Regular Paper

DNS Traffic Analysis — CDN and the World IPv6 Launch

Kazunori Fujiwara[1,2,a)] Akira Sato[1,b)] Kenichi Yoshida[1,c)]

Received: October 15, 2012, Accepted: March 1, 2013

Abstract: The Domain Name System (DNS) is a key naming system used in the Internet. Recently, the deployment of IPv6 (especially after the World IPv6 Launch) and DNS prefetching in web browsers has significantly changed DNS usage. Furthermore, content delivery networks (CDNs) use complicated DNS configurations together with small TTL values to control their traffic. These three factors significantly increase DNS traffic. Thus, the importance of DNS traffic analysis has been increasing to properly maintain DNS operations. This paper presents an analysis of DNS full resolver traffic at the University of Tsukuba in Japan. What we found are 1) The deployment of IPv6 has increased queries from clients as much as 41%, 2) The deployment of CDNs increases the use of small TTL values, the use of CNAME resource records and the use of out-of-bailiwick DNS server names. Since these increases are making the DNS cache hit rate low and the DNS response slow without recognition by Internet users, this paper seeks to warn application designers of potential system design risks in current Internet applications.

Keywords: DNS, IPv6, CDN, content delivery network

1. Introduction

The Domain Name System (DNS) [1], [2] is a key naming system used in the Internet. DNS translates human readable domain names into IP addresses. Although it is technically possible for recent Internet applications such as search engines to help naive Internet users to find appropriate IP addresses without the help of DNS, such usage is not common. Thus the importance of DNS will continue to increase as the importance of the Internet increases.

Although the importance of DNS has been increasing, the load on DNS and the potential of DNS problems are also increasing. For example, DNS prefetching is a common technique used to speed up Web browsers these days. Since prefetching multiplies DNS traffic, the increased DNS traffic impacts the DNS. To achieve fast access for Web sites world-wide, it is common to use content delivery networks (CDNs). A CDN uses CNAME resource records, out-of-bailiwick server names and small TTL values. These three factors also increase overall DNS traffic. Emerging IPv6 clients also increase DNS traffic.

The importance of DNS has been recognized since the beginning of the Internet, and there have been a number of research projects on DNS behavior [3], [4], [5], [6], [7], [8], [9]. These research projects have sought to clarify the behavior of DNS. The DNS cache hit rate and its effects on shortening response times have been studied extensively.

However, new issues related to DNS have been coming into existence with the deployment of new services. The effects of CDNs and IPv6 mentioned above are some examples of these new issues. Thus, this paper tries to analyze the effects of CDNs and IPv6 together with other statistics, e.g, overall cache hit rates and error rates, which have also been studied in previous studies. Although our careful observation of DNS behavior will continue, the recent characteristics of DNS traffic and issues of DNS operation required by CDNs and IPv6 are the findings that we are reporting in this paper.

This paper reports on the results of our recent analysis on DNS behavior. The impacts on DNS and IPv6 deployments (especially after the World IPv6 Launch) are the primary results reported in this paper. The rest of this paper is organized as follows. Section 2 summarizes past research on DNS traffic. Then, recent characteristics of DNS traffic are qualitatively described in Section 3. Section 4 explains the data collection system at our university that we used in this study. Section 5 reports our results. Section 6 discusses issues related to DNS operation, and Section 7 summarizes our findings.

2. Related Works

The increase of DNS traffic has required the reinforcement of each DNS server, and the reinforcement of DNS itself. As such, there has been a great deal of research analyzing DNS traffic. Since caching is an important mechanism used to mitigate heavy DNS traffic, not only the traffic itself but also the cache hit rates have been analyzed in such research.

Since most of the research has been conducted on large DNS servers, this section classifies previous studies according to the class of the DNS servers, i.e., root DNS servers, top-level domain (TLD) DNS servers, and full resolvers of large Internet service providers (ISPs), which have been used as data acquisition sites.

2.1 Root DNS Traffic

Since the traffic and the load caused by the traffic are important information for DNS equipment reinforcement planning, root

[1] University of Tsukuba, Tsukuba, Ibaraki 305–8577, Japan
[2] Japan Registry Services Co., Ltd., Chiyoda, Tokyo 101–0065, Japan
[a)] fujiwara@jprs.co.jp
[b)] akira@cc.tsukuba.ac.jp
[c)] yoshida@gssm.otsuka.tsukuba.ac.jp

DNS operators have been working with academia to carefully analyze root DNS traffic and behavior. The Cooperative Association for Internet Data Analysis (CAIDA) has coordinated DNS traffic collection events which have collected traffic for two whole days each year since Jan. 2006. Liu et al. reported their first results in "Two days in the life of the DNS anycast root servers" [5].

Zdrnja et al. [6] have reported that their DNS response data had included typo squatter domains, fast flux domains and domains being used (and abused) by spammers. They observed that current attempts to reduce spam increased the number of A records being resolved. They also observed that the data locality of DNS requests was diminished because of the domains advertised in the spam.

Castro et al. have reported on the DNS evolution process [7]. They analyzed root DNS server traffic and pointed out several difficulties.

2.2 TLD DNS Server

In 2006, Larson et al. reported on DNS full resolver behavior which caused a significant volume of queries to be sent to the root and TLD DNS servers [4]. These abnormal DNS queries were found during the operation of their largest TLD (i.e., .COM and .NET) DNS servers. They proposed a solution to developers of an iterative resolver to alleviate these unnecessary queries. Their recommendation was a direct by-product of their observation and analysis of abnormal query traffic. This abnormal traffic was found at two of the thirteen root name servers and all thirteen com and net TLD name servers. Their findings contributed to the normalization of DNS operations.

2.3 Full Resolver

In 2002, Jaeyeon et al. reported on DNS performance and the effectiveness of caching [3]. They collected all the network traffic at two universities for one week and analyzed it. They reported on the behavior of authoritative DNS servers from a client viewpoint and the effectiveness of caching. They reported that 23% of lookups received no answer, 13% of lookups received other errors, and 27% of the queries sent to the root name servers received errors. They also reported on the relationship between TTL and DNS cache effectiveness. Their findings were 1) Reducing the TTLs of A records to as low as a few hundred seconds had little adverse effect on cache hit rates. 2) Little benefit was obtained by sharing a forwarding DNS cache among more than 10 or 20 clients. Their findings suggested that client latency was not as dependent on aggressive caching as had been commonly believed. The widespread use of dynamic low-TTL A-record bindings should not greatly increase DNS-related wide-area network traffic.

