers may increase. While the `User-Agent` field could potentially be used to answer such questions, there are several challenges with using this field. First, as noted in Table I, there are many unique `User-Agent` values. Second, many of the values are not well-structured. A standardized approach for setting the values in this field seems like the best solution to these issues.

Other Questions. We investigated three questions of potential interest to network operators or researchers, and encountered numerous problems. As Web-based services become more important to organizations, a much broader set of questions are likely to be asked, which will result in an even larger set of issues. Future work will involve exploring what other important questions might be, so that solutions can be developed in unison with those for the issues identified above.

5. LONGITUDINAL ANALYSIS OF WEB WORKLOAD CHARACTERISTICS

During the past decade, the Web and its underlying infrastructure have changed significantly, including tremendous growth in services, the emergence of "Web 2.0" [Cormode and Krishnamurthy 2008], and new/improved Internet access capabilities. In this section we provide a longitudinal analysis of Web workload characteristics, to examine the effects such changes have had.
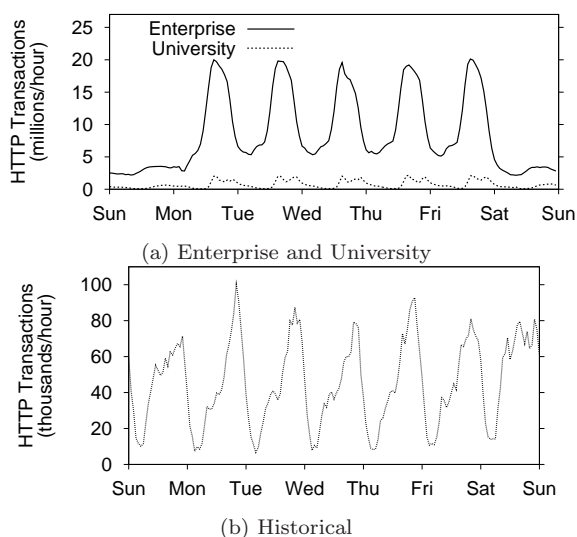

(a) Enterprise and University

(b) Historical

Fig. 5.   HTTP hourly transaction rates.

---

[8]https://www.mturk.com/mturk/welcome

Table VIII. Breakdown of methods.

Method	Enterprise			University			Historical		
	Trans. (%)	Req.* Bytes (%)	Rsp. Bytes (%)	Trans. (%)	Req. Bytes (%)	Rsp. Bytes (%)	Trans. (%)	Req. Bytes (%)	Rsp. Bytes (%)
GET	92.0	76.6	93.8	92.6	0.0	97.7	98.0	98.0	99.2
POST	7.7	23.0	6.2	6.7	99.2	2.0	1.6	1.8	0.7
Other	0.3	0.4	0.1	0.7	0.8	0.3	0.4	0.2	0.1

* The data sets differ in how the request and response bytes were counted. The enterprise and historical data sets count the HTTP headers as part of the requests and responses, while the university data set does not. This has only a minor effect on the percentages for response bytes, but is responsible for the noticeable difference between the data sets for request bytes.

5.1 Similarities in Transaction-level Statistics

By comparing transaction-level statistics of our traces with those reported in earlier studies, we have identified characteristics that have remained *invariant* over the past decade.

Strong diurnal pattern: Figure 5 shows the HTTP activity in each data set over a one-week period. There are strong diurnal components, as have been observed previously [Crovella and Bestavros 1997]. As in the past, the time of day and day of week non-stationarities observed are consistent with human activity. In particular, the enterprise and university data sets show peak workloads during work hours on weekdays, when users are at work or on campus. The historical data set (from a residential ISP) reveals peak use during evenings and weekends, when users are at home. Unlike a decade ago, however, the pervasiveness of portable devices (e.g., laptops) and changes to business policies (e.g., shutting down computers at night to reduce energy use) could result in automated use of Web-based services occurring during work hours.

