Scaling the Root

Report on the Impact on the DNS Root System
of Increasing the Size and Volatility of the Root Zone

Prepared by the Root Scaling Study Team
for the Root Scaling Steering Group

Jaap Akkerhuis
Lyman Chapin
Patrik Fältström
Glenn Kowack
Lars-Johan Liman
Bill Manning

Version 1.0
7 September 2009

Executive Summary

Until recently the root zone of the Domain Name System (DNS) has enjoyed two important stabilizing properties:

- it is relatively small— currently the root zone holds delegation information for 280 generic, country-code, and special-purpose top-level domains (TLDs), and the size of the root zone file is roughly 80,000 bytes; and
- it changes slowly—on average, the root zone absorbs fewer than one change per TLD per year, and the changes tend to be minor.

The root system has therefore evolved in an environment in which information about a small number of familiar TLDs remains stable for long periods of time. However, the type, amount, and volatility of the information that is contained in the root zone are expected to change as a result of the following four recent or pending policy decisions:

- support for DNS security (DNSSEC), or "signing the root";
- the addition of "internationalized" top-level domain names (IDN TLDs);
- support for the additional larger addresses associated with Internet Protocol version 6 (IPv6); and
- the addition of new TLDs.

This report presents the results of a study that was undertaken to determine if, how, and to what extent "scaling the root" will affect the management and operation of the root system. The principal results of the study are qualitative and quantitative models of the root system that show how its different parts are related and how it responds to changes in the parameters that define its environment. These models enable the static analysis of popular "what-if" scenarios—*e.g.*, "what would happen if the size of the root zone increased by three orders of magnitude (assuming that everything in the system remained as it is today)?"—but also a far more useful dynamic analysis of the way in which the system responds and adapts to changes in the DNS environment over time. The insights available from this dynamic analysis will allow the community to anticipate the consequences of scaling the root; identify and recognize "early warning signs" of system stress; and plan ahead for the mitigations that may be necessary to keep the system running smoothly if and when those signs appear.

The root is a highly decentralized dynamic system. The geographic and organizational decentralization of the root system arise from a deliberate design decision in favor of diversity and minimal fate-sharing coordination, which confers substantial stability and robustness benefits on the global Internet. Simple quantitative extrapolation from a baseline model of the

current root system does not predict realistic future states of the system beyond the very short term, because

- each part of the system adapts in different ways to changes in the quantity, type, and update frequency of information in the root, while also responding to changes in the rest of the Internet;
- these adaptations are not (and, in the current model of root system management, cannot be) effectively coordinated; and
- for some, if not all, of the actors non-quantifiable considerations dominate their individual adaptation behavior (both strategically, in a planning context, and tactically, in an operations context).

These attributes of the root system invalidate the popular "empirical evidence" analogies to monolithic commercial registries for large top-level domains such as COM, or DNS-management companies that operate second-level domains with millions of entries. The fact that it is technically feasible to reliably operate very large DNS zones is a necessary, but in no sense sufficient, condition for scaling the root.

Any increase in the size or volatility of the root zone involves risk. If the only objective of root system management were stability, no change to the current size or composition of the root would be justifiable. Whether or not—and how rapidly—to introduce specific changes to the root zone are therefore policy decisions that must balance the expected benefit of the change against the expected impact on root system stability. Adding support for DNSSEC, for example, has a negative impact on root system stability but a positive impact on DNS and Internet security. For policy-makers, the important question is not "will this affect the stability of the root system?" but "does the benefit justify the effort that will be necessary to ensure that any negative impact on root system stability can be effectively mitigated?"

Root system oversight should focus on "early warning" rather than threshold prediction. As the size, contents, and volatility of the root zone change, the important question for the community is dynamic and continuous—"as the root and the rest of the Internet evolve, how can we be sure that we will recognize approaching problems early enough to take appropriate mitigating actions?"—not static and predictive—"based on today's root system, how many new TLDs can we add to the root per year without breaking something?" An effective early warning system that can recognize and respond to approaching problems will require new agreements and channels for communication among the participants in the management of the root system. The focus of root zone management policy should be the establishment of effective mechanisms for detecting and mitigating risks as they become visible.

In order for "early warning" to be effective, changes to the root must be made gradually. Except in the very short term, we cannot confidently predict what the effect (and side-effects) of making a particular change to the root will be. In order for the strategy of "early warning" to succeed, the rate at which we introduce changes to the root must be gradual enough to allow the system to respond and adapt within the "early warning" horizon. The prudent automobile driver at night does not try to imagine all of the possible hazards that might lie in the dark road ahead; she adjusts her speed to the forward reach of her headlights, so that she has time to take evasive action if and when a hazard appears.

Regardless of how slowly or quickly changes to the root zone are made, each of the root system players will eventually encounter boundaries that require step-function process or configuration changes. These discontinuities represent the boundaries between steady-state or continuous-function operating regions for each of the players. Because each of the players adapts to changes in the root system differently, and also responds to changes in the behavior of the other players and changes in the rest of the Internet, it is not feasible to construct a simple composition of step-functions that would express the expected adaptive behavior of the root system as a whole.

On the provisioning side, the ability to scale the root is completely dominated by the steps that involve human intervention. Even after the new root zone workflow automation system is fully deployed, all three of the provisioning system actors (IANA, NTIA, and VeriSign) will retain at least one "human inspection" step for every change request. These bottlenecks govern the rate at which root zone changes can be processed; by comparison, every other factor is "in the noise."

On the publication side, scaling the root primarily affects poorly-connected Internet locations. A much larger root zone can easily be served from sites with high-bandwidth connections and ready access to servers and other computing infrastructure. It cannot easily be served from sites with poor connectivity or infrastructure. Scaling the root is likely to place additional demands on those operators who use IP anycast to deploy root servers in economically less-developed parts of the world.

The risks associated with an annual increase in the size of the root zone on the order of hundreds of new entries [*O*(100)] can be managed without changing any actor's current arrangements. A thorough analysis of the currently visible system suggests that the risks associated with doubling or tripling the size of the root zone can be mitigated without hitting a discontinuity in the scaling properties of any of the root system management functions. Note that this does *not* mean that "adding 999 new entries to the root is OK, but something will break

when you add the 1,000th entry." The discontinuities in a continuously evolving dynamic system do not arrive at statically predictable numeric boundaries; they lie in regions that can be surveyed only imprecisely from a distance. It also does not mean that "adding hundreds of new entries per year to the root is safe." Our ability to survey the regions in which discontinuities may lie for one or more of the root zone management functions is limited to assessment of risk, not absolute conclusions about "safety" (see "Any increase in the size or volatility of the root zone involves risk" above).

The risks associated with an annual increase in the size of the root zone on the order of thousands of new entries [*O*(1000)] can be managed only with changes to the current arrangements of one or more actors. A thorough analysis of the currently visible system suggests that the risks associated with short-term order-of-magnitude increases (from hundreds of entries to thousands of entries) in the size and volatility of the root zone cannot be mitigated entirely by the arrangements that are currently in place for one or more of the root zone management functions. Note that this does *not* mean that annual increases on this scale cannot be sustained—only that in order to do so, one or more actors must be willing and able to undertake a discontinuous upgrade or other "non-linear" change in resources or behavior.

The risks associated with adding DNSSEC, new TLDs, IDNs, and IPv6 addresses to the root simultaneously can be managed only with changes to the current arrangements of the root server operators. Signing the root would, by itself, immediately increase the size of the root zone by roughly a factor of 4 and increase the size of the priming response message. The consequences of these two effects could be absorbed by the root server operators without loss of service to the Internet, but would require them to substantially re-plan in order to recover lost headroom (deliberate defensive over-provisioning) in both server capacity and bandwidth. Adding new TLDs, IDNs, and IPv6 addresses would also increase the size of the root zone; adding IPv6 addresses would in addition increase the size of the priming response. With aggressive re-planning (some of which is already underway), the system is capable of managing the risks associated with adding either (a) DNSSEC or (b) new TLDs, IDNs, and IPv6 addresses over a period of 12-24 months—but not both.

If a choice must be made, DNSSEC should come first. The effects of signing the root would be felt immediately—a sudden jump in the size of the root zone, and a sudden increase in the volume and type (TCP vs. UDP) of root server query traffic. The effects of the other changes would be spread out over some period of time (longer or shorter depending on the rate at which the system is able to adapt). Because the step-function impact of signing the root will be proportionally greater the larger the root becomes, deploying DNSSEC before the other changes have increased the size of the root would significantly lower the risk it presents to DNS stability.

Contents

1 Introduction

Until recently the root zone of the Domain Name System (DNS), which contains the "top-level" domains (TLDs) that anchor the DNS naming hierarchy, has enjoyed two important stabilizing properties:

- it is relatively small—currently[1] the root zone holds delegation information for 280 generic, country-code, and special-purpose TLDs,[2] and the size of the root zone file is roughly 80,000 bytes;[3] and
- it changes slowly—on average, the root zone absorbs fewer than one change per TLD per year, and the changes tend to be minor.[4]

The system of name servers that respond to queries concerning the top-level domains (the "root servers") and the system that processes updates to the root zone (the "root zone management system")[5] have therefore evolved in an environment in which information about a small number of familiar TLDs remains stable for long periods of time.

The type, amount, and volatility[6] of the information that is contained in the root zone are expected to change as a result of the following four recent or pending policy decisions, with potential impact on the operation of the root system:

- support for DNS security (DNSSEC) [43], or "signing the root";[7]
- the addition of "internationalized" top-level domain names (IDN TLDs);[8]
- support for the larger addresses used with Internet Protocol version 6 (IPv6)[9] [7]; and
- the addition of new generic TLDs.[10]

[1] In this report "currently" means "25 August 2009" (root zone file serial number 2009082500).
[2] A list of all of the top-level domains in the root is published by the Internet Assigned Numbers Authority (IANA) at http://www.iana.org/domains/root/db.
[3] A public copy of the current root zone file is maintained at ftp.internic.org in the directory ~/domain. 80K (technically 80*1024 or 81,920 bytes) is a working estimate of the size of the file, which is slightly different when stored on or transported between different systems.
[4] http://www.isoc.org/briefings/020/changefile.shtml
[5] Collectively, "the root system." An introduction to the root system may be found in "DNS Root Name Servers Explained For Non-Experts" [9].
[6] "Volatility" in this context means "update frequency"—the rate at which information in the root zone changes.
[7] http://www.icann.org/en/announcements/dnssec-qaa-09oct08-en.htm
[8] http://www.icann.org/en/topics/idn
[9] http://www.iana.org/reports/2008/root-aaaa-announcement.html
[10] http://www.icann.org/topics/new-gtld-program.htm

This report presents the results of a study that was undertaken to determine if, how, and to what extent "scaling the root"—increasing the size and volatility of the root zone as a consequence of pursuing these policies—will affect the management and operation of the root system. These results are expected to inform the community's discussion of root zone and TLD policy alternatives by replacing anecdotes and speculation with data and analysis.[11]

The principal results of the study are qualitative and quantitative models of the root system that show how its different parts are related and how it responds to changes in the parameters that define its environment. These models enable the static simulation of popular "what-if" scenarios—*e.g.*, "what would happen if the size of the root zone increased by three orders of magnitude (assuming that everything in the system remained as it is today)?"—but also a far more useful dynamic analysis of the way in which the system responds and adapts to changes in the DNS environment over time. The insights available from this dynamic analysis will allow the community to anticipate the consequences of scaling the root; identify and recognize "early warning signs" of system stress; and plan ahead for the mitigations that may be necessary to keep the system running smoothly if and when signs of stress appear.

[11] Notwithstanding Raymond Wolfinger's brilliant aphorism "the plural of anecdote is data."

2 Background and context

2.1 The Domain Name System[12]

The DNS is fundamentally three things:

- a shared name space;
- the servers (name servers) that implement the name space; and
- the resolvers (intermediate caching servers) and end systems that send questions (queries) about the name space to the name servers.

The root zone defines the apex of the shared name space and the root nameservers are where this name space apex is implemented for the rest of the users of this namespace—*i.e.*, the Internet as we know it.

The billion or so computers that form the Internet of today would have to send all of their queries to these root name servers without two other architectural features of the DNS. The first is that it is designed to be hierarchical—parts of the name space can be and are distributed and delegated to other authoritative name servers in the Internet. This DNS feature allows for and has enabled the massive growth and scalability of the Internet in the past 20 years. The second is the use of DNS resolvers that cache responses from authoritative servers as a result of queries sent to them from their client end systems.

The DNS name space is implemented as a hierarchical distributed database, divided for management purposes into pieces, called *zones*. Each zone is served by one or more *name servers*,[13] which are synchronized to contain identical sets of data. The zones are hierarchically organized into a structure that is usually represented graphically as an inverted "tree", and the zones contain DNS information belonging to the corresponding name domains in the tree. The root zone constitutes the top of the inverted tree (level 0). Its name is, strictly speaking, an empty string (not "root"), but it is usually denoted with a single "." (period or "dot").[14]

The DNS data in a zone are usually stored in a file—a *zone file*. The servers serving the same file synchronize by sending the contents of the zone file from the *master* server to *slave* server(s).[15] This is known as a *zone transfer*. Masters and slaves are considered equal from a DNS "quality" or "authority" standpoint; the term *master* simply distinguishes the server at which changes to the zone in question are entered.

[12] Much of the background information in this section has been taken (in some cases verbatim) from [26].
[13] Both "name server" and "nameserver" are used, interchangeably, in descriptions of the DNS.
[14] Apologies to readers for whom this punctuation mark is called a "full stop."
[15] In old literature the terms *primary* and *secondary* are often used instead of *master* and *slave*.

The root zone contains pointers downwards in the DNS hierarchy. It contains the names of the existing domains one level below the root—level 1, or the top level domains (TLDs). This has two consequences:

- a TLD is visible to the public Internet only if it is listed in the root zone, and
- any DNS client (*resolver*) can always start its lookup process for any domain name by asking a server that carries the information in the root zone; the pointers in the root zone will lead the client to issue a series of subsequent queries which will eventually lead to the sought information.

For each TLD there is also a zone, which is served by one or more synchronized servers. The model repeats itself, level by level, downwards.

From a strictly technical standpoint, there is no difference between the root zone and any other zone in the DNS. They are all served by synchronized servers, and they all contains the same type of data. The only property that the root zone has, which the others don't, is that it sits at the top of the tree. From a network political standpoint, however, it is the most important zone of all, since it is the guaranteed entry point for a resolver that wants to look up any domain name in the public DNS, and since it literally defines the first (and arguably the most important) level of the entire DNS namespace, by listing all public TLDs.

2.2 The root zone file

Compared to the zone files of most active TLDs, the root zone file is very small. As of 25 August 2009, the number of TLDs listed therein is 280. Each such name is associated with a set of *resource records* that build up the pointer structures that will lead the clients onwards and downwards in the DNS hierarchy. There is also a small number of resource records that relate to the root itself. This leads to a total number of 2,942 records, or, stored as a text file on disk, roughly 80 KB.[16] The 280 TLDs divide as follows: 248 country code TLDs (ccTLDs), pertaining to geographical and political representations, 21 generic TLDs (gTLDs), and 11 test domains used for tests with internationalized domain names (IDNs, which are described in a later section).

The root zone file is public, and the actual and current file, as used by the official root server operators (see below), can be obtained from http://www.internic.net/zones/root.zone. After an instance (version) of the root zone file has been created by VeriSign from information in the root zone database (as described in a later section), it is not modified in any way by anyone in the provisioning or publication process. Its content remains exactly the same as at the time of

[16] Because the information in the root zone file is stored in different ways on different systems, and can be moved from one system to another using several different transfer syntaxes, different measures of "the size of the root zone file" yield different byte counts. "80K" is therefore a useful working estimate.

compilation until it is replaced by a later version. This means that later versions are never replaced with a previous version (or, to put it another way, the value of the serial number in the SOA record of the root zone file at all of the root servers increases uniformly and monotonically). It also means that during the lifetime of a root zone file (that is, until it is replaced by a file with a higher serial number), it is not updated from the Internet at large or by any of the servers that participate in the DNS. This fixed-state property of the root zone file distinguishes the operation of the DNS from, for example, the operation of the Internet's IP routing system, in which routers continuously and mutually update each other's BGP tables using router-to-router pathways defined by the protocol.

Entry of a top-level domain into the root zone file has become a subject of substantial economic and social importance. Consequently, who controls entry into the root, and by what means, have become controversial subjects.

The root zone and the root name servers are critical to the operation of the DNS. The effective and reliable operation of the DNS, and of the Internet, is entirely dependent on the accuracy and integrity of the root zone file (and its distribution) and the security, reliability, and efficiency of the root name servers. Fortunately, the root zone has proven highly resilient and resistant to errors or disruptions. One possible source of error is an incorrect entry—a mistyped domain name or an erroneous IP address—in the root zone file. A consequence could be misdirection of queries seeking a particular top-level domain (TLD) name server, say .NET. That could prevent users searching for the address of any domain name within that name server's TLD, say any address in .NET, from reaching it through the specific root name server containing the incorrect entry. If the error were updated in all copies of the root zone file, access would effectively be denied to all domain names in that TLD, except from name servers that had previously cached the correct address.[17] However, relatively simple error detection and correction procedures can prevent such errors. (For example, most large top-level domain administrators regularly query the root name servers to ensure that they properly route queries pertaining to their respective TLDs.)

Although, as noted, there have been instances in the past of errors in the root zone file that have had significant effects on the reliable operation of the DNS, there has been none in recent times. At least for the current rates of queries and of changes in the root zone file, the systems of error checking and correction and of capacity sharing appear to be working successfully.

[17] This exception would hold true only until the "time-to-live" (TTL) intervals of the cached addresses expired.

2.3 The root name servers

The *root name servers* (or simply *root servers*) are DNS name servers that carry and serve data from the root zone. There are 13 publicly accessible well-known IPv4[18] addresses on the Internet from which such service can be obtained. The servers are denoted by the letters A through M, and carry DNS hostnames of the form <*letter*>.*root-servers.net* (for example, a.root-servers.net). Some of them also provide service at IPv6 addresses.

Historically the master for the root zone was one of the regular root name servers. The others did zone transfers according to the specifications in the DNS protocol from that server, to maintain updated copies of the zone file. Starting in 1996 and achieving adoption by all slaves by the end of 2001, the role of the master was transferred to a distinct *distribution master* (also referred to as a "hidden master"), which is a server that is used to update the secondaries but which is not itself visible in the DNS system. This distribution master is operated by VeriSign, Inc. All of the public root name servers are now slaves, including the former master. Thus, all of the root name servers have equal status. Digital signatures and error checking are in place between the distribution master and the respective slave systems to minimize the chance of introducing errors or being successfully attacked during updates.

The home locations of some of the root servers were originally determined by analysis of network traffic flows and loads, seeking to have at least one server "close" in terms of message communication time to every location on the network. It is important to have root servers distributed so that they provide a sufficient level of service to all users across the network.

Considerations of this type are both complex and important, and have, as the Internet evolved, become increasingly so. Over time, these original locations have became less satisfactory, which has been one of the reasons for the proliferation by some operators—notably C, F, I, J, K, and M—of satellite sites at different locations. These satellite sites use a method called *anycast*, [19] which enables servers with the same IP address to be located at different points on the Internet. Copies of the F-root server, for example, can be placed at multiple locations around the world. The widespread distribution of anycast satellites of the root servers has improved the level of service provided to many previously less well served locations. An argument has been made that the 13 letters representing the distinct root name servers are too few to meet requirements for reliability and robustness, which requires sufficient capacity distributed widely enough to protect against system or network failures and attacks. Others have argued that more root servers are needed to satisfy institutional requirements. These concerns arise from the belief that to be a full

[18] IPv4 and IPv6 are different versions of the Internet Protocol, using different addressing schemes.
[19] See [29] for a description of IP anycast.

participant in the Internet, a nation or region must have its own root name server. While technical constraints make it difficult to increase the number of root name server IP addresses beyond 13,[20] the rapidly increasing use of anycast addressing has enabled the root name server system to respond to both the technical and institutional requirements through the addition of multiple geographically distributed anycast root server satellites. Even so, since it addresses the distribution only of servers and not of the operating institutions, the location issue (or, more strictly, the issue regarding the *identities* of the server operators) is likely to continue to add some political acrimony to any selection process that might follow from the withdrawal of one of the current operators.