In 2010, Iinou et al. presented information about the effect of DNS query traffic increases on caching DNS servers [8] at a DNS OARC meeting which was an operators' community workshop. Their main conclusion was that their customers' queries had increased by about 50% for that one year and the cause of the increase was Firefox's DNS prefetching. DNS prefetching doubled queries from Firefox. This might cause NAT box or firewall problems.

In 2011, Koc et al. presented a global reference model of the DNS [9] at a DNS EASY 2011 Workshop. They evaluated end user query trends, the behavior of major DNS server software and proposed their DNS model.

2.4 Recent Issues

Although enthusiastic studies have been conducted, new issues related to DNS have been coming to light with the deployment of new services. The effects from CDNs and IPv6 are examples of such new issues, and have not been covered by previous studies. In the following sections, we will report our findings on these issues.

Early versions of this paper were published as Ref. [10] in IEEE/IPSJ 12th International Symposium on Applications and the Internet. In addition to the facts reported in Ref. [10], this present paper also reports on the impact of the World IPv6 Launch on 6th June 2012. After the World IPv6 Launch, major ISPs, home networking equipment manufacturers, and web service companies around the world have enabled IPv6 services. How this event affects DNS behavior is also reported in this paper.

3. Recent DNS Issues

Recently, the deployment of IPv6 and DNS prefetching in web browsers has significantly changed DNS usage. Furthermore, CDNs use a complicated DNS configuration together with small TTL values to control their traffic. This also significantly increases DNS traffic. Before reporting on recent statistics, this section explains the effects of IPv6 and CDNs on DNS behavior.

3.1 AAAA (IPv6 Address) Queries

Ten years ago, there were only a few IPv6 end nodes. However, recent operating systems support IPv6 and recent clients send AAAA (IPv6 address) queries for DNS. Although there are a small number of IPv6 aware services and most Internet users are not aware of the fact, the half-finished deployment of IPv6 services creates useless DNS traffic. **Figure 1** shows the typical situation.

As shown in the figure, recent clients that support IPv6 send both A (IPv4 address) and AAAA (IPv6 address) queries to their full resolver. When a user types "http://www.example.jp/" in the browser, the client sends an A query to the full resolver first. Then

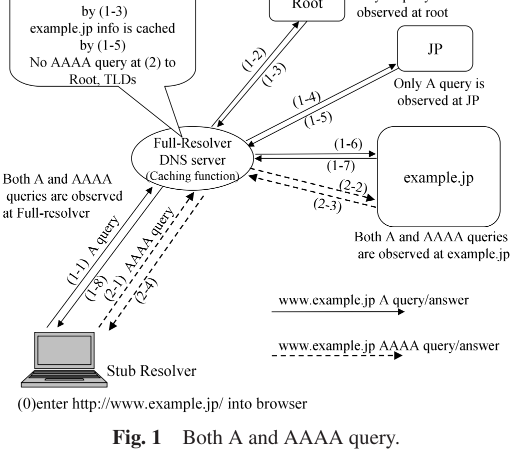

Fig. 1 Both A and AAAA query.

it sends an AAAA query in this example.

If the delay of an AAAA query from an A query is larger than the duration of the full resolver's action, "JP" and "example.jp" information is cached before the AAAA query from the client reaches the full resolver. In such a situation, the full resolver will not send the AAAA queries to the root and TLD DNS servers. It directly sends the AAAA queries to "example.jp" using the information cached by the previous A query.

As a result, AAAA queries are observed both at the full resolver and at the "example.jp" DNS servers. However, they are not observed at the root and TLD DNS servers. Using the full resolver installed at our university, we measured traffic-related AAAA queries. The results will be reported in Section 5.2. Section 5.2 also reports on the effect of AAAA queries on the DNS cache system.

Since the World IPv6 Launch, major web services have been supporting IPv6 services. Differences after the World IPv6 Launch are described in Section 5.5.

3.2 CDN Queries

Recently, the volume of CDN traffic has been rapidly increasing. CDNs try to guide user traffic to a server nearest the user, or to vacant servers. To support this, CDNs use small TTL values. CDNs sometimes use out-of-bailiwick DNS server names and CNAME resource records for easier system design and understanding. The use of small TTLs, the use of CNAMEs and the use of out-of-bailiwick DNS server names are three major concerns to DNS operations.

3.2.1 CNAMEs and Out-of-bailiwick DNS Server Names

A CNAME is an alias mechanism of DNS used to guide traffic to the specific CDN server from its original domain name.

When a full resolver confronts an alias, i.e., a CNAME, it has to restart the name resolution process from the alias. If the TLD of the original query and the TLD of the alias are different, a new resolution process has to start from the root DNS servers. Inefficient use of CNAMEs doubles queries for the root and TLD DNS servers. It also wastes full resolver DNS server resources.

What makes things worse is the use of out-of-bailiwick DNS server names (See **Fig. 2**). If a customer zone's DNS server name is outside the customer's zone, the DNS server is said to be "out-of-bailiwick." For example, if the DNS server for "example.co.jp" is "ns1.example.com," the JP DNS answers that "example.co.jp" DNS server is "ns1.example.com" without attaching "ns1.example.com" addresses because "example.com" is outside "JP."

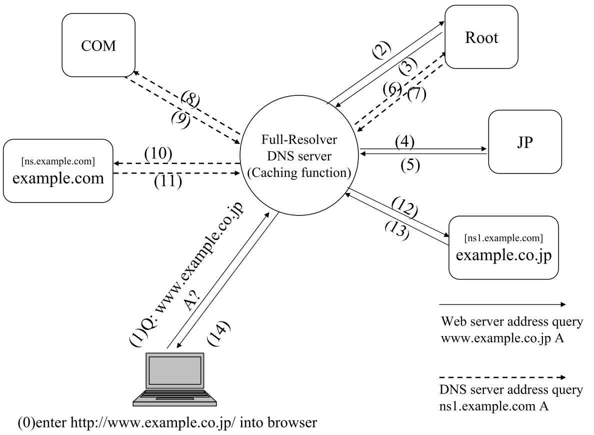

Fig. 2 Out-of-bailiwick example.

Figure 2 shows the process in detail. First, the full resolver starts with the root DNS servers (2). And then it sends the "www.example.co.jp" query to the JP DNS servers (4). The JP DNS knows that "example.co.jp" is hosted by "ns1.example.com" (i.e., out-of bailiwick), but it does not know "ns1.example.com" because "ns1.example.com" is outside JP. Then it has to resolve the "ns1.example.com" address from the root DNS servers (6), COM DNS servers (8), and "example.com" DNS servers (10). Finally, the full resolver gets the "ns1.example.com" IP address by receiving it (11), and sends a query about "www.example.co.jp" to "example.co.jp" DNS servers (12).

