On the problem of optimization of DNS root servers' placement

Tony Lee, Bradley Huffaker, Marina Fomenkov, kc claffy

Abstract—

The Domain Name System (DNS) is a critical component of the modern Internet. It provides a critical link between human users and Internet routing infrastructure by mapping host names to IP addresses. The DNS is a hierarchy of distributed system of servers anchored at 13 DNS root servers.

In this paper we examine the macroscopic connectivity between the DNS root servers and the worldwide population of their clients. We study the impact of the geographical locations of root servers on the latency of server-client connections. We also propose a methodology to estimate the effects of root servers' relocation.

We found that all root servers can be clustered in four groups that closely correlate with their geographical positions. Servers in the same group are nearly indistinguishable for their clients in terms of latency and can replace one another in providing DNS services to the clients. M-root, the only root server in Asia, is in a group of its own and, therefore, is the most crucial for its clients in terms of the latency increase in case of its unavailability. Clients in Europe appear to be relatively underprovisioned and may merit an additional root server. Clients in North America appear overprovisioned. One of the US servers may be a suitable candidate for relocation to a different region of the world.

Keywords—**DNS RTT root server placement**

I. Introduction

A. The Domain Name System

The Domain Name System is a fundamental and indispensable component of the modern Internet [1]. In essence, the DNS is a globally distributed and decentralized database of network identifiers. Its most common use by far is to *resolve* host names into Internet addresses. This mapping occurs continually, for example, every time a user visits a web page, sends an email message, or uses an instant messaging service. The DNS also serves a number of other purposes, including reverse mapping addresses to names, locating mail exchange servers, and several other albeit less common functions.

The DNS is based on a classic client-server scheme [2]. Programs called *name servers* constitute the server half of the mechanism; each name server is responsible (*authoritative*) for its own piece of the database. Clients (*resolvers*) create queries and send them across the network to name servers. In most cases, network applications such as web browsers and mail transfer agents have integral DNS resolver clients. DNS servers, on the other hand, are typically dedicated applications.

One of the most important properties of the DNS is its use of hierarchical namespaces. This hierarchy is manifest through the standard "dot" notation used in web site and domain names. For example, in order to reach a machine with the name "not.invisible.net", one must send a query to the DNS server responsible for machines and/or subdomains in the domain ".invisible.net." The authoritative machine for ".invisible.net" will be looked up by sending a query to the server authoritatively responsible for ".net". Such a server is called a *global top-level domain* (gTLD) server. Information on the appropriate gTLD server can be obtained from one of the *root* servers. Currently there are 11 gTLD servers and 13 root servers.

The recursive process of name resolution is transparent to an end user but may contribute significantly to the overall delay of establishing a connection [3]. The root servers experience heavy load because they are the starting points for DNS clients (applications) when resolving host names. A typical root server receives between 5000 and 8000 queries per second; this load appears to grow linearly in proportion to the number of registered domain names [4]. Clearly, proper, secure and efficient operation of the root servers is crucial for functioning of the Internet.

The most popular DNS implementation in use today on Unix systems is the Berkeley Internet Name Domain (BIND) software [5].[1] Other implementations of the DNS

CAIDA, San Diego Supercomputer Center, University of California, San Diego. E-mail: {tlee,bhuffake,marina,kc}@caida.org.

Support for this work is provided by WIDE (Jun Murai), NSF ANIR NCR-9711092 and DARPA NMS N66001-01-1-8909.

[1]There are several versions of BIND that propagate in the Internet infrastructure. Some versions are dramatically different, being a com-

specifications are also available, including *djbdns* [6], and Microsoft's DNS software bundled with its Windows operating systems [7].

Figure 1 shows the locations of existing root servers around the world. The geographical distribution is highly uneven, with six root servers on the US East coast, four on the US West coast, two in Europe, and one in Japan. Each DNS root server is administered independently by a separate organization and uses diverse types of hardware and operating systems.

Numerous organizations are interested in hosting a root name server since it brings prestige and, to some extent, control. The Root Server Selection Advisory Committee (RSSAC) is a technical advisory committee for the Internet Corporation for Assigned Names and Numbers (ICANN). One of its responsibilities is to advise ICANN on the placement of future root name servers. On behalf of RSSAC, CAIDA is gathering measurement data to help determine architecturally strategic locations and to provide unbiased recommendations on optimal servers' placement [8].

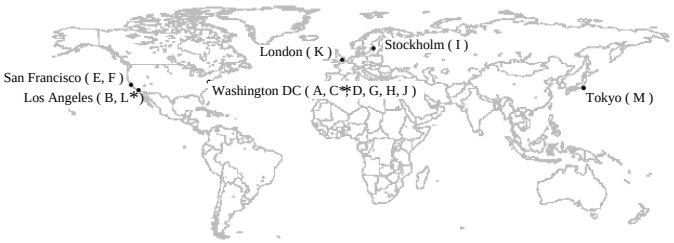

Fig. 1. The geographic locations of DNS root servers. Servers marked with '*' currently do not have co-located CAIDA *skitter* monitors.