There is no standard hardware and software implementation of the root name servers. Each of the responsible organizations uses its preferred equipment and operating and file systems, although most of them run some version of the BIND name server software.[21] Although there might be some operational advantages to having standard implementations, most Internet technologists believe that variation in the underlying hardware and software of the root name server system is highly desirable, since it ensures that an error in a particular piece of hardware or software will not lead to the simultaneous failure of all of the root servers. As the number and rate of queries (the *query load*) to the root name servers have increased, hardware and software upgrades have enabled the servers to keep up. However, the pace of inquiries is likely to continue to grow and it is conceivable that it could, in principle, exceed the capacity of a system comprising even the most powerful single computers. Because of anycast deployment, load balancing, and multiprocessing, through which a root server can comprise multiple processors, the number of computers at each root-server address is effectively unlimited. Moreover, it is plausible to expect continued improvements in computing and communications performance.

Although the root zone file is very small, a very large number of client resolvers need the information that it contains. Using I-root as an example, the combined (all anycast instances added together) 24-hour average query rate is (as of August 25, 2009) 15,600 queries per second, or 1,347,840,000 (1.35 billion) queries per 24-hour cycle. Not all servers receive the same amount of traffic, but the combined traffic to all servers of all operators ought to be in the very rough neighborhood of 10 billion queries per day. This is well within the computational capabilities of the servers, because of the substantial overprovisioning of the system.

[20] This is due to a very old restriction that limits DNS messages to 512 bytes. A maximum of 13 servers and their IPv4 addresses fits in 512 bytes. See [28] section 4.2.1.

[21] BIND—originally an acronym of "Berkeley Internet Name Domain"—is a managed open source software product maintained by the Internet Systems Consortium (https://www.isc.org/software/bind).

A possibly disruptive event would involve one or more of the root name servers being non-operational for hours or more. This might happen because of the unexpected failure of a computer or communication link (accidental or purposeful) or because of regular maintenance. However, since the capacity of each operator's server cluster is many times greater than its average load, and iterative resolvers can use any of the 13 root name servers addresses, other name servers can take up the load without degradation in the system's performance that is perceptible to users. In the past decade, the total number of server instances operated has increased by roughly a factor of 10 (one order of magnitude).

However, *sustained* problems with many instances will eventually lead to a degradation of service, which might well become evident to Internet end users. The most obvious reason for this would be the inability of one or more operators to "feed" updated versions of the zone file to the remote root servers, forcing the operator to shut down the operations in a certain part of the world. Another reason would be a sustained malicious attack on the system as a whole by a very large number of computers.

2.4 The root zone management process

Because the root domain, and its corresponding zone, is central and critical to the operation of the DNS, decisions about the root zone and the root name servers are of substantial importance, and the institutions that bear responsibility for them play an important role as stewards of the DNS root.

Those institutions carry out four critical functions:

- deciding what new or revised entries will be included in the root zone file;
- creating the root zone file, keeping it current, and distributing it to all of the root name servers;
- selecting the locations and the operators of the root name servers; and
- establishing and continually and reliably operating the root name servers.

The diverse collection of institutions that perform these functions includes a not-for-profit corporation (ICANN,[22] as the operator of the IANA[23]); a U.S. government agency (NTIA[24]); a corporation (VeriSign, Inc., as the contracted root zone administrator); and an informal group

[22] The Internet Corporation for Assigned Names and Numbers (http://www.icann.org).
[23] The Internet Assigned Numbers Authority (http://www.iana.org), operated under contract to the U.S. Department of Commerce [10] and a Memorandum of Understanding (MOU) with the IETF [42].
[24] The National Telecommunications and Information Agency of the U.S. Department of Commerce.

consisting of commercial, non-commercial, and governmental root name server operators. These players, and the roles they play in the root zone management process, are depicted in Figure 1.

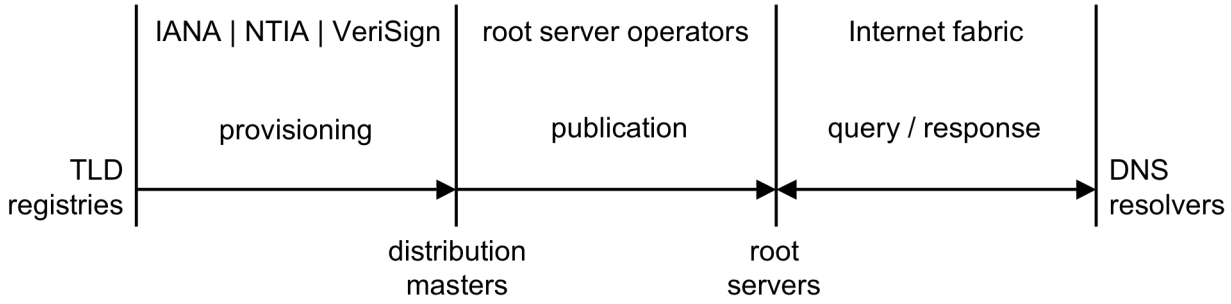

Figure 1—Root zone management players

2.4.1 The TLD registries

Each TLD has one sponsoring organization, a designated administrative contact, and a designated technical contact. These three are known jointly as the TLD manager. The administrative and technical contacts can be the same person, and either or both of these contacts can be pseudo-identities.[25] IANA tries to know each of the actual persons directly; the total number of these individuals currently known by IANA is between 400 and 500.

2.4.2 The Internet Assigned Numbers Authority

The Internet Assigned Numbers Authority (IANA) is responsible for coordinating the codes, numbers, and names that must be globally managed on behalf of the otherwise decentralized Internet [10] [42]. IANA's activities can be broadly grouped into three categories:

- Domain names—IANA manages the information contained in the DNS root, the .int and .arpa domains, and an IDN practices resource.
- Number resources—IANA coordinates the global pool of IP and AS[26] numbers, providing them to Regional Internet Registries.
- Protocol codes—IANA manages a variety of codes and numbering systems for Internet protocols, in conjunction with standards bodies such as the IETF[27].

IANA's role within the root zone management process is described in detail in Section 4.

[25] Occasionally, role identities are used, in which there is a named identity and the actual person behind the identity may change. IANA knows the vast majority, but not all, of the actual people behind these role identities. As with any human or manual process, the use of direct personal knowledge has significant implications for the ability of the system to scale, as described in Section 4.
[26] Autonomous System.
[27] Internet Engineering Task Force (http://www.ietf.org).

The IANA is currently operated by ICANN[28] under a contract from the U.S. Department of Commerce [10] and a Memorandum of Understanding (MOU) with the IETF [42].

2.4.3 The National Telecommunications and Information Agency

The U.S. Government's role in the management of the DNS root is exercised by the Office of International Affairs at the National Telecommunications and Information Agency (NTIA), which is part of the Department of Commerce. NTIA contracts with VeriSign, Inc. and ICANN for the performance of the root zone management functions that are controlled by the U.S. Government, and itself (in-house) performs the review and approval function that authorizes each individual change to the root zone.

NTIA's role within the root zone management process is described in detail in Section 4.

2.4.4 The VeriSign corporation

VeriSign acts as the "root zone administrator" under contract to the NTIA. Amendment 11 to the "Cooperative Agreement between NSI and U.S. Government" [2], dated October 7, 1998, defines the responsibility of the root zone administrator—then Network Solutions, Inc. (NSI), now VeriSign—within the root zone management process:

> *NSI agrees to continue to function as the administrator for the primary root server for the root server system and as a root zone administrator until such time as the USG instructs NSI in writing to transfer either or both of these functions to NewCo[29] or a specified alternate entity. While NSI continues to operate the primary root server, it shall request written direction from an authorized USG official before making or rejecting any modifications, additions or deletions to the root zone file.*

It is important to distinguish the separate roles played by VeriSign as the contracted root zone administrator and as the operator of two root servers (A and J). Those functions are distinct, are performed by different groups, and could be performed equally well by two separate organizations. There is no inherent requirement that they be performed either by VeriSign or by any one organization.

Both of VeriSign's roles within the root zone management process are described in detail in Section 4.

[28] Formally, ICANN is "the IANA functions operator."
[29] Now ICANN.

2.4.5 The root server operators

The current root name server operators were not selected through a formal evaluation and qualification process, although they play a fundamental role in ensuring the availability and reliability of the root. Rather, the group is the cumulative result of a sequence of separate decisions taken over the years since the establishment of the DNS. It is a loosely organized collection of autonomous institutions whose names are given in Table 1. Ten of them are based in the United States. Of those, three are associated with the U.S. government (National Aeronautics and Space Administration (NASA), Department of Defense (DoD), and the U.S. Army), two are universities (University of Maryland and University of Southern California), two are corporations (VeriSign, Inc. and Cogent Communications), and two are not-for-profits (Internet Systems Consortium, Inc. (ISC) and Internet Corporation for Assigned Names and Numbers (ICANN)). Three are based outside the United States: one in Sweden (Autonomica AB—a private not-for-profit corporation), one in the Netherlands (The RIPE Network Coordination Centre—a cooperative body of European Internet Service Providers), and one in Japan (the WIDE Project—an academic project).

As shown in Table 1, 10 of the 13 letters are hosted at multiple sites, in many countries, and today the root zone is served at more than 190 sites around the globe. A map showing the locations of all instances can be found at http://www.root-servers.org.

Server	Operator	Locations	IP Addresses
A	VeriSign, Inc.	**Sites: 4** Global: 4 Local: 0 **Los Angeles, CA, US**; **New York, NY, US***; **Palo Alto, CA, US***; **Ashburn, VA, US***	IPv4: 198.41.0.4 IPv6: 2001:503:BA3E::2:30
B	Information Sciences Institute	**Sites: 1** Global: 0 Local: 1 Earth	IPv4: 192.228.79.201 IPv6: 2001:478:65::53
C	Cogent Communications	**Sites: 6** Herndon, VA, US; Los Angeles, CA, US; New York, NY, US; Chicago, IL, US; Frankfurt, DE; Madrid, ES	IPv4: 192.33.4.12
D	University of Maryland	**Sites: 1** Global: 1 Local: 0 **College Park, MD, US**	IPv4: 128.8.10.90
E	NASA Ames Research Center	**Sites: 1** Global: 1 Local: 0 **Mountain View, CA, US**	IPv4: 192.203.230.10

F	Internet Systems Consortium, Inc.	**Sites: 49** Global: 2 Local: 47 Ottawa, Canada*; **Palo Alto, CA, US***; San Jose, CA, US; New York, NY, US*; **San Francisco, CA, US***; Madrid, ES; Hong Kong, HK; Los Angeles, CA, US*; Rome, Italy; Auckland, NZ*; Sao Paulo, BR; Beijing, CN; Seoul, KR*; Moscow, RU*; Taipei, TW; Dubai, AE; Paris, FR*; Singapore, SG; Brisbane, AU*; Toronto, CA*; Monterrey, MX; Lisbon, PT*; Johannesburg, ZA; Tel Aviv, IL; Jakarta, ID; Munich, DE*; Osaka, JP*; Prague, CZ*; Amsterdam, NL*; Barcelona, ES*; Nairobi, KE; Chennai, IN; London, UK*; Santiago de Chile, CL; Dhaka, BD; Karachi, PK; Torino, IT; Chicago, IL, US*; Buenos Aires, AR; Caracas, VE; Oslo, NO*; Panama, PA; Quito, EC; Kuala Lumpur, Malaysia*; Suva, Fiji; Cairo, Egypt; Atlanta, GA, US; Podgorica, ME; St. Maarten, AN*	IPv4: 192.5.5.241 IPv6: 2001:500:2f::f
G	U.S. DOD Network Information Center	**Sites: 6** Global: 6 Local: 0 **Columbus, OH, US**; **San Antonio, TX, US**; **Honolulu, HI, US**; **Fussa, JP**; **Vaihingen, DE**; **Naples, IT**	IPv4: 192.112.36.4
H	U.S. Army Research Lab	**Sites: 1** Global: 1 Local: 0 **Aberdeen Proving Ground, MD, US***	IPv4: 128.63.2.53 IPv6: 2001:500:1::803f:235
I	Autonomica	**Sites: 34** **Stockholm, SE**; Helsinki, FI; Milan, IT; London, UK; Geneva, CH; Amsterdam, NL; Oslo, NO; Bangkok, TH; Hong Kong, HK; Brussels, BE; Frankfurt, DE; Ankara, TR; Bucharest, RO; Chicago, IL, US; Washington, DC, US; Tokyo, JP; Kuala Lumpur, MY; Palo Alto, CA, US; Jakarta, ID; Wellington, NZ; Johannesburg, ZA; Perth, AU; San Francisco, CA, US; Singapore, SG; Miami, FL, US; Ashburn, VA, US; Mumbai, IN; Beijing, CN; Manila, PH; Doha, QA; Colombo, LK; Vienna, AT; Paris, FR; Taipei, TW	IPv4: 192.36.148.17
J	VeriSign, Inc.	**Sites: 62** Global: 55 Local: 5 **Dulles, VA, US (2 sites)**; Dulles, VA, US (1 sites); **Ashburn, VA, US***; **Vienna, VA, US**; **Miami, FL, US**; **Atlanta, GA, US**; Seattle, WA, US; Chicago, IL, US; **New York, NY, US***; **Honolulu, HI, US**; **Mountain View, CA, US (1 sites)**; Mountain View, CA, US (1 sites); **San Francisco, CA, US (2 sites)***; **Dallas, TX, US**; **Amsterdam, NL**; **London, UK**; **Stockholm, SE (2 sites)**; **Tokyo, JP**; **Seoul, KR**; Beijing, CN; **Singapore, SG**; **Dublin, IE**; **Kaunas, LT**; Nairobi, KE; **Montreal, CA**; **Sydney, AU**; **Cairo, EG**; Cairo, EG; **Warsaw, PL (2 sites)**; **Brasilia, BR**; **Sao Paulo, BR**; **Sofia, BG**; **Prague, CZ**; **Johannesburg, ZA**; **Toronto, CA**; **Buenos Aires, AR**; **Madrid, ES**; **Fribourg, CH**; **Hong Kong, HK (2 sites)**; **Turin, IT**; **Mumbai, IN**; **Oslo, NO**; **Brussels, BE**; **Paris, FR (2 sites)**; **Helsinki, FI**; **Frankfurt, DE**; **Riga, LV**; **Milan, IT**; **Rome, IT**; **Lisbon, PT**; **San Juan, PR**; **Edinburgh, UK**; **Tallin, EE**; **Taipei, TW**; **New York, NY, US***; **Palo Alto, CA, US***	IPv4: 192.58.128.30 IPv6: 2001:503:C27::2:30

K	RIPE NCC	**Sites: 18** Global: 6 Local: 12 **London, UK***; **Amsterdam, NL***; **Frankfurt, DE**; Athens, GR*; Doha, QA; Milan, IT*; Reykjavik, IS*; Helsinki, FI*; Geneva, CH*; Poznan, PL; Budapest, HU*; Abu Dhabi, AE; **Tokyo, JP**; Brisbane, AU*; **Miami, FL, US***; **Delhi, IN**; Novosibirsk, RU; Dar es Salaam, TZ	IPv4: 193.0.14.129 IPv6: 2001:7fd::1
L	ICANN	**Sites: 2** Global: 2 Local: 0 **Los Angeles, CA, US***; **Miami, FL, US***	IPv4: 199.7.83.42 IPv6: 2001:500:3::42
M	WIDE Project	**Sites: 6** Global: 5 Local: 1 **Tokyo, JP (3 sites)***; Seoul, KR; **Paris, FR***; **San Francisco, CA, US***	IPv4: 202.12.27.33 IPv6: 2001:dc3::35

Bold = Global site
* = IPv6 enabled site
191 servers in total

Table 1: Root servers and their locations

The role of the root server operators is to maintain reliable, secure, and accurate operation of the servers containing the current root zone on a 24-hour-a-day, 365 days-per-year basis. Each server is expected to have the capacity to respond to many times the rate of queries it receives and must increase its capacity at least as fast as the query rate increases, but also as the properties of the root zone itself changes, adapting to growth, new types of records, and new expected behavior. Attempts to define the responsibilities of the root name server operators were made first in RFC 2010 [27], issued in October 1996, and later again in RFC 2870 [14], issued in June 2000, but some of the ideals that they describe were outdated already when the documents were finally published, and the root servers system has evolved beyond that point long ago.

Historically, the operators of the root servers have not charged fees for resolving Internet address queries, instead obtaining support in other ways for the substantial costs they incur in providing the service. These operators choose to do so because they believe that operating a root server is an important public service (and sufficiently inexpensive that they can afford to absorb the cost) and/or that operating a root server conveys a business, or other, advantage to them.

Nevertheless, it is a valuable service, whose provision is a little-known and little-appreciated gift in kind to all users of the Internet.

The root server operators have no formal relationship to each other, and with one exception they have no specific contractual obligations to each other, to ICANN, or to NTIA (the exception is VeriSign, which has a contractual relationship with NTIA covering its operation of the A root

and its role as the root zone administrator). In the past 10 years, few changes to the stewardship of the root servers have been made—and then only in ways that were the result of "organic" developments of the root server operator organizations (*e.g.*, corporate mergers).

2.4.6 DNS resolvers

Throughout the global Internet, systems that need to discover the binding between a domain name and an IP address employ DNS resolvers to send queries ("where is the resource with the domain name *mangelwurzel.example.org*?") to name servers and receive the responses ("it's at the IP address 192.168.48.3").[30] The queries and responses are defined by the DNS protocol [28], and are usually carried across the Internet in User Datagram Protocol (UDP) packets (although under certain circumstances, described elsewhere in this report, the queries and/or responses may be carried over Transmission Control Protocol (TCP) connections).

Internet end systems send queries to a DNS resolver. The end system is configured with the IP address of the DNS resolver. The configuration is either static or dynamic (using for example DHCP). The DNS resolver is configured with the IP addresses of the root servers. At startup time, it sends a so called "priming query" (described in Section 6.8) to those IP addresses to find out the current set of root servers. After this priming of the cache in the DNS resolver, the DNS resolver is ready to respond to queries from end systems. The DNS resolver when getting a query first looks in its cache, and if the response is not there, it queries the authoritative servers in the world, starting with the root name servers, and places all responses in its cache, caching the responses according to so-called "time to live" information defined by the authoritative servers. In some cases the DNS resolver is configured to not send queries to the authoritative servers, but instead to some other DNS resolver, in which case this second DNS resolver views the first as an end system.

It is these DNS resolvers—also called forwarding servers, caching name servers, or Iterative Mode Resolvers (IMRs)—that send most of the queries from the Internet to the root servers. These systems are the "consumers" of the data in the root zone. As virtually anyone on the Internet can create a DNS resolver at any time, there is no way to precisely determine how many DNS resolvers are "out there," where they are, what software they are running, or other details of their configuration.

The characteristics and behavior of resolvers, and the way in which DNS query/response traffic flows through the fabric of the Internet, have important effects on the root system—specifically, on the query side of the publication system described in Section 4.3.

[30] The operation of the DNS is of course more complicated than this, but for the purposes of the root scaling study the simplified description given here suffices.

2.5 Impending changes to the root

The upcoming changes to the root described in Section 1—some of which are already well underway—will increase the size of the root zone file and, more importantly, its volatility—the growth rate of the root zone, and the frequency with which changes to it are made. This Section describes these changes; their effects on the root system are discussed in Section 6.

2.5.1 New top-level domains

Adding resource records to the root zone that define new top-level domains (TLDs) will obviously increase its size. In the current root zone, a TLD delegation consists of, on average, 238 bytes (or an average of 11 DNS records). The RRsets for new zones can be expected to be somewhat bigger, since the current root zone is dominated by two-character ccTLDs, whereas newer TLDs tend to be 3-6 character names, but the order of magnitude is comparable.

The very nature of the term "new TLDs" implies that they are added at a rate decided by the zone administrator. The zone administrator has a strong influence on the rate at which new names are introduced to the root, but pressure (*e.g.*, political, financial, or legal) exerted by other entities may in turn influence the zone administrator's ability to exercise control over that rate. It is also noted that the growth is virtually open ended. There is no natural boundary at which the growth is expected to stop.

2.5.2 Support for DNS security (DNSSEC)

Secure DNS (DNSSEC) [43][31] adds security extensions to the DNS system. To accomplish that, each normal DNS record is supplemented by one or more accompanying cryptographic signature records. To allow for validation of the signatures, separate extra records containing cryptographic keys also must be added.

Proposals for "signing the root" are currently being evaluated by NTIA, ICANN, and VeriSign.[32]

2.5.3 Support for IPv6

Internet Protocol version 6 (IPv6) [7] is a new packet-layer protocol for the Internet, intended to replace the current IP version 4 (IPv4). It utilizes a different addressing scheme. To enable DNS to contain IPv6 address information, and to use IPv6 for transportation of DNS data, the DNS database must contain these addresses. It is likely to become more and more important to "be visible" (from a DNS standpoint) both over IPv4 (traditional Internet technology) and IPv6 (new

[31] A large number of RFCs in addition to the original RFC 2525 have been produced by the IETF to document the DNS security extensions. These are listed at http://www.dnssec.net/rfc.
[32] http://www.icann.org/en/announcements/dnssec-qaa-09oct08-en.htm

Internet technology). To enable this, new records must be added for the name server hosts mentioned in the DNS databases.