GET is the dominant method: As was the case a decade ago [Cunha et al. 1995; Duska et al. 1997; Kelly and Mogul 2002; Mahanti et al. 2000; Wolman et al. 1999], GET is the most prevalent method, used for over 90% of the transactions and over 90% of the downloaded data. Most remaining transactions (6–8%) use the POST method for uploading data to the server. This method is used, for example, in (Web-based) email, posting comments, and Web-based instant messaging.

HTTP response codes: Table IX shows that like a decade ago, most HTTP transactions resulted in either a "Successful" (status 200) object transfer or a "Not Modified" (status 304) cache validation message. In our new data sets, these transactions account for 90-94% of all transactions, compared to 92% of all transactions in the historical data set. The next most common status codes are "Redirections" (e.g., status 301, 302, 303) at 4–5%, reflecting load-balancing across Web server farms or delivery infrastructures. The "Partial Content" (status 206) response, although seldom used, contributes a moderate fraction (2–7%) of the bytes downloaded in the new data sets. This is likely due to some services breaking large objects into smaller pieces prior to downloading. The proxy in our historical data set also handled FTP transactions; these are classified as part of "Other" status

Table IX.   Breakdown of status codes.

Status Code	Enterprise		University		Historical	
	% Trans.	% Bytes	% Trans.	% Bytes	% Trans.	% Bytes
Successful (200)	72.18	92.15	77.87	97.27	75.83	87.50
Not Modified (304)	21.44	0.43	11.98	0.08	15.81	0.15
Partial Content (206)	0.35	7.02	0.54	2.42	0.02	0.04
Redirection (30x)	3.73	0.14	5.14	0.10	3.96	0.11
Client Error (40x)	1.68	0.17	1.50	0.07	1.60	0.06
Server Error (50x)	0.19	0.01	0.17	0.01	1.51	0.06
Other	0.43	0.08	2.80	0.05	1.27	12.08

codes.

Zipf-like popularity: Figure 6 shows the rank-frequency plot (with logarithmic scale on both axes) for the total number of references to each unique object for each of our data sets.[9] The straight-line trend indicates strong Zipf-like behavior in all data sets, consistent with earlier Web studies [Breslau et al. 1999; Glassman 1994; Kelly and Mogul 2002; Mahanti et al. 2000]. Using a discrete version of *maximum likelihood estimation* [Clauset et al. 2009; Newman 2005], we estimated the Zipf exponent [Adamic 2009; Adamic and Huberman 2002] as $\theta \approx 1.0$ in each of our recent data sets, and as $\theta \approx 0.9$ in the historical data set. These results suggest that there have been a shift towards a higher skew in object popularity. Another reason for the higher Zipf exponent than suggested by previous studies may be due to the maximum likelihood estimation giving a more even weight to all parts of the distribution. Traditional regression techniques, which were used in some prior works to estimate the Zipf exponent, are subject to systematic subject to systematic (and potentially large) errors [Clauset et al. 2009].
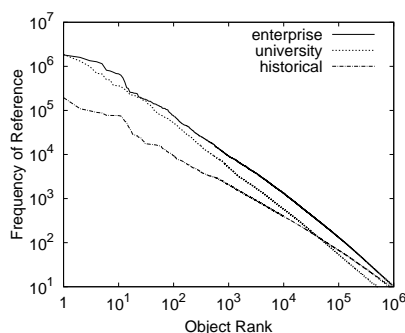

Fig. 6.   Object popularity.

Caching is effective: Wolman *et al.* [1999] considered the "ideal" cache request hit rate as a function of client population, and observed that 51–60% of requests

---

[9]For this analysis only, we examined the referencing activity of a single week day (Sep. 15th) in the enterprise data set, due to the space requirements of this analysis and the memory limitations of our analysis machine. We still used the entire university and historical data sets.
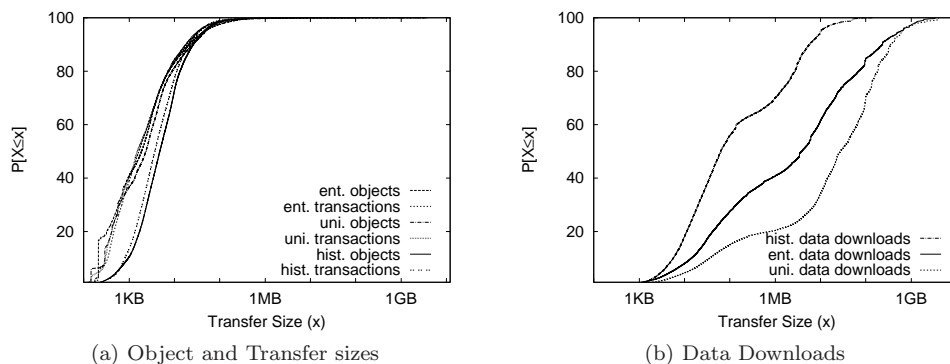
(a) Object and Transfer sizes