As shown above, the resolving process with out-of-bailiwick DNS servers is very complicated and requires full resolver resources. The use of the out-of-bailiwick DNS server name increases dependencies on other DNS servers and increases the load on the root DNS servers and TLD DNS servers.

3.2.2 Small TTLs

Each DNS response has a TTL parameter which is a 32 bit unsigned integer that specifies the length of time (in seconds) that the resource record will be cached. Each zone manager specifies TTL values for each DNS data. Most DNS data in the root, .com, and .net zones have a 2 day TTL. Most DNS data in .org and .jp zones have a 1 day TTL. These long TTLs are implemented to make DNS caching effective.

However, CDNs frequently change domain name and IP address mapping. Frequent changes require a small TTL value so that the full resolvers' cache does not interfere with the change. Thus, a small TTL value set by a CDN increases DNS traffic by making the DNS cache ineffective.

CDNs use larger TTL values for their DNS zone information, their DNS server addresses and negative responses other than their service TTL values.

3.2.3 Common Problems

The common problems of AAAA queries and CDN queries are not visible to Internet users. Few users notice the existence of unnecessary AAAA queries and complex out-of-bailiwick queries. Even worse, the traffic of AAAA queries is hidden from the operators of the root DNS servers and TLD DNS servers. To clarify these hidden problems, we analyzed the log data of the full resolver at our university. Since the traffic volume of our university reaches the same level as that of a medium-sized ISP, our results are useful for clarifying recent trends and hidden problems of DNS.

4. DNS Data Collection

We collected end user DNS queries and analyzed them. Our data collection method, an overview of the collected data and the definition of cache hit rate are described in this section.

4.1 System Configuration

We collected DNS queries by tapping a DNS full resolver op-

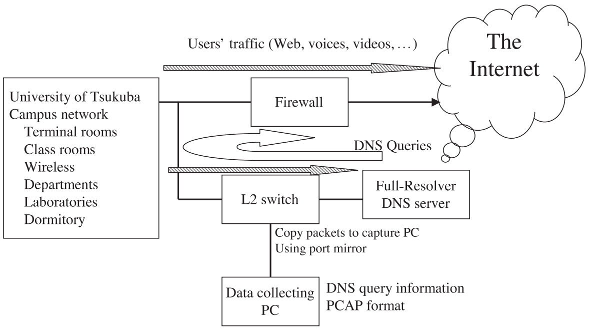

Fig. 3 Tapping DNS traffic at University of Tsukuba.

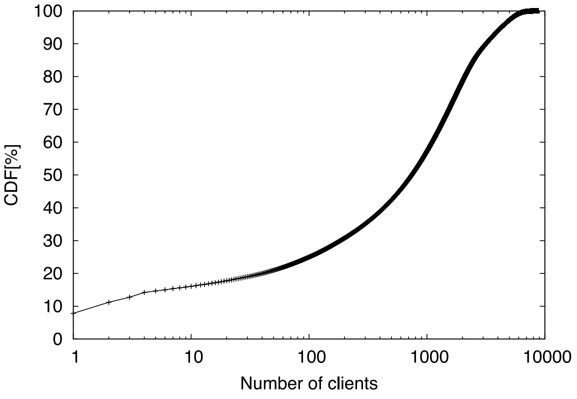

Fig. 4 Cumulative distribution of client queries.

erated by the Academic Computing & Communication Center of the University of Tsukuba for one year. The center provides the full resolver DNS server for the university. The tapping topology is shown in **Fig. 3**. At the tapping point, all incoming and outgoing packets of the full resolver are captured. The captured packets include 1) packets between stub resolvers and the full resolver, and 2) packets between the full resolver and authoritative DNS servers.

Although the DNS query information contains private end user information, we keep the information private by storing and analyzing captured data on only one machine. Only a limited number of researchers have access to this machine. The machine only exports analyzed statistics for this study.

4.2 Abnormal Traffic

A pilot survey found abnormal traffic in the collected data. **Figure 4** shows the cumulative distribution of client queries. As shown in the figure, the client that made the largest number of queries sent 8% of the total queries. It sent five "localhost.localdomain" queries every second. The client with the second largest number of queries sent repeated queries of some reverse domain names (near the query source IP address) every second. The third and fourth clients sent various queries and seem to be normal clients.

Since the behavior of the first and second clients seems to lack generality, we have removed the data on these two clients from our collected data.

4.3 Outline of Collected Data

After removing the abnormal traffic, we generated 3 one-month-long datasets. **Table 1** shows an outline of our datasets and compares them with Jaeyeon et al. [3]'s dataset. Although a direct comparison is almost meaningless since the data gathering methods are different, the total DNS query rate at the University of Tsukuba's November 2011 data is about 20 times larger than Ref. [3]'s dataset. The number of distinct query names is about 10 times larger. The number of clients is similar.

Table 1 DNS query dataset comparison.

Origin	Ref. [3]		Authors		
Place	MIT	KAIST	Tsukuba		
Year Month/Day	2000 12/04 –12/11	2001 05/18 –5/24	2010 11/1 –11/30	2011 11/1 –11/30	2012 7/7 –8/6
lookups	4,160,954	4,339,473	234,308,393	366,489,499	317,686,402
query/sec	6.88	8.37	90.40	141.39	122.56
query names	302,032	219,144	3,375,088	4,015,966	2,971,084
clients	1,216	8,605	6,556	8,815	11,662
query/sec /clients			0.01378	0.01603	0.01051

From November 2010 to November 2011, the total number of client queries increased 56% in our datasets. The number of query names also increased 19%. The query rate from each client increased 16% (from 0.01378 to 0.01603). The increase may indicate the increased use of browsers that support DNS prefetching.

From November 2011 to July 2012, the total number of client queries decreased 14%. We believe this decrease was caused by a decrease in activities due to examinations and the summer vacation.

4.4 Definition of Cache Hit Rate and TTL Value per Query

Since the cache mechanism is the most important mechanism of DNS for decreasing the response time of client queries, we evaluated the cache hit rate of the full resolver.

A single stub query, i.e., a DNS query from a client, might generate multiple queries from the full resolver. Before calculating the cache hit rate, we extracted packet captures as DNS query sequences that started with queries from end users and contained all iterative queries to resolve the end user queries. The query sequences might include CNAME aliases and out-of-bailiwick DNS server name queries.