IPv6 records are already present in the root zone, are being added at a steady pace, and currently represent 15% of all IP addresses in the root zone.

2.5.4 Internationalized domain names (IDNs)

Domain names and the DNS system were designed in an era when communication across the Internet was conducted almost exclusively in English, with computers, software and operating systems that used 7 bits per character, and without any concept of "character sets." The one and only character set being used was ASCII, and it had 128 defined symbols. The design of the DNS allowed because of these reasons and more only for the use of the 26 ASCII letters [a-z], the ten digits [0-9], and hyphen [-]. It was further defined that two domain names were equivalent regardless of the case of the letter.

In today's internationalized Internet, this is obviously quite insufficient, and this has led to the development of Internationalized Domain Names (IDNs). Using IDNs, a vast number of characters used in a great variety of scripts can be encoded, stored, and retrieved in the DNS system. All characters must though come from (a subset of) the Unicode Character Set, as DNS (see above) do not have any concept of being able to handle multiple character sets. The encoding is such that a label with one or more characters not in the earlier defined set is transformed into a string of ASCII. It is prefixed with the string "xn--" for two reasons:

- to ensure that there is no collision in the name space with non-encoded strings, or other potential future encodings; and
- to indicate to an IDN-aware application which receives a label that starts with "xn--" that it can transform the string of ASCII letters (back) into a string of Unicode letters.

As a result of this encoding scheme, the strings that are actually stored in and retrieved from the DNS system (as well as in other protocols that uses domain names as protocol identifiers, like SMTP) are still just made from the subset of ASCII consisting of letters, digits and hyphen, [-a-z0-9] (also called LDH). From a purely DNS-technical standpoint, these domain names are just normal domain names, although they have the special indicator "xn--" in such labels. For example, "fältström.example.com" is transformed into "xn--fltstrm-5wa1o.example.com".

It should be noted that due to the normalization of strings in Unicode that is required by the IDN standards, the mapping from a random (not rejected) Unicode string to its ASCII form (starting with xn--) and back is not always transitive. This is specifically the case with the current IDNA standard (defined in 2003) [38], and the reason is that it is defined how certain codepoints should

be mapped to other codepoints before the encoding. An updated version of the standard is finalized by the IETF as of August 2009, and although a mapping step is still recommended in many situations, the new version of the standard explicitly defines the terms A-label and U-label such that the mapping is transitive. The A-label (starting with "xn--") is what is used in the DNS namespace, and by the DNS protocol; the U-label is the equivalent set of Unicode codepoints.

3 The root scaling study

3.1 Motivation

Until recently the root zone of the DNS has enjoyed two important stabilizing properties:

- it is relatively small—the root zone currently holds delegation information for 280 generic, country-code, and special-purpose TLDs, and the size of the root zone file is roughly 80,000 bytes; and
- it changes slowly—on average, the root zone absorbs fewer than one change per TLD per year, and the changes tend to be minor.

The root system has therefore evolved in an environment in which information about a small number of familiar TLDs remains stable for long periods of time.

Adding IDN TLDs, support for DNSSEC, IPv6 address data, and new gTLDs will change the type, amount, and volatility of the information that is contained in the root zone. Because these changes are unprecedented, ICANN and the other organizations that participate in the root system considered it prudent to study their potential effects in order to determine if, how, and to what extent increasing the size and volatility of the root zone will affect the management and operation of the root system.

3.2 Previous work

Previous studies of the root system and the way in which it has evolved have for the most part focused either on specific individual elements of the system (such as the use of IP anycast to distribute root server instances geographically and topologically) or on the DNS as a whole. The work reported here is believed to be the first comprehensive, thorough, and authoritative study of the entire root system and the way in which it exists and evolves within the context of the DNS and the Internet.

3.2.1 History

In the first five years or so of the existence of the DNS, there were a large number of adaptations, modifications, and changes as the system became better understood and utilized. The use of a standard "SBELT" system to bootstrap the DNS was agreed on, a variety of transports and protocols were explored and abandoned, and to the extent that the DNS was visible or useful in the policy arena, there was some "jockeying" for control.

It was not entirely clear in those early days how the namespace itself should be structured—there were a number of TLDs that came into existence and then disappeared (NATO, SAT, *etc.*). This

period of DNS evolution could be considered the "Cambrian Era," in which many forms emerged and competed, and eventually some forms became dominant.

The last major change to the root zone occurred in the early 1990s with the introduction of so-called "ccTLDs" based on the ISO-3166 country codes. This addition of over 250 new TLDs was massive, since the previous root zone contained 15 TLDs. In nearly every case, the operation of these new TLDs was assigned by the IANA, Jon Postel, to one or more technically adept folks around the world, as stewards—even when, for the most part, these stewards did not reside in or were not a part of the political or administrative bodies in the target countries. This selection of stewards was not surprising, since the root system needed technical expertise—and reasonable connectivity.

Larry Landweber began a study to examine the penetration of the Internet on a global scale in the early 1990s. It showed that initially, the best connectivity was in North America and western Europe. Parts of the Internet—not using normal telephony—extended IP to parts of Asia, Mexico, and elsewhere. Larry periodically re-ran his surveys, which documented the growth of Internet technologies throughout the world.[33]

In the mid-1990s, it became clear that the location of the original root name servers, with eight in the USA and one in Europe, was not optimal or desirable for what was becoming a global network. In 1995, Mark Kosters and Bill Manning devised a plan to change the names of the root servers, which would allow (using DNS label compression) the addition of five more root servers and operators. The IANA approved four, and in consultation with the existing root operators the change was made in the names of the servers and four new roots were added: J, K, L, and M. These were initially housed at VeriSign (J and K) and ISI (L and M). This work was completed in 1996.

This left the IANA with the task of selecting operators for these new servers. The general method chosen was to find a regional organization which had the support of the Internet/ISP community. The first selection was the RIPE-NCC, which was tasked with the operation of K. This node function was moved from VeriSign to RIPE-NCC and renumbered. The second was the M node. It was sent to the WIDE project in Japan—the first root in Asia—and renumbered. Then in 1998 Jon Postel passed away, ICANN was formed, and the remaining two nodes were not moved.

This particular path for managing emergent growth appeared to have closed. The existing operators, either unilaterally or in small groups, began to explore how to better provide DNS

[33] http://pages.cs.wisc.edu/~lhl/maps

service to the now nearly completely saturated globe.[34] What was settled on as a means to get DNS service in play in underserved areas was the use of IPv4 anycast. This is a technique which depends on Internet routing and the fact that to date, most DNS transactions are UDP based. Anycast started as early as 2000 and by 2003 was in widespread use. As described elsewhere in this report, by using anycast nearly 200 root name servers have been deployed on all continents except Antartica.[35]

3.2.2 Antecedents

Peter Danzig *et.al.* [39] conducted one of the first directed studies of the DNS root system in 1992, with a focus on cache behavior. Much of this work focused on the development of HTTP-based proxies such as Squid, and less on the structure, modification, and evolution of the root system itself.

About a decade passed before the academic research community picked up the DNS itself as a topic worthy of serious investigation. Shortly after the active deployment of anycast, a number of reports, studies, and presentations emerged—many based on Danzig's work—that focused on traffic flows to and from the root servers, topological placement of root servers, and several novel suggestions for the mitigation of a perceived threat of capture or removal of a specific root server (CHORD, BEEHIVE, etc.).

The "Signposts in Cyberspace" [26] report from the National Research Council appears to have been the first broad analytical examination of the DNS and the root system. Several rigorous measurement-based studies of the DNS and the root system have been conducted by the Cooperative Association for Internet Data Analysis (CAIDA)[36] [22], the DNS Operations, Analysis, and Research Center (DNS-OARC)[37] [24], the Widely Integrated Distributed Environment (WIDE)[38] Project [15] [16], and NLnet Labs[39] [25], and others.

3.3 Scope

As previously defined, the term "root system" encompasses the entire spectrum of actors and activities involved in the maintenance and operation of the root zone. Figure 1, in Section 2.4, illustrates this spectrum, which extends from the registries that originate requests to add, delete,

[34] http://pages.cs.wisc.edu/~lhl/maps
[35] At one time a root server was in fact deployed in Antarctica, but the bandwidth requirements made it infeasible to maintain it.
[36] http://www.caida.org
[37] https://www.dns-oarc.net
[38] http://www.wide.ad.jp
[39] www.nlnetlabs.nl

or change root zone information to the DNS resolvers throughout the Internet that submit queries and receive responses.

The scope of the present root scaling study extends from the interaction between IANA and the TLD registries, at the beginning of the root zone update process, through the operation of the root servers (and, where applicable, their anycast satellites) that deliver root zone information in responses to queries from resolvers. Specifically excluded from this scope are

- the TLD registries, except as the sources of add/delete/change requests;
- the DNS resolvers, except as the sources of queries; and
- the Internet fabric through which queries and responses are exchanged.

This means, for example, that the query load on a root server is relevant (in scope), but the mechanism through which those queries arrive at the server (*e.g.*, the characteristics of the path between a resolver and a root server, the intermediation of caching resolvers, or other resolver details) is not.

3.4 Terms of Reference and study timeline

A resolution of the ICANN Board adopted on 3 February 2009[40] noted that "over a short period of time, significant changes to root zone operations are anticipated, including the still-recent addition of IPv6 records to the root, and planned implementation of DNSSEC, IDNs, new cc IDNs, and new generic top level domains," and called for the Security and Stability Advisory Committee (SSAC), the Root Server System Advisory Committee (RSSAC), and ICANN Staff to jointly undertake a formal analysis of "the aggregate impact of these anticipated changes … for impacts on scalability and stability":

> *Given that these changes—IPv6 records in the root zone, DNSSEC, IDNs, and new TLDs (country code and generic)—have not been analyzed for their combined impact on root zone operations, the ICANN Board requests the SSAC and RSSAC jointly conduct a study analyzing the impact to security and stability within the DNS root server system of these proposed implementations. The analysis should address the implications of initial implementation of these changes occurring during a compressed time period. ICANN must ensure that potential changes in the technical management of the root zone and scope of activity at the TLD level within the DNS will not pose significant risks to the security and stability of the system. The study should address the capacity and scaling of the root server system to address a stressing range of technical challenges and*

[40] http://www.icann.org/en/minutes/prelim-report-03feb09.htm

operational demands that might emerge as part of the implementation of proposed changes.

Responding to this resolution, RSSAC, SSAC, and Staff created a Root Scaling Steering Group consisting of 4 representatives from each organization. This steering group developed Terms of Reference [6] for a root scaling study, and posted these for public comment[41] on 29 May 2009.

The steering group engaged Lyman Chapin to assemble and lead a team of experts to conduct the study specified by the Terms of Reference. The study team consisted of the following members:[42]

- Jaap Akkerhuis, NLnet Labs
- Lyman Chapin, Interisle Consulting Group
- Patrik Fältström, Cisco
- Glenn Kowack, RiverOnce
- Lars-Johan Liman, Autonomica
- Bill Manning, ep.net

The study team began its work in mid-May 2009, and completed the study on 31 August 2009.

3.5 Methodology

Figure 2 illustrates the process that the study team followed to gather and analyze information about the root system and construct qualitative and quantitative models of the root.

[41] http://www.icann.org/en/public-comment/#root-scaling

[42] Organizational affiliations are provided for transparency only; each study team member participated as an individual expert, not as a representative of any organization.

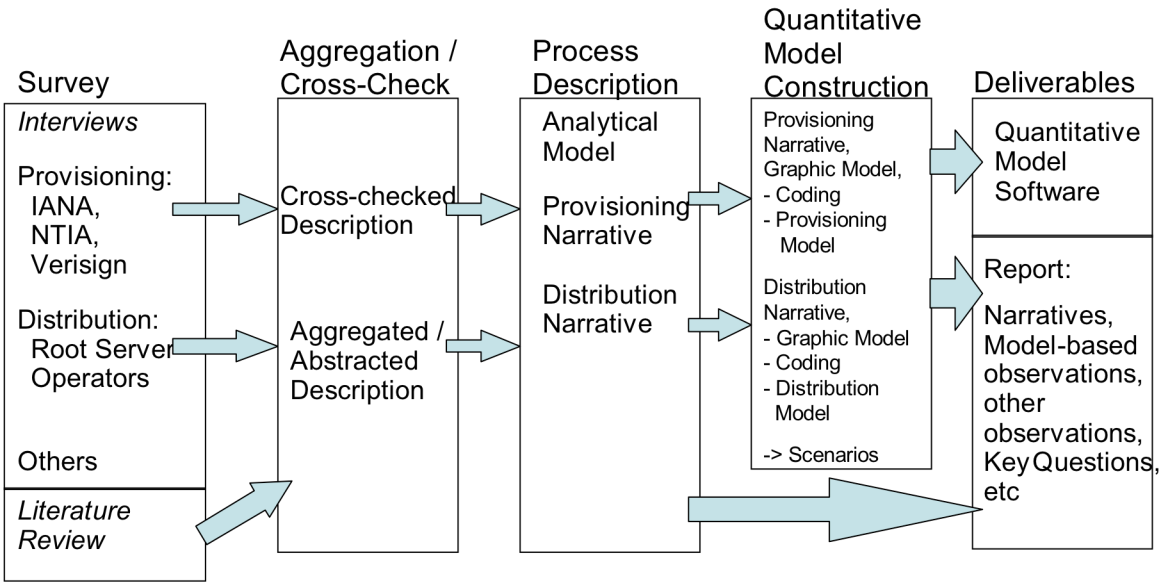

Figure 2—Root scaling study process

3.5.1 Sources

The principal documentary sources used by the study team are listed in Section 9. The team also gathered information from and corroborated elements of its analysis with a broad range of individuals with insight into one or more aspects of root system operation:

- ICANN staff responsible for the IANA task,
- ICANN staff responsible for root server operations,
- Other ICANN staff,
- Department of Commerce—NTIA staff,
- VeriSign staff responsible for root zone generation and publication,
- VeriSign staff responsible for root server operations, and
- Staff responsible for root server operations at University of Southern California, Cogent Communications, University of Maryland, NASA AMES and its contractors, Internet Systems Consortium, United States Department of Defense—Network Operations Center, United States Army—Aberdeen Proving Grounds, Autonomica, RIPE-NCC, and the WIDE project with JPRS.

Additional conversations were held with selected TLD operators, DNS service providers, ISPs, registries, registrars, and government officials.

3.5.2 Outreach

Presentations and outreach to the Internet community were conducted in the following meetings:

- Study Team meeting with Microsoft
- Study Team meeting with Boeing
- RIPE-58 (Amsterdam)
- NANOG-46 (Philadelphia)
- DNSSEC Symposium (Reston, VA)
- ICANN-35 (Sydney)
- SANOG-14 (Chennai)
- LACNIC XII (Panama)
- IETF-75 (Stockholm)

3.5.3 Public comments

On 29 May 2009 ICANN opened a public comment period for the root scaling study,[43] which closed on 31 July 2009. During that period five substantive[44] comments were submitted. One comment expressed support for the Terms of Reference governing the root scaling study; one comment raised a list of issues that either are already covered by the Terms of Reference or are out of scope for this study (dealing with new gTLDs rather than with root scaling *per se*). The other three comments raised the following specific issues:

- Assuming that the root system will have to absorb the effect of priming and normal responses larger than 512 bytes in order to implement DNSSEC and to add IPv6 address data to the root zone, would it be possible to reconsider the issue of expanding the system to include more than 13 root servers?
- The small size of the current root zone makes it easy "for caches to regularly and efficiently fetch compressed and digitally signed copies out of band (*e.g.*, using HTTP, rsync, or BitTorrent), and caches that do so benefit from improved DNS resolution efficiency while also lessening load and dependence on central network infrastructure." Containing the growth of the root zone within Moore's Law might be a reasonable way to preserve this property.
- Allowing the size of the root zone to increase may frustrate future innovations.

The study team was able to take all of the substantive comments into account in conducting its investigation and preparing this report.

[43] http://www.icann.org/en/public-comment/public-comment-200907.html#root-scaling
[44] In this context "substantive" simply means "other than subscribe requests."

4 Analytical model of the root system

The analytical (or qualitative) model of the root system is constructed from a set of six primary actors—types and instances of organizations that perform the functions necessary to operate the root—and narrative descriptions of the processes in which those actors participate.

As illustrated in Figure 3, the model identifies four regions in which root system activities take place:

- the **TLD registries**, which originate requests to add, delete, or modify information in the root zone;
- the **provisioning** system, which processes change requests, maintains the authoritative database of root zone information, and periodically creates a root zone file that contains the authoritative root zone information;
- the **publication**[45] system, which distributes the root zone file to the root servers (including anycast clones) and arranges for the information in the root zone file to be available for the construction of responses to queries; and
- the **DNS resolvers** distributed throughout the global Internet, which send queries (requests for information) to the root servers and receive responses from them.

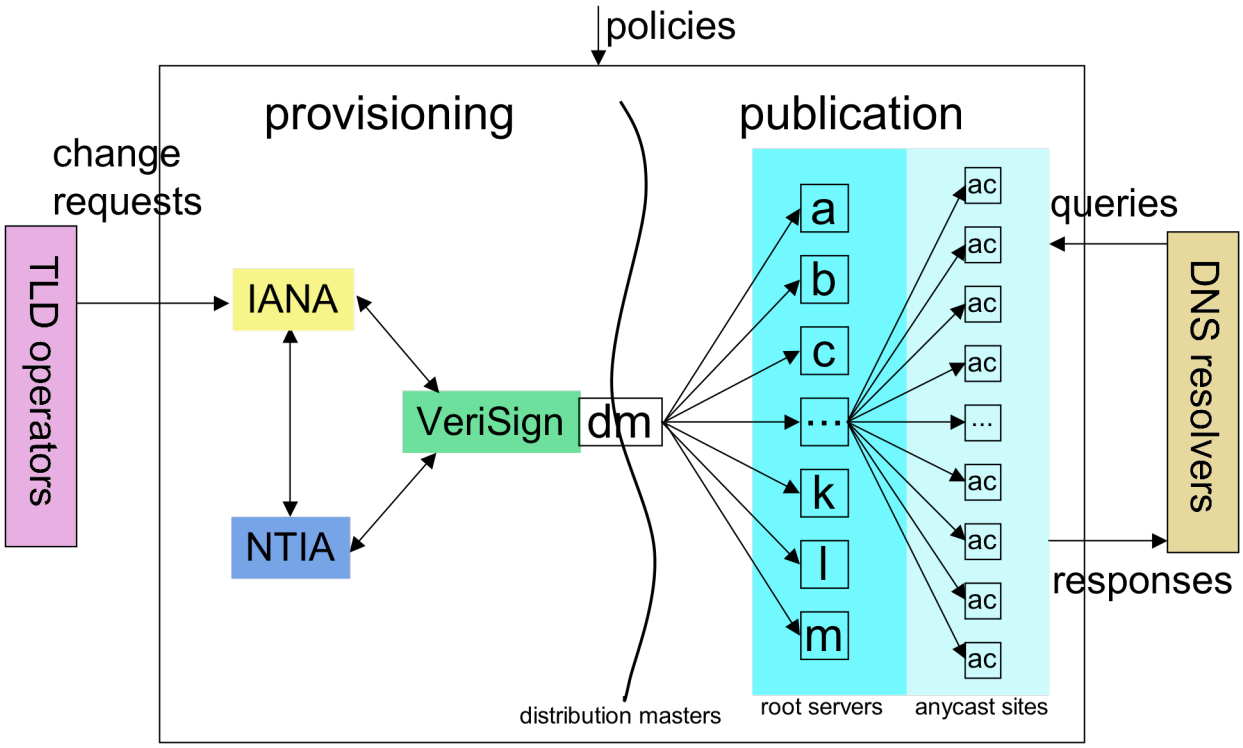

Figure 3—The root system model

[45] The terms "publication" and "distribution" are used interchangeably to refer to this process.

As explained in Section 2, the TLD registries and the DNS resolvers are mostly outside the scope of this model. Their impact on the root system is felt at the margins of the provisioning and publication systems.

