University of Paderborn
Computer Networks Group

Report on Locality in DNS Requests – Evaluation and Impact on Future Internet Architectures

Christian Dannewitz, Holger Karl, Aditya Yadav

cdannewitz@upb.de, hkarl@upb.de, y.aditya@iitg.ac.in

July 2012

Technical Report TR-RI-12-323

Technical report

Abstract

To develop a feasible future Internet architecture, it is important to base the architecture on realistic assumptions that should be based on recent measurements if possible. In this paper, we focus on the locality of user requests, i.e., on the portion of requests that users pose for local content, which we call *neighborhood effect*. We argue and have shown in our related research [1] that the neighborhood effect can have a strong influence on the scalability of a network architecture, on latency, overall network traffic, and on inter-domain traffic. Although there are plenty of traffic measurements available, we are not aware of any recent measurement that investigates the neighborhood effect. To evaluate this effect in more detail, we have performed Domain Name System (DNS) measurements in two different DNS zones at the University of Paderborn for almost four months in total, comprising over 2.5 billion DNS requests. We evaluate the magnitude and characteristics of the neighborhood effect, the influence of requests originating from user devices and servers, and the sub-zone relationship between the two observed DNS zones. Our results show a strong neighborhood effect with 71% (university-wide) and 40% (computer science department) requests for local hosts, respectively. Other work indicates that similar results can also be expected in networks by other institutions, companies, and ISPs. As a consequence, we argue that this effect can have a significant impact on future Internet architectures in general and on information-centric networking (ICN) in particular, especially for name resolution, name-based routing, and caching.

1 Introduction

Future Internet architectures are an important research area. Basing a new architecture on realistic assumptions, including user request patterns that are validated via (recent) measurements, is important to ensure an architecture's feasibility.

In this paper, we are interested in evaluating the locality pattern of content requests, i.e., the relationship between the (network) location of the requester and the "location" of the requested content. We define "local content" as content that has a close semantic relationship to the requester's network, e.g., a company's web page or intranet content. In this paper, we identify this relationship via the content's host name, i.e., a close relationship exists if the content is named using a local domain name like *www.uni-paderborn.de/someContent*. This definition makes no statement about *where* the content is hosted. In today's Internet architecture, "local content" is in fact not necessarily hosted locally. In the course of this paper, we argue that hosting local content locally has several benefits. This does not preclude content being hosted in multiple networks nor does it imply that this content is then always local in all networks.

We call the interest of users in local content *neighborhood effect*. Note that the neighborhood effect differs from the effect that homogeneous user groups tend to share a common interest in *similar* content as this lacks the locality aspect. We see an important influence of the neighborhood effect on several aspects of network architectures, especially on *name resolution*, *name-based routing*, and *caching*. A large neighborhood effect has the potential to increase an architecture's scalability and reduce latency, overall network traffic, and costly inter-domain traffic. This positive influence stems from the potential to keep a significant amount of data traffic and/or resolution requests within the local network vicinity.

Our main focus is on future information-centric networking (ICN) architectures that put the information at the center of the architecture and shift the communication model from the node-centric paradigm that focuses on conversations to the information-centric paradigm that focuses on information dissemination. Here, we expect a strong influence of the neighborhood effect as all three components – name resolution, name-based routing, and caching – play a major role in ICN. We have evaluated the influence of the neighborhood effect on ICN name resolution in more detail in reference [1].

We evaluate the neighborhood effect based on Domain Name System (DNS) request patterns. Several DNS evaluations exist already. Most evaluations focus either on the DNS top level or on the popularity distribution of the requested hosts, e.g., to evaluation cachability. However, we are not aware of any recent DNS evaluation that focuses on the locality of requests, i.e., on the neighborhood effect. Hence, in this paper, we evaluate this effect in detail.

This evaluation might appear trivial at first glance and a strong neighborhood effect might be expected based on gut feeling. However, due to the lack of quantitative results, we believe that this kind of evaluation is essential as a basis for future ICN

architectures.

Our DNS evaluation is based on two independent measurements that we have performed at the internal DNS servers of the University of Paderborn during 2010 and 2011, covering almost four months of DNS traffic in sum and containing more than 2.5 billion DNS requests. Each measurement covers a distinct set of DNS servers: the DNS servers responsible for the main university domains and the separate DNS servers of the computer science subdomain.

We focus on the following research questions: What are the characteristics and the magnitude of the neighborhood effect? What is the influence of requests originating directly from user devices and originating from servers (e.g., mail server, web server) on the overall request patterns? What is the influence of the sub-zone relationship between the two evaluated DNS zones?

In Section 2, we discuss the measurement setup. In Section 3, we evaluate the measurement results for both DNS zones. Section 4 discusses related work before we discuss the consequences of our results for future Internet architectures with a focus on ICN architectures in Section 5.

2 Measurement Setup

The first data set, called *university* (*Uni*) DNS zone, includes all authoritative DNS servers for our university's principal DNS zone, mainly including the domains *uni-paderborn.de* and its alias *upb.de* as well as some department-specific second-level domains. The data set contains the full DNS traffic of an 11 weeks period between December 2009 and February 2010, including more than 2.5 billion DNS requests.