Since queries have to be sent to multiple authoritative DNS servers including the root and TLD servers, a single stub query causes multiple queries. In this paper, the DNS cache hit rate is defined as:

$$\frac{\text{Number of stub DNS queries which do not cause}\atop\text{any query to authoritative DNS servers}}{\text{Number of stub DNS queries from clients}}$$

Other stub DNS queries which cause queries to authoritative DNS servers are considered to be cache misses.

Each stub query generates queries to multiple authoritative DNS servers and receives multiple TTL values. This paper treats the minimum TTL value received as the TTL value for the query.

5. Analysis of University of Tsukuba Data

In this section, November 2011 data is analyzed in detail first. Later, July 2012 data is analyzed to show the impact of the World IPv6 Launch.

Table 2 Cache hit rate and effect on authoritative DNS servers.

Case	Ratio	Cache	Authoritative queries			Latency
	in the	hit	divided by Stub queries			
	whole	rate	Root	TLDs	All	(ave.)
	[%]	[%]				[ms]
ALL	100.0	75.1	0.00079	0.025	0.31	28.0
A	61.6	72.7	0.00100	0.037	0.36	30.7
AAAA	29.1	74.1	0.00012	0.007	0.27	28.8
with CNAME	53.9	73.8	0.00070	0.025	0.34	30.4
without CNAME	43.2	76.7	0.00096	0.028	0.26	24.9
Normal Answer	57.2	72.3	0.00071	0.039	0.37	31.4
Error	42.8	78.9	0.00090	0.007	0.22	23.4
·Server Failure	2.9	77.8	0.00007	0.002	0.54	89.5
·No Data	27.3	75.4	0.00006	0.007	0.24	23.4
·Name Error	11.5	93.5	0.00315	0.008	0.06	5.3
·Name Error (wo TLD error)	11.1	93.5	0.00008	0.008	0.06	5.3
·TLD error	0.5	92.6	0.07303	0.000	0.07	5.4
Forward lookup	92.4	73.5	0.00044	0.027	0.33	29.8
Reverse lookup	7.3	95.7	0.00035	0.006	0.07	6.1
$TTL \leq 300$	44.0	67.3	0.00033	0.021	0.40	31.7
$TTL > 300$	56.0	81.3	0.00115	0.029	0.23	25.1

5.1 Cache Effectiveness

Table 2 shows the results of our analysis of our 2011 data (See Table 1). The first column classifies the stub queries from various aspects, i.e., the use of CNAMEs, any error they encounter, the use of reverse lookups, and the TTL value in the lookup sequence. The "ALL" row shows the average of all data. "with CNAME" and "without CNAME"[*1] mean that the response contains a CNAME or not. "Normal Answer" means that the stub resolver receives what it wants. "Error" means that the stub resolver does not receive what it wants. "Server Failure" means that the full resolver is unable to process the query due to configuration errors. "No DATA" means that the query name exists, but the name does not contain the query type entry. For example, a domain name has an IPv4 address, but does not have an IPv6 address entry. "Name Error" is an error in which the query name does not exist. "TLD error" is an error in which the TLD of the query name does not exist. "Name error" includes "TLD error" in principle. The "Forward lookup" row shows that the TLD of the query name is not "ARPA." The "Reverse lookup" row shows that the TLD of the query name is "ARPA." The "$TTL \leq 300$" row shows that the response TTL value is equal to or lower than 300. The "$TTL > 300$" row shows that the response TTL value is larger than 300.

The second column shows their ratios. The third column shows the cache hit rate as a percentage. The fourth column shows the number of queries for the root DNS servers divided by the number of stub queries. The fifth column shows the number of queries for the TLD DNS servers divided by the number of stub queries. The sixth column is the number of queries for all authoritative DNS servers divided by the number of stub queries. The seventh column is the latency to clients.

The average cache hit rate is 75.1%. Each stub query generates 0.00079 root DNS server queries, 0.025 TLD DNS server queries, and 0.31 authoritative DNS server queries on average. Here, a 0.31 authoritative DNS server query includes the root DNS server queries and TLD DNS server queries. Clients receive answers in 28.0 milliseconds on average.

The existence of a CNAME ("with CNAME") lowers the cache hit rate slightly and increases the number of authoritative queries a little. However, the differences are limited.

Small TTL (≤ 300) values seem to make the cache hit rate low. Since small TTLs show the effects of CDNs, a detailed analysis of them is described later.

Other characteristics which should be noted are:

- 57.2% of stub queries receive "Normal Answer"'s and the remainder (i.e., 42.8%) receive some errors. The stub queries that encounter errors have a slightly higher cache hit rate, i.e., 78.9% compared to the "Normal Answer" of 72.3%. This seems to result in a faster response of 23.4 milliseconds compared to a normal response of 31.4 milliseconds.
- Although a "Server Failure" has a slower response (89.5 milliseconds), they account for only 2.9%, so it does not cause a large problem.
- Both "Name Error" and "TLD error" have a high cache hit rate, i.e., 93.5% and 92.6% respectively. These high cache hit rates result in fast responses to their queries (5.3 and 5.4 milliseconds on average). These results do not suggest any problem. However, if we pay attention to the number of queries to the root DNS servers, "TLD error" does indicate a problem. Recall that the average stub query only generates a 0.00079 root query. A single stub query that encounters a TLD error generates 0.07303 root queries, which is 92 times larger than the average. To clarify this, "Name Error (without TLD error)" shows results that exclude "TLD error" from "Name error." As shown in the table, the average number of root queries for "Name Error (without TLD error)" decreases down to 0.00008.
- Viewed from the number of queries for the root DNS servers, the full resolver made 289,492 root DNS server queries in the month. However, for "TLD error," only 0.5% of stub queries made 45% of the root DNS server queries.

5.2 Increase of AAAA Queries

Ten years ago, most operating systems (OS) did not support IPv6 and it can be presumed that there were few AAAA queries. Recent OSs, e.g., Windows Vista, Windows 7, Mac OS X, Linux and *BSDs, send both A (IPv4 address) and AAAA queries at the same time even if they don't have IPv6 connectivity[*2]. **Figure 5** shows the query types from clients that our full resolver observed. 62% of queries are IPv4 address (A) queries, and 29% of queries are IPv6 address (AAAA) queries. 53% of queries are both IPv4 and IPv6 address queries occurring at the same time.

Thus, compared with ten-years ago, the deployment of IPv6-aware OSs has increased queries from clients up to 41% (29%/71%) and the increases are AAAA queries. Here, PTR queries (8%) are used to find the domain name from an IP address. The other 1% are such queries as finding mail server (MX) information.