4.1 Root system roles

The root server operators operate a service that has developed and keeps developing in a dynamic fashion. They try to measure their environment, and to respond to changes as needed, but their view of the world is limited. Two of the most important parameters the root server operators need to take into their equations when deciding on resources is the size and the update frequency of the root zone. Until recently it has been understood to be very stable, with few updates, and very limited expansion of TLDs. The operators maintain a certain "headroom" between the observed data (the actual size of the zone, and the actual update frequency) and the capacity of their systems in that respect. Unless someone gives them advance warning, they will use the observed data as input to the process that tells them how they need to provision for the service.

An illustrative analogy is provided by the situation of a pilot flying an airplane in fog, with a vertical radar measuring the distance to the ground. When you observe (through the radar) that the ground rises, you tell the pilot to climb to maintain the safety gap (the headroom). However, if the ground rises too quickly, you may not be able to react quickly enough, or your airplane may not be able to climb quickly enough, or you may have reached the maximum altitude for that type of airplane, and will have to rebuild it in flight.

The root server operators are in a similar situation. If the root zone properties change slowly enough, the root server operators' normal processes for operations, administration, maintenance, and provisioning (the familiar "OAM&P" of any information technology enterprise) will allow them to adapt, and find ways to keep providing good service with sufficient headroom. If the rate of change (increase in the size of the zone, or number of changes to the zone, or other properties) becomes too steep, the root server operators will not be able to follow without changing their established "normal state" OAM&P.

The same situation is faced, with different details, by each of the root system actors. Figure 4 illustrates the relationship between the "steepness of the curve" representing rate of change and the discontinuities between operating regions for any individual actor.

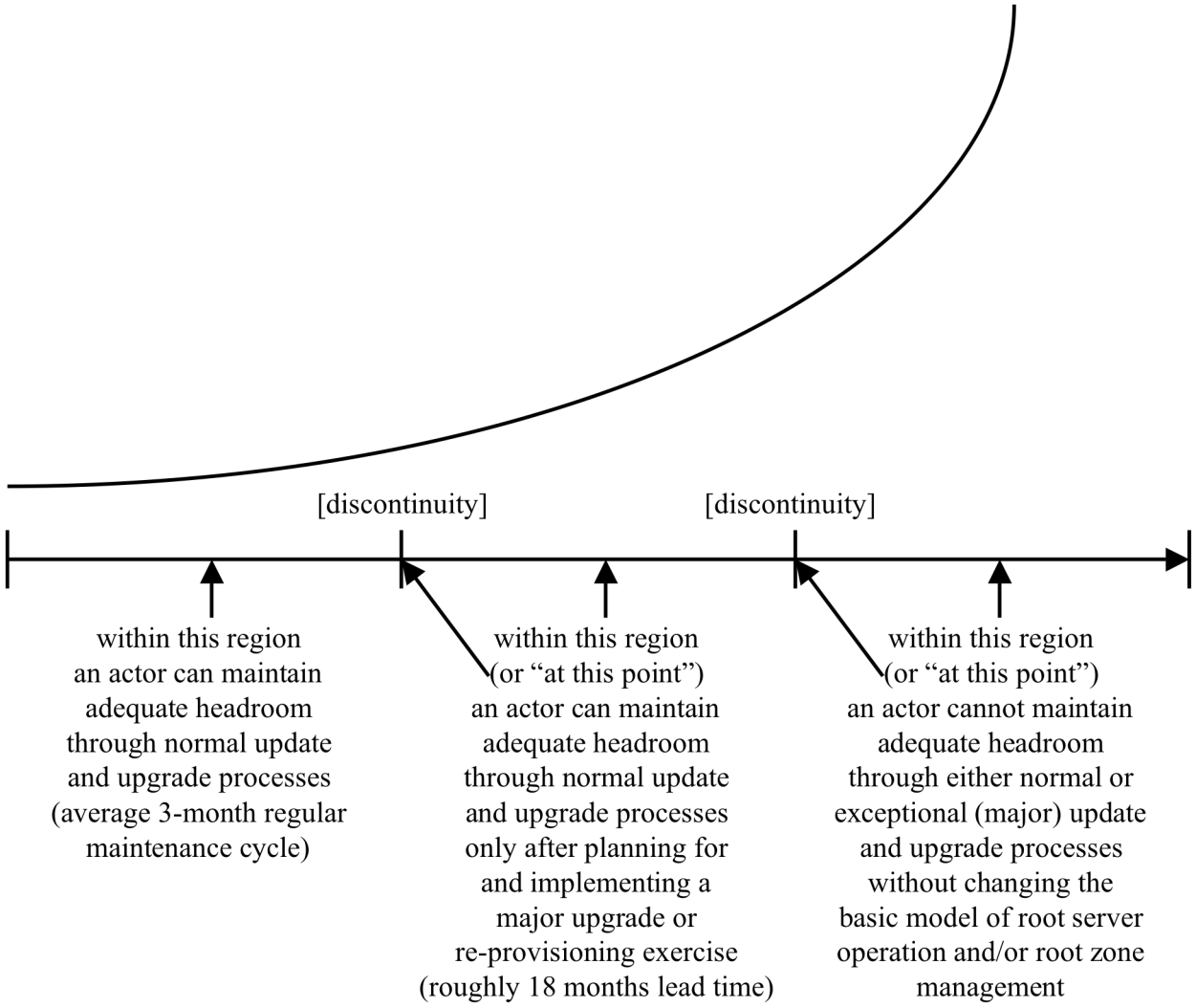

Figure 4—Dynamic operating regions

The analytical model identifies three "regions" in this dynamic view of the adaptability of the root system actors:

- **Normal operations**. The changes to the root zone are within the limits that the operators expect from the provisioning side. The operators are able to adapt within their normal upgrade processes, and funding is handled through their normal respective channels. Response time seems to be around 3 months.
- **Replanning**. The changes to the root zone are such that they require the operators to plan and perform a major upgrade *beforehand*. The lead time for issues like this seems to be on the order of 18 months, and advance warning is necessary.
- **Restructuring**. The changes to the root zone become so big and rapid that the entire root system must be restructured, with new funding models, new data channels all the way from the root zone administrator to the end server nodes (possibly even to the

client nodes!), new designs using new equipment, new software, new protocols, etc. This is a major undertaking that is likely to require several years of planning and activity.

Each player in the root system is responsible for its own headroom, and makes its own judgment regarding its own systems. The points where an operator goes from one region to the next varies from operator to operator. Also, different operators have different limitations in their systems, so the respective systems react differently to different kinds of stress. Furthermore, the other actors in the system (IANA, NTIA, andVeriSign) also have headroom, limitations, and ways to react to stress. This is normally considered a strength of the system (it doesn't all break at once), but it also makes it *very* difficult to predict the capacity of the entire system in any specific way. Increase in zone size may have very different effects on the IANA, on NTIA, on one operator, and on a different operator.

4.1.1 IANA

As described in Section 2.4.2, The Internet Assigned Numbers Authority (IANA) is responsible for coordinating the codes, numbers, and names that must be globally managed on behalf of the otherwise decentralized Internet.[46]

The critical resource for IANA's root zone management role is manpower—the staff time that is required to execute those parts of the process that require direct human involvement. Currently the demand for this resource increases linearly with the frequency of arrival of change requests.[47] An increase in the complexity of change requests—an increase in the number or inexperience of TLD registries, perhaps—would add a (probably small) super-linear component to the equation.[48]

Figure 5[49] graphs the root zone change request processing queues at IANA from August 2008 through July 2009. Averaged over a full year, IANA processes roughly one root zone change request per day.

[46] The policies that currently govern ICANN's administration of the IANA root zone management functions may be found at http://www.icann.org/en/icp/icp-1.htm.
[47] The size of the root zone does not significantly affect the demand for this resource.
[48] IANA invokes complexity in its explanation of the growth in the request processing queue: "IANA is handling increasingly complex root zone change requests, including addition of IDN TLDs to the root zone. Over time, the outstanding queue grows due to the increased time required to process these requests."
[49] http://idashboard.icann.org/idashboards/engine.swf?dashID=89&serverURL= http://idashboard.icann.org/idashboards&guest=icannguest

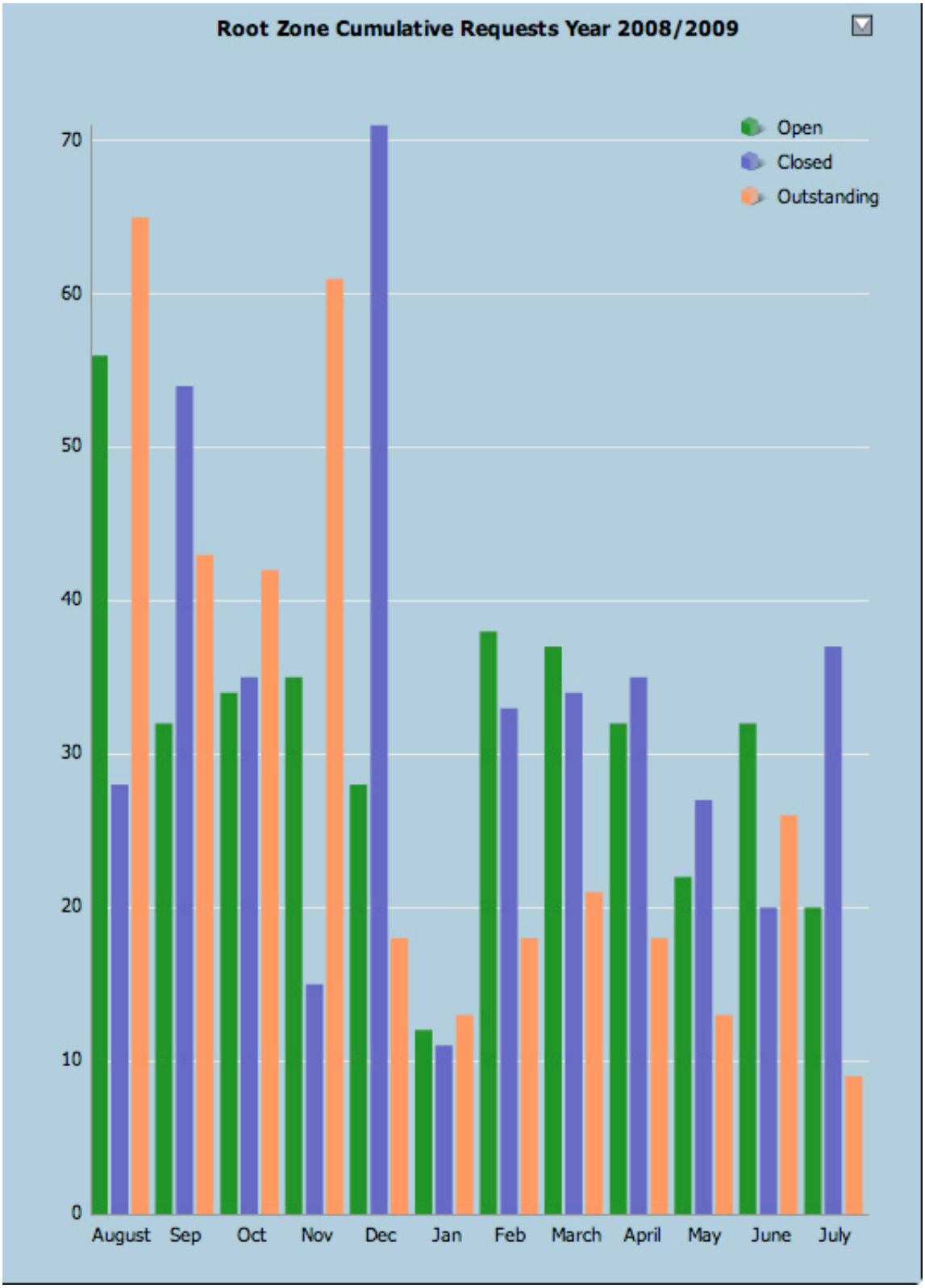

Figure 5—IANA root zone change request processing load

Additional manpower resources are required out-of-band with respect to the arrival of change requests, to maintain the trust relationships between IANA and the TLD registry administrative and technical contacts, administer the on-line change request submission tool (including, particularly, the password-based access control mechanism), and perform other management functions. The demand equation for this resource consists of a fixed overhead term and a variable term that increases linearly with the number of TLD registries.[50]

Many TLDs now use a DNS provider to host and manage their name servers, with the result that some name their DNS provider as their Technical Contact. Where the TLD has done this, the DNS provider can authorize a change along with the administrative contact. If the DNS provider is not a named contact, they can still lodge a request, but IANA will have to get authorization from the two named contacts. If the rules were changed to allow DNS providers to submit and authorize name server changes, the result would be about 20 large DNS providers and about 1500 smaller providers. Most of those DNS providers offer secondary services for the TLDs. This number of additional contacts would place substantial additional demand on IANA's ability to scale the "personal knowledge" aspect of the trust relationship.

4.1.2 NTIA

As described in Section 2.4.3, NTIA contracts with VeriSign, Inc. and ICANN for the performance of the root zone management functions that are controlled by the U.S. Government, and itself (in-house) performs the review and approval function that authorizes each individual change to the root zone.

The critical resource for NTIA's root zone management role is also manpower—the staff time that is required to execute those parts of the process that require direct human involvement. As for IANA, the demand for this resource increases linearly with the frequency of arrival of change requests; the demand equation would become super-linear if the complexity[51] of change requests increased.

The expectation that the increase in processing load will be linear (or, with added request complexity, super-linear) assumes that NTIA continues to apply the same type and level of scrutiny to individual change requests as the rate at which requests are received increases. As far as this study was able to determine, NTIA does not plan to change the way in which it exercises its oversight role in order to accomodate an increase in the frequency of arrival of root zone change requests.

[50] Strictly speaking, it increases linearly with the number of distinct administrative and technical contacts.
[51] For NTIA, "complexity" arises from the potential, with an increase in the number of TLD registries, for change requests to be received from previously unknown sources.

4.1.3 VeriSign

As described in Section 2.4.4, VeriSign acts as the "root zone administrator" under contract to the NTIA.

The critical resources for VeriSign's root zone management role[52] are manpower, server capacity, and bandwidth. As for IANA and NTIA, the demand for manpower increases linearly with the frequency of arrival of change requests, but unlike IANA and NTIA is not sensitive to the complexity of those requests. Because VeriSign's role in the root zone management process could be fulfilled by a workflow that involved less or no human involvement, its ability to scale does not necessarily depend on adding additional manpower resources.

The server capacity and bandwidth resources required by the provisioning system and by VeriSign's role in the publication system are negligible, and although increasing the frequency of change requests would place additional demands on both, the effect would be too small to be of any significance.

4.1.4 Root server operators

As described in Section 2.4.5, the role of the operators of the root name servers is to maintain reliable, secure, and accurate operation of the DNS name servers that publish the root zone for the Internet.

The critical resources for the organizations that host root server operations are institutional commitment, manpower, server capacity, and bandwidth.

Institutional commitment is not easily quantified, but it is perhaps the most important critical resource for root server operation, which takes place in the absence of formal contracts or return-on-investment expectations. Each operator's sponsoring organization is committed to providing sufficient resources to ensure continuous reliable operation of the root name server function. During the root scaling study, many operators told the study team that they were prepared to do "whatever it takes" to operate their root server.

In order to achieve extremely high availability even under adverse conditions (such as a denial of service (DOS) attack directed at the root), most operators have set their normal operating parameters so that any signal of utilization above 1% of capacity triggers a review of the affected system. They also observe the prudent network engineering rule of thumb that critical systems should maintain at least a 10% overhead in capacity to absorb either attacks or unanticipated increases in legitimate demand.[53]

[52] As distinct, in this context, from its role as a root server operator.
[53] It is often impossible for an operator to distinguish a new, legitimate source of traffic from an attack.

Some operators run with larger buffer margins, some with slightly less, depending on the state of their particular service sector, but as a whole the system has tended to keep between 50% and 100% overhead in critical resources available at all times. The trigger for capacity upgrades is usually when the demand on a resource experiences a 10% change (either up or down). This is due to the fact that in a normal operational refresh cycle, it is usually the case that lead times for hardware availability, software debugging, and configuration management are between 90 and 180 days.

Effects that drive rate of change faster than a normal operational refresh cycle hit what we call the first order discontinuity, where the operators move beyond a normal refresh cycle and have to consider either architectural changes or operational practices to maintain their commitment to serve the root to the Internet. Such a "re-planning" cycle may involve 15-18 months to plan, implement, test, and deploy new facilities or capabilities.

4.2 The provisioning system

The actors in the provisioning system are IANA, NTIA, and the VeriSign Corporation, which are described in Section 2.

4.2.1 Provisioning system data

The following data[54] are maintained in the provisioning process.

Data maintained by IANA

- A root zone information database, which is the authoritative source of the data in VeriSign's official root zone database.
- The social data database, consisting of contacts and addresses. IANA derives the published "whois" data from this database. These "whois" data are a subset of the information in the social database. IANA also maintains additional contact details and any "special instructions" for a TLD in its social data database.[55]
- An archive of email requests in its ticketing system. Prior to the creation of the ticketing system, email requests were kept in online folders. For email records from pre-ICANN management of IANA, physical folders hold copies of much of the communication with IANA, but not a complete set. IANA keeps these paper files backed up in scanned pdf files.

[54] Data are important; how they are stored and managed is important only to the extent that the peculiarities or limitations of a particular storage/management system might affect scaling. Otherwise, this section refers to the existence of specific databases only for convenience and completeness.
[55] In practice, IANA maintains a single database for the root zone and social data, so the "two databases" are conceptual rather than physical.

Data maintained by NTIA

- NTIA maintains an archive of mail requests, and strictly in the original request format. It is not known how far back NTIA keeps mail requests, but the files may go back as far as 1998.

Data maintained by VeriSign

- The root zone database containing the root zone information received from IANA.
- The root zone file, which is generated from the root zone database and made available to the root server operators on a cluster of "distribution master" servers.

4.2.2 Change requests

IANA receives and manages all requests to change information in the root zone.[56] **Name server changes** and **social data changes** originate from TLD authorities. **Operational changes** originate from within the root zone management system.

Name server changes are changes to information concerning the primary or secondary name servers. They affect the contents of the IANA database, the root zone database, and the root zone file (produced at VeriSign and then distributed to the root server operators).

The term "delegation" has historically been used in several different ways, including the assignment of name server associations to a specific TLD in the root zone, and can be confusing. If one has control over a TLD's primary name severs, then one effectively has control of that TLD. However, IANA now reserves the term **delegation change** (or "redelegation") to mean specifically a change of authority over the management of the TLD itself. IANA no longer uses the term "delegation change" to mean the addition of or changes to a name server in a TLD zone; these are now referred to simply as "name server changes," as above.

Social data changes are changes to information concerning the administrative or technical contacts or to information concerning the sponsoring organization (except for the identity of the sponsoring organization, changing which would involve a redelegation of authority). These changes affect the contents of the whois database[57] but have no effect on the contents of the root zone database or root zone file. From a process-load standpoint, they affect IANA and NTIA to the extent that resources must be provided to process them; they are in the same category with respect to the scope of our study as query/response load on the root name servers.

[56] The current change request template may be found at http://www.iana.org/domains/root/tld-change-template.txt.
[57] The whois data mentioned here applies strictly to TLDs.

Social data changes can be combined with name server changes in such a way that the net effect is a "stealth" redelegation—that is, they effectively change the management of the zone without an explicit delegation change request. This situation is described in Section 4.2.3.

Operational changes, also known as "internal changes" or "modifications of change request processes," consist of changes to the way in which IANA operates, and affect the operating characteristics of the provisioning system (how root zone change requests are processed) but do not directly affect the contents of the root zone database or root zone file. For example, adding DNSSEC to the root requires such a change of process, because it requires the use and management of new data: the various keys.

Operational changes are closely associated with policy input to the root zone management process. Operational changes could include, for example, what level of PGP[58] security to use in exchanging email, what data to hold for DS RRsets, and other particulars related to management of the root zone data.

4.2.3 Change request processing

Automated root zone management

IANA, NTIA, and VeriSign are currently implementing a root zone workflow automation system (RZ-WAS) that will streamline the processing of change requests. The differences between the former (all-manual) system and the RZ-WAS are significant with respect to the way in which the root zone management process adapts to root scaling; for this reason, and because the implementation of RZ-WAS is expected to be completed in roughly the same timeframe as the completion of the root scaling study, the qualitative and quantitative models assume that RZ-WAS is deployed.[59]

Initiating change requests

TLD authorities ask IANA to make name server and social data changes via a custom web-based on-line tool that is part of RZ-WAS. IANA strongly encourages use of the automated web tool by all TLD managers. However, IANA will accept a change request in any form that manages to get to IANA, including by email, fax, and phone. If for any reason a TLD, or anyone, is unable to get to the automated web tool (*e.g,*. they do not have sufficient bandwidth to support a web

[58] "Pretty good privacy" is a public-key cryptography system that is often used to sign and/or encrypt email messages. It was originally developed by Philip Zimmerman in 1991, and is currently the subject of an IETF standard [47].

