End-User Mapping: Next Generation Request Routing for Content Delivery

Fangfei Chen
Akamai Technologies
150 Broadway
Cambridge, MA
fachen@akamai.com

Ramesh K. Sitaraman
University of
Massachusetts, Amherst
& Akamai Technologies
ramesh@cs.umass.edu

Marcelo Torres
Akamai Technologies
150 Broadway
Cambridge, MA
mtorres@akamai.com

ABSTRACT

Content Delivery Networks (CDNs) deliver much of the world's web, video, and application content on the Internet today. A key component of a CDN is the mapping system that uses the DNS protocol to route each client's request to a "proximal" server that serves the requested content. While traditional mapping systems identify a client using the IP of its name server, we describe our experience in building and rolling-out a novel system called end-user mapping that identifies the client directly by using a prefix of the client's IP address. Using measurements from Akamai's production network during the roll-out, we show that end-user mapping provides significant performance benefits for clients who use public resolvers, including an eight-fold decrease in mapping distance, a two-fold decrease in RTT and content download time, and a 30% improvement in the time-to-first-byte. We also quantify the scaling challenges in implementing end-user mapping such as the 8-fold increase in DNS queries. Finally, we show that a CDN with a larger number of deployment locations is likely to benefit more from end-user mapping than a CDN with a smaller number of deployments.

1. INTRODUCTION

Content Delivery Networks (CDNs) deliver much of the world's web sites, video portals, e-commerce applications, social networks, and file downloads. As an example, Akamai's CDN currently serves 15-30% of all web traffic from a large distributed platform of over 170,000 servers deployed in over 102 countries and 1300 ISPs around the world [2]. The CDN hosts and delivers content on behalf of thousands of enterprises and organizations that represent a microcosm of the Internet as a whole, including business services, financial services, travel, manufacturing, automotive, media, entertainment, e-commerce, software, gaming, and the public sector. The clients[1] who access content on Akamai's CDN are a representative sample of Internet users from nearly every country, every major ISP, and use every major device.

SIGCOMM '15, August 17 - 21, 2015, London, United Kingdom

 ISBN 978-1-4503-3542-3/15/08…$15.00

DOI: http://dx.doi.org/10.1145/2785956.2787500

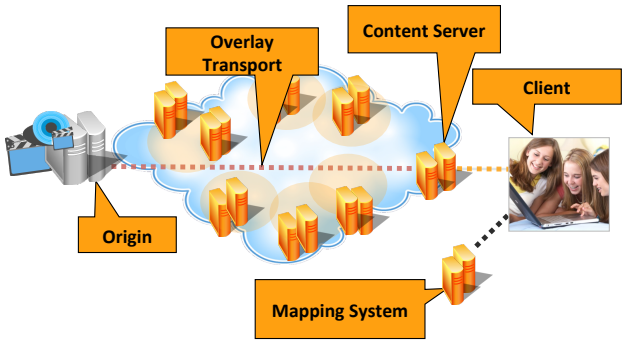


Figure 1: A Content Delivery Network

The goal of a CDN is to host and deliver content and applications to clients around the world with high availability, performance, and scalability [13, 21]. Akamai's CDN achieves its goal by deploying a large number of servers in hundreds of data centers around the world, so as to be "proximal" in a network sense to clients. To understand the overall architecture of the CDN, we enumerate the steps involved when a client accesses content hosted on the CDN. As shown in Figure 1, when the client accesses a Web page, the domain name of the Web page is translated to the IP address (shortened to "IP" in this paper) of a server that is live, has sufficient capacity, and is proximal to the client. The domain name translation is provided by the CDN's mapping system that we study in this paper. The client requests content from the server assigned to it by the mapping system. If the server has the content in cache, it serves the content to the client. Otherwise, the server requests the content from the origin servers that are operated by the content provider

[1]In this paper, we use the term "client" to denote the end-user or his/her device such as a cell phone, desktop or laptop that is connected to the Internet and is running software such as a browser capable of accessing Web pages.

and and serves it to the client. For a more detailed discussion of CDN evolution and architecture, we refer to [21].

A central component of Akamai's CDN is its *mapping system*. The goal of the mapping system is to maximize the performance experienced by the client by ensuring quicker downloads of the accessed content. To speedup the downloads, the mapping system routes each client request to a "proximal" server that can be reached by the client with low latency and loss. Further, the mapping system ensures that the chosen server is live, not overloaded, and is likely to contain the requested content. The last consideration reduces the probability of a cache miss at the server that would result in the content being fetched from origin with longer response times. Conceptually, the mapping system can be viewed as computing the following complex time-varying function:

$$\mathrm{MAP_t} : \Sigma_{\mathrm{Internet}} \times \Sigma_{\mathrm{Akam}} \times \mathrm{Domain} \times \mathrm{LDNS} \rightarrow \mathrm{IPs}. \quad (1)$$

At each time t, $\mathrm{MAP_t}$ takes as input the current state of the global Internet $\Sigma_{\mathrm{Internet}}$, including detailed real-time knowledge of connectivity, liveness, latency, loss, and throughput information; the current state of Akamai's CDN Σ_{Akam}, including real-time knowledge of liveness, load, and other information about servers and routers of the CDN; the domain name of the content that is being accessed by the client; and, the local recursive domain name server (LDNS) that makes the request for domain name resolution on behalf of the client. The mapping system returns two or more IPs[2] of the CDN's servers that can serve the requested content.

Akamai's mapping system routes trillions of client requests per day, controlling tens of terabits per second of content traffic served to clients world-wide. On a typical day, there are 6.4 million LDNS servers located in 240 countries making 1.6 million DNS queries per second (cf. Figure 2), representing 30 million client requests per second around the globe. Ever since the first CDNs were built at Akamai nearly sixteen years ago [13], the mapping system has been the subject of much research to improve its responsiveness and accuracy. However, as represented in Equation 1, traditional mapping systems at CDNs make request routing decisions based on the identity of the client's LDNS rather than that of the client itself. We call this *NS-based mapping* and can be inaccurate in cases when the LDNS is not in a "similar" location as the client, i.e., when the network characteristics of the LDNS is not a good proxy for that of the client. This fundamental limitation arises from the use of the domain name system (DNS) protocol that allows the mapping system to learn the identity of the LDNS but not the client on whose behalf the domain name translation request is made.

To rectify the limitations of the DNS protocol, Google, Akamai and others industry players have recently proposed an extension to the DNS protocol that allows a recursive name server to specify a prefix of the client's IP (usually a /24 prefix) when requesting domain name translations on behalf of a client [11]. For the first time, this mechanism allows the novel possibility of building a mapping system that has *direct* knowledge about the client and uses it to provide more accurate mappings. We call such a system *end-user mapping*. The insights gained from building and rolling-out Akamai's end-user (EU) mapping system is our focus.

[2]While the mapping system checks liveness before returning the IP of a server, more than one server is returned as a additional precaution against transient failures.

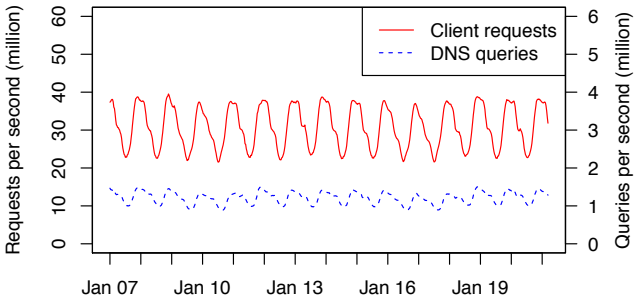


Figure 2: Client requests served (left axis) and DNS queries resolved (right axis) by the mapping system. When the mapping system resolves a DNS query from a LDNS, multiple content requests from clients that use that LDNS may follow.

Our contributions. We list key contributions below.

1. The limitations of NS-based mapping caused by discrepancies in the locations of clients and LDNSes have been known for over a decade [24]. However, we provide the first public analysis of clients and their LDNSes at a global scale across the entire Internet using data from 584 thousand name servers and 3.76 million /24 client IP blocks across 37294 AS'es and 238 countries across the world.

2. Our work presents the architecture and real-world roll-out of Akamai's end-user mapping, a major conceptual advance in mapping technology. We capture the performance impact of the roll-out on actual clients around the world. Web performance is a complex phenomena that is influenced by the global state of the Internet, the connectivity of the client, properties of Web sites and their hosting infrastructure, and a multitude of other factors. Our work captures the impact of the new mapping paradigm in a real-world setting providing insights that are hard to obtain in a controlled experimental setting.

3. End-user mapping requires both measurements and analysis to be performed at a much larger scale, as mapping decisions are made at a much finer granularity. Using extensive data from clients and their LDNS architectures in the global Internet and measurements taken during the end-user mapping roll-out, we provide insights into the scaling considerations in using the EDNS0 client-subnet extension of the DNS protocol.

4. Using latency measurements from over 2500+ server deployment locations around the world to 8K representative client IP blocks that generate the most traffic on the Internet, we study the important question of how deployments impact the performance of traditional NS-based and end-user mapping. We show that end-user mapping provides more incremental benefits for a CDN with servers in a large number of deployed locations than a CDN deployed in fewer locations. Further, we expose an inherent limitation of NS-based mapping in reducing latencies for the worst 1% of clients.

Roadmap. In Section 2, we describe the architecture of a traditional NS-based mapping system and how end-user mapping can be incorporated into this architecture using the EDNS0 client-subnet extension. In Section 3, we analyze the relative locations of clients and their LDNSes in the global Internet with the view of understanding the benefits that we are likely to see from rolling-out end-user mapping. In Sections 4 and 5, we analyze the performance and scalability impact of our roll-out of end-user mapping system to clients who use public resolvers. In Section 6, we study the role of server deployments in end-user mapping. In Section 7 we present related work and conclude in Section 8.