(b) Data Downloads

Fig. 7.   Transfer size distributions.

Table X.   Breakdown of cache actions.

Cache action	Enterprise		Historical	
	% Trans.	% Bytes	% Trans.	% Bytes
Hit	46.8	24.8	26.9	22.8
Not Cacheable	38.0	30.7	8.6	4.7
Miss	12.1	37.8	43.5	72.3
Other	3.1	6.8	21.0	0.2

are to cacheable documents. Examining the *cache action* fields (i.e., the action taken by the cache for each requested object, based on the accompanying HTTP cache control headers) in our enterprise data set suggests that caching properties remain similar to a decade ago. In the enterprise data set, 38% of cache misses due to objects marked as uncacheable, compared to 40–49% reported by Wolman *et al.* [1999]. Our historical data set did not include detailed *cache action* information. We established a lower bound (8.6%) using the number of requests for server-side scripts (e.g., `cgi-bin`).

The set of enterprise proxies achieve a 47% cache hit rate (which is consistent with the achievable hit ratios reported a decade ago by both Wolman *et al.* [1999] and Mahanti, *et al.* [2000]. Our historical data set observed a much lower hit rate of 26.9%, but this was largely due to a conservative consistency mechanism, as an additional 14.3% of requests resulted in Not Modified responses from the origin servers. This means the cache in the historical data set had an effective hit rate of 41.2%. The "byte hit rate" is substantially lower (25% in the enterprise data set, 23% in the historical data set), indicating that hits are typically for small objects.

Small objects dominate: Figure 7(a) shows the empirical CDF for the sizes of distinct objects. It also shows the transfer size distributions, since there may be many transactions for the same object. Figure 7(b) shows the cumulative percentage of the total data volume for object transfers, as a function of the object threshold size. (When interpreting these results it is important to note that the per-object and per-transaction curves in Figure 7(a) are with regards to frequencies, while the download curves in Figure 7(b) pertain to data volume.)

The size distributions for unique objects and for transactions are similar, both within and across the three data sets. In addition, most transfers are for small objects: about 85% of the transfers are for objects smaller than 10 KB in the two new data sets, and 77% in the historical data set. Not only do small file sizes still dominate, but Figure 7(a) shows that most objects and transfers in the new data sets are smaller than they were in the historical data set. This is at least partially due to many Web sites splitting their pages into many (smaller) objects that are hosted and delivered by many different servers.

Heavy tails: The "data download" curves in Figure 7(b) show that large object transfers account for most of the data volume. For example, while only 15% of the object transfers in the new data sets are larger than 10 KB, these transfers together account for over 90% of the total data downloaded. In the historical data set, 24% of transfers were larger than 10 KB, accounting for 82% of the total data downloaded. Similarly, large objects account for a disproportional fraction of the total data volume. 50% of the total data volume is contributed by objects exceeding 4 MB in the enterprise data set and 28 MB in the university data set. The slightly higher skew in the university data set is due to substantially more video traffic from services such as YouTube. These large files only account for less than 0.1% of the transactions. This skew is similar to observations from a decade ago, although larger objects are having a greater effect. In the historical data set, 50% of the total data volume was contributed by objects larger than 64 KB, which accounted for 2% of transactions.

5.2 Differences in Transaction-level Statistics

Compared to Web data sets from a decade ago, we have also identified some *differences* that provide insights into the evolution of the Web. Our data sets provide evidence to quantify these trends.