[59] ICANN and VeriSign are currently engaged in a testing process in which change requests are managed through both the existing manual process and the RZ-WAS automated process. Only when these tests have been successfully completed will IANA switch to uniform public use of the automated tools.

data-entry session, or there are other infrastructure issues that impede access), IANA will act as intermediary and, upon receipt of the request, and confirmation that the source is legitimate, will load the information into the automated web tool. Thus, requests for changes will always ultimately be entered using the web tool.

IANA will not act on a request (howsoever received) until they confirm it with both the administrative contact and the technical contact at the requesting TLD. This "confirmation check" is an authorization/authentication check that also helps prevent "stealth" redelegations, which are described below.

User authentication

The technical details of how IANA employs a userID/password system are not significant to our model. The manner in which TLD managers get their UIDs and passwords is not a high-frequency event, and it has no impact on the parameters and variables associated with the "IANA" box in the model. Furthermore, it is not important that the authentication credential is UID/password; if IANA changed to biometric authentication, or some other system, it would have no effect on the role they play in the root zone management process.

The only scenario in which UID/password allocation might be important would be one in which large numbers of new TLD managers sought authorization within a small time window—but even then, the rate at which new TLD managers obtain authorization to use the RZ-WAS does not appear to be a significant variable.

Password distribution

IANA currently practices a "high overhead" password distribution method with the ccTLD managers. This distribution method requires face-to-face contact with each TLD authority when userIDs and passwords are allocated. Reissuing passwords, including cases where a password was lost, would require an additional direct interaction. This could incur significant lag for a given TLD wishing to make a change.

The RZ-WAS employs separate front-end servers and back-end servers; for security reasons they do not share access. For data security, a TLD's userID and password enable access only to the front end.

New gTLD password distribution

IANA intends that new gTLDs will establish a trust relationship by being under contract to IANA/ICANN. For these new, non-ccTLD authorities, password allocation will not need to be face-to-face in the vast majority of cases. Furthermore, the allocations will occur as the new

TLDs are added incrementally, rather than in batches. It will consequently be a much simpler, highly automated process.

IANA will thus continue to recognize two classes of TLD manager—the ccTLDs and the gTLDs—and will treat them differently for some purposes.

Real-time syntactic checks

While the TLD authority (or, on their behalf, IANA staff) is entering data, the web tool will perform in real time a variety of syntactic checks on the information being entered, and reject change requests that are incomplete (e.g., specific fields are missing) or otherwise "unacceptable." Unacceptable requests are those that do not conform to the formal requirements of that data element; for example, an IP number that is incorrect due to being too large, or containing a nonsense character.

Real-world checks

Additional checks are performed in real-time as data are entered. Some of these check the IANA database, for example, to see that a specific zone exists. Others are real-world technical checks for name servers, URLs, WHOIS servers, and DNSSEC data.[60] For example, the RZ-WAS will check that the specified zone is actually served by the specified name server.

Authentication confirmation

A request that completes data entry via the tool (directly by a TLD or by IANA acting as an intermediate) by passing all checks, is called a **submitted** or **pending request**. At that point, the web tool automatically generates an email to the requesting party that contains a code unique to the requesting party and this submission. The administrative contact and the technical contact respond by entering that unique code in the web tool. If they do not have access to the website for whatever reason, they can respond via email with the code embedded in the email. This will cause an "exception" in processing and default to manual review. This second confirmation ensures that the request comes from the TLD, that they authorize it, and that it conforms to the authorization request held by IANA. The request now has '**valid'** status.

Human actions

At this stage, there are a variety of human actions performed by IANA. Human actions are especially significant in our model: they can be more prone to error, and have an impact on manpower (and hence cost) requirements. Human actions may be operational—that is, they may be a process step consisting of one or more actions—or they may be a "check" to ensure that a prior step was done correctly, or that the sum of a number of steps is not having an undesirable

[60] DNSSEC is not currently in the root zone.

consequence (see "stealth delegation detection" below). Each of the following states can be initiated by human review, followed by a human action.

OPT-OUTs

There may be cases in which a particular TLD is allowed to "opt out"—that is, to receive a waiver of IANA technical requirements—based on the TLD's specific circumstances. For example, IANA requires diversity in AS paths to the TLD name servers. However, some TLDs cannot yet satisfy that requirement; IANA will not block their submitting a request. Any request that does not meet a specific technical requirement will subsequently be reviewed to determine if an exception is warranted.

Direct contact with TLDs

If necessary, IANA will attempt to resolve problems that arise at any stage by communicating directly with the source of the change request.

Timeouts / administrative closure

If a change process fails due to timeout, the request is deemed not valid. Administrative closure occurs when a request cannot be completed in the timeframe established by the contract for IANA functions, at which point the request expires. An expired request cannot be re-started; a new request must be initiated.

Deletion of requests

The administrative or technical contact can abort a change request at any time up to the moment when the EPP transaction is sent to VeriSign. The EPP transaction is only formulated after the authorizations from both administrative and technical contacts are received.

Other affected parties

In those cases in which a request has an effect on a shared resource (that is, where another authority is involved, as for example when two TLDs share the same name server), then every one of these authorities will be contacted for approval of the change. This human action can add significant delay to the process. Unanimity is required: if any authority refuses the change, the request is rejected. If approval is not obtained with the time-out period, the request goes into "administrative closure."

IANA check procedure

After all of the checks have successfully completed, and all of the information has been confirmed and is in place, IANA performs a manual review known as the "IANA check procedure." In some cases, this may be the first time a given request has a human action. The

check procedure ensures that every request has had a "human in the loop" somewhere. This procedure determines the nature of the request: is it a substantial change – that is, a change of authority of the TLD (a "delegation change")—or not? If it is a change of authority, a delegation evaluation must be performed. IANA also checks to see if special handling requirements mandate special instructions or actions. For example, some combinations of changes at any one time might put stability at risk; these could be broken down into individual changes performed in sequence.

Stealth delegation detection

This step is especially important since it is the last point at which a "stealth delegation request" may be detected. This occurs (both intentionally and otherwise) when the net effect of a number of individually innocuous changes amounts to a change in authority.

If there is no change to delegation (in which case the change is called "simple"), the request goes to the "Supplementary Technical Check."

Supplementary technical check

Because a significant amount of time may have passed between the last check and approval, the details of the request may no longer be valid or have changed. This check confirms that no such change has occurred since the original request was made. All technical checks are performed again, followed by review by a staff member. If there has been no change, the request is approved. If there is a change and is acceptable, it is made in concert with the steps above; once all the changes are made, an additional supplementary technical check is done; at some point a final supplementary check passes and the request is approved. If not, the request is denied. Waivers may also be granted at this time.

VeriSign notification

At the positive conclusion of all IANA checks, the change is now ready for implementation within the root zone, and the change moves to VeriSign. IANA sends a provisioning request to VeriSign via an EPP[61] (Extensible Provisioning Protocol [31]) session, initiating a transaction.

[61] Extensible Provisioning Protocol was designed for allocating objects within registries over the Internet. Based on XML, it is robust and flexible enough to be used as a general-purpose messaging facility. An EPP message *may* automatically create a pending event waiting on completion of a specific transaction; it is used in this way here. Because root zone changes are grouped and have to be treated as an atomic unit with respect to updating the database, VeriSign has designed and implemented a proprietary extension to EPP called a "change object" that groups multiple operations into a single unit. This extension is used by IANA and VeriSign.

VeriSign processing

VeriSign receives the EPP request and immediately sets a holding timer for 10 days (as specified by [11]). In most cases, this is VeriSign's first contact with and knowledge of this specific root zone change request and results in a pending event in the VeriSign workflow database. VeriSign acknowledges receipt in the same EPP session by sending an EPP response back to IANA containing a "submission identifier" unique to this request (the submission ID is generated by VeriSign's proprietary extensions to EPP). VeriSign also begins to perform its own checks (see below). However, VeriSign will not implement a change until it receives authorization from NTIA (see below).

NTIA review

IANA, upon receipt of VeriSign's EPP response, composes and sends a PGP-signed email change request to NTIA tagged with the unique submission ID. Upon receipt of IANA's email, NTIA begins its own internal authorization checks.

VeriSign technical checks

During this time, VeriSign performs a number of technical checks on the change request, which may be in parallel with authorization checks being performed at NTIA. These checks overlap, but do not necessarily precisely duplicate the technical checks that IANA made before it initiated the EPP transaction.

NTIA request for clarification

While performing its checks, NTIA may make a request for clarification from IANA via an email tagged with the unique submission ID. Upon receipt of a clarification request, IANA aborts the EPP transaction with VeriSign, which effectively aborts the change request. VeriSign, upon receipt of that EPP notification, terminates work on the change request and removes change-related data from its database and the corresponding event from its workflow system. Clarification between IANA and NTIA is done by an exchange of PGP-signed emails. If and when clarification is made and is acceptable to NTIA, IANA initiates a new EPP transaction.

Timeout exception

In rare and exceptional cases, VeriSign may not receive NTIA authorization before the (10-day) holding timer expires.[62] Were this to occur, VeriSign would abort the EPP transaction, signaling IANA that implementation of the change request did not occur. VeriSign would also purge the

[62] This event is so rare and exceptional that it has, so far, never occurred.

change information from storage, remove the change event from its workflow system, and take no further action with respect to the aborted change request.

NTIA authorization

Once NTIA has successfully completed its own checks, and does so before the holding timer expires, NTIA authorizes the change by sending a PGP signed email tagged with the submission ID to VeriSign, with a copy to IANA.

Root zone database updated

Upon receipt of NTIA authorization, VeriSign installs the change in the root zone database. As with all of the information transmitted among the parties in machine-readable format, the database update is expected to occur "within seconds" of the arrival of a "yes" from NTIA. VeriSign then performs a "human eyes" look immediately after the NTIA "yes." This must be completed before the next step is permitted. (VeriSign requires that at some point "human eyes" look at every change request.)

Currently, VeriSign builds a new zone file and moves it to the distribution masters roughly every twelve (12) hours. Since multiple changes may accumulate during any single twelve hour interval, a single new root zone file many include any number of individual changes.[63]

After VeriSign generates a new zone file and publishes it to the distribution masters (servers located in VeriSign facilities), it queries the "a" root server repeatedly until it has confirmed that the update has reached the "a" root.[64] Only then does VeriSign signal completion of the change request to IANA as part of the EPP transaction. Upon receiving the close of the session, IANA reports the change to the original requestor.

Social data changes

Social data change requests and delegation data change requests are processed differently. A valid social data change request does not trigger an EPP transaction to VeriSign. Instead, IANA sends a similarly formulated request in an email message to NTIA, and waits for NTIA authorization.

4.3 The publication system

This section describes the system that publishes, or distributes, a current root zone file from the distribution masters maintained by VeriSign to each of the root server operators and, as required,

[63] The automated system is able to publish a new zone file "on demand," should the need arise.
[64] VeriSign and IANA both consider a change to be complete only after it is observed as having arrived at the "a" root.

across their own infrastructure onto each of their root zone servers. Those servers then participate in the query/response process and the subsequent flow of root zone data out to the Internet. The publication system consists of the actors who operate the system; the systems of servers, links, and software they use; and the processes they employ. Considered at a high level, publication is both exceedingly simple and straightforward, and highly defined by existing protocols.

4.3.1 Publication system data and actors

The only set of data managed by the publication system that is specific to the root system is the current root zone file. Other operational data sets include monitoring and audit data, but these data sets are not directly germane to the publication process.

The actors in the publication system are the root server operators (operators); they are described in Section 2.4.5. They in turn frequently have relationships with external facilities managers, who house many of the servers, and external telecommunications operators, who provide transmission capacity. However, in every case in which a function is outsourced to another party, the operators direct all significant activities. There are no operationally significant non-operator participants involved in managing the root zone data.

4.3.2 Name server functions

The root servers support two query-related functions:

- they serve as the source of authoritative data about the root of the DNS name space, on which systems located anywhere on the Internet fabric may rely; and
- they function as the backstop or "servers of last resort" for many DNS queries not otherwise answerable from elsewhere in the DNS name space.

The second category has a surprising consequence: because of the structure of the DNS protocol, many failed queries (which could, for example, be badly-formed and hence nonsensical) or queries outside the Internet name space are passed progressively up the DNS hierarchy until they reach the root servers. In this way, only at the very top of the DNS hierarchy—the root—is it possible to determine that a particular request is invalid. As a result, the proportion of (often unintentionally) bad or malformed queries to the root can be surprisingly large. Unfortunately, none of the answers to these queries is cache-able because the answer is "NXDOMAIN"—"not in this domain." It is possible that with an increase in the number of new delegations that some types of these queries, particularly those outside the Internet name space, will find themselves included in the Internet name space and therefore cache-able. The number of NXDOMAIN

replies at the root is reported by the root server operators to be approximately 95% of the total number of responses.

4.3.3 The publication process

The publication (distribution) process includes the steps included in transferring the zone from distribution masters (DMs) to the authoritative servers operated by each root server operator.

In some cases, an operator may have staging capabilities between the DMs and the authoritative servers they run. These staging capabilities are used for several purposes—as an archive of the root zone data; to achieve better control of the transit or distribution network (*e.g.*, ensuring that the paths between the staging platform and the target authoritative servers is completely under the management and supervision of the operator); and as a means to ensure the integrity of the zone data as it passes from one actor to another. The holding time in a staging area can range from a few seconds to as long as 20 minutes.

4.3.4 The mechanisms of distribution

Updating a zone involves communication over both TCP and UDP. The initial messages (the zone NOTIFY and zone transfer request) are small, and generally fit comfortably in UDP packets, while the zone transfer itself might be large, and require TCP.

Currently, normal distribution of the zone is done in all cases using standard DNS zone transfers over the Internet. All such transfers are controlled by use of access control of both the IP address of the source and destination IP addresses as well as DNS cryptographic checks using the TSIG (transaction signature) features of the DNS [35]. In exceptional conditions, distribution of the zone may be done using the FTP protocol or by other means. The operator specifies a set of target nodes to the distribution master manager, which includes those IP addresses in the access control lists. Operators meet three times a year, and agree on a new transaction signature key. When a new copy of the root zone data is available from the DM, those nodes automatically send a DNS NOTIFY signal to all the nodes in the access control list. Each target node then generates a zone transfer request and the zone data are transferred. This process is "backstopped" by the normal DNS retry/expire timers in the Start of Authority (SOA) record.

4.3.5 Root server architecture

Traditionally, each root server operator ran only one server. Over the years, and especially with the development of anycast, many moved to multiple machines. Often explicitly following the principle of intentional diversity described elsewhere in this report, they have evolved to support a variety of different distribution architectures, as illustrated in Figure 6.

	local cluster	anycast	single server
staged	B	I	
unstaged	A, E, H	C,F,G,J,K,L,M	D

Figure 6: Distribution Architecture

The dimensions are (a) whether or not the operator stages[65] updates to the root zone file, and (b) whether the operator deploys a single root server, a local cluster of servers, or an anycast cloud of servers distributed geographically and topologically.

4.3.6 Publication system variables

The following variables are relevant to the distribution process:

- propagation delay from the distribution master to the operator's staging server (if staged) or root server (if unstaged);
- propagation delay from the staging server to the authoritative servers;
- RTT and packet loss experience during the notify sequence;
- size of the data to be transferred;
- bandwidth in various parts of the system;
- success rate of XFR attempts in the various stages; and
- the impact of response size on bandwidth consumption.

4.4 Timing data

During the study period, interviews and measurements provided the following data points for baseline use in both a quantitative model and as input to an analytic discussion.

Transfer of the root zone file from the distribution masters to a root server operator:

- 5-7 minutes

[65] "Staging" refers to the process used to obtain and distribute updates to the root zone file from VeriSign's distribution masters. A "staged" root server first loads an update to a local server, and distributes it to any other servers (either local or remote) from there. An "unstaged" root server pulls updates directly to each of its root server systems.

In some cases, the operator uses a staging server. If staging is used, **transfer from the staging server to the authoritative servers**:

- 5-30 minutes

After the zone has arrived at the authoritative server, the zone has to be loaded in the name server itself. **Time to load the zone**:

- < 30 seconds

Sometimes, the zone fails to load, and that has to be detected, and acted on—either manually (requiring human interaction) or automatic, somewhat depending on what the problem is. The **response time when there is a zone load error**:

- 5 minutes—36 hours

The **size of the root zone file**:

- approximately 80K bytes

As normal DNS zone transfers are used, it is interesting to look at the **SOA times**, in the case a Notify message is dropped in transfer:

- Refresh—1800 seconds (interval between verification of serial number)
- Retry—900 seconds (retries if refresh fails)
- Ignore—604800 seconds (after what time should no responses be given)

5 Quantitative model

Qualitative models describe the structure and properties of things or events and the relationships among them. Qualitative modeling is concerned with building an abstract representation of a system, which can then be manipulated logically to analyse the functional behavior of the system.

A qualitative model is a prerequisite for a *quantitative* model, which is concerned with expressing the parameters and interactions that define the dynamic behavior of a system in numeric and algorithmic terms, which can then be manipulated mathematically. Quantitative models are constructed from observations and measurements of a system that can be expressed as absolute or relative magnitudes. Ideally, the simulations enabled by a quantitative model complement and confirm the reasoning about system behavior that is based on the corresponding qualitative model.

The root scaling study included the development of a quantitative model, which was used to simulate a variety of root scaling scenarios. In many cases the results obtained from these simulations validated the analytical model; in others, the results disagreed with the analytical model. Some of the differences could only have been detected by simulation, and resulted in corrections to the analytical model. Some show how limited the simulation is at this point in its development, and point toward areas of possible future refinement (see Section 8).

5.1 Role and scope

The quantitative model of the root server system plays two roles with respect to the outcome and impact of the root scaling study:

- it confirms and validates many (but not all—see below) of the findings derived from reasoning based on the qualitative model described in Section 4, and
- it represents a potentially valuable first step toward a continuously updated and expanded model that could be part of the "early warning system" anticipated in Section 8.1.

The scope of the quantitative model is identical to that of the general scope of the root scaling study (see Section 3.3), with the exception that it does not model the dynamic behavior of the root servers in response to changes in query load. It is important to recognize that it is also a model of the system *as it exists today*.[66] In order to simulate the behavior of future states of the

[66] With one exception: at the time of publication of these study results, provisioning is largely accomplished by manual processes, some of which will be replaced by automated processes when the root

root system, after it has absorbed changes (such as the addition of new TLDs) and its actors and processes have made corresponding adaptive changes, it would be necessary to update the model to account for the changes.

Based on data and knowledge of the root system compiled from the study's information gathering and analysis, the model expresses the quantitative relationships among the following root system parameters:

- the number of TLDs in the root zone file, one of the four drivers of root zone file expansion;
- the rate at which TLD change requests are submitted (requesting changes to either delegation information or social information);
- the timing and error characteristics of the processes through which each of the parties involved in the root zone management system carries out its individual designated function(s);
- the timing and error characteristics of the communication channels that carry information among the parties (*e.g.*, the email communication channel between IANA and NTIA, or the distribution channel between a root server operator's main servers and its anycast instances);[67] and
- the likelihood of error[68] (a) between the submission of a change request and the corresponding update to the root zone distribution master (provisioning subsystem), and (b) between the distribution master and all of the anycast servers (publication subsystem).

5.2 Limitations of quantitative modeling

George Box [44] introduced the now well-known aphorism[69] "All models are wrong. Some models are useful." All models are wrong because they are, by definition, formal abstractions of real systems; the only perfectly complete and accurate model of a system would be the system itself. The value of a model lies not in how closely it resembles the corresponding real system, but in how effectively it contributes to understanding the system and reasoning about its likely behavior in response to changes in its environment. The important question, therefore, is not "is the model valid?" but rather "is the model useful?"

---

zone workflow automation system (RZ-WAS, described in Secction 4.2.3) is fully implemented. The root scaling models assume that the RZ-WAS has already been fully deployed.

[67] The actual network protocols are not simulated.

[68] In this context the term "error" refers to the propagation to one or more name servers of a TLD change that should not be made.

[69] Often incorrectly attributed to W. Edwards Deming.

The usefulness of the root scaling study quantitative model is limited by two factors:

- the intrinsic limitations of all quantitative models of complex dynamic systems, in which the number of variables, the precision with which they can be measured, and the way in which they depend on each other make modeling of all but the simplest system behaviors mathematically infeasible; and
- the limitations of the model developed for this study, which was constructed from incomplete information over a relatively short period of time.