2. THE MAPPING SYSTEM

A Web site hosted on Akamai typically delegates authority for its domain names to authoritative name servers that are part of the mapping system. Further, each client uses a "local" domain name server[3] (LDNS) that works in a recursive fashion to provide domain name translations for the client. The LDNS that provides domain name service for the client is typically hosted by the Internet Service Provider (ISP) who provides Internet connectivity to the client. Alternately, the LDNS could be an *public resolver* that is a name server deployed by a third-party provider that can be used by the client. The location of the LDNS with respect to the client depends on the DNS architecture of the name service provider, whether it be an ISP or a public resolver provider such as Google DNS [5] or OpenDNS [7].

To better illustrate Akamai's mapping system, we trace through the steps of how a client interacts with the system to obtain a domain name resolution (see Figure 3).

(1) Suppose that the client wants to access content at some Web site that is hosted on Akamai. The client requests its LDNS to resolve the domain name of the Web site.

(2) LDNS works in a *recursive* mode as follows. If the LDNS has a valid name resolution for the requested domain in its cache, it responds to the client with the relevant IPs. Otherwise, the LDNS forwards the request to an authoritative name server for the requested domain.

(3) The authoritative name server responds with a valid resolution to the LDNS. LDNS caches the response and in turn forwards the response to the client.

Note that a DNS response from an authoritative name server is associated with a TTL (time-to-live) that dictates how long the response is valid. TTL's are tracked and enforced as the response is forwarded and cached downstream by name servers and resolvers, including the LDNS and the client's resolver. When the TTL expires, the cached entry is made invalid, requiring a new DNS resolution.

NS-based versus end-user mapping. In a traditional NS-based mapping system, the LDNS does not forward any information about the client when it contacts the authoritative name servers in step (2) above. Hence, the mapping system does not know the IP of the client that requested the name resolution and assigns edge servers entirely based on the IP of its LDNS. However, in end-user mapping, the LDNS forwards a prefix of the client's IP to the authoritative name servers in step (2) above using the newly-proposed EDNS0 client-subnet extension. This enables the end-user mapping to use additional client information in providing domain name translations as we describe next.

[3]Despite its name, a LDNS may not be very "local" to the client, the key rationale for end-user mapping.

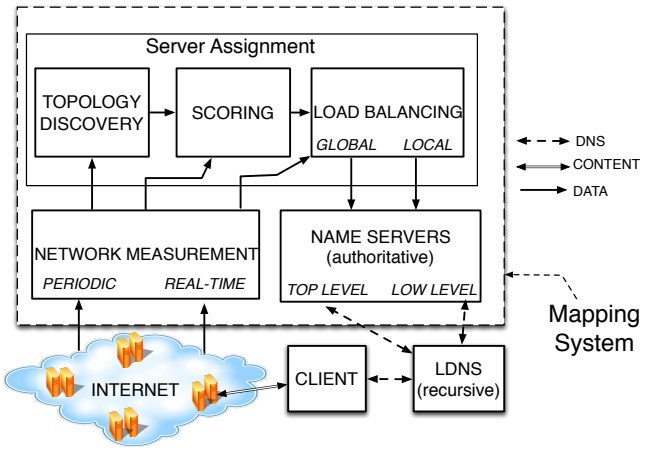


Figure 3: The architecture of the mapping system

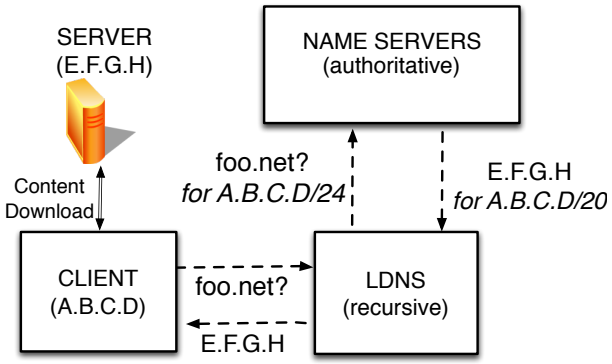


Figure 4: Example of interaction between the client, LDNS, and Akamai's name servers with the EDNS0 extension.

2.1 End-User Mapping

End-user mapping deployed recently at Akamai uses the identity of the client rather than its LDNS. Conceptually, end-user mapping (EUMAP) computes the following time varying function.

$$\text{EUMAP}_t : \Sigma_{\text{Internet}} \times \Sigma_{\text{Akam}} \times \text{Domain} \times \text{Client} \rightarrow \text{IPs}. \quad (2)$$

Compared to NS-based mapping (see Equation 1), end-user mapping uses the client information to make more accurate mapping decisions, even in cases where the LDNS and the client are not proximal to each other. A key enabler for our end-user mapping design is a recent Internet draft to extend the DNS protocol called the EDNS0 client-subnet extension that allows recursive name servers to forward information about the client as a part of their DNS request [11]. Figure 4 shows the interaction between the client, recursive, and authoritative name servers for an example domain `foo.net`

when the name servers support the EDNS0 protocol extension. The client with IP `A.B.C.D` contacts its LDNS to resolve `foo.net`. With extension support, when the LDNS forwards the request for `foo.net` to an authoritative name server it can append a /x prefix of the IP of the client who initiated the request, where the prefix[4] used is generally /24. (By /x prefix we mean the first x bits of the IP.) The authoritative name servers, which in the case of a domain hosted on Akamai is part of the mapping system, responds with server IPs appropriate for a /y prefix of the client's IP where $y \leq x$, i.e., the name server can return a resolution that is valid for a superset of the client's /x IP block. (By client's /x IP block, we mean the set of IPs that have same first x bits as the client's IP.) The DNS resolution provided by the authoritative name server can be cached for the duration of the TTL by downstream recursive name servers such as the LDNS. However, the cached resolution is only valid for the IP block for which it was provided and not for any client IPs that do not belong to the block.

2.2 Mapping System Architecture

The mapping system consists of three major functional components as shown in Figure 3 that we describe in turn. We also use data collected from the network measurement component below for our analysis.

1) Network Measurement. Both the global Internet and Akamai's CDN are monitored and measured. The data that needs to be collected on both counts is enormous and varied. The Internet is a large "patchwork" of 71K autonomous systems (AS's) that interconnect with each other in complex and ever-changing ways. The server and network components of Akamai's CDN are deployed in clusters in more than a thousand networks around the globe. A few major sources of data collected include:

(i) AS-level information is collected by Akamai's BGP collectors installed around the Internet that initiate BGP sessions with ISP's and periodically records the BGP session state. This information is used to understand which IPs belong to which AS, how AS'es connect with each other, etc.

(ii) Geographic information such as the city, state, country, and continent is deduced for IPs around the world using various data sources and geolocation methods [1].

(iii) Name server information is collected using the DNS request logs for Akamai-hosted domains from name servers (i.e., LDNSes) around the world.

(iv) Network-level measurements include path information, latency, loss, and throughput between different points on the Internet.

(v) Liveness and load. Liveness and load information of all components of Akamai's CDN is collected in real-time, including servers and routers.

2) Server Assignment. The server assignment component uses network measurement data to create a real-time topological map of the Internet that captures how well the different parts of the Internet connect with each other, a process called *topology discovery*. The topological map is then used to evaluate what performance clients of each LDNS is likely to see if they are assigned to each Akamai server cluster, a process called *scoring*. Different scoring functions that incorporate bandwidth, latency, packet loss, etc can be used for different traffic classes (web, video, applications, etc). The *load balancing* module assigns servers to each client request in two hierarchical steps: first it assigns a server cluster for each client, a process called *global load balancing*. Next, it assigns server(s) within the chosen cluster, a process called *local load balancing*. To perform these tasks, the load balancer uses the output of scoring to evaluate candidate server choices that yield the highest performance for each client request and combines that information with liveness, capacity, and other real-time information about the CDN. The load balancing algorithms are described in greater detail in [19].

3) Name Servers. Akamai has a large distributed system of name servers around the world that act as authorities for Akamai-hosted domain names. For example, a content provider hosted on Akamai can CNAME their domain to an Akamai domain, for example, `www.whitehouse.gov` could be CNAME'd to the Akamai domain of e2561.b.akamaiedge.net. The authority for the latter domain is in turn delegated to an Akamai name server that is typically located in an Akamai cluster that is close to the client's LDNS. This delegation step implements the global load balancer choice of cluster for the client's LDNS, so different clients could receive different name server delegations. Finally, the delegated name server returns "A" records for two or more server IPs to be used by the client for the download, implementing the choices made by the local load balancer.

3. UNDERSTANDING CLIENTS AND THEIR NAME SERVERS

To motivate the need for end-user mapping, we start by analyzing the locations of clients relative to their recursive name servers (i.e., LDNS) in the global Internet. To obtain an accurate picture we need to match a large characteristic set of clients around the world with their respective LDNSes. The matched client-LDNS pairs can then be located using our geo-location database [1] to provide the geographic location and network information needed for the analysis.

3.1 Collecting Client-LDNS pairs

Associating a client with its LDNS has some intrinsic difficulties. Both the LDNS's request for a domain name resolution and the client's subsequent request for an URL on that domain are logged at Akamai's authoritative name servers and content servers respectively. One potential approach is to match these requests to obtain client-LDNS pairings. However, matching the requests is tricky and inexact since the two requests can be spaced within a time window equal to the TTL of the domain name. Further, when the client receives a cached response from its LDNS, the LDNS makes no corresponding downstream request to Akamai's authoritative name servers. While there are heuristic ways of obtaining a smaller sample of client-LDNS pairs [24], our chal-

[4]A prefix longer than /24 is discouraged to retain client's privacy.

lenge is to obtain a large characteristic and definitive set of pairs that have good coverage of the clients who generate traffic on the global Internet.