B. Related work

The indispensable role of the DNS in Internet functioning and its unparalleled scale prompted multiple studies of DNS performance *per se* [9], [10],[11],[12] and of its contribution to overall web performance [13],[3],[14]. In these studies, measurements are usually taken at a limited number of locations in the Internet topology and analysis is focused on effects of errors in DNS implementations and on caching. In an ongoing project, Cho, *et al.* [15] monitor the DNS root name server performance from various parts of the Internet by active probing. They seek to develop technical methods for assessing the root name server system performance and for planning its future reconfigurations.

Liston, *et al.* [16] identified various DNS performance metrics (completion and success rates of resolving names, the mean response time for completed lookups, the root and gTLD servers that are favored by the sites, the distribution of TTLs across names), and studied location-related variations of these metrics. The measurements were obtained from 75 different Internet locations in 21 countries. Liston, *et al.* conclude that the greatest performance enhancements can be achieved by reducing the response time of intermediate-level servers rather than the top-level root and gTLD servers. They state, however, that a more equitable choice of placement of the gTLD servers in particular has the potential to signficantly affect user-perceived performance. Note that although the results presented in our paper deal with the placement of the root servers, our measurements and approach can be expanded to evaluate the gTLD servers as well.

Other studies have considered the DNS in conjunction with the more general problem of nearest server selection. Shaikh, *et al.* [17] evaluated the effectiveness of DNS-based server selection. They found that DNS-based schemes typically disable client-side caching of name resolution results. The negative consequences of this policy are two-fold: a) considerable increase of name resolution overhead for the client, especially when the number of embedded objects, e.g., images and advertisements, served from multiple sources increases; b) growth of the number of queries to authoritative DNS servers and the network traffic incurred by these queries. Shaikh, *et al.* propose modifications to the DNS protocol to improve the accuracy of the DNS-based server selection technique.

Somegawa, *et al.* [18] examined server selection mechanisms employed by different DNS implementations (reciprocal algorithm in BIND-8, best server in BIND-9, uniform algorithm in *djbdns* and Windows 2000) [2] as a case study for the general problem of best server placement and selection. They used data collected by Cho, *et al.* [15] and simulated effects of different server selection mechanisms. Somegawa, *et al.* found that the reciprocal algorithm is more suitable for the Internet environment than the other two currently implemented algorithms. They also showed that the proper use of server selection algorithms is essential for the stability of the DNS service.

II. Methodology

A. Data collection

For this study we use the data obtained as part of CAIDA's macroscopic topology probing project [19].

plete rewrite from scratch.

[2]The reciprocal algorithm selects a server with a probability reciprocal to a certain metric. The best server algorithm chooses a server with the best metric. The uniform algorithm selects each server with uniform probability.

CAIDA has deployed its topology probing tool *skitter* [20] at hosts co-located with the DNS root servers. At the time of this study we have instrumented 11 out of 13 root servers; at this time A and J roots were co-located (they no longer are) and so shared a monitor. The administration of the C root server has not responded to our request to host a *skitter* monitor at their site. The monitor deployed at the L root server has had intermittent troubles.

The *skitter* tool actively measures connectivity and performance of the network between the monitor host and a pre-determined target list of destinations. It iteratively sends 52-byte ICMP echo request packets, incrementally increasing their time-to-live values until a packet reaches the target host. Each trace is a record of the IP addresses of responding intermediate routers on the forward path from the source to the target destination, as well as the round-trip time (RTT) to the destination. Such measurements, typically made from 1 to 15 times daily (the frequency depending primarily on the size of the list), characterize macroscopic connectivity between the topology monitor and the destination hosts on its probe list.

B. Target list

In order to study the global connectivity of the root servers to their clients, we needed a representative target list. Ideally we would like to monitor a destination in each /24 prefix, but this is impossible. We attempted to find a destination in each globally routable prefix[3] from a vast pool of IP addresses sending messages to the DNS root name servers. We also restricted the size of the target list to between 100 and 200 thousand addresses. The size restriction ensures that a typical topology monitor polls each destinations at least 3-5 times in a 24 hour period thus making RTT measurements less sensitive to diurnal variations, but avoiding over-sampling.

We used the CAIDA *dnsstat* utility [22] to passively monitor DNS queries at the A, D, E, F, H, I, K, and M root servers. On each root server, numbers of messages and number of queries (but not the subjects of queries) were counted for 24 hours and recorded together with source IP addresses originating these messages. These aggregated statistics yielded nearly 2 million client addresses representing, however, only about 52K routable prefixes out of 118K prefixes in the BGP table from March 18, 2002. Therefore we could afford to increase the coverage of large prefixes still within the optimal list size. In order to add destinations uniformly across the IPv4 space, we started by splitting each /8 prefix into two /9 prefixes and searching for a destination in each half. We then repeated this process by splitting each /9 prefix into two /10 prefixes, and so on, and counting available addresses at each level of granularity.