POST method more frequently used: POST currently accounts for only 6–8% of the transactions, although this is an increase from a decade ago. For example, less than 2% of transactions in our historical data set used the POST method. This change is important to note, as it affects the amount of data uploaded. In the enterprise data set, the POST method was responsible for 23% of all uploaded data (including HTTP request headers). In the university data set (which excludes HTTP request headers from the count), the POST method accounts for essentially all of the uploaded data. As HTTP is used to provide additional functionality via Web-based services, use of the POST method may become more prevalent, in which case the volume of data transferred to servers will increase.

Scripts are more prevalent: In the Web's first decade, image and text (e.g., HTML) objects accounted for more than 90% of all transactions [Cunha et al. 1995; Kelly and Mogul 2002; Mahanti et al. 2000]. Today, *application* types are also prevalent, due to the extensive use of scripts (e.g., Javascript) to provide more sophisticated interfaces to Web-based services. In fact, 50–75% of the transactions of type "application" are for Javascript objects (i.e., declared to be of content type Application/Javascript by the server). Table XI provide a high-level breakdown

Table XI.   Breakdown of content types

Content Type	Enterprise		University		Historical	
	% Trans.	% Bytes	% Trans.	% Bytes	% Trans.	% Bytes
Images	51.87	14.81	39.00	9.96	73.11	47.58
Text	26.50	13.95	33.56	13.69	12.56	4.97
Application	18.71	40.14	13.75	28.33	8.61	4.71
Video	0.11	24.46	0.20	40.54	0.17	19.87
Audio	0.04	3.98	0.05	6.21	0.60	3.92
Other	2.76	2.65	13.44	1.27	4.95	18.95

of the content types observed.[10]

Video traffic is growing: The percentage of transactions involving audio and video remains quite low ($< 1\%$). However, audio and video types do account for 24–48% of the downloaded bytes, compared to 20% in the historical data set. These percentages may become more significant as high definition (HD) quality video becomes more prevalent. We expect this could happen in the near future, as it is already possible to buy a HD-quality video camera for around $100 USD, and commercial sites (such as YouTube[11], Dailymotion[12] and Smugmug[13]) already offer upload, download, storage, and viewing of HD content. Clearly, if such types do become more common, organizations will feel the effects on their networks.

Another change relates to video formats. A decade ago, the most common video formats included MPEG and AVI [Cunha et al. 1995; Kelly and Mogul 2002; Mahanti et al. 2000]. In our data sets, Flash Video (the format used by many popular video sharing services) accounts for 20% of the total data downloaded at the enterprise, and 25% at the university.

Heavier heavy tails: Comparing Figure 7(b) with those observed by Mahanti *et al.* [2000], the CDF of the unique object-size distribution is similar to a decade ago, though significantly larger transfer sizes are seen now; i.e., the tail is longer and shifted to the right. Should this change continue, it could diminish the effectiveness of caches to reduce network bandwidth consumption.

Software tools and size constraints: Finally, we note that there is a noticeable "step" in the transfer size distribution for both data sets near 100 MB. This "step" is due to some sites and/or software tools splitting large files into chunks of "smaller" sizes, as well as sites (e.g., YouTube) imposing file size limitations. As opposed to the general trend pushing to the right, these constraints truncate the

---

[10]These statistics rely on information in the content-type field for the new data sets, and file extensions and substrings in the historical data set. The 13% transactions in the "other" category in the University data set are a reflection of many objects not being labeled by the content providers. While the majority of these transactions can be manually placed into traditional categories such as applications, text and images, we note that a non-negligible fraction corresponds to live-updates, such as automatic score-board updates, which content providers may find more difficult to label into traditional categories. The "application" category reflects client-side scripts in the new data sets (e.g., `Javascript`) but server-side scripts (e.g., `cgi-bin`) for the historical data set.

[11]www.youtube.com

[12]www.dailymotion.com

[13]www.smugmug.com

tail of the distribution. However, we note that the tail now contains substantially larger files than observed a decade ago. For example, while files larger than 100 MB accounted for a negligible portion of the bytes observed by Mahanti *et al.* [2000] a decade ago, these files now account for 16–30% of the total bytes transferred in our data sets. As network bandwidths and storage capacities increase (and the corresponding costs decrease), the size-based constraints will likely move further "to the right", and may eventually disappear.