To obtain a large set of pairs, we use Akamai's download manager called NetSession [3]. NetSession is installed on client devices and is used to perform downloads in a faster and more reliable fashion. Software and media publishers opt-in to use Netsession features to improve http delivery performance for their content. Once they opt-in, clients use Netsession to download that content. Thus, Netsession has a large, representative installed base of clients around the world, making it an ideal measurement platform for our analysis. More than 30 million unique NetSession clients perform transactions every month.

NetSession was instrumented to collect LDNS information as follows. Each NetSession client maintains a persistent connection with a NetSession control plane. Even if the client is behind a NAT, it can reliably learn its external client IP from this persistent connection. NetSession clients also found their LDNS server performing a "dig" command on a special Akamai name `whoami.akamai.net`. The client-LDNS association was then sent to Akamai's cloud storage for processing. The LDNS information for clients around the world were then aggregated in the cloud to the granularity of /24 client IP blocks. Specifically, for each /24 client IP block, the process generates the set of IPs corresponding to the LDNSes used by the clients in that address block. For each LDNS in the set, the relative frequency with which that LDNS appeared was computed.

Using the above process, we collected LDNS data from March 24 to April 7, 2014. On average, about 14.8 million records were processed per day during the course of our data collection. Client-LDNS association data for a total of 3.76 million /24 client IP blocks was computed in aggregate. While the clients that use NetSession are generally a fraction of the total active clients in any given /24 client IP block, our coverage of /24 client IP blocks is representative and significant of the overall Internet. *In particular, the /24 client IP blocks in our dataset account for about 84.6% of the total global client demand*[5] *served by Akamai.* The number of distinct LDNSes in our data set was just over 584,000. Thus, our data set is a large representative cross-section of clients and LDNSes in the global Internet.

To derive client-LDNS distance estimates, we use Akamai's Edgescape [1] geo-location database that uses registry data and network data distilled from transactions handled by over 170,000 Akamai servers in 102 countries and over a thousand ISP deployments around the world to establish geographical location and network information for IPs around the world. Edgescape can provide the latitude, longitude, country and autonomous system (AS) for an IP. For IPs in mobile networks, the mobile gateway location is used as the reference location. To derive the distance between a client-LDNS pair we use the latitude and longitude information to compute the great circle distance between the two locations.

[5]Client demand is a measure of the amount of content traffic downloaded by a client (or by clients in an IP block).

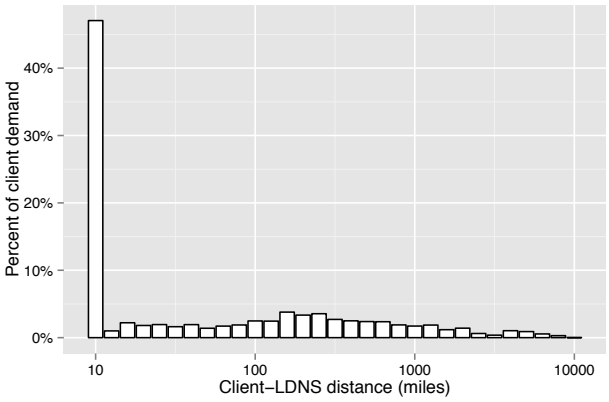


Figure 5: Histogram of client-LDNS distance for clients across the global Internet.

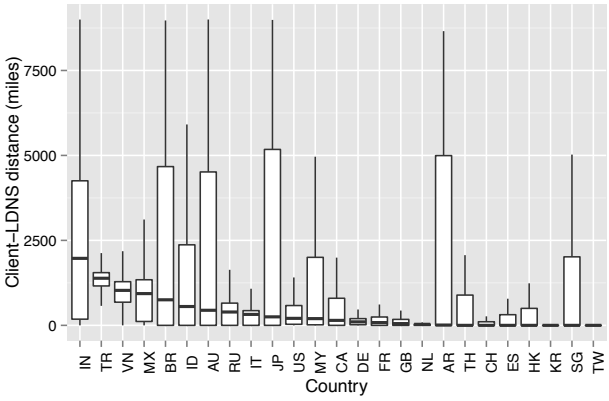


Figure 6: Client-LDNS distances by country.

3.2 How far are clients from their LDNSes

Figure 5 shows the overall global distribution of client LDNS distances. Nearly half of the client population is located very close to its LDNS. The most typical distance lies in a range that is no greater than the diameter of a metropolitan area. At around 200-300 miles, there is a noteworthy increase in the marginal distribution. At around 5000 miles, there is another increase that can be attributed to the small number of clients that use LDNS that are either across the Atlantic or Pacific oceans.

Breakdown by country. Breaking these distances down by country, Figure 6 is a box-plot[6] representing the 5th, 25th, median, 75th, and 95th quantiles of the per-country distributions. We list data for the top 25 countries as measured by aggregate client demand. Overall, most countries have a median distance that is fairly small, though India, Turkey, Vietnam and Mexico have median distances over 1000 miles. India, Brazil, Australia, and Argentina have significant populations whose LDNSes are very far away as over a quarter of the population is served by LDNSes whose distance is over

[6]All box plots in this paper show 5^{th}, 25^{th}, 50^{th}, 75^{th} and 95^{th} percentiles.

4500 miles. Western Europe sees low distances appearing in a small band. However, Korea and Taiwan are significant in having the smallest distances. This is not surprising considering the well-developed Internet infrastructure and the concentration of populations within a small geographical area in the major cities in these countries. Japan has a small median distance but a significant fraction of clients have LDNSes that are far away. One reason is clients at multi-national corporations with centralized LDNSes deployed outside Japan.

Public resolvers. We now evaluate the client-LDNS distance for public resolvers where a client uses an LDNS provided by a public third-party provider such as Google Public DNS or OpenDNS. Such providers have a distributed name server infrastructure and use IP anycast [14] to route clients to the "closest" LDNS deployment. However, the public resolvers use their unicast addresses when communicating with Akamai's authoritative name servers allowing us to geo-locate the public LDNSes. Figure 7 shows client-LDNS distance for clients that use public resolvers. We see that the client-LDNS distances are significantly higher with median distance at 1028 miles, compared to a median distance of 162 miles in the overall client population. This reflects the fact that the LDNS deployments of a public DNS provider may often not be local to the client.

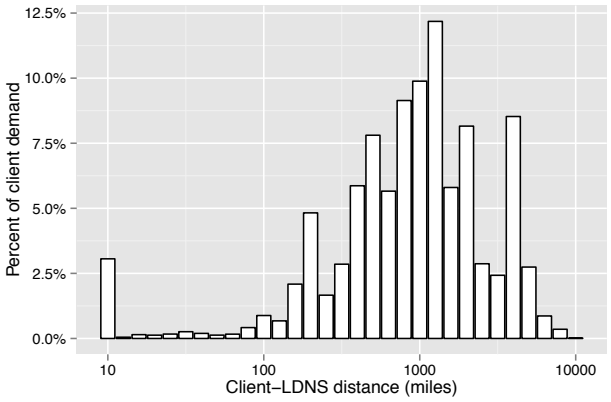


Figure 7: Histogram of the client-LDNS distance for clients who use public resolvers.

The country breakdown in Figure 8 shows disproportionately large distances for clients using public resolvers in some countries in South America, South East Asia and Oceania. The two South American countries of Argentina and Brazil had the largest distances. In this regard, it is notable that the largest public resolver provider, Google Public DNS, does not currently have a presence in many South American countries. Singapore and Malaysia are well served by the public resolvers hosted in Singapore. However, presumably due to peering arrangements, many clients in these countries are routed to more distant public resolvers. Clients who use public resolvers in Western Europe, Hong Kong and Taiwan are relatively close to their LDNS in comparison with other countries, though they are much more distant when compared to clients in those same countries who do not use public resolvers.

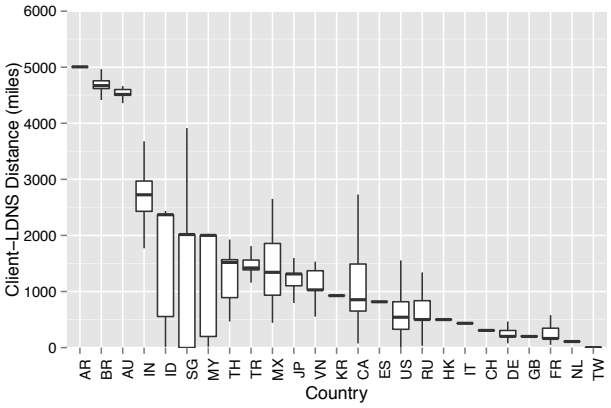


Figure 8: Client-LDNS distance for clients who use public resolvers.

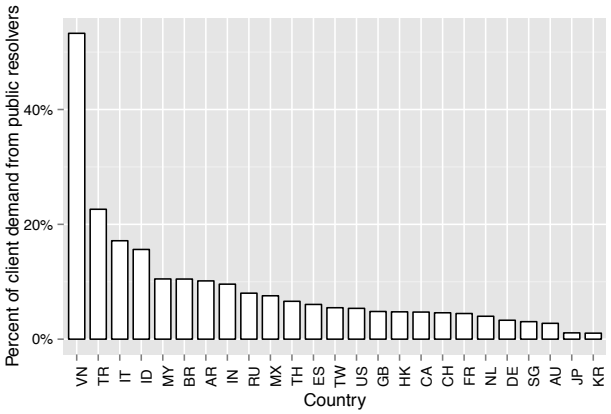


Figure 9: Percent of client demand originating from public resolvers, by country.

The adoption of public resolvers also vary country by country. Figure 9 shows the percentage of client demand originating from public resolvers broken down by country as seen in our NetSession data set. Clients in Vietnam and Turkey are very heavy users of public resolvers. Remarkably, despite the significant client-LDNS distances, a significant fraction of clients in India, Brazil, and Argentina use public resolvers. Overall, percent of client demand from public resolvers approaches 8 percent worldwide.