We made our final selection of destinations when we reached the /21 level because the number of hosts available at the next level /22 exceeded our desired limit. If multiple destinations in the same prefix were present in the collected *dnsstat* files, we selected one based on the following criteria:

- prefer IP addresses from the old DNS Clients list used in our previous DNS related studies in 2000-2001 [23], [24].
- prefer IP addresses seen by the largest number of the DNS root servers.

The resulting *DNS Clients* list has about 140K destinations. We have been monitoring this list since the end of March 2002.

C. RTT measurements

When a client addresses a root server with a request, the response time is a sum of two components: the RTT between the client and the server, and the request processing time. Several other studies have considered the actual response time of the DNS roots [3],[10],[11]. In this paper we focus strictly on the RTT component that is due to the packet propagation in the infrastructure. We have examined how the location (both geographical and virtual) of root servers with respect to their clients affects the latency of client-server connections.

Our analysis is based on the following assumptions.

1. Although the size of the DNS Clients list is tiny in comparison with the total number of name servers in the Internet, this list is representative of the overall population of the root servers' clients. Therefore, our conclusions drawn from measuring the limited sample of clients are arguably representative of the global DNS system.
2. RTTs collected by our topology monitors for probe ICMP packets are approximately the same as DNS response times actually experienced by root servers' clients. This approximation is valid if the request processing time is small in comparison with the propagation time. Measurements comparing the ICMP probe RTT and the DNS response times are available at [25]. Both times seem to be in sufficient agreement to validate this assumption.
3. When choosing among root servers, a client selects the root server with the lowest RTT and always addresses it with its DNS requests. If this best server becomes unavailable, the client switches to the second best and so on, the rank number increasing in the increasing order of RTTs. This assumption is a simplification of the actual algorithm for server selection used by BIND, which makes sure that

[3]We consider a prefix 'globally routable' if it is present in the combined Route Views BGP table [21] at the time we compiled the destination list.

eventually a client will address all root servers, not only the best one with the shortest response time. However, the best server will be used most frequently. This property of the BIND algorithm lends credibility to our approach.

4. Our topology monitors normally collect three values of RTT for each client in a 24 hour period. We use *median RTT* as a representative metric of the latency between clients and root servers. Generally, RTTs are influenced by diurnal network patterns and by multiple random short-lived factors (e.g., link congestion, queuing, routing changes). However, we have shown [24] that the median RTT derived from a certain set of previously observed values is a stable and reliable metric of the proximity between two Internet hosts. [4] While prior information can not predict the absolute value of RTT, it consistently achieves a high positive correlation to the current latency of a connection.

III. Results

We have analyzed one week of traces collected by topology monitors co-located with the A, B, D, E, F, G, H, I, K, and M root servers from July 14 to July 20, 2002. At that time, about 108.5K destinations in our list were responding to `skitter` ICMP probes.[5] For each replying destinations, our monitors collected between 3 and 7 RTTs per day.

A. Significance of individual root servers

BIND implements an affinity algorithm that causes client name servers to select intelligently among all available root servers. It chooses a random starting point, cycles through the root servers, remembering the response time for each, and sorts the root servers into groups based on the observed values of RTT. Subsequent queries are directed to servers in the closest group in a round robin fashion. As a result, a client that is 'close' (in terms of latency) to a particular root server will query that server most of the time, only occasionally querying other root servers that are further away. Accordingly, each root server acquires a set of client hosts who prefer its services over those of other roots.

Should a root server become unavailable, its clients would experience an increase in response time to their DNS requests related to how far the 'second closest' root server is from each client. We analyze a root server's 'importance' based on the greater the number of clients who would experience increased latency in such a situation, and the magnitude of those increases.

In our measurements, the topology monitors simulate root servers, and target destinations in the DNS Clients list represent the general population of clients. For each destination $client_n, n = 1..N$ in our list we did the following. From a week of data we determined median RTTs to each monitor $\{mRTT_n^{S_i}\}, i = 1..10$, rounded them to integers, and ranked them in increasing order. The difference between the lowest RTT and the second lowest RTT is the increased latency that this client would experience if its best ('closest' in latency) server became unavailable.

$$\Delta RTT_n = RTT_n^{second\_lowest} - RTT_n^{lowest}$$

We then grouped destinations into ten subgroups by their best servers and calculated the complementary cumulative distribution functions (CCDFs) of $\{\Delta RTT_n\}$ in each subgroup (figure 2). Here the x-axis is the increase in latency; $y(x)$ is the count of clients for which the increase in latency due to removal of their best root server is greater than x. The higher the curve, the more clients will be adversely affected by the removal of that server. In the legend the servers are sorted by the average height of the corresponding curves.