[*1] The existence of a CNAME cannot be determined in the case of a "Server Failure." Thus 53.9 (with CNAME) + 43.2 (without CNAME) + 2.9 (Server Failure) makes 100% in Table 2.

[*2] Recent versions of Microsoft Windows try to use IPv6 tunnels if they don't have IPv6 connectivity. They send AAAA queries if they have IPv6 connectivity even if it is a tunnel.

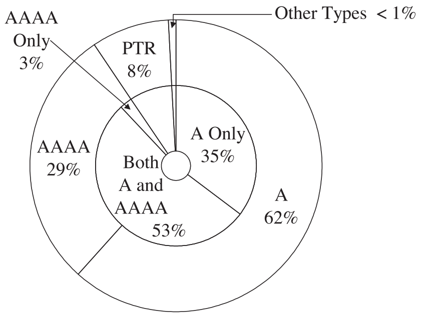

Fig. 5 Query types observed by full resolver, Nov. 2011.

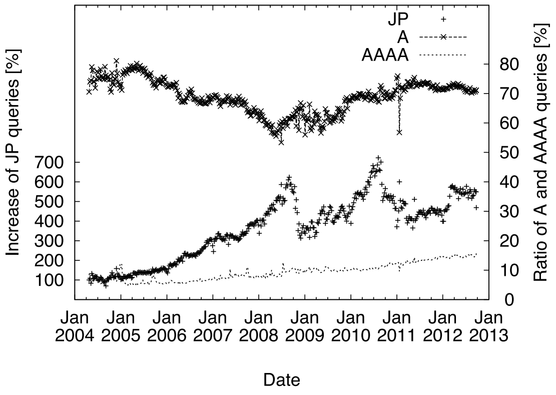

Fig. 6 JP query data sent to A.DNS.JP.

Table 3 A, AAAA query ratio seen at Root and JP.

Destination	Query type	Number of authoritative queries divided by number of stub queries
Root	A	0.007559
Root	AAAA	0.000116
Root	ALL	0.004445
JP	A	0.043914
JP	AAAA	0.009056
JP	ALL	0.030236

Note that this increase of AAAA queries is not observed at the TLD DNS servers. **Figure 6** shows the recent trend of DNS queries observed at the TLD servers. We use data collected at A.DNS.JP which is one of the JP TLD DNS servers and operated by JP TLD operator JPRS. The left side Y axis, "Increase of JP queries" shown by "+" marks, shows the number of observed queries divided by the number of queries as of April 2004. Since the JP TLD DNS has been built up twice, the line is folded. However, it shows the growth of queries for 8 years. The ratio of AAAA queries has been increasing each year from 7% in 2004 to 14% in 2011. However these numbers are far less than the 29% that is observed at the full resolver.

Table 3 suggests the reason why the increase of AAAA queries is not observed at the TLD DNS servers. It shows the ratio of queries which the full resolver received and generated. As shown in the table, the full resolver does not forward AAAA queries (0.000116 root queries and 0.009056 JP queries) compared to A queries (0.007559 and 0.043914). This decreases the outward appearance of AAAA queries at the root and TLD DNS servers.

Table 4 suggests the reason why the full resolver does not forward AAAA queries. As shown in Table 4 November 2011 data [*3], 88.2% of clients send both A and AAAA queries. However, most of them (79.3%) first send A queries. If the delay between the AAAA query and A query is larger than the duration of the full resolver's action, information from the root and TLD DNS servers is cached before the AAAA query. In such a situation, the full resolver will not send AAAA queries to the root and TLD DNS servers. The full resolver directly sends the AAAA query to an organization's authoritative DNS server using the information cached by the previous A query. Since most clients (79.3%) first send A queries, the caching mechanism seems to reduce the AAAA queries sent to the root and TLD DNS servers.

Table 4 Number of clients that send A and AAAA queries.

	Nov. 2011		July 2012	
Case	Hosts	%	Hosts	%
Total Hosts	8,815	100.0	11,662	100.0
Hosts, send A queries	8,707	98.8	11,588	99.4
Hosts, send AAAA queries	7,772	88.2	10,247	87.9
Hosts, send both A and AAAA queries	7,720	87.6	10,202	87.5
send A, then AAAA	6,993	79.3	9,303	79.8
send AAAA, then A	100	1.1	160	1.4
mix	627	7.1	739	6.3

Table 5 Answer types for queries, Nov. 2011.

Query Type	Total	A	AAAA	PTR	Other
Number of queries [million]	366.5 100%	225.7 62%	106.6 29%	31.1 8%	3.0 1%
Answer Type	%	%	%	%	%
Server Failure	2.9	0.6	1.8	23.3	1.3
Name Error	11.5	8.1	3.4	59.1	67.7
Refused	0	0	0	0	0.3
Normal Answer	57.2	89.3	2.0	17.4	17.1
No Data	27.5	0.7	92.6	0.1	12.2
Timeout	0.9	1.3	0.2	0.1	1.4

Table 5 classifies received answers for A, AAAA, PTR and other queries. "Timeout" means that the server did not answer. "Name Error" and "No DATA" are cached with a negative cache TTL value if the cache capacity allows. "Server Failure" and "Time out" are not cached.

As shown in Table 5, 92.6% of AAAA queries receive a "No DATA" response. This is clearly different from the situation for A queries, where 89.3% of A queries receive a "Normal Answer." Since there exists few IPv6 services, only 2.0% of AAAA queries receive IPv6 addresses.

Summaries of our findings on AAAA queries are:

(1) 88.2% of clients support IPv6 and send both A and AAAA queries, and 79.3% of clients send A queries first, and then send AAAA queries. As a result of this query order and the effect of the DNS cache, a small increase of AAAA queries is observed at the root and JP DNS servers. However, an increase of AAAA queries is clearly observed at full resolvers.

(2) There are small services that support IPv6, and most AAAA queries receive "No DATA" responses. In many cases, the negative cache TTL value is higher than or equal to the TTL value of normal (A) responses. The "No DATA" responses are cached and the effects on authoritative DNS servers are the same as "Normal Answer" responses.

Since IPv6 services are increasing, we expect that the portion

[*3] Table 4 July 2012 data will be discussed at Section 5.5.

of "No DATA" responses will change to "Normal Answer" soon. However, the caching mechanism and the query order of clients may keep hiding the increase of AAAA queries from the root and TLD DNS servers.

5.3 Short TTLs of CDNs

As shown in Table 2, the TTL value seems to affect the cache hit rate. As a further analysis of its effects, **Fig. 7** shows the cumulative distribution of TTL values, and **Fig. 8** shows the effects of TTL values on the cache hit rate.