Breakdown by AS. Figure 10 shows the distribution of the client-LDNS distance as a function of the AS size, where AS size is the client demand originating from that AS as a percentage of the total client demand served by Akamai, i.e., an AS with size 2^{-1} has clients that account for 0.5% of the total client demand served by Akamai. A total of 37,294 ASes with the most demand were analyzed.

As can be seen in the figure, when the AS size is small, the client-LDNS distances are large, especially the higher percentiles of the distance. This may seem counter-intuitive.

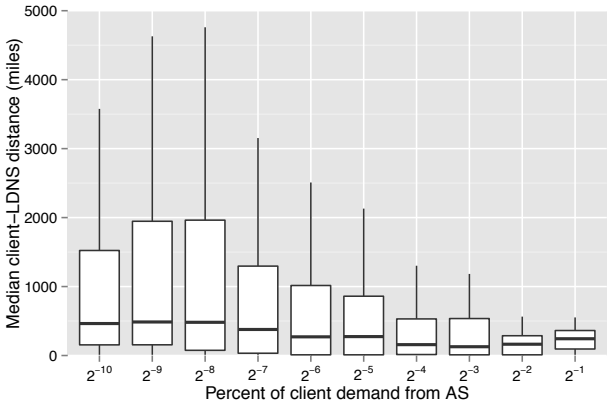


Figure 10: Client-LDNS distance as a function of AS size.

But, the reason is that smaller AS'es include small local ISPs who are more likely to "outsource" their name server infrastructure to other providers. The reason for the outsourcing is economic in nature as the ISP may not want to own and operate a name server infrastructure. So, the ISP may choose the inexpensive option of using a public resolver operated by a provider such as Google Public DNS, OpenDNS, Level 3, UltraDNS, etc. The "outsourcing" of DNS services often causes the LDNSes to be non-local, leading to larger client-LDNS distances. A different category of small AS'es with large client-LDNS distances are enterprises with geographically diverse branch offices who for operational convenience use a centralized name server infrastructure deployed in only one of those offices. Given the large client-LDNS distances, we expect end-user mapping to benefit a large fraction of clients of small AS'es.

Large ISPs typically operate their own name server infrastructures for their clients. Such infrastructure often consists of LDNSes that are deployed in multiple geographically distributed locations. To direct clients to the "nearest" LDNS, the IP anycast [16, 15] mechanism is often used. This explains the smaller values of client-LDNS distance despite the large geographical area covered by these global ISPs. However, IP anycast has many known limitations [23] that can result in a fraction of the clients being routed to far away LDNS locations. Thus, end-user mapping may be beneficial for clients of large ISPs also.

3.3 How far are clients that use the same LDNS from each other?

A *client cluster* is a set of clients that use the same LDNS. The clients on the Internet can be partitioned into client clusters, one cluster for each LDNS. We define the *radius* of a client cluster to be the mean distance of the clients in the cluster to the centroid of the cluster[7]. In traditional NS-based mapping, a client cluster is the unit for making server assignment decisions, i.e., all clients in a client cluster are assigned the same set of server IPs, since they use the same LDNS (cf. Equation 1). If a client cluster of a LDNS has a small radius, i.e., the clients are close together, a more sophisticated form of NS-based mapping could still be effective, *even if the client-LDNS distances are large.* The reason is that the mapping system could discover the client cluster and assign servers that provide good performance for the entire cluster. However, if the client cluster has a large radius, i.e., the clients are far away from each other, there may be no single server assignment for the entire cluster that is optimal for all clients in it. Thus, it is *inherently* difficult for NS-based mapping to perform well when the client cluster has a large radius, even knowing client-LDNS pairings.

Figure 11 reaffirms that on an overall basis a large fraction of clients are close to their LDNSes and the cluster radii are small. However, focusing on the subset of LDNSes that are public resolvers, we see that not only are client-LDNS distances large, but cluster radii are large as well. In fact, 99% of the public resolver demand originates from client clusters with radii between 470 to 3800 miles. The figure also shows that for public resolvers the mean cluster-LDNS distance tends to be larger than the cluster radius. This implies that the LDNS is often not deployed at a "central" location within the client cluster that it serves, i.e., near the centroid that minimize the mean client-LDNS distance. This is in part due to the fact that a public resolver provider does not have fine-grained control over which clients in which locations use their service. e.g., clients from countries where the provider has no deployments often use the service.

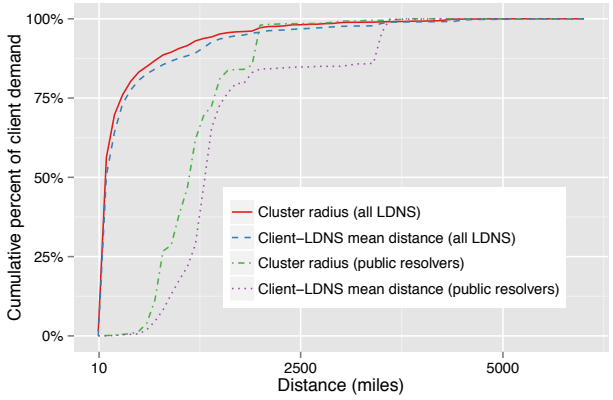


Figure 11: CDFs of mean client-LDNS distance and cluster radius for all LDNSes and for the subset that are public resolvers.

4. PERFORMANCE IMPACT

We present our experience and insights obtained in deploying end-user mapping for clients around the world in the first half of 2014. During this period, Akamai began the rollout of end-user mapping for clients who use public resolvers such as Google Public DNS and OpenDNS. The reasons for initially targeting clients who use public resolvers were twofold. Based on our analysis of client-LDNS distances in Section 3, we concluded that clients who use public resolvers

[7]Distances are computed using the latitude and longitude of the clients from our geo-location database. The radius and centroid use client demands as the weights.

are more likely to benefit from end-user mapping, since they tend to be farther away from their LDNSes (cf. Figure 7) and also had large client cluster radii (cf. Figure 11). Further, public resolver providers such as Google Public DNS and OpenDNS support the EDNS0 client-subnet extension that is required for end-user mapping. The end-user mapping roll-out[8] started on March 28th 2014 and completed on April 15th 2014. We present insights based on performance measurements made before, during, and after the roll-out.

4.1 Performance metrics

The performance experienced by clients who download web content can be characterized in many different but complimentary ways. We use the following four metrics measured from real-world clients downloading content from Akamai to evaluate the performance. Each metric sheds light on a different facet of mapping and client-perceived performance. Note that we expect all these metrics to decrease (smaller is better) when end-user mapping is rolled out.

1) Mapping distance is the great circle distance between a client and the server to which it was assigned by the mapping system. This is a purely geographical metric with no network-related component.

2) Round trip time (RTT) between the client and the server to which it was assigned. This is simply the TCP RTT measured from the server's TCP stack. This is purely a network-related metric.

3) Time to first byte (TTFB) is the duration from when the client makes a HTTP request for the base web page to when the first byte of the requested web page was received by the client. This quantity is measured from the client's browser and includes three components: (i) the time for the client's request to reach the server, (ii) time for the server to construct the web page, and (iii) time for the first chunk of the web page to reach the client. Note that end-user mapping is expected to decrease both the first and third component of TTFB above by reducing the server-client RTT. However, since many base web pages are "dynamic" and need to be personalized for the client, the second component of constructing the web page may involve fetching personalized elements from the origin. Overlay transport is used to speedup origin-server communication [26], though such transport is not impacted by the end-user mapping roll-out. Thus, we expect TTFB to show more modest reductions as end-user mapping impacts only some of its time components.

4) Content download time is the duration from the receiving of the first byte of the page to completing the download of the rest of the web page, including the content embedded in the page. This metric is also measured from the client's browser. The embedded content of web pages are typically more static and cacheable and includes CSS, images, and JavaScript that are not personalized to the client. Thus, unlike TTFB, we expect this metric to be significantly impacted by the end-user mapping roll-out as this metric is dominated by client-server latencies.

[8]We are unaware of any other Akamai software releases or Internet events happening during the roll-out period that could confound our measurements and conclusions.

4.1.1 High and low expectation countries

To better understand the performance impact, we classify the countries into two groups: a "high expectation" group where we expect end-user mapping to have a greater impact and a "low expectation" group where we expect the impact to be lower. Our client-LDNS analysis in Section 3.2 gives us an idea of what benefits to expect in which countries. Specifically, Figure 8 shows the proximity of clients to their LDNS for major countries. Using this analysis, we split the major countries into two halves. We define the high expectation group to be those clients who reside in countries where the median distance to a public resolver is more than 1000 miles and the low expectation group to be those whose median distance is under 1000 miles. We aggregate and present the performance metrics separately for these two groups, as we expect them to show different behaviors.

4.2 Collecting performance information

We collected performance metrics from a large and characteristic set of clients around the world before, during, and after the end-user mapping roll-out. We used Akamai's Real User Measurement (RUM) system [4] for our client-side performance measurements. RUM inserts JavaScript into select web pages delivered by Akamai. That JavaScript runs inside the client's browser when the page is downloaded by the client. The performance measurement is made using the industry-standard navigation timing [6] and resource timing APIs [8]. Using these APIs, the JavaScript running inside the client's browser collects precise timing information when the page download is in progress, including when the DNS lookup started and completed, when the TCP connection was initiated, when the fetch request was sent out, when the first byte of the response was received, and when all the page content was fully downloaded. Using these timing milestones, metrics such as TTFB and content download time can be computed. The timing measurements performed in client browsers around the world was sent to a backend cloud storage system and was subsequently analyzed to produce the aggregate statistics we provide in this section.