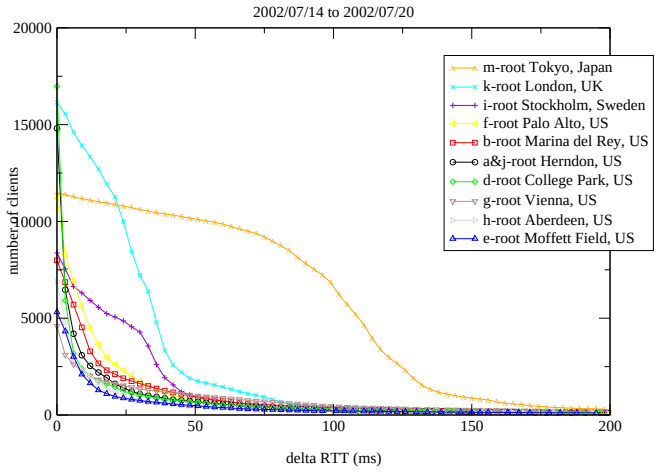

Fig. 2. Increase of latency caused by a root server removal. The curves are CCDFs of the number of clients.

The M root server in Tokyo is the one with the highest curve. A steep drop of its CCDF at ΔRTT of about 100 ms means that disabling the M root would cause a noticeable increase in latency of about 100 ms or more for the vast majority of its clients. Most other CCDF curves drop steeply at small values of ΔRTT. The faster a curve drops along the x-axis, the fewer clients of this root server are affected by its removal from service and the smaller in-

[4] Defined as the latency of connection between these hosts.

[5] From our previous experience with monitoring of other destination lists, we know that the number of replying destinations decreases with time at a typical decay rate of 2-3% per month. [26]

crease in latency they would experience. For example, if E root or H root became unavailable, for 80% of their clients the increase of RTT would be less than 20 ms.

It is not surprising that the M root server stands out in global importance since it is the only root server in Asia. Clients for which it is the best server are also most likely located in Asia (cf. [19], [24] about correlation between geographic distance and RTT) and thus are far from all other root servers. Without M root their DNS service would degrade significantly. However, if one of the root servers in the US is down, the other US root servers are nearby and provide an acceptable backup with a minimal increase of RTT to the US clients.

B. Root server clusters

We have defined and studied the metric of *distance* between a given pair of root servers S_1 and S_2. We consider a subset of destinations $\{client_k\}, k = 1..K$ that respond to both topology monitors co-located with these servers. For each destination in this subset we find the median RTTs to each of the two monitors derived from a week of measurements: $mRTT_k^{S_1}$ and $mRTT_k^{S_2}$. The distance between a pair of root servers $D(S_1, S_2)$ is:

$$D(S_1, S_2) = \frac{1}{K} \sum_{k=1}^{K} \left| mRTT_k^{S_1} - mRTT_k^{S_2} \right| \quad (1)$$

This metric measures the average absolute deviation between the two sets of RTTs and represents the virtual distance between the two root servers as viewed by the destinations in the DNS Clients list. The closer the resemblance between the two RTT distributions, the shorter the distance, and the more indistinguishable in terms of latency variations these servers are to the clients that have connectivity to both of them.

Next we identified clusters of the root servers based on their virtual proximity in terms of the metric above, and thus determined root server groups ("root families"). The resulting clusters (table I) satisfy two requirements:

- For each server, its closest neighbor is always in the same group.
- All distances between members of the same group are lower than to members of other groups.

Table I is diagonally symmetric. Four clusters of servers that we found in virtual space correlate remarkably well with the servers' geographical location. Therefore the name of each group reflects the smallest geographic region that includes all the servers in the same group.

Servers in Group 1 (Europe) are less similar to each other than those in Group 2 and 3 (US), possibly because European servers are geographically more spread out. All servers of the group 2 (US-East) are very close to Washington DC, while all servers of the group 3 (US-West) are in California. The M root server is in a Group 4 of its own because it is geographically so remote from all other root servers. Unsurprisingly, it appears that within each cluster, any one server can functionally replace another one with the least RTT increase experienced by their clients.

C. Root server clusters and their clients

As previously mentioned, we assume that a client uses only its best server for lookups and if this server becomes unavailable, the client switches to the second best server. In Section III-B we found that servers attributed to the same group are close to each other in virtual space and nearly indistinguishable to their clients in terms of connection latencies. Therefore if the best server of a particular client belongs to Group X and that group consists of more than one server, then the second best server of this client most likely is in the same root family. In other words, a client depends primarily on root servers from a certain group for the DNS service.