5.3 Summary and Discussion

To summarize, by comparing the current access characteristics to those previously observed in the literature, we have identified similarities and differences to the characteristics observed a decade ago. The primary differences are more prevalent use of the POST method, increased use of scripts (e.g., `Javascript`), a longer tail of large files (e.g., videos, software distribution), and the popularity of new video formats. These differences are consistent with general Web trends: increased interactivity of Web 2.0, increased availability of video on the Internet, and use of new object formats (e.g., Shockwave Flash video) to enable new functionality.

A necessary component of workload characterization is empirical data. This tends to be difficult to obtain, especially from enterprise environments. Interestingly, the use of Web-based services in the university traces considered here exhibits many similar properties as the enterprise traces. There are two reasons why we think these similarities exist. First, many of the users in a university environment will find employment with enterprises. When they do, they will "drag along" their usage behaviors (even if policies at the enterprises must first be revised to permit usage of certain Web-based services). Second, universities are a type of enterprise; they generate revenue (e.g., via tuition), and they incur expenses (e.g., to construct and operate buildings, etc.). As such, universities are faced with managing their expenses, and may become early adopters of Web-based services. In fact, numerous universities are already using Web-based services to provide students with email and other software functionalities. For example, Google currently offers free "communication and collaboration tools" to educational institutions [Google Apps Education Edition 2009].

6. CONCLUSIONS

As the Web evolves, and as organizations adopt Web-based services to provide critical business functionality, these organizations will have a greater need for information on the types of Web-based services they use, who provides them, who is using them, and why. In this paper, we took an initial step towards understanding how to answer such questions, by examining traces of Web-based service use within a large enterprise and a large university. To build this understanding, we examined techniques for identifying service instances, service providers, brands, and service classes. We highlighted the challenges for discerning the functionalities in use on the Web today, assessed the strengths and weaknesses of the techniques used, and provided initial insights on what popular functionalities the Web provides to organizations today.

In addition to presenting a new dimension to Web workload characterization, we also revisited the traditional object-level properties considered in previous work.

Our results show that while the Web has undergone fundamental changes with respect to the services it provides, the underlying object-level properties have not significantly changed. While there are some differences observed between the enterprise and university data sets, we note that many of the properties observed appear to be similar. This raises the question if the Web activity collected at large universities may provide useful insights for enterprise-level networks as well. However, a community-wide effort would be required to fully answer such a question. Identifying such invariants would be particularly valuable, as many academic researchers do not have access to enterprise data.

This paper works towards enabling organizations to better understand their use of Web-based services. A key contribution of our work is the discovery of the many gaps that exist in composing useful information in a scalable, automated and verifiable manner. Addressing these and related gaps are open research questions.

Acknowledgments

REFERENCES

ADAMIC, L. 2009. Zipf, power-laws, and Pareto - a ranking tutorial. `http://www.hpl.hp.com/research/idl/papers/ranking/ranking.html`.

ADAMIC, L. AND HUBERMAN, B. 2002. Zipf's law and the Internet. *Glottometrics* 3, 143–150.

ARLITT, M., FRIEDRICH, R., AND JIN, T. 1999. Workload characterization of a Web proxy in a cable modem environment. *ACM SIGMETRICS Performance Evaluation Review 27,* 2 (September), 25–36.

ATKINSON, K. 2008. Kevin's word list page (12 dicts package). `http://wordlist.sourceforge.net/`.

BAEZA-YATES, R., CASTILLO, C., AND EFTHIMIADIS, E. 2007. Characterization of national Web domains. *ACM TOIT 7,* 2 (May).

BENT, L., RABINOVICH, M., VOELKER, G., AND XIAO, Z. 2006. Characterization of a large Web site population with implications for content delivery. *WWW Journal 9,* 4 (December), 505–536.

BERNERS-LEE, T., CAILLIAU, R., LUOTONEN, A., FRYSTYK-NIELSEN, H., AND SECRET, A. 1994. The world wide Web. *Communications of the ACM 37,* 8 (August), 76–82.