We collected RUM measurements from a wide selection of Web sites and clients around the world from Jan 1, 2014 to June 30th, 2014, a period that includes the end-user mapping rollout from March 28th to April 15th. Since the roll-out only impacts clients who use public resolvers, we identified such clients using our client-LDNS pairing data described in Section 3.1 and extracted RUM data from only those qualified clients. Figure 12 shows the total number of qualified RUM measurements collected and used in our analysis from both high and low expectation countries. Our data set has 33 million to 58 million measurements per month, each month from Jan to June 2014, for a total of 273 million measurements. The measurement volume shows an increasing trend on account of more downloads from qualified clients of the pages measured by RUM.

Our goal is to measure performance for a large and char-

acteristic cross section of clients, Web sites, devices, and connectivities across the global Internet. To achieve that we measured 6,388 domain names and 2.5 million unique URLs accessed by 149,826 unique clients. Our data set includes all major client platforms such as Windows, FreeBSD, Linux, Android, iOS, and game consoles, and all major browsers including Firefox, Opera, Chrome, and IE. Further, our clients use a variety of ways to access the Internet including cellular, WiFi, 3G, 4G, DSL, cable modem, and fiber.

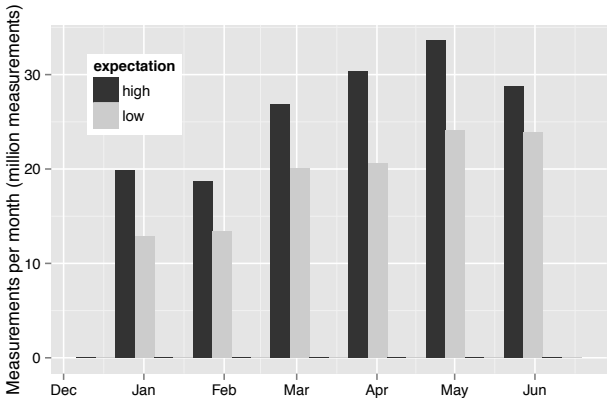


Figure 12: Number of RUM measurements per month.

4.3 Performance Analysis

We analyze the mapping distance, RTT, TTFB, and content download time for clients who use public resolvers before, during, and after the roll-out.

1) Mapping distance. Mapping distance shows a significant improvement during the roll-out period of March 28th to April 15th. Figure 13 shows for the high expectation group, the mean mapping distance dropped from over 2000 miles on average to around 250 miles. Even the low expectation countries experienced shorter mapping distance: the average mapping distance went from 400 miles to 200 miles.

Figure 14 shows the CDF of the mapping distances for both high and low expectation countries both before and after the roll-out is completed. The period after the roll-out is April 15th or later and the period before the roll-out is March 28th or earlier. Note that all percentiles see improvement. But, there is a drastic decrease in the mapping distance around the 90^{th} percentile for high expectation countries from 4573 miles to 936 miles. The decrease is due to improved mapping distance for clients in large countries like India and Brazil who use public resolvers located in Southeast Asia and North America respectively (cf. Figure 8).

2) RTT. Recall that RTT measures the latency between the client and the server assigned to that client. Unlike mapping distance, RTT reflects the state of the network path such as propagation delay, and congestion. As shown in Figure 15, the average RTT for the high expectation group dropped from 200ms to 100ms, a significant 50% decrease. But, the improvement for the low expectation group was modest. Figure 16 shows the CDF of the RTT for both high and low expectation countries before and after the roll-out. All

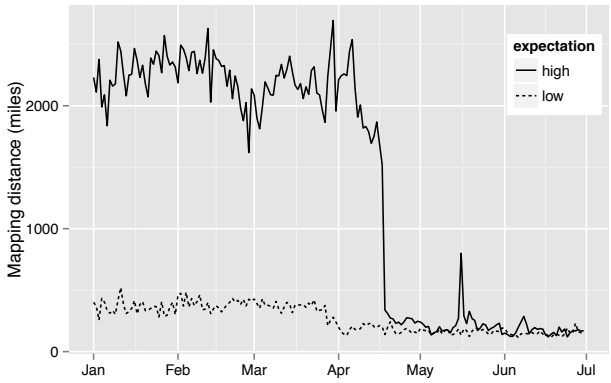


Figure 13: Daily mean of mapping distance.

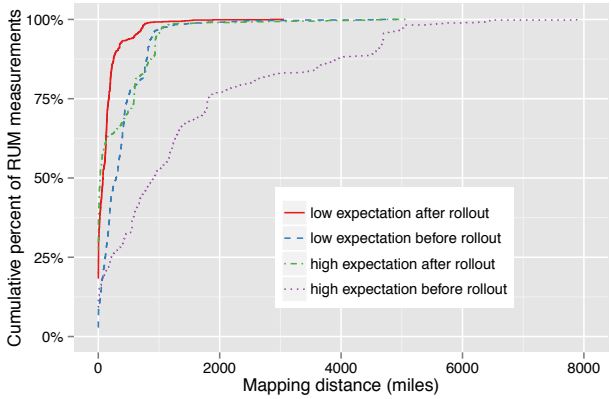


Figure 14: CDFs of mapping distance.

percentiles show improvement. For instance, the 75^{th} percentile of the RTT decreases significantly from 220 ms to 137 ms for the high expectation countries.

3) Time-to-First-Byte. As noted earlier, TTFB includes aspects that are not impacted by better mapping decisions, such as the computation time to generate and transmit a dynamic web page. Consequently, the gains expressed as a percentage are lower but still significant. Figure 17 shows that the mean TTFB of the high expectation countries decreased from around 1000 ms to 700 ms, a 30% improvement. Figure 18 shows the CDF of the TTFB for both high and low expectation countries before and after the roll-out. All percentiles show improvement. For instance, the 75^{th} percentile of the TTFB decreases from 1399 ms to 1072 ms for the high expectation countries and from 830 ms to 667 ms for the low expectation ones.

4) Content Download Time. Figure 19 shows a reduction from 300 ms to 150 ms for the high expectation countries, a 50% reduction. Recall that content download time is dominated by server-client latencies and the decrease is more correlated with corresponding decrease in RTT. The improvement for the low expectation group is small as the download time is already small. Figure 20 shows the CDF of the content download time for high and low expectation countries before and after the roll-out. All percentiles show improvement, e.g., the 75^{th} percentile of the download time reduces

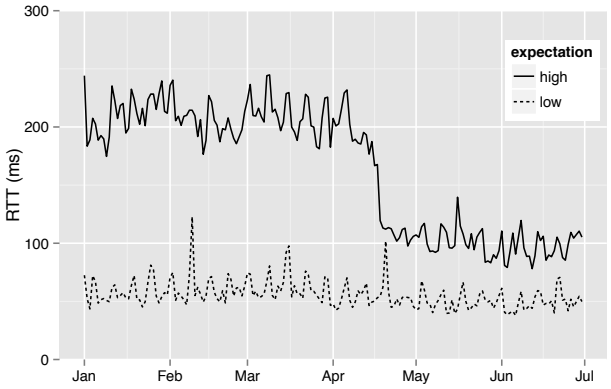


Figure 15: Daily Mean of Round Trip Time (RTT).

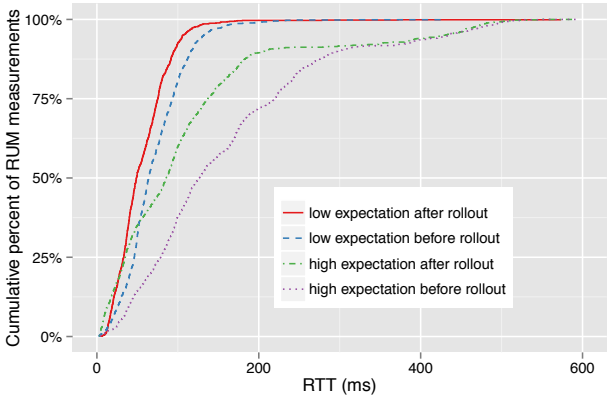


Figure 16: CDFs of Round Trip Time (RTT).

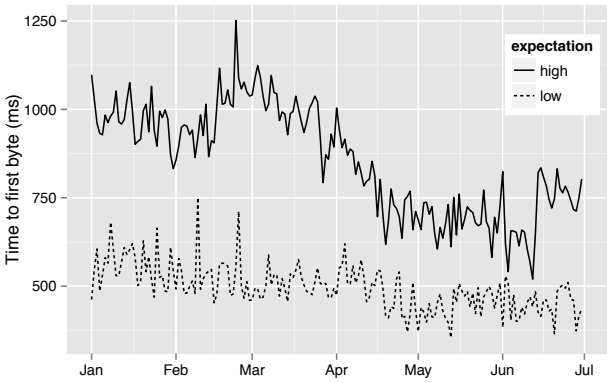


Figure 17: Daily Mean of Time to First Byte (TTFB).

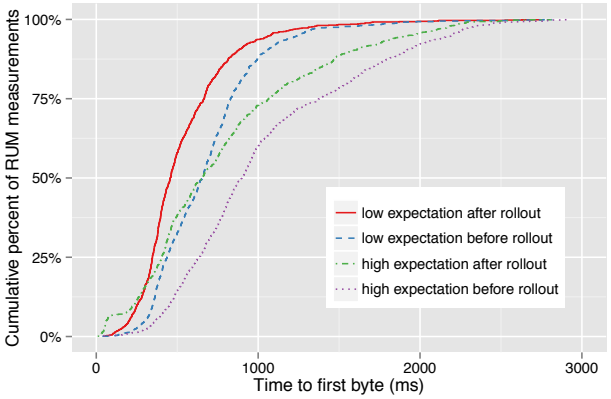


Figure 18: CDFs of Time to First Byte (TTFB).

from 272 ms to 157 ms for the high expectation group and from 192 ms to 102 ms for the low expectation one.