We subdivided all hosts in our target list into four subsets corresponding to four groups of root servers in Table I. We associated a host with a given group if its median RTT (derived from a week of observations) is lowest to one of the root servers in this group. Columns 1, 2, and 3 in Table II show root families, the number of monitored root servers in each family, and the number of destinations in the subset associated with this family.

We then studied the geographical distribution of clients in each subset. We used the commercially available tool `IxMapper` [27] in order to determine the geographic location of each host in our target list. Figure 3 shows the distribution of the four subsets of destinations by continents and countries. The left column lists the four root families. Each horizontal bar is colored by continents and, when space allows, names the largest contributing countries within each continent. Note that the `IxMapper`'s placement can be imprecise (cf. discussion in [24]). The proposed location is obviously wrong if the RTT from at least one of our topology monitors to this destination is lower than the propagation time of the speed of light in fiber. This problem tarnished about 10% of destinations in the DNS Clients list and they were excluded from Figure 3.

As expected, we found a strong correlation between the geographical location of clients and the geographical group of servers they prefer (i.e. have lowest RTTs to). The clients of one root family tend to be geographically closer to servers in that group than to the others. This cor-

	Group 1 Europe		Group 2 US-East				Group 3 US-West			Group 4 Tokyo, Japan
	k-root	i-root	a&j-root	g-root	h-root	d-root	f-root	e-root	b-root	m-root
k-root	0.0	128.6	191.7	171.0	154.7	151.6	161.1	176.2	194.6	235.7
i-root	128.6	0.0	167.2	174.9	172.4	170.1	181.1	182.7	190.8	232.7
a&j-root	191.7	167.2	0.0	96.5	98.1	97.7	132.6	134.8	141.7	251.9
g-root	171.0	174.9	96.5	0.0	95.5	91.5	128.4	133.5	134.0	231.6
h-root	154.7	172.4	98.1	95.5	0.0	91.5	115.0	120.3	135.9	225.0
d-root	151.6	170.1	97.7	91.5	91.5	0.0	128.3	127.3	138.5	229.3
f-root	161.1	181.1	132.6	128.4	115.0	128.3	0.0	90.2	95.9	196.8
e-root	176.2	182.7	134.8	133.5	120.3	127.3	90.2	0.0	104.2	209.7
b-root	194.6	190.8	141.7	134.0	135.9	138.5	95.9	104.2	0.0	206.1
m-root	235.7	232.7	251.9	231.6	225.0	229.3	196.8	209.7	206.1	0.0

TABLE I
ROOT FAMILIES.

Groups	Monitored roots servers	Destinations preferred	All root servers
1. Europe	2 (18.2%)	24,387 (23.7%)	2 (15.4%)
2. US-East	5 (45.5%)	42,978 (41.7%)	6 (46.2%)
3. US-West	3 (27.3%)	24,343 (23.6%)	4 (30.8%)
4. Tokyo, Japan	1 (9.1%)	11,386 (11.0%)	1 (7.7%)
Total	11 (100%)	103,094 (100%)	13 (100%)

TABLE II
ROOT FAMILIES AND CORRESPONDING SUBSETS OF DESTINATIONS. PERCENTAGES ARE RELATIVE TO THE TOTAL OF EACH COLUMN.

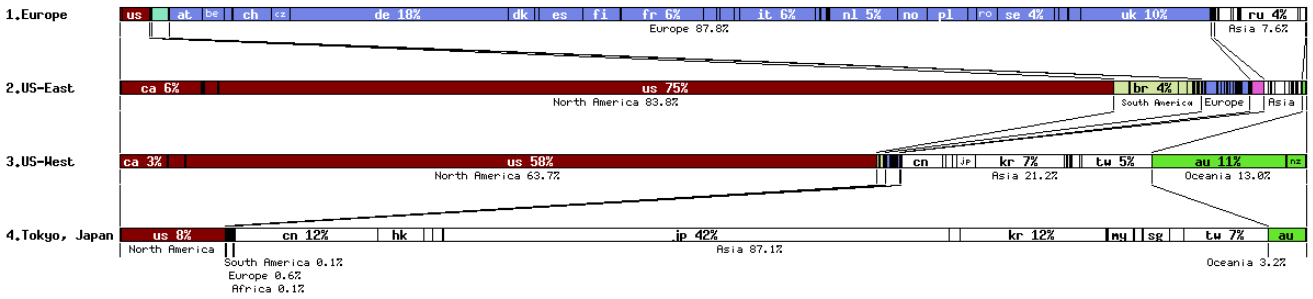

Fig. 3. Geographical makeup of destination subsets by continents and countries. Percentages of destinations in a given country are relative to the number of destinations in each individual subset.

relation implies that the location of servers with respect to the prevalent locations of clients is a significant factor affecting overall DNS performance.