As shown in Fig. 7, very short TTLs are in use, e.g., 6.4% have TTLs of 20 seconds and 6.8% have TTLs of 30 seconds. The volume of DNS queries that receives answers with TTLs of less than 31 seconds is 14.0%, and this creates a significant DNS load. These short TTLs are used by major content service providers and content delivery networks. For example, Akamai uses a TTL of 20 seconds. twitter.com and www.facebook.com use a TTL of 30 seconds. Yahoo uses a TTL of 60 seconds and Google uses a TTL of 300 seconds.

Figure 8 shows the cache hit rates at a given query frequency and TTL. In this figure, DNS queries are also classified into five average frequencies: 1 per day or less, 1 per day to 1 per hour, 1 per hour to 1 per minute, 1 per minute to 3 per minute, 3 per minute or more.

As expected, short TTLs tend to result in a low cache hit rate. However, high access frequency mitigates this tendency. For example, the cache hit rate of a TTL of 20 seconds which has more than a 3 per minute access is 76.1%. Since major content service providers have frequent accesses, the decrease in cache hit rate by the short TTL is hidden.

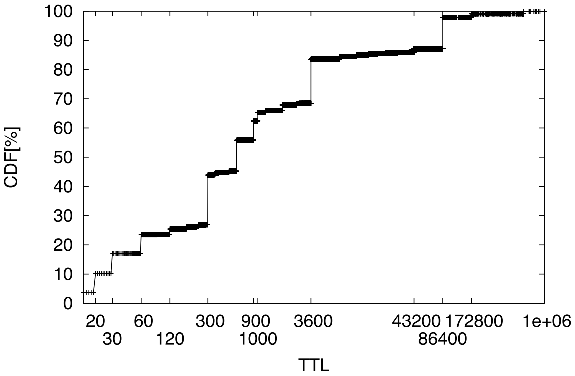

Fig. 7 Cumulative distribution of answer TTL.

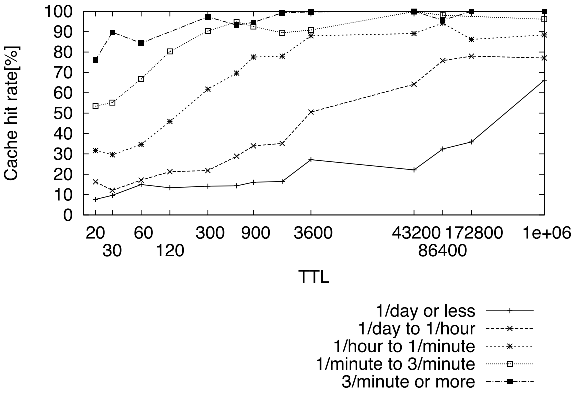

Fig. 8 Cache hit rate at given average query frequency and TTL.

Note that the mitigation of the high frequency does not mean a simplification of the DNS problem. Unnecessary high access frequency itself is an issue. If the TTL were increased from 20 seconds to 5 minutes, it would increase the cache hit rate 28% (76.1% to 97.3%) on a frequency of 3 per minute or more, and 69% (53.5% to 90.4%) on a frequency of 1 per minute to 3 per minute.

Since it also decreases the load of the full resolver and authoritative DNS servers, the service provider has to choose an appropriate TTL for their services. If a service provider with a low access frequency uses a short TTL, their customers will observe a low cache hit rate with a slow DNS response.

5.4 Effect of CNAMEs and Out-of-bailiwick DNS Servers

The use of CNAMEs and out-of-bailiwick DNS server names causes disordered DNS traffic as described in Section 3.2. **Figure 9** and **Table 6** show the related statistics.

As shown in Fig. 9, the use of CNAMEs lowers the cache hit rate by about 10%, and increases queries to the root DNS servers and TLD DNS servers [4].

This doubles queries for authoritative DNS servers. In the most frequent case (over 3 accesses per minute), the use of CNAMEs

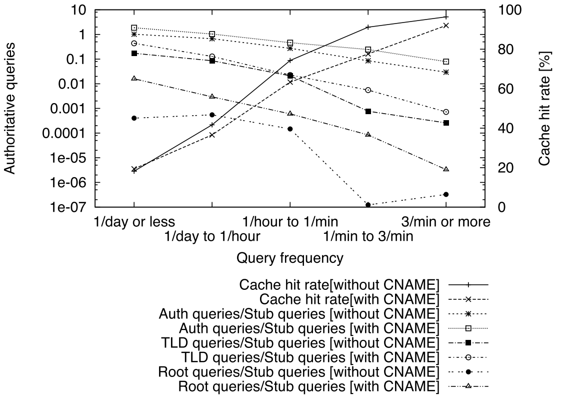

Fig. 9 Authoritative queries at given query frequency and CNAME.

Table 6 Cache hit rate and use of out-of-bailiwick DNS server names.

Case	Rate in the whole	Cache hit rate [%]	Authoritative queries divided by Stub queries			Latency to clients (average) [ms]
			Root	TLDs	All	
Total	100.0	75.1	0.00079	0.025	0.31	28.0
All servers are In-bailiwick	27.5	78.1	0.00035	0.009	0.24	22.5
All servers are Out-of-bailiwick (except .arpa)	60.1	71.5	0.00042	0.032	0.36	33.1
Some servers are Out-of-bailiwick	3.7	69.4	0.00118	0.039	0.40	28.6
Reverse lookup (.arpa)	7.3	95.7	0.00035	0.006	0.07	6.1
Undetermined	1.5	81.2	0.02559	0.116	0.29	31.5

[4] As shown in Table 2, "TLD error" responses generate many queries to the root DNS servers. In fact, 45% of queries to the root DNS servers is sent by "TLD error" responses. This disturbs the analysis shown in Table 2. It would seem to imply that the use of CNAMEs decreases queries to the root DNS servers and TLD DNS servers. However, this interpretation is misleading. Figure 8 shows results that exclude the effects of "TLD error" and "Name error" responses.

increases queries to the root DNS servers 10 times and queries to the TLD DNS servers 2.8 times.

Similar to the effects of the access frequency shown in Fig. 8, frequent access increases the cache hit rate. For example, a cache hit rate with a frequency of 1 ∼ 3 accesses per minute is 75% when CNAMEs are used. Although 75% is not a low figure, the cache hit rate without CNAMEs is 92%. The difference, i.e., 17%, amounts to the degradation of performance when CNAMEs are used. This degradation exists when the frequency is more than 1 access per day.