The long-term usefulness of the model ultimately depends on if and how it is used and improved for purposes beyond the current root scaling study. Acceptance of the model by the community that expects to use it as the basis for exploration of issues, formulation of strategies, and agreement on decisions is therefore as important as its content.

5.3 Applying quantitative modeling to the root system

5.3.1 Key attributes of provisioning, distribution, and actors

Dynamic and adaptive RSOs

The root server operators constantly adjust their systems, sometimes in response to changes, sometimes due to continuing refinement of their understanding of the distribution process and its requirements; they are fundamentally dynamic in their anticipation of and response to change. During our surveys, we repeatedly asked the RSOs if and how they could adapt to a specific change. The often-heard answer was: "we can adapt to that, given the time; the details of how we would adapt would depend upon the specific circumstances."

Adaptability, cost, and time frame

Each of the root server operators has its own processes for adapting to change, and ranges of how they respond. We observe that these may occur at four levels. First, modest changes and increases are those that are within the scope of their existing budget, operations book and change management process. These are where the responses are reasonably straightforward on the part of the RSOs, and for which our model is quite accurate. These can take up to several months to implement. Second, where there are changes that require a planning and budgeting cycle; these usually link to an annual planning cycle and thus take on the order of 18 months. Third, where a fundamental organizational change to a specific RSO is required. This should be expected to take on the order of several years. Finally, there are changes of a type and order of magnitude that could require a significant restructuring across the entire root server system, which would be done with coordination among the parties. This would also be an undertaking of at least several years.

Related to this is an important attribute of RSO funding: none of the root server operators directly charges for general root server services provided to the community at large. Some of the RSOs establish cooperative relations with entities that wish to have a root server within their networks or within their facilities; these sometimes include cost-sharing or even revenues beyond simple cost-recovery. Sometimes these servers are for use within a closed (*e.g.* corporate) network; others are for open use across the Internet. However, in any event, the DNS does not support charging for queries made to root server instances.

Each root server operates within an organization that is willing to fund their root server for reasons, collateral and otherwise, and where the costs of doing so are a reasonably small fraction of overall organizational costs. Some of their motivations for running an RSO relate to marketing and stature, others consider it a strategic value for their business, still others see it within their role as being members of the Internet technical community.

Whatever their motivations, they have real limits to the extent to which they can fund future growth, especially when it grows so large that "the tail begins to wag the dog", which it could readily do as the RSS grows by orders of magnitude. Providing external funding may not be as simple as is may sound. The RSOs are independent actors, and they guard that status fervently.

Architectural and design choices

Once the ability of the RSOs to immediately support some level of change is exceeded, there are many dimensions along which root server operators may satisfy the demands of increasing scale. That is, the **number of possible design and architectural choices is very large**. Furthermore, the combinations of changes that each individual RSO might take, and the number of variants they might choose at the same time, is combinatorially huge. Predicting in sufficient detail all of the many choices the RSOs might make and the number of choices they might take in combination is a massive undertaking, far beyond the scope of this study and model. It is not reasonable to pretend that a particular design solution will be used to solve the problems of growth in the future; similarly, any one design solution will have limited generality—it will not reliably predict the behavior of other design solutions. Consequently, the model will rely on rough assumptions about how the system will change as it grows, and will employ abstractions of how it will adapt. These use attributes and measures to drive the model rather than detailed internal structures—for example, by using linear (or other) multipliers for each "resource" in the model.

Diversity by intent

Many of the root servers operators practice **intentional diversity**. The root server system is a critical control point in the Internet, and an attractive target of attack. The greater the operational

consistency among the root servers operators, the easier it is for an attacker to do damage. The root server operators confer regularly; many of them subsequently modify their systems to avoid cross-operator commonality. This diversity is expressed in nearly every aspect of their systems: operating system(s), hardware, load balancers, link capacity, media and providers, number and variety of physical locations, and so on. This makes modeling even more difficult and time-consuming: the number of common practices is relatively low. It is even more so because the individual RSOs also practice internal diversity along the same dimensions mentioned above, thus avoiding single points of failure due to technical monoculture.

Root server adaptation

As we have illustrated above, the manner in which root servers adapt to change is not simple to predict. However, there are categories of how they react. The first example, made above, illustrated when different degrees of change drive different levels of reaction: simple change in magnitude, significant changes require architectural changes, changing requiring new levels of funding, and changes that require a significant organizational change. Another category has to do with the manner of specific technical change. Some changes are relatively continuous or incremental. For example, adding another server to an already large server farm. However, some changes consist unavoidably of "bigger steps." There are two excellent historical examples. The first, is the transition from unicast to anycast for root zone file distribution. The second, now underway, is the IANA transition from its largely manual style of work to the RZ-WAS automated web tool described in Section 4.2.3. Both were critical to the root management system's ability to adapt to growth and change. Current "stepped" increases at RSOs could include large steps in capacity (especially satellite links to remote instances).

Steepness of growth curve

The "steepness of the curve" is a compound function of the size of the root zone and its rate of growth. This may be the result of, for example, the number of TLDs, or the size of each entry in bytes, or both. Steepness also determines how rapidly one may approach one of the boundaries, or "discontinuities." Another change from the model is for example how many changes per year each TLD makes; in the simulation, based on interviews, we have estimated that each TLD makes one change per year.

Root server adaptation matrix

Our analysis suggests that additional planning and coordination between policy-makers and RSOs will be required to determine the viability of any proposed increases in root zone size. This suggests the development of a matrix of the projected impact and consequences of changes as follows: per change, there should be a response from each letter server. The matrix would thus be

expressed as 13 x (number of distinct changes). The matrix could become a useful tool for organizing understanding of the impact of changes on a per-RSO basis, across all RSOs, and also to anticipate where changes across the entire system might be warranted.

5.3.2 Applying modeling to root system management

Determining parameters

Constructing this model work follows our analytical efforts (see section 4) to reveal the components, structures, and processes in provisioning and distribution. In discovering those, we also sought to distinguish the parameters that, when manipulated, will allow policy makers and others to investigate a range of possibilities of how the system could respond to change and growth. These parameters will include, for example, the capacity of internal distribution links used by the root server operators.

Modeling the root system—baseline scenario

When the parameters are set to currently seen real-world values, when complete, debugged, and validated, the model should show the current state of root server operations. This is the **baseline scenario**. As the four drivers of growth are increased (IPv6 glue records, IDNs, DNSSEC and new gTLDs), singly and in combination, and we adjust various parameters of the system, the goal is for the model to accurately show system behavior. Armed with these illustrations, policy makers are in a better position to determine when and how the system will begin to show stress, and thus have a guide for when and how much to increase these drivers of root zone file growth.

Complexity of combinations

The four drivers of root zone growth are not independent in their effects. Rather, they combine is subtle and complex ways. One example is deploying DNSSEC while also increasing zone size to the point that the RSOs move to incremental zone updates (from AXFR to IXFR). Since DNSSEC can be configured to incrementally update signatures (for example, $1/30^{th}$ of the zone file could be redistributed every day of the month), the two kinds of incremental updates must be carefully managed to preclude incompatibilities. Further, when doing AXFR, the time a transfer takes will depend on the size of the zone. When doing IXFR, the time depends on the number of changes (and size of the individual records) since the previous IXFR. If the rate of change to the zone increases, but the rate of IXFR does not increase, the size of each one of these updates will increase. For slow links, or links with bad characteristics, the choice between using IXFR or AXFR and how often they should be made are things that an RSO must calculate carefully. This will add complexity, and gives at the same time one example of where the model used for simulation can be improved.

5.4 The TNO modeling process

TNO[70] defines the art of modeling as the process of including all relevant parameters and relations between them that capture the dynamic behavior of the modeled system, while omitting all other aspects that are not relevant to the goal of the model based analysis. Typically the development of such a quantitative model is an iterative process with several model adjustment and fine-tuning steps. In order to support a flexible model development approach TNO uses its model development approach "PerfICT."

Two specific features of this approach are its hierarchical concept and the decoupling of workflows and resources. The hierarchical modeling concept enables the creation of an initial model in which, for example, IANA is modeled as a black-box; that box can later be worked out in more detail during the modeling process without having to repeat the process of modeling the interaction between IANA and other systems in the provisioning process. Likewise, the decoupling of resources and workflows enables the swapping of a modeled human resource by an automated one, without having to modify other, neighboring modeled resources or workflows. Besides providing flexibility during model development the PerfICT approach also enables flexible creation of alternative models for future variants of the Root system. We hope that using the PerfICT approach will make it most useful as the basis for future root server system modeling efforts.

The model first of all presents a comprehensive overview of the dynamical behavior of the provisioning and publication processes of the Root system. Time did not permit us to visualize the dynamic behavior of the system using graphical animation; that option is available in the future. Further, by performing sensitivity analyses on the input parameters the model enables policy-makers and community members to answer scalability questions such as what amount of resources are required to keep a near-zero error rate for specific scenarios of change request rates, how effective would it be to further automate specific provisioning actions, and so on.

5.5 The modeling software

The root system quantitative model runs on the ExtendSIM™ suite of software, a product of Imagine That, Incorporated.[71] The model consists of a set of instructions that can be executed by the ExtendSim modeling software. ExtendSim software uses a visual style of programming; it has extensive libraries used to deal with elements commonly seen in simulations, such as various

[70] TNO, the Netherlands Organization for Applied Scientific Research, is an independent research organisation with headquarters in Delft (http://www.tno.nl). The root scaling study subcontracted the development of the quantitative root system model to TNO.
[71] http://www.extendsim.com

types of queuing systems. In particular, the model is implemented in "ExtendSim OR" (Operations Research), a tool for researching operational performance.

The model we produced for this study is a "pure baseline model," through which we have sought to describe current practices. The model concentrates on the global picture of root zone management and consists of two major and distinct parts: provisioning and publication. It will be straightforward to evaluate these two parts separately in the future if there is a desire to do a more detailed study or model of either or both of these area separately.

Our emphasis has been on developing the model. Unfortunately, the there has not been sufficient time during this study period to develop a proper user interface for the operator. Some of the major parameters can be inserted via input files, but many detailed parameters are scattered across the different parts of the program. This makes the implementation and evaluation of scenarios a non-trivial task.

The use of the PerfICT method can be seen in the modeling software. The TNO implementation consists of two main sections, corresponding to the provisioning and publication parts of the root maintenance operation. Each of those sections is subdivided into "blocks" and, as required, further subdivided into sub-blocks, and so on. On the provisioning side, three distinct blocks describe the three different actors, IANA, NTIA, and VeriSign. Each of those blocks in turn consist of smaller blocks, each describing various functions in the process; they themselves are build up from smaller blocks. The publication part has four distinct main blocks, each representing the different styles of RSO operations. Again, those are made of corresponding sub-blocks. The entire model consists of approximately 3000 blocks.

Resource requirements for running the model are modest and on a scale readily available even to a casual computer user. Our entire ExtendSim model occupies approximately 7.2 MB of disk space and may be run on a single-processor, off-the-shelf PC.

5.6 Attributes

The model simulates the time it takes to complete a change request. This is output as the "lead time" for a change request. This gives an impression about the global performance of the system and also change request throughput. The "root zone load time" denotes the loading time of the root zone in the name servers for the various types of systems. More specifically, it models the delay between the production of the zone file and the loading of that file into the various type of name servers. There is also an error rate computation based on the so-called "reward model."

The ExtendSim software allows a 2D-animation of events in the model and the blocks. Available now, this makes it easy to follow the flow of requests within and across the various parts of the

model. By varying the speed of animation, it is possible to quickly debug the data flow of the model.

5.7 Parameters

As mentioned above, a key element in the RSST process is to identify the parameters used by the model. In the model, some parameters apply to both provisioning and distribution, others are specific to one or the other. Other parameters, such as working hours in a day, or working days in the week, are manifest in only one, or a small number of blocks in the model. Some, such as the levels of human resources within the actors, are directly set by "knobs" inside the model; these are treated as reasonably static.

Among the identified parameters are number of TLDs (affecting both provisioning and publication), bandwidth variations in the publication infrastructure (affecting only publication), and processing time of various steps. Other parameters define the mix of requests for social data or name server changes, and time waiting on actions from actors outside the system, such as delays in getting confirmation from zone managers for change requests.

Most of these types of parameters are read into the model from input files. These input files are in the form of CSV (Comma Separated Values). Each row in these files defines the parameters for a single run. The virtual duration of the events being simulated (in hours) is a manual setting in the ExtendSim itself, just as how many and which of defined the runs actually will take place.

A description of the actual parameters can be found in the description of the model by TNO.[72]

The parameters which deal with basic network simulation, notably for the zone transfer, are bound to a documented model of TCP/IP throughput. Other parameters come in an abstract form because there are different reasons why they might change.

5.8 Status

From the start of this effort it was well-understood that building a model including all possible attributes in the time given would be highly ambitious. The optimistic expectation was that by the time the current study was completed, it would be possible to have the basic model implemented, debugged, and validated. In that case it would be possible to use it to exercise some of the scenarios and gain some experience in operating the model.

That goal turned out to have been unrealistic (see Section 8). The current status is that the basic model, especially of the provisioning process, is functioning and sufficiently debugged that it implements the workflow as outlined the description.

[72] TNO's report is published separately.

Work is progressing on the validation of the model. However, there is a problem that not all data are presently known or available in a form suitable to put in the model. We have just begun to determine where the model requires more precise data about how much time certain events take or within what range. For example, the time needed for validation checks was available only at the last moment and is consequently implemented in the model with an exponential distribution; such distributions may need to be refined.

The model is sufficiently developed to run various scenarios, and the results look promising. Although currently limited, the model usefully describes basic root server maintenance process. Current results do not appear to contradict the corresponding analytical model. However, given the early state of the model, these results should best be taken as examples of what type of things the model might simulate. The reader should be even more careful than usual when interpreting current results of the model and basing any conclusions on these results. Doing so would be premature.

5.9 Limitations of the model

The workflow in the provisioning part is currently static rather than variable. There are also some limitations to what may be modeled. As the description states, there is a simplification of how individual tasks are done by, for instance, NTIA: if an action is started, it is continued until finished without taking into account limitations in working hours (the normal business workday), so a 4 hour process begun at 16:30 would "complete" at 20:30, but the same process arriving at 17:30 would not begin until 08:00 the following day. Similarly, the assumption that the reactions for confirmation to the initial request towards follow the distribution as used in the model will likely need refinement. Whether these things are actually needed can only determined after further experience with the model and the results are properly analyzed.

Likewise, the publication part is a simplified model of the actual zone file transfer process, as implemented using AXFR (for zone transfer). Other approaches might be preferred in certain circumstances but these are better first studied separately before incorporating them into a global model like this.

Using the model in its current state of development will be a challenge for the uninitiated user. The user interface is quite primitive and often requires that one look inside its various blocks to find out how parameters are used, and thus defined. For example, the model currently represents zone file size with '.1', an undefined number. By looking at the block where this number is input, it is possible to determine what this represents. It would have been better to have this defined differently, which we anticipate will be done in a future version of the model.

If one "overloads" the model it will just "fast forward" without indicating that things are out of bounds. This can lead to results which may be invalid, but not obviously so. On the other hand, by the time that starts to happen (with current settings, it may first be seen around 5,000 TLDs, and is unambiguously observed by 8,000 TLDs) the average time for a change approaches approximately 400 hours. This high lead time is probably already outside the boundaries of what is desirable.

5.10 Scenarios

Any model needs to be debugged—a process of checking the correctness of the software implementation. Beyond debugging, and more fundamentally, one must exercise the simulation to confirm to some level of trust that the model's assumptions and abstractions are not too far from reality. That is, the model also needs to be validated. TNO has developed and run two validation scenarios, in which they have changed only two different variables: the number of TLDs and the size of the zone file. They have left many other parameters in their default settings.

It must be stressed that the motivation for these scenarios is the validation of the model. One should be wary about using these results as the basis for anything more than beginning to validate the model. As we just mentioned, the assumptions made for the default parameters have not yet been tested. Nevertheless, some initial, rough observations may be drawn from these simulations.

The first scenario describes the current situation of 280 TLDs, with roughly one change per year per TLD. This corresponds to the baseline scenario mentioned in Section 5.3. In this case, it takes about 2 days for a change to become effective (that is, to be present across the root server instances). This is within the range of what is expected; detailed analysis of this case is given in the TNO model description. Growth to 1,120 TLDs, without a change in any other parameters, does not yet cause a significant change in the outcome. By 4,480 TLDs, change output (referred to in the TNO report as "lead time") starts to slow down and, as mentioned before, the whole simulation begins to become overloaded. There is one last additional run of 8,960 TLDs; at this TLD value the model becomes fully overloaded. With each step of the increase from 280 to 8,960 TLDs, the zone file grows and the loading time of the name servers that are connected by "good" and "bad" links increases.[73] To better illustrate the influence of the size of the zone file in loading the root servers, the same runs were made with a much bigger zone file, disproportionately large for the parameter setting for the number of TLDs in these runs. The

[73] There is a typo in the TNO model description. It twice refers to good connectivity in their description of this part, while the corresponding red line in the graph shown is actually bad connectivity.

outputs show that it is likely that some tuning of the parameters describing the "bad" links will be required in future runs.

It is encouraging that these outcomes are consistent with our analyses to date. However, they should not be taken as confirmation of any specific behaviors at this time. These scenarios were developed for validation of the model, not for reliable insights into root zone maintenance. Real, meaningful simulation runs will require more carefully thought out scenarios, additional fine tuning of the parameters, and a more fully debugged and validated model.

5.11 Future scenarios

The following classification of scenarios may be useful for future modeling of the root system. These suggestions are based on requirements in the Terms of Reference, and on our own experience.

We divide scenarios into four groups. *Individual growth drivers* look at the effect of increasing, in isolation, each of the four root zone growth drivers, *e.g.* TLDs. Although practically speaking little or no isolated growth will occur, we stand to gain basic knowledge of the character of growth effects. *Combined drivers* permits evaluation of various combinations of interest of the four drivers or growth. For example, this could include DNSSEC and IDNs together, or DNSSEC, IPv6, and IDNs. Those in turn could be organized into several "doublings" of the zone file, as a means of arranging the classes of effects. This suggests another approach, that of constructing *effect-oriented scenarios*. These would look at the scenario concept from "the far end," that, is from their consequences. For example, what combinations of drivers and values would result in a specific size of zone file? Finally, as described in the Terms of Reference, one could look specifically at *decimal order of magnitude growth scenarios* with respect to the number of TLDs. This of course requires that decisions be made concerning which drivers, and their values, are input. The reader is cautioned to be aware that some of these scenarios may require changes to the actual structure of the model. For example, some issues regarding DNSSEC processes currently under discussion could require process changes in provisioning, and hence in the model.

6 Findings and recommendations

6.1 Effects of combinations

Each of the four changes described in Section 2.5 has an effect on the growth of the root zone, and they can be combined in different ways to produce compound effects. If we add more TLDs to the zone, let every TLD have a couple of IDN mirror names, let every name server for each of the domain names (both ASCII and IDN) have not only an IPv4 address but also an IPv6 address, and then sign the root zone using DNSSEC, the compound effects tend to be felt multiplicatively rather than additively. If adding a TLD to a normal zone means a growth factor of 1.0, adding the same name to a zone that is signed with DNSSEC could mean a 4 times bigger change to the zone than if it wasn't signed. If a TLD is added to an unsigned zone, but with IPv6 records for its name servers, the change may be 1.25 times what it was without IPv6. If you add the TLD, with IPv6, to a zone that is signed with DNSSEC, the growth will be 1.25 x 4 = 5 times the base example.[74]

Following this line of reasoning, it is desirable to add changes that have a sudden and large impact on the root zone as early as possible, whereas more gradual changes can be added at later stages, as the absolute numbers can be kept low by the effects of the rate limiting. As DNSSEC represents the most pronounced "step," it would seem prudent to add DNSSEC to the root zone before any steps to increase the size by adding substantial amounts of new names are taken.

6.1.1 Adding new TLDs to the root zone

This increases both the number of entries in the root zone and the size of the root zone. It does not change the average size of each entry, however, so an incremental change of say 3 entries would have the same size regardless of the number of entries in the zone. An increase in the number of TLDs is not expected to change the number of requests per year per TLD.[75]

6.1.2 Adding DNSSEC support to the root zone

This increases the size of the zone, but not the number of TLDs. It increases the number of resource records in the root zone. It also increases the size of an incremental change to the zone after changes in the data related to one TLD. Finally, it increases the number of changes to the root zone per TLD and year as the number of reasons a TLD has to change data in the root increases (not only changes in glue and NS records trigger communication, but also changes of key signing keys).