BRESLAU, L., CAO, P., FAN, L., PHILLIPS, G., AND SHENKER, S. 1999. Web caching and Zipf-like distributions: Evidence and implications. In *Proc. IEEE INFOCOM.* New York, NY.

BRO INTRUSION DETECTION SYSTEM. 2008. `http://www.bro-ids.org/`.

CLAUSET, A., SHALIZI, C., AND NEWMAN, M. 2009. Power-law distributions in empirical data. *SIAM Review 51,* 4 (November), 661–703.

CORMODE, G. AND KRISHNAMURTHY, B. 2008. Key differences between Web 1.0 and Web 2.0. *First Monday.*

CROVELLA, M. AND BESTAVROS, A. 1997. Self-similarity in world wide Web traffic: Evidence and possible causes. *IEEE/ACM Transactions on Networking 5,* 6 (December), 835–846.

CUNHA, C., BESTAVROS, A., AND CROVELLA, M. 1995. Characteristics of world wide Web client-based traces. Tech. Rep. BUCS-TR-1995-010, Boston University, CS Dept, Boston, MA. April.

DUSKA, B., MARWOOD, D., AND FREELEY, M. 1997. The measured access characteristics of world wide Web client proxy caches. In *Proc. USITS.* Monterey, CA.

Fetterly, D., Manasse, M., Najork, M., and Wiener, J. 2003. A large-scale study of the evolution of Web pages. In *Proc. WWW*. Budapest, Hungary.

Glassman, S. 1994. A caching relay for the world wide Web. *Computer Networks and ISDN Systems 27,* 2 (November), 69–76.

Google Apps Education Edition. 2009. `http://www.google.com/educators/p_apps.html`.

Han, E. and Karypis, G. 2000. Centroid-based document classification: Analysis and experimental results. In *Proc. PKDD*. Lyons, France.

Kelly, T. and Mogul, J. 2002. Aliasing on the world wide Web: Prevalence and performance implications. In *Proc. WWW*. Honolulu, HI.

Krishnamurthy, B. and Wills, C. 2006a. Cat and mouse: Content delivery tradeoffs in Web access. In *Proc. WWW*. Edinburgh, Scotland.

Krishnamurthy, B. and Wills, C. 2006b. Generating a privacy footprint on the Internet. In *Proc. IMC*. Rio de Janeiro, Brazil.

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

Kwan, O. and Lee, J. 2003. Text categorization based on k-nearest neighbors approach for Web site classification. *Information Processing and Management 39,* 1 (January), 25–44.

Li, W., Moore, A., and Canini, M. 2008. Classifying http traffic in the new age. In *Proc. ACM SIGCOMM (poster)*. Seattle, WA.

Ma, J., Levchenko, K., Kreibich, C., Savage, S., and Voelker, G. 2006. Unexpected means of protocol inference. In *Proc. IMC*. Rio de Janeiro, Brazil.

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

Manning, C., Raghavan, P., and Schütze, H. 2009. *An Introduction to Information Retrieval*. Cambridge University Press, Cambridge, England.

Newman, M. 2005. Power laws, Pareto distributions and Zipf's law. *Contemporary Physics 46,* 5 (September/October), 323–351.

Qi, X. and Davison, B. 2007. Web page classification: Features and algorithms. Tech. Rep. LU-CSE-07-010, Lehigh University. June.

Schneider, F., Agarwal, S., Alpcan, T., and Feldmann, A. 2008. The new Web: Characterizing Ajax traffic. In *Proc. PAM*. Cleveland, OH.

Trestian, I., Ranjan, S., Kuzmanovic, A., and Nucci, A. 2008. Unconstrained endpoint profiling (googling the Internet). In *Proc. ACM SIGCOMM*. Seattle, WA.

Wikipedia Article. 2009. Domain hack. `http://en.wikipedia.org/wiki/Domain_hack`.

Williams, A., Arlitt, M., Williamson, C., and Barker, K. 2005. Web workload characterization: Ten years later. *Web Content Delivery*, 3–21.

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