Is it possible to improve the efficiency of the DNS service by optimizing the placement of existing root servers? On one hand, if a server can be placed in the vicinity of those clients that are geographically far from their current best root servers, then their DNS service will improve. This result further supports findings of [23] that root server clients in Africa, South America, and to a lesser degree Asia appear to be underserved in comparison with North America and Europe. On the other hand, it would be wrong to distribute the existing root servers "uniformly" around the globe. At present, the number of Internet users in North America and Europe exceeds the number of users in Africa, South America, and Asia by an order of mag-

nitude [28]. Moving servers to these latter continents will improve the DNS service for a smaller number of clients, but would likely degrade it for a much larger number of clients.

Is the current geographic distribution of servers around the world close to optimal then? The data we have to answer this question are somewhat incomplete since two root servers (C and L) remain uninstrumented. However, C is on the East coast, and L on the West coast of the US (cf. Figure 1). For the sake of estimation, we attributed these root servers to the corresponding root families thus adding column 4 to Table II. Assuming that our DNS Clients list is representative of the worldwide population of clients, we compare the percentage of servers in each root family with the percentage of destinations served by this family (columns 4 and 3). The comparison shows that Group 1 (Europe) is most seriously underprovisioned, while both US groups have proportionally more servers then clients. Therefore if the total number of root servers remains the same in the future, US servers are (again, unsurprisingly) the best candidates for relocation to other regions of the world.

D. Impact of a root server relocation

We developed and tested a methodology to simulate the effect of a possible server relocation. Suppose that one of the existing root servers is moved elsewhere. How would this move affect the DNS performance for different groups of clients?

In the example that follows, we considered Amsterdam, NL as a possible site for relocating one of the existing root servers. A backup server for the K-root, K-peer, is located in Amsterdam. Although currently the K-peer is not providing DNS services, it is provisioned with the necessary hardware and software and is sufficiently well-connected to the network that it could easily replace the services of K-root if necessary. These characteristics make K-peer a suitable candidate for our simulation. We installed a topology probing monitor at K-peer and polled the DNS Clients destination list from this host. We used a week of data collected in July 2002 at root servers and at K-peer to estimate quantitatively the potential changes in macroscopic DNS performance.

As in Section III-A, for each destination in the DNS Clients list we calculated (rounded to integers) median RTTs to each monitor $\{mRTT_n^{S_i}\}, i = 1..10$, and ranked them in increasing order. We also found the median RTT between each destination and K-peer - $mRTT_n^{K-peer}$. Suppose that K-peer becomes a root server instead of one of the existing servers S_i. If $mRTT_n^{K-peer}$ is smaller than $mRTT_n^{S_i}$ then the client will experience an improvement in service as the result of this change. Clients for whom $mRTT_n^{K-peer}$ is smaller than the smallest RTT from the set of RTTs to existing roots, will always benefit regardless of which of the current root servers were replaced by the K-peer. However, if $mRTT_n^{K-peer}$ is larger than the minimum RTT from the set, then root DNS service for this client would deteriorate if its current best root server were moved to Amsterdam.

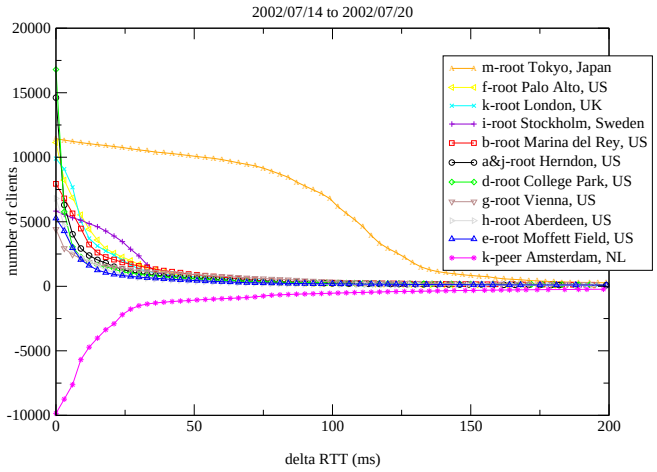

Fig. 4. Latency change caused by a root server relocation. The curves are CCDFs of the number of clients. The negative curve labeled as "k-peer Amsterdam, NL" shows the clients whose connection to the K-peer would have a latency lower than to any root server. The positive curves show a latency increase due to a relocation of the corresponding root server to Amsterdam, NL.

Figure 4 shows the number of clients affected by a hypothetical relocation of one root server (where we try each one at a time) to Amsterdam. The x-axis is the absolute value of the latency change and the y-axis is the count of clients. The curves are CCDFs. Negative values indicate the number of clients for which the latency decreased by a given value of x or more. The positive values indicate the number of clients for which the latency increased by a given value of x or more.