[74] These numbers are intended to be understood as examples, not as actual measured data points.
[75] The number of requests per year per TLD might change if the character of the root zone changes—if, for example, it becomes more like .COM—but the current study did not try to anticipate this.

6.1.3 Adding IDN TLDs to the root zone

This is very similar to addition of TLDs (see 6.1.1), with the addition that the size of each entry in the root zone changes. This because the average length of the domain names in the DNS end up being longer when being encoded with Punycode [46] than non-encoded domain names.

6.1.4 Adding IPv6 address data to the root zone

This adds glue records for the name servers for each TLD, which implies the amount of data in the root zone for each TLD increases. It also might increase the number of changes each year for each TLD.

6.1.5 Compound effects

Table 2 summarizes the compound effects of the changes described individually in Sections 6.1.1 through 6.1.4.

	New TLDs	DNSSEC	IDNs	IPv6 addresses
Increases number of TLD entries in the root zone	X		X	
Increases size of the root zone file	X	X	X	X
Increases amount of data per TLD		X	X	X
Increases number of variables per TLD		X		X
Increases number of changes per TLD per year		X		X

Table 2—Compound effects of root zone changes

Table 3 summarizes the impact of each of these effects on the root zone management system.

	Increased number of TLD entries in the root zone	Increased size of the root zone file	Increased amount of data per TLD	Increased number of variables per TLD	Increased number of changes per TLD per year
Impact on IANA/NTIA/VeriSign automatic processing	X			X	X
Impact on IANA/NTIA/VeriSign manual processing	X			X	X
Impact on AXFR in publication system		X			
Impact on IXFR in publication system	X		X		X
Impact on root server memory requirements		X			
Impact on root server CPU requirements	X				
Impact on root server bandwidth requirements	X (query load)		X (response size)	X (response size and query load)	

Table 3—Impact of compound effects

6.2 Qualitative and quantitative effects

New TLDs and IDNs have, *per se*, primarily quantitative effects on the root system—they are essentially just "more of the same." In the cases of adding IPv6 address records and DNSSEC, however, the change to the root zone is qualitative as well as quantitative. A name server that is expected to serve IPv6 records needs to have its behavior changed so that it understands to provide to the client not only IPv4 addresses for the listed name servers for a specific sub-domain, but also its IPv6 addresses (if they exist). This means more work for the processor to retrieve and pack the information when responding to a query. In the case of DNSSEC this is even more pronounced, since even more and bigger records are expected by the client.

An effect which is both quantitative and qualitative pertains to IDNs, IPv6, and (even more so) DNSSEC. In these cases more information has to be carried in the packets that are returned in response to a query. That means that the required amount of network bandwidth needed to

support the operations of the server increases. As the DNS messages get bigger, they will no longer fit in single 512-byte packets forwarded by Internet's UDP (User Datagram Protocol) transport mechanism. This will lead to clients being forced to resend their queries using the TCP (Transmission Control Protocol) transport mechanism—a mechanism which has much more overhead and requires the end nodes to maintain much more state information. It also has much more overhead in terms of "extra packets" sent just to keep things on track. The benefit is, of course, that it can carry much larger pieces of information.

Moving the root system from its default UDP behavior to TCP will not only have the undesirable effects mentioned above; it will also affect the current trend of deploying root servers using IP anycast. Anycast works well with single packet transactions (like UDP), but is much less well suited to handle TCP packet streams. If TCP transactions become more prevalent, the anycast architecture for root zone distribution may require changes.

6.3 The effect of adding DNSSEC to the root

These signature and key records vastly exceed the original DNS data in size. The enabling of DNSSEC is a transaction performed on an entire zone, meaning that it is impossible to enable it gradually. The result is that the act of enabling DNSSEC on a zone leads to the sudden addition of a substantial number of new and very big DNS records to the zone. The growth factor depends on a number of parameters, but "a factor of 4" is generally accepted as a reasonable estimate (see Section 6.9.4).

Due to the two facts that the growth factor is significant for DNSSEC, and that it is impossible to introduce it gradually, it is desirable to enable DNSSEC on a zone while it is small, to minimize the sudden size impact that the act will have. The sudden growth, measured in absolute numbers, and hence the impact on the system, will be much bigger if DNSSEC is enabled on a large zone.

DNSSEC changes the nature of the zone being signed in three significant ways:

- The "atomic unit" is no longer the individual resource record, it is a group of resource records. The size of the group is based on administrative choices by the zone administrator in the selection of signing algorithms, key lengths, validity intervals and rollover periods. These variables can be tuned but not with instantaneous effect, and regardless of their values, the group of records will always be significantly larger than the original unsigned data. Therefore the amount of data carried in the zone will be much bigger, leading to much bigger zone transfers.
- The responses to DNS queries for signed data will have to include the signatures and other security related information. As a consequence the answers will have to be much bigger than the corresponding answer not containing DNSSEC data, requiring

the servers to use more network bandwidth resources, and also to use TCP transportation, rather than the customary UDP transportation.

- The signatures used in DNSSEC have the property that they expire. This is a measure taken to mitigate the risk of "replay attacks", where an attacker tries to fool a client, by sending it old data that was recorded at an earlier stage. The signature records have an "expiration date" (actually an expiration *second*), and to avoid serving bad data from the authoritative servers (the root servers, in our case), the signature records need to be updated on a regular basis. The interval depends on the signature validity time. The longer a signature record is valid, the more seldom it needs to be updated, but the bigger the risk it being used in replay attacks. The decision on validity periods is a tradeoff.

An effect of the signature expiration property is that the zone needs to be updated on a regular basis *just because the signatures expire*. Even if no other data are changed in the zone, time will pass, and the signatures will have to be updated, thereby creating the need for zone transfers. Hence, as an effect of the introduction of DNSSEC, the zone doesn't only get bigger—thereby increasing the *size* of the zone transfers—but it also needs to be transferred *more often*, due to the cyclic creation of new (and "fresh") signature records.

The response size issue has an effect on the client side that should be mentioned. In certain client configurations, where firewalls are incorrectly configured [4], the following scenario can occur:

A resolver inside the misconfigured firewall receives a DNS request which it can not satisfy locally. The query is sent to the root servers, usually over UDP, and the root servers responds to this query with a referral, also over UDP. Today, this response will fit nicely in 512 bytes. It is also true that for the past six years, ISC has been anticipating DNSSEC and has shipped resolver code that, by default, requests DNSSEC data. Once the root is signed, the response will no longer fit into a 512 byte message. Estimates from NIST, using standard key lengths, indicate that DNSSEC will push the response to at least 2048 bytes or larger. This larger response will not be able to get past a misconfigured firewall which restricts DNS packets to 512 bytes, not recognizing the more modern extensions to the protocol that allow for bigger packets.

Upon not receiving the answer, the resolver on the inside will then retry the query, setting the buffer size to 512 bytes. The root will resend the response using smaller packets, but since it doesn't fit in a 512 byte packet, will fragment the response into a series of 512 byte replies, and the root server will set the "fragmented" and "truncated" flags in the packets, indicating to the resolver that the answer was fragmented and truncated, and encouraging the resolver to retry the query once more using TCP transport. The resolver will do so, and the root server will respond

using TCP, but the misconfigured firewall also rejects DNS over TCP, since this has not been considered a normal or widely used transport for DNS queries.

In this worst case, a node will be unable to get DNS resolution once the root zone is signed and there will be three times the DNS traffic, including one round in which TCP state must be maintained between the server and the resolver. There are of course ways around this problem, the most apparent ones being to configure the firewall correctly, or to configure the resolver to not ask for DNSSEC records. The process to achieve those work-arounds can be cumbersome and expensive, and in any case is outside the scope of the current study.

6.4 The effect of adding IPv6 to the root

In the future, it must be expected that every server to which a sub-domain is delegated (e.g., a TLD delegated from the root) need both an IPv4 address and an IPv6 address, thereby requiring two DNS records, as opposed to the common case of only IPv4 (i.e., one record) today. This will lead to an increase in zone size.

As opposed to the DNSSEC case above, IPv6 addresses can be added gradually, so there is no "sudden impact" expected in this case. However, should IPv6 suddenly become the subject of rapid deployment, it must be expected that the DNS in general, and the root zone in particular, will receive a similarly sudden increase in the number of IPv6 records.

IPv6 in the root zone requires the support of the AAAA Resource Record type and adds to the size of the zone by 96 bytes for every A Resource Record it replaces or 128 bytes if it is a straight addition. This does not have a significant effect on the size of the root zone or the rate of change in the root zone. The primary effect is again in the size of the priming response when it exceeds 512 bytes. The specifics are covered in Section 6.7.

6.5 The effect of adding IDNs to the root

From a growth perspective, the effect of adding IDN versions of existing TLDs is roughly equivalent to the effect of adding new TLDs—because that is what they are. In two respects the effect of IDN TLDs and other new TLDs is different:

- IDN labels are in many cases longer than (in some cases much longer than) non-IDN labels; and
- some observers expect that IDN TLDs will "mirror" non-IDN TLDs in different languages or scripts,

With respect to the second point, "mirroring" is something the DNS cannot implement directly.[76] Mirroring can be implemented only through policies implemented by the registry(ies) involved. The consequence of the registration of IDN versions of an existing TLD may well be that 3 or 4 new TLDs need to be registered, each being managed by a registry—potentially the same registry for all of them, and also potentially the same registry that is the registry for the original TLD.

As in the case of new TLDs, the (root) zone owner is expected to be able to exercise a high level of control over the growth rate, but for the reasons described above, the introduction of IDN versions of existing TLDs has the same character as the introduction of new TLDs. The introduction of IDN TLDs must therefore be included in the same growth calculations as the introduction of new TLDs. For these reasons, we have two growth factors regarding IDN TLDs: the length of the IDN version of the TLD compared with its existing TLD equivalent, and the addition of the IDN TLD itself.

This might in turn imply larger responses from the root servers, which is described as an effect in other places in this report.

6.6 The effect of adding more TLDs to the root

The primary effect of adding new TLDs to the root will be felt in the distribution of the zone data to the root servers themselves, but a secondary expected effect is increased traffic to the root name servers, due to the reshaping of the name space. New TLDs will have a much smaller effect on the size of any response to a query since, as delegation records, they are not signed in a DNSSEC enabled zone.

A larger zone has impact on the relationship between anycast end nodes and their distribution masters (regardless of where that server is located), and very remote areas of the Internet, that *can* be served today, could possibly fall off the list of possible sites, just because there isn't enough bandwidth to "push" the zone into the remote anycast servers if the zone grows to be too big or is updated too often.

In some corners of the world, a local root name server can help keep traffic local to the region, since the entire chain of DNS servers can be found locally—root server, TLD server, and second-level domain server. There are such corners, where the international bandwidth needed to keep the root server updated is horrendously expensive. The often very limited bandwidth is shared with the "payload" (web, mail, etc.) traffic from normal users. If the update traffic to the

[76] DNAME resource records are sometimes cited as adding a mirroring capability to the DNS, but the situation is more complicated than that, and beyond the scope of this discussion.

root server increases, it will "eat" a bigger share of the available bandwidth, thereby diminishing the amount of bandwidth available to the paying customers. Alternatively, the paying customers will push the heavier update traffic out from the shared link, thus making it impossible to keep the remote root name server properly updated. The consequence of such a process would be the forced shutdown of the local root server, and the result of that would be that the community needs to send all its root related DNS queries to a different root server, probably across the very expensive link. This must be considered a loss for all parties.

The DNS was designed to be a hierarchical system, partly to offload traffic from the root name servers by taking advantage of the caching mechanism. The more top-level domains there are, the less effective the caching mechanism will be, and the more queries the root servers will receive.

6.7 The priming sequence

Modification of the root zone to accomodate IDNs, IPv6, and DNSSEC has the potential to change the size of the response to a priming query. [77] The addition of new TLDs to the root zone will not affect the priming sequence directly, since this is the "bootstrap" phase for a node in the Internet to participate in the DNS.[78]

The basic DNS protocol specifies that clients, resolvers, and servers be capable of handling messages sizes of at least 512 bytes. They may support larger message sizes, but are not required to do so. The 512 byte "minimal maximum" was the original reason for having only nine root servers. In 1996 Bill Manning, Mark Kosters, and Paul Vixie presented a plan to Jon Postel to change the naming of the root name servers to take advantage of DNS label compression and allow the creation of four more authoritative name servers for the root zone. The outcome was the root name server convention as it stands today.

The use of 13 "letters" left a few unused bytes in the priming response, which were left there to allow for changes—which soon arrived. With the advent of IPv6 addressing for the root servers, it was no longer possible to include both an IPv4 "A" record and an IPv6 "AAAA" record for every root server in the priming response without truncation; AAAA records for only two servers could be included without exceeding the 512 byte limit. Fortunately the root system was able to

[77] When a validating resolver is first started, it uses a hints file or other initial "guess" to find a root server, and then it asks that root server for the current list of root servers (this is the "priming query"). The answer is the full list of thirteen root servers and their addresses (this is the "priming response").
[78] RFC 1034 [30] calls this the "SBELT" phase of DNS operation.

rely on the practical circumstance that any node asking for IPv6 address information also supported EDNS0 [40].[79]

DNSSEC also increases the size of the priming response, particularly since there are now more records in the RRset and those records are larger. In [32] the authors make the following observation: "The resolver MAY choose to use DNSSEC OK [RFC4033], in which case it MUST announce and handle a message size of at least 1220 octets."

Delegations in referral responses are not signed, so following this model there would be no need to require a signed root NS RRSet and, equally important, signed A and AAAA RRSet for the root name servers' names. On the other hand, a poisoned priming response could drastically influence the resolver's operations. If the priming response should be secured by DNSSEC, then it should also be self contained, *i.e.*, the whole validation chain should be present in the priming response.

If there were a desire to actually protect against a poisoned priming response, then the current root server naming constructs would have to be reconsidered. Presuming this was desired and implemented, then the next impact on the root system would be that the response itself would be too large to pass back through the Internet. This would trigger additional priming queries, generating additional query and response traffic to the root servers.

6.8 Critical regions

A particularly important finding concerns the special case of "critical regions" or "atomic units" as they relate to root zone expansion. The potential addition of "famous marks" as new gTLDs illustrates this point. Assuming that there are on the order of 40,000 generally recognized such marks, and that it is politically infeasible for any of these marks to be admitted as gTLDs without offering the same opportunity—simultaneously—to every other mark, adding the first "famous mark" gTLD would represent a commitment to add as many as 40,000 new gTLDs over a relatively short period of time. Such a circumstance, in which either none or an entire block of names must be added, a "critical region." Not only must the entire block be added, but once begun the process cannot be reversed, nor can it be stopped or even slowed for a significant period of time. It is even possible that ICANN will not be able to control the rate at which it must add new famous mark gTLDs. Critical regions will be of great concern due to the possibility that they might exceed the ability of the root system to adapt, and thus threaten the stability of the DNS. Policy makers must carefully anticipate critical regions, and ensure that, once entered, they

[79] EDNS0 is an extension to the DNS protocol that supports the negotiation between a client and server of "minimum maximum" message sizes larger than 512 bytes.

can be completed without encountering a discontinuity that cannot be resolved within the timeframe available for completion of the events inside the critical region.

6.9 Limits to growth

6.9.1 Publication system headroom

The publication system has some amount of "headroom" (deliberately provisioned excess capacity) that allows the root server operators to remain within the range of normal operations (see Section 4.1 and Figure 4) with respect to the distribution of root zone file updates to their root servers (including geographically distributed anycast instances). This headroom can be roughly calculated as follows:[80]

Given an 80K-byte root zone file, a modal propagation time of 38 minutes from the distribution masters to the authoritative root name servers, and a "jitter" or settling time of 12 minutes, updates to the root zone are generally available to the Internet at large in 50 minutes.

With an update to the distribution masters on about 12 hour intervals, there is slightly more than 10x buffering in the update cycle with the zone at the current size.

The propagation time within the distribution channel is currently dominated by staging effects. Staging has provided a valuable backstop in the past and should not be abandoned without good cause. Comparing the difference between nodes that stage and those that do not gives a nominal propagation time of about 8 minutes for an 80K root zone file without staging.

Presuming that we wish to maintain normal operations and retain the 10x buffer, we can extrapolate how the effect of each feature in isolation will affect the distribution channel.

6.9.2 New delegations

A canonical ("ordinary") new TLD delegation will consist of some number of name server (NS) and A (address) records; for example:

```
example.   in ns pdc.example.
           in ns sdc.example.
           in ns foo.example2.
           in ns bar.example2.
           in ns kip.example3.

pdc.example. in a 192.168.42.53
sdc.example. in a 192.168.53.53
foo.example2. in a 172.17.80.53
```

[80] The data used for these calculations are averages from statistics provided by the 12 root server operators.

bar.example2. in a 172.18.80.80
kip.example3. in a 10.10.10.10

Such a record set consists of roughly 350 bytes. According to the root server operators, the current distribution system could absorb a growth of 20K to 40K in the size of the root zone file without noticeable effect on propagation delay or jitter and without pushing any root server operator out of its "normal operations" region. Presuming ordinary delegations, no labels larger than four characters, no more than five name servers, no IPv6, and no DNSSEC, simple arithmetic suggests that adding between 60 and 120 new TLD delegations would effectively be absorbed in normal operations.

6.9.3 IPv6

Changing the canonical delegation to include an equal number of IPv6 records would change the size of the record set to roughly 500 bytes. Adding IPv6 therefore reduces the number of new delegations that can be absorbed by the distribution system without noticeable effect.

6.9.4 DNSSEC

Adding DNSSEC records to the RRset of any TLD delegation will result in a significant increase in the size of the root zone. Although the details for "signing the root" are still uncertain, it is possible to extrapolate from current NIST guidelines [45] and empirical data from the IANA DNSSEC testbed[81] to reach the following "best guesses" at the value of the following parameters:

- key signing key (KSK) length: 2048 bits
- KSK re-signing interval: 1 week
- number of KSKs: 1
- number of zone signing keys (ZSKs): unspecified

Very recent data from VeriSign suggest that the complexity of DNSSEC in the root will be greater than can be expressed by a few simple variables. Current testing considers not only the steady state in which root zone data are signed, but also situations in which keys are changed or rolled over; and it recognizes distinctions between the types of keys and their validity periods.

In VeriSign's testing, full responses ranged from 1509 to 2601 bytes; minimal responses ranged from 736 to 1700 bytes. All of the responses exceeded the non-EDNS0 limit of 512 bytes.

Extrapolation from the data incorporated into the root system analytical model suggests that in a signed root every delegation RRset will grow by a factor of 4 for delegations without DS records

[81] https://ns.iana.org/dnssec/status.html

and by a factor of 8 for delegations with DS records. Currently only about a dozen delegations have DS records. This number will grow as TLD registries begin signing their own data. That growth would increase the delay in the distribution channel (without staging) from 8 minutes to 32 minutes. With staging, the delay would increase from 50 minutes to roughly 75 minutes. The latest test results from VeriSign suggest that the delay will be higher than these extrapolations.

6.10 Combining DNSSEC with other root zone changes

The substantial over-provisioning of the root servers with respect to their ability to handle query load, and the headroom that currently exists in the distribution channel for updating the root zone file, implies that adding a small number of new delegations to the root zone would be absorbed by the publication system with minimal effect. This observation applies most directly to new delegations that have the characteristics of the canonical delegation example above. IDNs and additional NS, A, or IPv6 (AAAA) records in new delegations would produce a larger effect on propagation delay, but could still be accommodated by the distribution and query components of the root system without an operational disruption or OAM&P discontinuity for the root server operators.

However, adding DNSSEC, either to the root zone as presently constituted or in combination with the other changes described in Section 2.5, would immediately push the root server operators into a re-planning discontinuity (see Figure 4 in Section 4.1) in order to deal with the expected increase in TCP query load associated with non-EDNS0 capable resolvers and other systems that cannot handle DNS responses larger than 512 bytes.[82] The root server operators have spent roughly 3 months in this re-planning phase since the 3 June 2009 announcement by NTIA, ICANN, and VeriSign that the root would be signed "by year's end."[83]

This suggests that adding DNSSEC to the existing root zone would cause enough direct and indirect impact to the root system and the Internet at large that would preclude combinatorially adding any other feature until the operations have returned to normal ... DNSSEC will take some time, perhaps another 12 to 15 months to amalgamate into the system.