4.4 Why Download Performance Matters

From our results above, we can conclude that end-user mapping provides significant performance benefits to clients who use public resolvers, especially in those countries where client-LDNS distances are high. Faster download times such as those provided by end-user mapping are key to a better Internet experience, resulting in web pages that load more quickly and videos that start playing sooner. Better download performance enhances the client's experience of a content provider's Web site, more satisfied clients in turn favorably impact the business of the content provider, allowing the content provider to invest in even greater performance enhancements, forming a "virtuous cycle" [25]. As an example, an oft-cited recent study by Walmart labs [12] concluded that the download time of Web pages in Walmart's e-commerce site impacts the buying behavior of its users. By correlating RUM performance measurements collected for Walmart.com with back-end business metrics, the study concluded that a 100 ms decrease in web page download time can result in a 1% increase in revenue and a 1 second decrease can result in up to a 2% increase in conversion rates[9]. Numerous other studies show how even a few 100 ms increase in page download times of a Web site can decrease revenues, page views, searches per user, etc [9]. In fact, it is widely held in industry that a Web site that is faster than its competing sites by as little as 250 ms has a significant business advantage to be reckoned with [18]. In addition, search engines rank faster Web sites ahead of slower ones and clients often associate greater brand reputation with faster Web sites. Thus, the "need for speed" is a singular focus for content providers and the CDNs alike and "shaving off" even tens of milliseconds of Web download times for a cross-section of clients is deemed worthy and important. Besides faster download times, the decrease in mapping distance and RTT due to end-user mapping often means that the client-server path crosses fewer AS boundaries, peering points and transnational cable links, hence reducing the likelihood of congestion and failure. Thus, end-user mapping may result in more stable and reliable client-server paths.

4.5 The Benefits of EDNS0 Adoption

To deploy end-user mapping beyond the current set of clients, the client's ISP needs to adopt the EDNS0 exten-

[9]Conversion rate is a key metric for e-commerce sites and is the percentage of visitors to the site who buy a product.

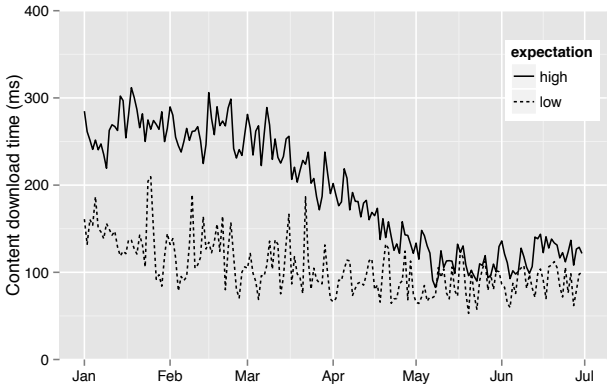


Figure 19: Daily mean of content download time.

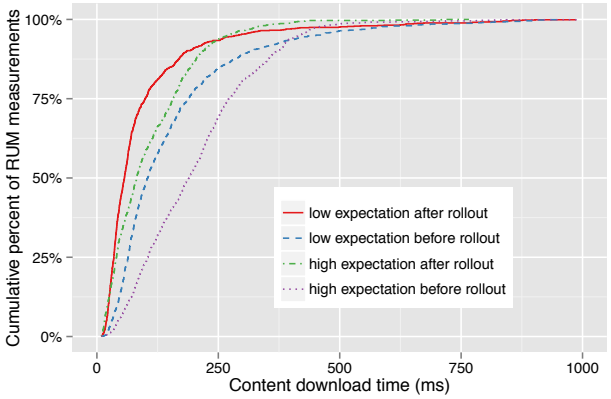


Figure 20: CDFs of content download time.

sion for their DNS services. Our results shed light on the performance benefits such adoption would yield and provides a strong impetus for its adoption. For instance, excluding the public resolvers, we know that 6.2% of the remaining client demand originates from clients whose LDNS are at least 1000 miles away. Extrapolating from our results for similar clients using public resolvers, we could expect a similar 50% reduction in RTT and content download time for these clients. Likewise, excluding the public resolvers, clients with LDNSes between 500 to 1000 miles account for 5.3% of remaining client demand. Extrapolating from similar clients who use public resolvers, we can speculate that these clients will see a 24% decrease in RTTs and content download times. Of course, 54% of the remaining client demand will see no benefit at all from end-user mapping, since they have local LDNSes. However, the fact that at least 11.5% of the remaining client demand will see a significant enough performance improvement is sufficient impetus to EDNS0 adoption.

5. SCALING CHALLENGES

End-user mapping is challenging since it makes mapping decisions at potentially a much finer granularity than traditional NS-based mapping. There are orders of magnitude more clients than there are name servers on the global Internet. An end-user mapping system must perform more fine-grain network measurements and provide resolutions at a finer scale across the global Internet than a NS-based mapping, leading to scaling considerations discussed below.

5.1 Tradeoffs in choosing the mapping units

A mapping unit is the finest-grain set of client IPs for which server assignment decisions are made by the mapping system. A traditional NS-based mapping system uses a LDNS as the mapping unit, i.e., all clients in the client cluster that use a LDNS are mapped together as a unit. An end-user mapping system could use /x client IP blocks that partition the client IP space, where $x \leq 24$. A natural first choice is /24 client IP blocks since LDNSes that support the EDNS0 extension currently use /24 IP blocks in their queries.

To understand the scaling issues in switching from NS-based to end-user mapping, let us first examine the number of relevant /24 client IP blocks on the Internet in comparison to the number of relevant LDNSes. We use our NetSession data to first compute the demand generated by clients in each /24 client IP block. We also computed the demand generated by each LDNS, where the LDNS demand is simply the demand generated by clients who use that LDNS. We then sorted all the /24 client IP blocks (resp., LDNSes) in decreasing order of demand and plotted a CDF of the demand in Figure 21. In the data set, the total number of /24 client IP blocks with non-zero demand is 3.76 million, while 584 thousand LDNSes have non-zero demand. Suppose the mapping system is required to measure and provide mapping decisions for 95% of the total client demand on the Internet. As the figure shows, an NS-based mapping system need only measure and analyze the top 25,000 LDNSes with the most demand, whereas an end-user mapping system must measure and analyze the top 2.2 million /24 client IP blocks, which is several magnitudes higher. Likewise, to cover 50% of the total client demand, the top 1800 LDNSes with the most demand suffice, whereas nearly 430,000 of the /24 client IP blocks with the most demand are needed.

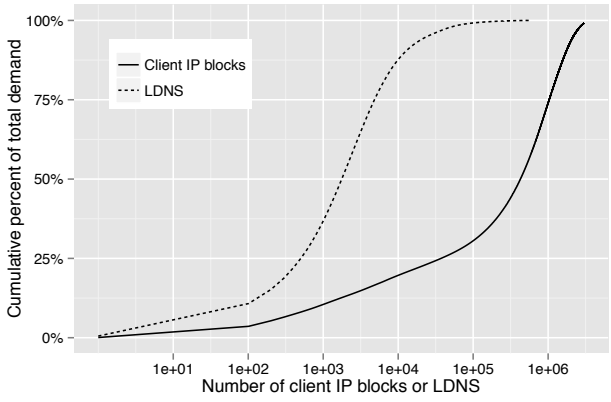
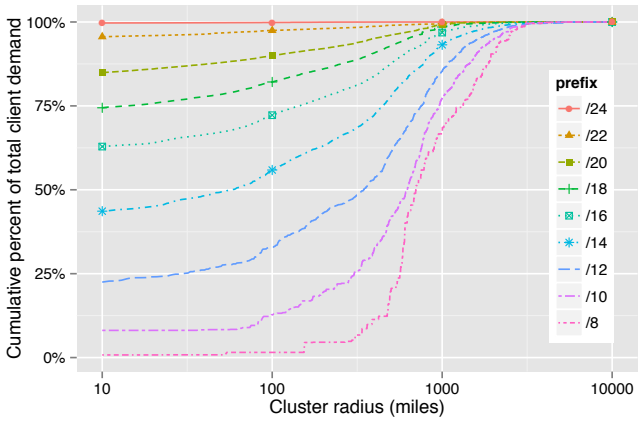
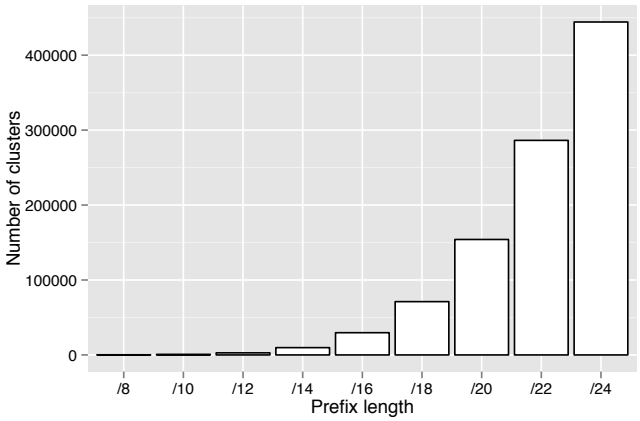


Figure 21: Number of /24 client IP blocks or LDNSes that produce a given percent of total global demand.

One heuristic approach to reducing the number of map-



(a) Histogram of the cluster radius for /x client IP blocks.



(b) Number of /x client IP blocks with non-zero demand.