The single curve of negative values corresponds to the case of $mRTT_n^{K-peer} < mRTT_n^{S_i}$ for any $i = 1..10$, that is, an RTT to the K-peer being lower than to any root server S_i. For these clients, the service would always improve if one existing root server were moved in Amsterdam. The amount of decreased latency for them is always the difference between $mRTT_n^{K-peer}$ and $mRTT_n^{lowest}$. Therefore, the negative curve remains the same regardless of which server is hypothetically relocated.

Curves of positive values correspond to the case when the mRTT to K-peer is not the smallest among all. If the best server for a client is hypothetically removed, then the increased latency would be the difference between the original lowest RTT in the set and the second lowest RTT determined after the $mRTT_n^{K-peer}$ is added to the set. Otherwise the clients are not affected at all and thus not accounted for in Figure 4.

Since all the curves in the negative region are the same, the fewer the number of clients in the positive region as well as the less the amount of RTT increase incurred by those clients, the more beneficial the relocation of that server is overall. For each server, we calculate the net effect of its imaginary relocation to Amsterdam by combining resulting latency increases and decreases together. Figure 5 shows the 5%, 25%, 50%, 75%, and 95% quartiles of the resulting distributions. If the bulk of the $\{\Delta RTT_n\}$ distribution for a given server is below the x-axis, then relocation of this server will have an overall positive effect. If the bulk of the distribution is above the x-axis, then moving this server would degrade the overall DNS service. If the distribution is centered around the x-axis, then the numbers of clients positively and negatively affected by relocating this server would be approximately equal.

We note that from our measurements root servers E, G, and H may be suitable candidates for the relocation. First, these servers have the fewest number of clients who actually use them for DNS lookups (their positive curves are lowest in figure 4). Second, the corresponding combined latency change distributions in figure 5 are mostly below the x-axis. The number of their clients for whom RTT would deteriorate (positive ΔRTT) is insignificant, and the increase is usually less than 25 ms.

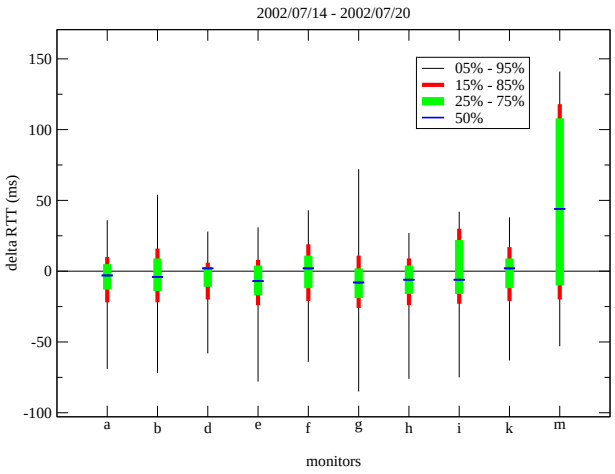

Fig. 5. Combined distributions of latency change due to potential relocation of root name servers.

IV. Conclusion and Future Work

We have used CAIDA topology probe monitors and lists of clients gathered at DNS root servers in order to analyze the connectivity between the roots and the worldwide population of their clients. We have considered how the geographical locations of root servers with respect to those of the clients they serve influence observed performance. We also developed and tested a methodology simulating the effects of root server relocation. Our main conclusions follow.

1. In terms of the impact of latency increase on clients of the root server system if it were removed, the most crucial root server by far is M-root, the only root server in Asia. M-root serves the majority of Asian clients and if it became unavailable, its clients would have to use other root servers in the US or Europe. For a large percentage of those clients the latency would increase by a significant amount, e.g., more than 100 ms. In contrast, for the two groups of US roots, their geographic affinity to each other provides any given client with a chance to switch to another root with little increased latency if that client's best ('closest') server becomes unavailable.

2. The geographical distribution of root servers plays the most important role for overall performance. Clients normally have lower RTTs to geographically nearby servers and so naturally use these servers for lookups. If the root servers were distributed in accordance with the current geographic distribution of their clients, it would benefit clients that are currently far away from our 13 root servers; all clients would have the opportunity to use a root server that is geographically close to them.

3. Our analysis based on geographically grouping servers and clients into four groups demonstrated the unsurprising result that US root clients appear to be overprovisioned. Therefore, if it is impossible to add new root servers, RSSAC/ICANN should relocate some US root servers to Asia and Europe. As an example, we simulated such a relocation of each root server, one at a time, to where K-peer is currently located (Amsterdam, Netherlands). The simulation showed that out of 11 root servers that CAIDA monitored for this study, G, E, and H-root are the most suitable candidates for relocation.