Although the stability risk associated with adding DNSSEC to the root is greater than for any of the other changes described in Section 2.5, it does not follow that prudent policy should favor the "less risky" introduction of new TLDs, IDNs, and IPv6 addresses over the deployment of DNSSEC. The effects of signing the root would be felt immediately—a sudden jump in the size of the root zone, and a sudden increase in the volume and type (TCP vs. UDP) of root server

[82] A description of the spike in TCP query traffic at .ORG when it was signed on 2 June 2009 is at https://www.dns-oarc.net/node/199.
[83] http://www.ntia.doc.gov/press/2009/OIA_DNSSEC_090603.html

query traffic. The effects of the other changes would be spread out over some period of time (longer or shorter depending on the rate at which the system is able to adapt). Because the step-function impact of signing the root will be proportionally greater the larger the root becomes, deploying DNSSEC *before* the other changes have increased the size of the root would significantly lower the risk it presents to DNS stability. Figure 6 illustrates the effect on the root server operators' defensive headroom—which bounds their ability to absorb sudden increases without service disruption—of signing the root when it is still relatively small versus waiting to sign the root until it has grown substantially.

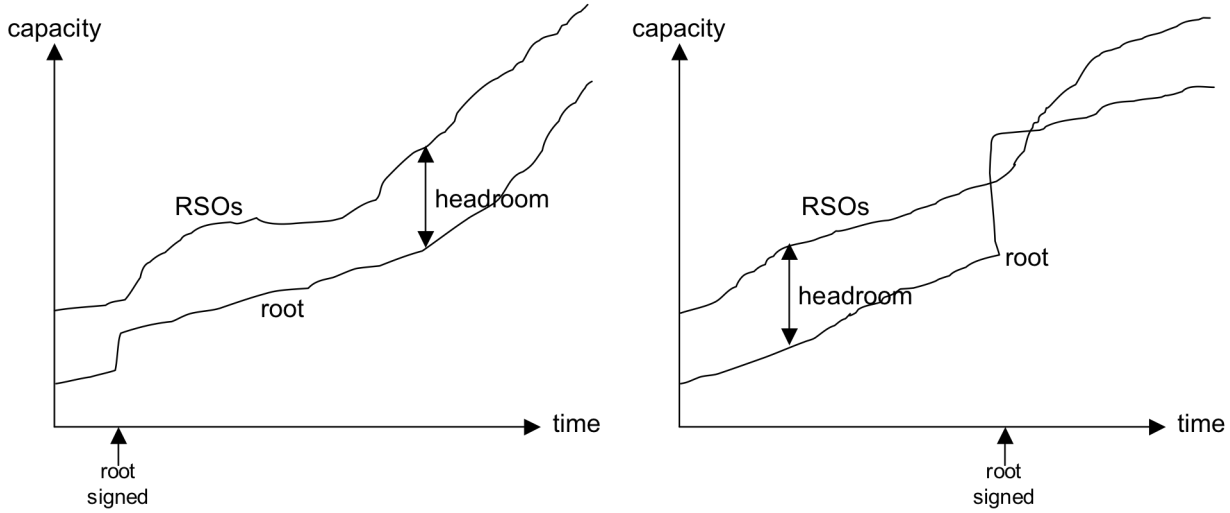

Figure 6—Root signing effect on root server operator headroom

6.11 Implications

6.11.1 Priming query and response

Although the details are still being debated in the IETF and elsewhere, it is clear that when the size of the priming query response grows above 512 bytes, the root system will encounter *some* set of issues. We will hit this regardless of what else is happening to the root zone. We can patch for IPv6 glue and possibly for some other changes, but those measures will only delay the time at which the size of the response to the priming query grows above 512 bytes.

6.11.2 Zone transfer

Already today some of the root server operators report occasional or persistent problems with zone transfer over the Internet to some "distant" (poorly connected or poorly supplied) anycast nodes. This suggests that over time, as the zone grows, if the growth is faster than the average improvement in Internet connectivity throughout the world, more existing or potential anycast

sites will become untenable. If the growth is slower than the average quality increase, root server operators will be able to add nodes.

Root server operators and DNS management companies report that normal zone transfer to any site (not just poorly connected "distant" sites) becomes infeasible when the number of records in a zone file reaches approximately 20,000,000. The operators of large ccTLDs give similar reports. The experience of these operators suggests that managing a zone with 1,000,000 records is readily accomplished with today's technologies and root server architecture, and that at some point after a zone grows to 10,000,000 records it becomes unmanageable without significant change to the server architecture.

6.11.3 Human intervention and EPP

Processes that include steps involving human inspection can scale only by adding people (FTEs); and beyond a certain point, organizational dynamics limits the effectiveness, and eventually even the possibility, of adding people to a process. Early simulation results suggest that the human-inspection bottlenecks at IANA and NTIA break down when the root zone contains entries for between 3,000 and 8,000 TLDs, depending on other variables. The current requirements for multiple human intervention steps in the root zone management process therefore appear to limit the growth of the root zone to $O(100)$ entries.

Experience with the operation of large ccTLD zones suggests that the way in which EPP is used today by IANA and VeriSign will not scale beyond $O(10{,}000)$ root zone entries. The .SE operators report that they had to move to an agreed-upon database model and remove pre-screening of change and registration requests in 2001, when the .SE zone reached approximately 100,000 domains. Similar data from .NL show that human intervention became impossible after 70,000 entries, long before which they had simply "hired 10 students to hit the continue buttons"; the .NL domain started to experience problems when it was managed by humans at around 10,000 entries.

7 Frequently asked questions

This section collects, in no particular order, some of the questions that have been asked about the root system and the root scaling study. Where the answers are to be found in this report, the pointers are included here.

Q1 Is there a difference between a "hidden master" and a "distribution master"?

A1 A name server that is authoritative for a zone is "hidden" when there is no NS record referring to it (*i.e.*, the zone is not delegated to that name server). The term "master" (or "primary") implies that the server has direct access to the actual data, compared to a "slave" (or "secondary") server which obtains the data from some other authoritative server. A distribution master is therefore a server from which slave servers can fetch data. Distribution masters are normally hidden, and in the case of the root zone, the VeriSign distribution master is hidden.

Q2 Is there a difference between a "DNS request" and a "DNS Query"?

A2 The DNS standards use the term "query" for messages from a client to a server, and the term "response" for messages from a server to a client. The standards also talk about different types of queries, where one of the types is "a query" as compared to the type "notify" (about zone file changes) or "update" (with an update request about the zone). Because of this, the term "request" is sometimes used to refer collectively to all types of messages a client and server might exchange, and "query" for the queries that actually are of type "query."

Q3 What is a "root server"?

A3 The term "root server" refers to the entire system an organisation runs when providing root name service for one of the "letters" (A-M). In some cases this implies multiple servers (physical hosts) which the queries are load balanced over. In some cases the different servers are also located in geographically different locations. The latter is also called "anycast", where the same IP address is presented in different locations in the network topology, just like someone that is multihomed. See "DNS Root Name Servers Explained For Non-Experts" [9].

Q4 What is the "root system"?

A4 The term "root system" refers collectively to every process and every organisation involved in the care and feeding of the DNS root, from the submission of information to IANA from the TLD operators to the delivery of responses to DNS queries by the root name servers.

Q5 Why do you talk about only one root and one DNS name space?

A5 The importance of a single root and common name space is explained in [41] and in an IAB[84] letter to ICANN on 27 September 1999.[85]

Q6 Does the addition of TLDs have any impact on "alternative root zones"?

A6 When TLDs are added to the authoritative root, the risk for name space collisions with entries in alternative roots increases. See [Q5], specifically section 1.1 (Maintenance of a Common Symbol Set) of [41].

Q7 Who is responsible for the root zone?

A7 The responsibilities are shared. See this report for more information on how the sharing is currently set up.

Q8 Who decides whether a TLD can be added to the root?

A8 Decisions on new TLDs are ultimately made by the Board of Directors of ICANN, following a process[86] that was developed over a period of several years of consultation with the community.

Q9 How many root servers are there?

A9 The root system currently has 13 root servers, identified by the latin-alphabet letters A through M. These 13 servers are operated by 12 organisations. Collectively, the 13 root operators maintain root servers at 191[87] sites throughout the world. See http://www.root-servers.org for more information and up to date status.

Q10 Who pays for all of this?

A10 The root system consists of many processes and many organisations are involved. Each organisation has responsibility for the piece of the puzzle it operates, and each has its own business model for financing it. ICANN finances the IANA functions; the U.S. Government finances the functions that NTIA performs; and each root server operator finances its operations individually. There is no central funding agency or authority for the root system.

[84] Internet Architecture Board (http://www.iab.org).
[85] http://www.icann.org/correspondence/iab-tech-comment-27sept99.htm
[86] http://www.icann.org/en/topics/new-gtld-program.htm
[87] As of 25 August 2009.

Q11 What is the difference between IANA and ICANN?

A11 IANA is a function operated by ICANN under a contract with NTIA. ICANN also has a Memorandum of Understanding with the IETF regarding the IANA function [42]. A supplement to [42] was published in December 2006.[88]

Q12 What does "priming" mean?

A12 When a validating resolver is first started, it uses a hints file or other statically pre-configured initial "guess" to find a root server, and then it asks that root server for the current list of root servers (this is the "priming query"). The answer is the full list of thirteen root servers and their addresses (this is the "priming response").

Q13 Why can't we just add as many TLDs as we want to the root?

A13 Adding new TLDs increases the work load on the organizations that perform the many functions necessary to manage the information in the root zone. These organizations can adapt smoothly to increases in work load ("scale up") if those increases occur over time rather than all at once. If the increases happen too quickly, one or more of the organizations responsible for the root system might not be able to scale up fast enough without making significant changes to its normal mode of operation—changes that might require many months or years of planning and implementation.

Q14 How can we decide how many new TLDs to add each year?

A14 Because the root system is decentralized, each of its parts responds differently to increases in load. Beyond the very near term, we can't know in advance exactly how many TLDs can be added to the root, or how fast they can be added, because as soon as you start to add entries to the root each of the root system components adapts and changes in ways that cannot be predicted or effectively coordinated. That's why it's so important to build an "early warning system" that can (a) detect signs that one or more of the root system actors is reaching its limit for absorbing changes without major re-planning, and (b) take effective mitigating action when those signs are detected.

Q15 Can we add all of the "famous marks"?

A15 Not all at once. The number of "famous marks" is on the order of 40,000. Absorbing all of them while maintaining all parts of the root system within their normal operating regions would take many years. Re-planning to scale up the root system components into new operating regions capable of absorbing 40,000 new TLDs more rapidly would itself take at least 18 months.

[88] http://www.icann.org/en/general/ietf-iana-agreement-v8.htm

8 Topics for future study

The results of the root scaling study suggest that the following topics should be the subject of additional investigation and analysis.

8.1 Early warning system

This study has revealed the importance of being able to recognize and respond to signs of stress in the root system as it evolves dynamically in response to root zone changes and other changes in the Internet. Further study of the following issues should be undertaken:

- What are the relevant signs of stress in each of the root zone management functions? How can those be detected or measured?
- How should the community arrange for communication among the root zone management system actors to ensure that (a) timely intelligence is available to support the recognition of approaching discontinuities, and (b) effective cooperative action can be taken to mitigate the effects of discontinuities before they create problems?

8.2 Quantitative model

The quantitative model developed during the current study is a first step toward being able to simulate the effect of changes to the root on the operation of the root system. Further work to refine and calibrate this model would greatly increase its usefulness as a component of an early warning system.

8.3 Effect of root zone changes on the Internet

The focus of this study has been the effect of changes to the root zone on the root system itself. It is clear, however, that the changes described in Section 2.5 will also affect other parts of the Internet, including (for example) end-system applications such as web browsers; intermediary "middleboxes" that perform traffic shaping, firewall, and caching functions; and ISPs that "manage" the DNS services provided to customers. Some of these effects are mentioned in connection with the findings reported in Section 6, but as effects beyond the root system were out of scope, they have not been investigated thoroughly by this study.

9 References

1. Network Information Services Manager(s) for NSFNET and the NREN: INTERNIC Registration Services. NSF Cooperative Agreement No. NCR-9218742, 1 January 1993 (http://www.icann.org/en/financials/tax/us/appendix-11.htm).

2. Amendment 11 to Financial Assistance Award NCR-9218742: Cooperative Agreement Between NSI and U.S. Government. 7 October 1998 (http://www.icann.org/en/nsi/coopagmt-amend11-07oct98.htm).

3. DNS Stability: The Effect of New Generic Top Level Domains on the Internet Domain Name System. 6 February 2008 (http://www.icann.org/en/topics/dns-stability-draft-paper-06feb08.pdf).

4. Test Report: DNSSEC Impact on Broadband Routers and Firewalls [SAC035]. 16 September 2008 (http://www.icann.org/en/committees/security/ssac-documents.htm).

5. Announcement by ICANN of the Root Server System Root Scaling Study. 29 May 2009 (http://www.icann.org/en/announcements/announcement-29may09-en.htm).

6. Root Scaling Study Terms of Reference. 5 May 2009 (http://www.icann.org/en/committees/dns-root/root-scaling-study-tor-05may09-en.htm).

7. Accommodating IP Version 6 Address Resource Records for the Root of the Domain Name System—A Joint Report from the ICANN Security and Stability Advisory and Root Server System Advisory Committees [SAC018]. 26 January 2007 (http://www.icann.org/en/committees/security/sac018.pdf).

8. RSSAC Statement on ICANN's Proposed Next Steps for IDN Deployment in the Root Zone. 12 June 2007 (http://www.icann.org/en/committees/dns-root/rssac-idn-statement.htm).

9. DNS Root Name Servers Explained For Non-Experts—Internet Society Member Briefing 19. Daniel Karrenberg, September 2007 (http://www.isoc.org/briefings/019).

10. IANA Functions Contract SA1301-06-CN-0048 between NTIA and ICANN. 14 August 2006 (http://www.icann.org/general/iana-contract-14aug06.pdf).

11. Cooperative Research and Development Agreement Between ICANN and US Department of Commerce—Improvements to Management of the Internet Root Server System. 15 May 1999 (http://www.icann.org/en/committees/dns-root/crada.htm).

12. SSAC Response to IDN Program Director regarding ICANN's proposal for IDN deployment at the root level of the DNS [SAC020]. 23 July 2007 (http://www.icann.org/en/committees/security/sac020.pdf).

13. Public Comments on the ICANN Root Scaling Study. 29 May—31 July 2009 (http://forum.icann.org/lists/scaling).

14. Root Name Server Operational Requirements [RFC 2870, BCP 40]. June 2000 (http://www.ietf.org/rfc/rfc2870.txt).

15. A Study on the Performance of the Root Name Servers. Kenjiro Cho, Akira Kato, Yutaka Nakamura, Ryuji Somegawa Yuji Sekiya, Tatsuya Jinmei, Shigeya Suzuki, and Jun Murai (WIDE Project), 3 April 2003 (http://mawi.wide.ad.jp/mawi/dnsprobe).

16. NeTraMet in WIDE Project—Real-time Measurement of Round-trip Time for Queries to and Responses from Root DNS Servers. (http://dnstap.nc.u-tokyo.ac.jp/NeTraMet).

17. NeTraMet—a Network Traffic Flow Measurement Tool. Nevil Brownlee (http://www.caida.org/tools/measurement/netramet).

18. Implementing DNSSEC at the Root—ICANN DNSSEC Workshop. Ashley Heineman/NTIA, 24 June 2009 (http://syd.icann.org/files/meetings/sydney2009/presentation-dnssec-workshop-heineman-24jun09-en.pdf).

19. Scaling Up The Root Zone—The Good, The Bad and The Scary. Johan Ihrén, May 2009 (http://www.ripe.net/ripe/meetings/ripe-58/content/presentations/Ihren-Root_Scaling_Issues.pdf).

20. ICANN Root Zone Scaling Study Workshop [video recording and transcript]. 22 June 2009 (http://syd.icann.org/node/3806).

21. IANA Technical Requirements for Authoritative Name Servers. 8 June 2009 (http://www.iana.org/procedures/nameserver-requirements.html).

22. A Day at the Root of the Internet. Sebastian Castro, Duane Wessels, Marina Fomenkov, Kimberly Claffy. ACM SIGCOMM Computer Communication Review Volume 38, Number 5, October 2008 (http://portal.acm.org/citation.cfm?id=1452341).

23. Adding IPv6 Glue to the Root Zone. Ronald van der Pol, Daniel Karrenberg, October 2003 (http://www.nlnetlabs.nl/downloads/publications/ipv6/v6rootglue.pdf).

24. Root Zone Augmentation and Impact Analysis. Duane Wessels, Geoffrey Sisson, July 2009 (not yet available).

25. Resolver Analysis for a Signed Root. NLnet Labs, 22 June 2009 (http://www.nlnetlabs.nl/downloads/publications/rs_analysis_06.pdf).

26. Signposts in Cyberspace: The Domain Name System and Internet Navigation. Committee on Internet Navigation and the Domain Name System: Technical Alternatives and Policy Implications, National Research Council, April 2005 (http://www.nap.edu/catalog/11258.html).

27. Operational Criteria for Root Name Servers [RFC 2010]. October 1996 (http://www.ietf.org/rfc/rfc2010.txt).

28. Domain Names—Implementation and Specification [RFC 1035]. November 1987 (http://www.ietf.org/rfc/rfc1035.txt).

29. Distributing Authoritative Name Servers via Shared Unicast Addresses [RFC 3258]. April 2002 (http://www.ietf.org/rfc/rfc3258.txt).

30. Domain Names—Concepts and Facilities [RFC 1034]. November 1987 (http://www.ietf.org/rfc/rfc1034.txt).

31. Extensible Provisioning Protocol [RFC 4930]. May 2007 (http://www.ietf.org/rfc/rfc4930.txt).

32. Initializing a DNS Resolver with Priming Queries [Internet Draft—expired]. July 2008 (http://tools.ietf.org/id/draft-ietf-dnsop-resolver-priming-01.txt).

33. DNS Referral Response Size Issues [Internet Draft—expired]. July 2008 (http://tools.ietf.org/id/draft-ietf-dnsop-respsize-11.txt).

34. DNS Security Introduction and Requirements [RFC 4033]. March 2005 (http://www.ietf.org/rfc/rfc4033.txt).

35. DNSSEC and Transaction Signatures (TSIG). Bill Manning, Akira Kato, and Mark Kosters. 10 December 2001 (http://www.rssac.org/notes_papers/2001rs-tsig6.html).

36. Traffic Impact—A Presentation to the Signed Root Deployment Symposium (Reston, VA, 11-12 June 2009). Wouter Wijngaards and Olaf Kolkman ().

37. Evaluating the effects of anycast on DNS root name servers. Lorenzo Colitti, Erik Romijn, Henk Uijterwaal, and Andrei Robachevsky, October 2006 (ftp://ftp.ripe.net/ripe/docs/ripe-393.pdf).

38. Internationalizing Domain Names in Applications (IDNA) [RFC 3490]. March 2003 (http://www.ietf.org/rfc/rfc3490.txt).

39. An analysis of wide-area name server traffic: a study of the Internet Domain Name System. Peter Danzig, Katia Obraczka, and Anant Kumar, ACM SIGCOMM Computer Communication Review 22:4, October 1992 (http://doi.acm.org/10.1145/144191.144301).

40. Extension Mechanisms for DNS (EDNS0) [RFC 2671]. August 1999 (http://www.ietf.org/rfc/rfc2671.txt).

41. IAB Technical Comment on the Unique DNS Root [RFC 2826]. May 2000 (http://www.ietf.org/rfc/rfc2826.txt).

42. Memorandum of Understanding Concerning the Technical Work of the Internet Assigned Numbers Authority [RFC 2860]. June 2000 (http://www.ietf.org/rfc/rfc2860.txt).

43. Domain Name System Security Extensions [RFC 2535]. March 1999 (http://www.ietf.org/rfc/rfc2535.txt).

44. Robustness in the strategy of scientific model building. George E. P. Box, in *Robustness in Statistics*, R.L. Launer and G.N. Wilkinson, Editors. 1979, Academic Press: New York.

45. Secure Domain Name System (DNS) Deployment Guide. NIST Special Publication 800-81r1 (draft for review), 26 August 2009 (http://csrc.nist.gov/publications/drafts/800-81-rev1/nist_draft_sp800-81r1-round2.pdf).

46. Punycode: A Bootstring encoding of Unicode for Internationalized Domain Names in Applications (IDNA) [RFC 3492]. March 2003 (http://www.ietf.org/rfc/rfc3492.txt).

47. OpenPGP Message Format [RFC 4880]. November 2007 (http://www.ietf.org/rfc/rfc4880.txt).