Figure 22: A smaller value of x yields fewer mapping units but larger cluster radius with less mapping accuracy.

ping units for end-user mapping is to use the IP blocks (i.e., CIDRs) in BGP feeds that are the units for routing in the Internet. In particular, if a set of /24 IP blocks belong within the same BGP CIDR, these blocks can be combined since they are likely proximal in the network sense. We extracted 517K unique CIDRs with non-zero traffic from BGP feeds across the Internet from the network measurement component of the mapping system. By combining /24 IP blocks whenever they belong to the same BGP CIDR, we reduce the number of mapping units from 3.76 million to 444K. Note that the same technique may be applied to reduce the number of mapping units for /x IP blocks, for any value of x.

After applying the BGP CIDRs to reduce the number of mapping units, there is still the tradeoff of what /x client IP blocks to choose as the mapping unit. One could reduce the number of mapping units by using coarser /x client IP blocks, i.e., by choosing a smaller value of x. However, when coarser IP blocks are used, the set of clients in a given block is larger and span a larger geographical area. This reduces mapping accuracy as the client clusters that are the units of mapping have a larger radius. Figure 22 provides the exact tradeoff between the cluster radius which is a proxy for mapping accuracy and the number of clusters that need to be measured and analyzed. It can be seen that /20 client IP blocks are a worthy option as they reduce the number of mapping units by a factor of 3 in comparison to /24 blocks. However, the clusters are still relatively small with 87.3% of the clusters having a radius of no more than 100 miles.

5.2 Dealing with greater DNS query rates

In NS-based mapping, each LDNS stores one resolution per domain name. However, with end-user mapping, different client IP blocks within the *same client cluster* may get different resolutions for the same domain name. Thus, an LDNS that serves multiple client IP blocks may store multiple entries for the same domain name. Therefore, an LDNS may make multiple requests to an authoritative name server for the domain name, one for each client IP block. This can lead to a sharp increase in the LDNS queries seen by the authoritative name servers of the mapping system. Figure 23 shows the total DNS queries per second served by the mapping system before, during, and after enabling end-user mapping for clients who use public resolvers. Prior to the roll-out, the total queries per second served by Akamai's name servers was 870K queries per second of which public resolvers targeted by the roll-out accounted for roughly 33.5K queries per second. But after the rollout, the total queries per second on the Akamai network was 1.17 million queries per second of which public resolvers accounted for 270K queries per second. Thus, the queries from public resolvers increased by a factor of 270K/33.5K = 8, an increase largely attributable to the roll-out[10]. The gradual increase in query rate seen outside of the roll-out window is simply due to the normal increase in Internet traffic over time.

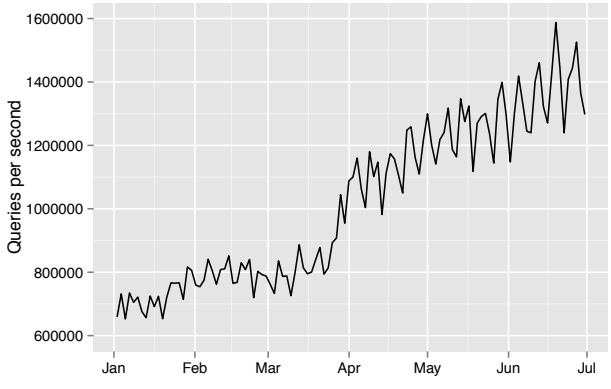


Figure 23: DNS queries received by Akamai's name servers from LDNSes showed a significant increase during the end-user mapping rollout.

[10]DNS queries increase when public resolvers turn on the EDNS0 extension. But, the performance improvements in Section 4.3 occur when Akamai gradually turned on end-user mapping for these public resolvers.

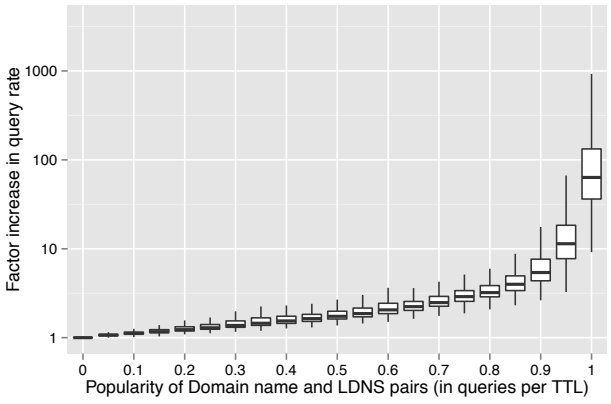


Figure 24: More popular domain name and LDNS pairs show a greater increase in query rate after the roll-out.

The popularity of a domain name among the clients of an LDNS influences the factor increase in DNS queries for that domain name when EDNS0 and end-user mapping are turned on. Prior to the end-user mapping roll-out, the query rate for a domain name from a particular LDNS is at most one query per TTL, since the LDNS can cache the translation for the time of the TTL. We bucket each domain name and LDNS pair according to the number of queries received per TTL *prior* to the roll-out. Figure 24 shows the factor increase in query rate for domain name and LDNS pairs that fall into each bucket. Note that the more popular domain name and LDNS pairs that have pre-roll-out query rates close to 1 query per TTL saw the largest increase in query rate when end-user mapping was rolled out, while less popular domains saw little or no increase. The reason is that a more popular domain name is more likely to be accessed by clients in multiple client IP blocks of the LDNS's client cluster, each IP block requiring a separate domain name resolution when EDNS0 is used. Fortunately, the domain name and LDNS pairs in the highest popularity bucket in Figure 24 accounted for only 11% of total pre-roll-out queries.

6. ROLE OF SERVER DEPLOYMENTS

Server deployments play an important role in determining client performance. More server deployment locations mean better performance for clients, since the mapping system has more options to choose a proximal server for each client. But, what role do deployments play in determining the additional performance benefits provided by end-user mapping over NS-based mapping? Should a CDN with a small number of deployment locations adopt end-user mapping? For a CDN with a given set of deployment locations, what is more beneficial: adding more deployment locations or incorporating end-user mapping? How much can NS-based mapping be improved by making it client-aware?

To provide intuition on these key what-if questions, consider a simplified model. Let a CDN have N deployment locations. The deployments partition the IP address space of the global Internet into sets $E_i, 1 \leq i \leq N$, such that E_i is the set of IPs for whom the i^{th} deployment location is the most proximal among all deployments. Observe that for any client c, if c and its LDNS are *both* in some set E_i, both end-user mapping and traditional NS-based mapping will pick a server in the i^{th} deployment location for client c, i.e., there is no additional benefit for client c from using end-user mapping. Thus, if a CDN has fewer deployments, each set E_i is likely larger and is hence more likely to contain both the client and its LDNS. Thus, we would expect a *a CDN with fewer deployments to benefit less from end-user mapping than a CDN with more deployments.* We quantify answers to this and other key questions using simulations.

Simulation Methodology. We create a universe U of possible deployment locations by using 2642 different locations around the globe with Akamai servers. These deployments are spread over 100 countries and were chosen to provide good coverage of the global Internet. Next, we choose around 20K /24 IP blocks that account for most of the load on the Internet and further cluster them into 8K "ping targets", so as to cover all major geographical areas and networks around the world. We then perform latency measurements using `pings` from each deployment U to each of the 8K ping targets. For any client or LDNS, we find the closest of the 8K ping targets and use that as a proxy for latency measurements, i.e., the latency measurements to the ping target are assumed to be the latency measurements to the client or LDNS. Using the ping latency measurements described above, we simulate three mapping schemes, each with a varying number of deployment locations.

(1) NS-based mapping (NS): Map client to the deployment location that has the least latency to the LDNS of that client.

(2) End-user mapping (EU): Map client to the deployment location that has the least latency to the client's /24 IP block.

(3) Client-Aware NS-based Mapping (CANS): For each client, find the cluster of clients that shares its LDNS. Map client to the deployment location that minimizes the traffic-weighted average of the latencies from the deployment to its cluster of clients.

Note that CANS mapping is an enhancement of pure NS mapping by using the latency measurements to the clients of the LDNS, rather than just the latency measurement to the LDNS. In situations where LDNS is far away from its clients, but its clients are themselves relatively close together, CANS mapping could provide low latency mappings. CANs requires tracking client-LDNS associations on an ongoing basis on the global Internet, an additional complexity in comparison with NS mapping. However, CANS can be viewed as a hybrid between NS and EU that uses client measurements but requires no specific knowledge about the client's IP, i.e., it does not require the EDNS0 protocol extension.

We simulated the three mapping schemes above for a varying number of deployment locations N chosen from the universe U. We performed 100 random runs of our simulation, where we do the following in each run. We randomly order the deployments in U. Then, for each N, we simulate all three mapping schemes assuming the first N deployments in the random ordering. The simulation computes the traffic-weighted mean, 95^{th}, and 99^{th} percentile latencies achieved by the three schemes. Finally, for each value of N, we av-

eraged the metrics obtained across the 100 simulation runs and those values are reported in Figure 25.

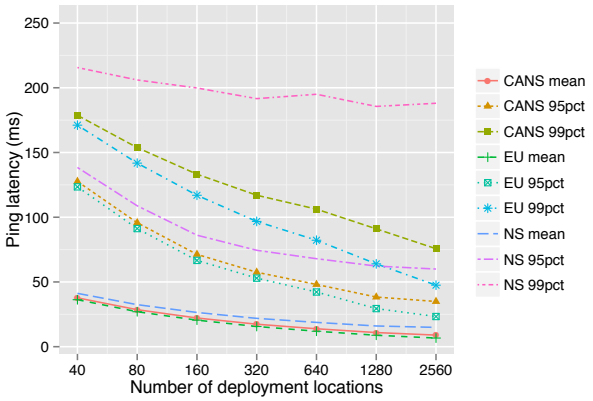


Figure 25: Latencies achieved by EU, CANS, and NS mapping as a function of CDN deployment locations.