The analysis presented in this paper does not take into account traffic load of the servers or load balancing factors; these are areas for valuable future study. We also plan to switch some other CAIDA topology monitors, i.e., those not co-located with root servers, to probe the desti-

nations in our DNS Clients list for about 2-3 weeks. With these new traces from those monitors located in other geographic regions, we can expand our analysis to simulate a variety of scenarios of potential future root server locations. Indeed, instrumenting any potential root server location with such a topology probe monitor would allow this sort of simulation and provide empirical basis for what has become an increasingly politically sensitive policy decision.

REFERENCES

[1] H. Rood, "What is in a name, what is in a number: some characteristics of identifiers on electronic networks," *Telecommunications Policy*, vol. 24, 2000.

[2] P. Albitz and C. Liu, *DNS and BIND*, O'Reilly and Associates, 1998.

[3] C. Huitema and S. Weerhandi, "Internet Measurements: the Rising Tide and the DNS Snag," in *Monterey ITC Workshop, September 2000*, 200.

[4] M. Kosters, "Massive scale name management: Lessons learned from the .com namespace," TWIST 99 Workshop, UC Irvine, `http://www.ics.uci.edu/IRUS/twist/twist99/presentations/kosters/kosters%.ppt`.

[5] "Bind website," `http://www.isc.org/products/BIND/`.

[6] "djbdns," `http://www.djbdns.org`.

[7] "How to install and configure Microsoft DNS server," `http://support.microsoft.com/default.aspx?scid=KB;EN-US;Q172953\&`.

[8] "Analysis of the DNS root and gTLD nameserver system: status and progress report," `http://www.caida.org/projects/dns-analysis/`.

[9] P. Danzig, K. Obraczka, and A. Kumar, "An analysis of wide-area name server traffic: a study of the Internet Domain Name System," in *Proc. ACM SIGCOMM*, 1992.

[10] Nevil Brownlee, k claffy, and Evi Nemeth, "DNS Root/gTLD Performance Measurements," in *Proc. Passive and Active Measurement workshop (PAM)*, 2001.

[11] Nevil Brownlee and Ilze Ziedins, "Response time distributions for global name servers," in *Proc. Passive and Active Measurement workshop (PAM)*, 2002.

[12] J. Jung, E. Sit, H. Balakrishnan, and R. Morris, "DNS performance and the effectiveness of caching," in *Proc. ACM SIGCOMM Internet Measurement Workshop*, 2001.

[13] C. Wills and H. Shang, "The contribution of DNS lookup costs to web object retrieval," 2000, Tech. Rep. TR-00-12, Worcester Polytechnic Institute.

[14] G. Chandranmenon and G. Varghese, "Reducing web latency using reference point caching," in *Proc. IEEE Infocom*, 2001.

[15] K. et al. Cho, "A study on the performance of the root name servers," `http://mawi.wide.ad.jp/mawi/dnsprobe/`.

[16] R. Liston, S. Srinivasan, and E. Zegura, "Diversity in DNS performance measures," in *Proc. Internet Measurement Workshop*, 2002.

[17] A. Shaikh, R. Tewari, and M. Agrawal, "On the Effectiveness of DNS-based Server Selection," in *Proc. of IEEE INFOCOM*, 2001.

[18] R. Somegawa, K. Cho, Y. Sekiya, and Yamaguchi S., "The effects of server placement and server selection for Internet services," 2002, IEICE.

[19] B. Huffaker, M. Fomenkov, D. Moore, and k. claffy, "Macroscopic analyses of the infrastructure: measurement and visualization of Internet connectivity and performance," in *Proc. Passive and Active Measurement workshop (PAM)*, 2001, `http://www.caida.org/outreach/papers/2001/Rssac2001a/`.

[20] "CAIDA topology mapping tool," `http://www.caida.org/tools/measurement/skitter/`.

[21] D. Meyer, "University of Oregon Route Views Project," `http://www.routeviews.org/`.

[22] "CAIDA DNS statistics utility," `http://www.caida.org/tools/utilities/dnsstat/`.

[23] M. Fomenkov, k. claffy, B. Huffaker, and E. Moore, D. Nemeth, "Macroscopic Internet Topology and Performance Measurements from the DNS Root Name Servers," in *Usenix LISA*, Dec. 2001, `http://www.caida.org/outreach/papers/2001/Rssac2001a/`.

[24] B. Huffaker, M. Fomenkov, D. Plummer, D. Moore, and k. claffy, "Distance Metrics in the Internet," in *IEEE International Telecommunications Symposium*, Sept. 2002, `http://www.caida.org/outreach/papers/2002/Distance/`.

[25] R. Thomas, "DNS Data Page," `http://www.cymru.com/DNS/`.

[26] M. Fomenkov, "Decay of destination lists," private communication.

[27] Ixiacom IxMapping, "Ixmapping," `http://www.ipmapper.com`.

[28] "CAIDA BGP Geopolitical Analysis," `http://www.caida.org/analysis/geopolitical/bgp2country/`.