Table 6 examines the effects of out-of-bailiwick DNS server names on the cache hit rate. In this table, "Reverse lookup" has a separate row. Although a DNS reverse lookup uses out-of-bailiwick DNS servers, it is not caused by CDN services and should be handled separately because in-addr.arpa and ip6.arpa do not offer in-bailiwick DNS server name registrations.

The table shows the fact that the use of out-of-bailiwick names is common in today's Internet, and 60% of answers contain out-of-bailiwick name delegations. The use of out-of-bailiwick DNS server names increases queries for the root DNS servers and TLD DNS servers: a 20% increase (i.e., 0.00042 from 0.00035) of root DNS server queries, means 3.7 times larger (i.e., 0.032/0.009) TLD DNS server queries.

The most important degradation in performance due to the use of out-of-bailiwick names is the latency of DNS answers, which has a 50% degradation (33.1 milliseconds from 22.5 milliseconds). The use of an in-bailiwick DNS server can decrease this delay (See the next Section for details).

5.5 Changes after the World IPv6 Launch

Figure 10 shows query types from clients that our full resolver observed during July 2012. This is almost the same client behavior that can be seen in Fig. 5.

Table 4 and Fig. 6 also show similar client behavior after the World IPv6 Launch. Table 4 shows that the ratio of query patterns from clients did not change between November 2011 and July 2012.

On the contrary, **Table 7** shows clear changes in answer types for clients. If we compare Table 7 with Table 5, "Name Error" increases from 11.5% to 17.3%. "No Data" decreases from 27.5% to 19.6%. Significant differences exist in AAAA answers. "AAAA Normal Answer" which contains IPv6 addresses increases from 2.0% to 21.2%. "No Data" answer which does not contain IPv6 address decreases from 92.6% to 71.9%. After the World IPv6 Launch, hundreds of popular WWW sites configured AAAA resource records. This increased the AAAA answers that contain AAAA records.

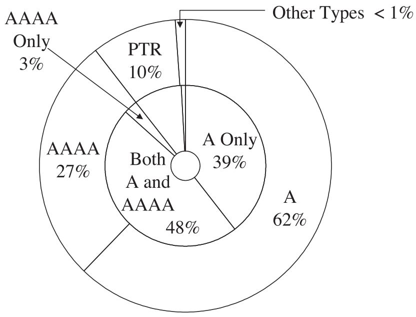

Fig. 10 Query types observed by full resolver, July 2012.

Table 8 shows the cache hit rate, queries to authoritative DNS servers and latency from the July 2012 dataset. Even if we compare Table 8 with Table 2, most of the tendencies are similar. In fact, although

- Queries for authoritative DNS servers slightly decreased
- Cache hit rates for AAAA queries slightly decreased from 74.1% to 71.6%
- Queries for authoritative DNS servers slightly increased

the differences are not significant.

Figure 11 shows the cumulative distribution of query names before and after the World IPv6 Launch for AAAA queries, "AAAA Normal Answer" and "AAAA No Data" answers. Queries have the same tendencies before and after the event. Before the event, query names that have IPv6 address answers do not occupy a significant number of queries.

After the event, a hundred query names that have IPv6 addresses occupy 16.5% of AAAA queries, a thousand query names occupy 20.0% of AAAA queries.

The most frequent query names that answer AAAA with IPv6 addresses are "www.facebook.com," "dns.msftncsi.com"[*5] and domain names used by Google services.

The impacts of the World IPv6 Launch are summarized as: "After the World IPv6 Launch, hundreds of popular query names are configured with AAAA resource records. Thus 21.2% of

Table 7 Answer types for [July 2012] queries.

Query Type	Total	A	AAAA	PTR	Other
Number of queries	317.7	200.2	84.3	30.1	3.1
[million]	100%	63%	26.5%	9.5%	1%
Answer Type	%	%	%	%	%
Server Failure	2.1	0.9	1.2	12.4	1.0
Name Error	17.3	14.8	5.1	64.7	47.3
Refused	0	0	0	0	0.0
Normal Answer	60.7	83.4	21.2	22.8	31.6
No Data	19.6	0.6	71.9	0.1	4.3
Timeout	0.4	0.3	0.1	0.1	15.6

Table 8 Cache hit rate and effect to authoritative DNS servers, July 2012.

Case	Ratio in the whole [%]	Cache hit rate [%]	Authoritative queries divided by Stub queries			Latency (ave.) [ms]
			Root	TLDs	All	
ALL	100.0	75.6	0.00063	0.020	0.29	26.8
A	63.0	74.1	0.00071	0.029	0.34	29.0
AAAA	26.5	71.6	0.00017	0.005	0.28	29.9
with CNAME	55.2	71.9	0.00052	0.020	0.36	30.8
without CNAME	44.4	80.9	0.00077	0.020	0.21	20.6
Normal Answer	60.7	73.0	0.00049	0.030	0.35	29.5
Error	39.3	79.7	0.00084	0.005	0.21	22.6
·Server Failure	2.1	59.4	0.00009	0.002	0.83	109.0
·No Data	19.5	69.2	0.00006	0.006	0.29	30.1
·Name Error	17.3	96.2	0.00183	0.004	0.04	3.0
·Name Error (wo TLD error)	16.8	96.3	0.00004	0.004	0.04	2.9
·TLD error	0.5	93.5	0.06387	0.000	0.06	5.6
Forward lookup	91.7	73.6	0.00033	0.021	0.32	29.0
Reverse lookup	7.9	97.9	0.00019	0.002	0.03	2.3
$TTL \leq 300$	46.1	65.8	0.00024	0.017	0.41	32.4
$TTL > 300$	53.9	84.1	0.00097	0.022	0.19	22.0

[*5] dns.msftncsi.com is used for network testing by Microsoft Windows.

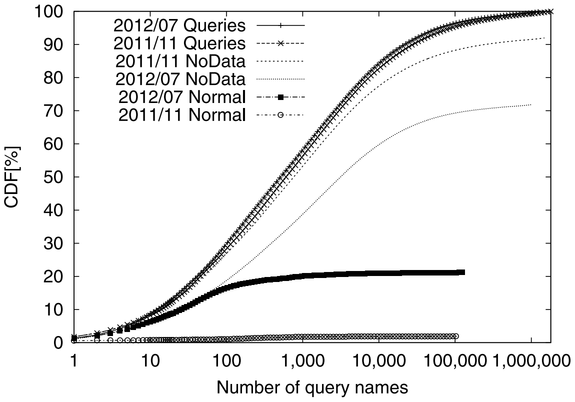

Fig. 11 Cumulative distribution of query names before and after the World IPv6 Launch.