An important caveat in interpreting Figure 25 is that the *ping latencies shown are an underestimate of the actual latency or RTT from the server to the client*, since only a ping target (typically a router) enroute to the client is "pinged". So, while the absolute values of the ping latencies are less meaningful except as a lower bound on the actual latencies, the relative values are still meaningful. As shown in the figure, all mapping schemes provide smaller ping latencies with a larger deployment. Further, mean ping latency is nearly identical for all three mapping schemes, reflecting the fact that in many cases a client and its LDNS are proximal to each other and LDNS is a good proxy for the client. Even so, EU performed the best of the three with mean ping latency dropping from 35 ms for a small deployment to under 10 ms as the deployments increase.

However, *mean* latency across all clients on the globe is less interesting than latency of the *worst-performing* clients. In fact, both CDNs and content providers are focused on improving the performance of the worst-performing client. Thus, we computed the 95^{th} and 99^{th} percentiles of the latencies, i.e, latencies for 1-5% of the worst clients. It is clear that EU provides a large benefit over the other schemes for higher percentiles of ping latency. In particular, NS-based mapping provides diminishing benefits beyond 160 deployment locations for the 99^{th} percentile latency, and is in particular unable to reduce it below 186 ms even with 1280 deployment locations. The reason is that NS-based mapping does not work well for clients whose LDNSes are not proximal who are likely among the worst-performing clients. However, EU continues to reduce the latencies with increasing deployments, even beyond 1280 deployments. It can also be seen in the figure that a CDN with larger deployment locations sees a proportionally larger reduction in higher percentiles of ping latency by switching to EU from NS than a CDN with smaller deployments. CAN mapping provides an intermediate point between the extremes of NS and EUM. In particular, the knowledge of latencies to clients behind a given LDNS provides sufficient knowledge to improve NS-based mapping for higher percentiles for latency.

7. RELATED WORK

While the EDSN0 extension provides a systematic mechanism for end-user mapping implementation, other mechanisms have been explored in limited ways in industry. A video CDN at Akamai in circa 2000 used metafile redirection to implement end-user mapping. When a client starts a video, the media player fetches a metafile that contains the server's IP from which to download the video. The server IP embedded in the metafile is dynamically generated by the mapping system using the client's IP derived from the metafile download. However, such a mechanism is hard to extend to the Web and other traffic that do not use metafiles.

Analogous to metafile redirection, systems that use http redirection have also been built where the client is first assigned a server using NS-based mapping. The first server uses its knowledge of the client's IP to redirect the client to a "better" second server if appropriate. The second server then serves the content to the client. However, this process incurs a redirection penalty that is acceptable only for larger downloads such as media files and software downloads.

Tools for discovering client-LDNS pairings have also existed in industry for the past 15 years. In principle, such pairings can be used to create a client-aware NS-based mapping system (cf. Section 6), though it will not be effective for LDNSes with large client clusters (cf. Section 3.3).

We think that the EDNS0 extension is key to building large-scale end-user mapping that overcomes the short comings of prior implementations. The EDNS0 extension removes the overhead of explicit client-LDNS discovery, avoids a redirection performance penalty, and is effective even for LDNSes with large geo-distributed client clusters.

From a research perspective, client-LDNS distances and their potential impact on server selection has been studied in [24], and subsequently in [20, 17]. The prior literature observed larger client-LDNS distances and poorer performance for clients using public resolvers that are increasingly in use [22]. Our measurement study of client-LDNS distances in Section 3 is based on a much wider global cross-section of clients and LDNSes than prior work and largely confirm prior conclusions on public resolvers. However, we go a step further by describing an end-user mapping system to remedy the issue. The EDNS0 extension has also been studied as tool for figuring out deployments of CDN providers who support the extension such as Google [10, 27]. Extensions other than EDSN0 for overcoming client-LDNS mismatch have also been proposed [16].

8. CONCLUSION

In this paper, we described our experience in rolling-out a new mapping system called end-user mapping. By analyzing clients and LDNSes from around the world, we showed that a significant fraction of clients have LDNSes that are not in their proximity and could benefit from end-user mapping. We confirmed the performance benefits by measuring map-

ping distance, RTT, Time-To-First-Byte (TTFB), and content download time during the roll-out. We showed that for "high-expectation" countries, clients using public resolvers saw an eight-fold decrease in mean mapping distance, a two-fold decrease in RTT and content download time, and a 30% improvement in the TTFB. We also quantified the scaling challenges in implementing end-user mapping such as the 8-fold increase in DNS queries and the greater number of mapping units that need to be measured and analyzed. Finally, we shed light on the role of deployments and showed that a CDN with a larger number of deployment locations is likely to benefit more from end-user mapping than a CDN with a smaller number. While we only describe the roll-out of end-user mapping to clients who are using public resolvers, our analysis shows that a broad roll-out of this technology across the entire Internet population will be quite beneficial. For such a roll-out to occur, more ISPs would need to support the EDNS0 extension. We expect our work that quantifies the real-world benefits of end-user mapping to provide impetus to a broader adoption of the EDNS0 extension.

9. ACKNOWLEDGEMENTS

First and foremost, we thank the many engineers at Akamai who designed, implemented and rolled-out end-user mapping, making it possible for us to evaluate its impact. Special thanks to Mike Conlen who helped collect DNS query data, to Pablo Alvarez who made key contributions to end-user mapping scoring, and to Jason Moreau who made major contributions to name server design. We thank our anonymous referees for copious reviews that helped improve the paper. A special thanks to our shepherd Ethan Katz-Bassett who provided lots of great feedback that strengthened the paper.

10. REFERENCES

[1] Akamai Edgescape. http://goo.gl/P68U6q.
[2] Akamai Facts & Figures. http://goo.gl/Megx1b.
[3] Akamai NetSession Interface. http://goo.gl/FOtjlz.
[4] Akamai Real User Monitoring. http://goo.gl/8oiQyC.
[5] Google Public DNS. https://goo.gl/p8cfJm.
[6] Navigation Timing. http://goo.gl/ePcQrG.
[7] OpenDNS. https://www.opendns.com/.
[8] Resource Timing. http://goo.gl/5eYQtL.
[9] Velocity and the bottom line. http://goo.gl/KTlcYR.
[10] M. Calder, X. Fan, Z. Hu, E. Katz-Bassett, J. Heidemann, and R. Govindan. Mapping the expansion of Google's serving infrastructure. In *Proceedings of the ACM Internet Measurement Conference*, pages 313–326, 2013.
[11] C. Contavalli, W. van der Gaast, D. Lawrence, and W. Kumari. Client subnet in DNS requests. *IETF Internet Draft*, Nov. 2014.
[12] C. Crocker, A. Kulick, and B. Ram. Real user monitoring at walmart.com: A story in three parts. In *San Francisco and Silicon Valley Web Performance Group*, Feb 2012. http://minus.com/msM8y8nyh.
[13] J. Dilley, B. M. Maggs, J. Parikh, H. Prokop, R. K. Sitaraman, and W. E. Weihl. Globally distributed content delivery. *IEEE Internet Computing*, 6(5):50–58, 2002.
[14] X. Fan, J. Heidemann, and R. Govindan. Evaluating anycast in the domain name system. In *Proceedings of the IEEE INFOCOM*, pages 1681–1689, 2013.
[15] T. Hardie. Distributing authoritative name servers via shared unicast addresses. *RFC 3258*, Apr. 2002.
[16] C. Huang, I. Batanov, and J. Li. A practical solution to the client-LDNS mismatch problem. *SIGCOMM Comput. Commun. Rev.*, 42(2):35–41, Mar. 2012.
[17] C. Huang, D. A. Maltz, J. Li, and A. Greenberg. Public DNS system and global traffic management. In *Proceedings of the IEEE INFOCOM*, pages 2615–2623, 2011.
[18] S. Lohr. For impatient web users, an eye blink is just too long to wait. *New York Times*, Feb 2012. http://goo.gl/y70JgH.
[19] B. M. Maggs and R. K. Sitaraman. Algorithmic nuggets in content delivery. *SIGCOMM Comput. Commun. Rev.*, July 2015.
[20] Z. M. Mao, C. D. Cranor, F. Douglis, M. Rabinovich, O. Spatscheck, and J. Wang. A precise and efficient evaluation of the proximity between Web clients and their local DNS servers. In *USENIX Annual Technical Conference, General Track*, pages 229–242, 2002.
[21] E. Nygren, R. Sitaraman, and J. Sun. The Akamai Network: A platform for high-performance Internet applications. *ACM SIGOPS Operating Systems Review*, 44(3):2–19, 2010.
[22] J. S. Otto, M. A. Sánchez, J. P. Rula, and F. E. Bustamante. Content delivery and the natural evolution of DNS: remote DNS trends, performance issues and alternative solutions. In *Proceedings of the ACM Internet Measurement Conference*, pages 523–536, 2012.
[23] S. Sarat, V. Pappas, and A. Terzis. On the use of anycast in DNS. In *Proceedings of the IEEE ICCCN*, pages 71–78, 2006.
[24] A. Shaikh, R. Tewari, and M. Agrawal. On the effectiveness of DNS-based server selection. In *Proceedings of the IEEE INFOCOM*, volume 3, pages 1801–1810, 2001.
[25] R. K. Sitaraman. Network performance: Does it really matter to users and by how much? In *Fifth International Conference on Communication Systems and Networks (COMSNETS)*, pages 1–10. IEEE, 2013.
[26] R. K. Sitaraman, M. Kasbekar, W. Lichtenstein, and M. Jain. Overlay networks: An Akamai perspective. In *Advanced Content Delivery, Streaming, and Cloud Services*. John Wiley & Sons, 2014.
[27] F. Streibelt, J. Böttger, N. Chatzis, G. Smaragdakis, and A. Feldmann. Exploring EDNS-client-subnet adopters in your free time. In *Proceedings of the ACM Internet Measurement Conference*, 2013.