AAAA answers contains AAAA records after the World IPv6 Launch. However, this fact does not affect DNS behavior significantly."

6. Countermeasures against Inefficient Use

As observed and shown in Fig. 5, recent end node clients make IPv6 address (AAAA) queries to their full resolvers even if they don't have IPv6 connectivity. This has already increased their full resolver loads and authoritative DNS server loads. Although this default client behavior can be changed by users to disable AAAA queries, very few users seem to change the default setting. If vendors were to modify the default configuration of their operating systems so that they did not generate unnecessary AAAA queries when the node lacks IPv6 connectivity, this small modification would reduce DNS packets by 29% at the full resolver. It would also reduce the access line traffic. Our findings can be used to improve the mechanism of client OSs which seem to be implemented by the insufficient understanding of DNS traffic. This fact also shows the importance of continuous traffic measurement researches which has possibility to improve the performance of the Internet software and systems.

As shown in the previous section, the use of out-of-bailiwick DNS server names results in a significant degradation in response time. It also increases the queries for the root DNS servers and TLD DNS servers. This inappropriate configuration seems to be caused by insufficient understanding of WWW server operators. However, the use of out-of-bailiwick DNS server names is easily fixed by changing the DNS server names by adding new in-bailiwick DNS server names for the DNS server IP addresses, and this is well known. By removing the accesses shown as dashed lines in Fig. 2, this countermeasure decreases queries to authoritative DNS servers by some 50%. According to the results shown in Table 6, this countermeasure can reduce DNS response times from 33 ms to 22.5 ms.

Suppression of the unnecessary CNAMEs can also reduce DNS response times from 30.4 ms to 24.9 ms. The usage of CNAMEs is also important. If both the owner name of a CNAME and its aliased name are included in the same domain, the CNAME does not increase queries for the root DNS servers and TLD DNS servers. For example, if we use "www.example.jp IN CNAME www.l.example.jp," both "www.example.jp" and "www.l.example.jp" are sub-domain names of "example.jp." By changing the aliased name to a name in the same domain like this example, we can reduce queries to the root DNS servers and TLD DNS servers.

The appropriate selection of a TTL can also reduce the DNS response time. The above issues related to out-of-bailiwick DNS server names, CNAMEs, and TTLs can be fixed by server operators. This paper intends to give them the basic information for that purpose.

7. Conclusion

Considering the recent deployment of IPv6 and CDNs, we have analyzed current DNS traffic and reported the following findings:

- The increase of IPv6 support increases IPv6 address (AAAA) queries. Today 29% of requests that a full resolver receives are AAAA queries.

  Although most of the AAAA answers are empty, they are well-cached. Because of the high cache hit rate and the query order of A and AAAA records, the increase of AAAA queries is not observed by the root and TLD DNS servers.
- After the World IPv6 Launch, hundreds of popular query names are configured with AAAA resource records. Thus 21.2% of AAAA answers contain AAAA records after the World IPv6 Launch. However, this fact does not affect DNS behavior significantly.
- The low TTL values used by CDN servers makes the DNS cache hit rate low. However, high-frequency access to major content providers hides this phenomenon.

  The volume of DNS queries that receives answers with TTLs of less than 31 seconds is 14.0%, which creates a significant DNS load.
- 60% of DNS traffic uses out-of-bailiwick DNS server names. This slows down the DNS response from 22.5 milliseconds to 33.1 milliseconds on average.
- 45% of queries to the root DNS servers came from wrong name queries from clients, and they are only 0.5% of all client queries.
- We have discussed countermeasures for unnecessary AAAA queries, out-of-bailiwick DNS server names, the use of CNAMEs and short TTLs.

DNSSEC [11] is a recent important change in DNS. Since the deployment is still underway, this paper does not analyze the issue of DNSSEC deployment. Thus observation of DNSSEC deployment remains an important issue for the near future.

References

[1] Mockapetris, P.: Domain names — Concepts and facilities, RFC 1034 (Standard) (1987).

[2] Mockapetris, P.: Domain names — Implementation and specification, RFC 1035 (Standard) (1987).

[3] Jung, J., Sit, E., Balakrishnan, H. and Morris, R.: DNS Performance and the Effectiveness of Caching, *IEEE/ACM Trans. Networking*, Vol.10, No.5, pp.589–603 (2002).

[4] Larson, M. and Barber, P.: Observed DNS Resolution Misbehavior, RFC 4697 (Best Current Practice) (2006).

[5] Liu, Z., Huffaker, B., Fomenkov, M., Brownlee, N. and Claffy, K.: Two days in the life of the DNS anycast root servers, *Passive and Active Network Measurement*, pp.125–134 (2007).

[6] Zdrnja, B., Brownlee, N. and Wessels, D.: Passive monitoring of DNS

anomalies, *Detection of Intrusions and Malware, and Vulnerability Assessment*, pp.129–139 (2007).

[7] Castro, S., Zhang, M., John, W., Wessels, D. and Claffy, K.: Understanding and preparing for DNS evolution, *Traffic Monitoring and Analysis*, pp.1–16 (2010).

[8] Iinou, H., Zushi, M., Nishida, H. and Sato, K.: An analysis of DNS queries sent from hosts to caching servers, *DNS OARC* (2010).

[9] Koc, Y., Jamakovic, A. and Hijsen, B.: A Global Reference Model of the DNS, *DNS EASY 2011 Workshop* (2011).

[10] Fujiwara, K., Sato, A. and Yoshida, K.: DNS traffic analysis — Issues of IPv6 and CDN, *IEEE/IPSJ 12th International Symposium on Applications and the Internet*, pp.129–137 (2012).

[11] Arends, R., Austein, R., Larson, M., Massey, D. and Rose, S.: DNS Security Introduction and Requirements, RFC 4033 (Proposed Standard) (2005).

Kazunori Fujiwara received his M.E. degree from Waseda University in 1991. He worked as a research associate at Waseda University. Since 2002, he has been a researcher at Japan Registry Services Co., Ltd. Since 2010, he has been a graduate student at the University of Tsukuba. His research interest are DNS and other Internet protocols. He is a member of IPSJ and IEICE.

Akira Sato received his Ph.D. from University of Tsukuba in 1998. He is a lecturer in the Department of Computer Science at University of Tsukuba. His current research interest is an operation of networks. He is a member of IPSJ.

Kenichi Yoshida received his Ph.D. from Osaka University in 1992. In 1980, he joined Hitachi Ltd., and is working for University of Tsukuba from 2002. His current research interest includes application of Internet and application of machine learning techniques.