The second data set, the *computer science* (called *IRB*) DNS zone, has been captured at the authoritative DNS servers of the computer science department, mainly consisting of the subdomain *cs.uni-paderborn.de* and its alias *cs.upb.de*. The data set has been captured during June/July 2011 for a four weeks period, containing about 39 million DNS requests.

All data has been collected using a syslog-ng logging server and some custom python scripts to perform data anonymization prior to logging to protect the users' privacy. Figure 1 illustrates the logging results using an example of the computer science log file. The third and fourth columns describe the requester. All requests that originate from a *computer science department-internal server* are marked `irb_intern` and its IP address is shown in plain text in the fourth column. All requests originating from *user devices* (also simply referred to as *users* in the following) *within the university network* are marked `upb_intern` and requests from *outside the university network* are marked `extern`. In both cases, the requester's IP address is removed and substituted with a random number for data privacy. To further protect the users' privacy, these client numbers are reset every 24 hours, i.e., a requester gets a new random number every 24 hours. This allows us to identify short term patterns, e.g., redundant/duplicated requests by the same client, while preventing us from identifying long term personal

request patterns[1].

Columns five and six identify the requested host name. Similar to above, hosts with an IP address internal to the computer science department are marked `irb_intern`, requests for university-internal domains `upb_intern`, and all other host names `extern`. All host names are hashed and stored in column six. Again, the hashing is performed for privacy reasons, yet it still allows us to extract valuable information concerning request patterns. Finally, columns 7–9 contain more details about the type of the query, e.g., address record lookup (`A`), reverse lookup (`PTR`), or service lookup (`SRV`).

```
---Date---------Time------Req.Type--Req.no./IP----HostType-Hash---Info--
30-May-2011 18:05:41.023; irb_int 131.234.24.147; irb_int 40... IN PTR +
30-May-2011 18:05:41.149; upb_int 74070213;       irb_int de... IN SRV +
30-May-2011 18:05:41.300; upb_int 41683357;       upb_int 2b... IN A   +
30-May-2011 18:05:51.328; irb_int 131.234.24.148; irb_int 40... IN A   +
30-May-2011 18:05:51.872; upb_int 63098679;       extern  86... IN A   +
30-May-2011 18:05:49.630; extern  96311156;       upb_int 40... IN A   -
```

Figure 1: DNS log example: computer science DNS zone (full hash values elided in figure for space reasons).

3 Data evaluation

3.1 Data Preprocessing

To gain a better insight into the request patterns, we have eliminated side effects in the logging data as much as possible as described subsequently.

Our main interest with this evaluation is to analyze the neighborhood effect of DNS requests. Hence, we are interested in the locality properties of requests by requesters *within the university network*. Therefore, we have filtered requests from clients external to the university network. As we are interested in the general request patterns during regular operations, we have also filtered requests that resulted from irregular situations, e.g., generated by a temporary university-internal Planetlab experiment that generated many requests. Likewise, we have filtered requests from a few obviously misconfigured clients posing requests with duplicated domain names like `hostname.upb.de.upb.de`. We have also filtered redundant requests resulting, e.g., from clients first requesting the IPv6 address and subsequently requesting the IPv4 address for the same host name within 1 s as this represents a temporary special situation due to the IPv4–IPv6 transition.

Finally, we have eliminated reverse lookups as our focus is on the host-name to IP-address lookups. These are the lookups that correspond to, e.g., object name resolution in an ICN.

[1]The measures to protect the users' privacy have been performed in coordination with the local data protection officer.

We evaluate the filtered DNS requests separately in the following sections. All other figures exclude the aforementioned DNS requests unless explicitly stated. As we are not interested in fluctuations over the course of the day in this evaluation, all values are averaged over 24 hours.

In Section 3.2, we first evaluate the data of the university-wide DNS servers. Subsequently, Section 3.3 evaluates the results of the computer science DNS servers.

3.2 University DNS Zone

This section is subdivided into the evaluation of the overall DNS requests (Section 3.2.1), the analysis of requests by university-internal servers (Section 3.2.2) such as web servers, mail servers, etc., the analysis of requests originating from user devices (Section 3.2.3), an analysis of the influence of our data preprocessing on these results (Section 3.2.4), and a summary of the results (Section 3.2.5).

3.2.1 All Requests (Filtered)

Figure 2 shows all DNS requests (excluding the filtered requests as previously described) received by the authoritative DNS servers for the university DNS zone, separated into requests for *internal domains* (i.e., university-internal hosts) and *external domains* (i.e., university-external hosts). Figure 2 reveals that the university DNS servers receive significantly more requests for internal host names (10312 req./min) than for external host names (4227 req./min). Hence, approximately 71% of the overall requests are for internal host names.

Next, we analyse the ratio between *user requests* (i.e., requests originating directly from user devices) and *server requests*. Figure 3 shows that the overall requests are dominated by server requests for internal hosts (8819 req./min), followed by user requests for external hosts (2410 req./min), server requests for external hosts (1817 req./min), and user requests for internal hosts (1493 req./min). In sum, server requests dominate the overall DNS requests (61%). In the following two sections, we evaluate server requests and user requests separately in more detail.

3.2.2 Server Requests

Let us start by evaluating the server requests as servers generate the majority of requests. Figure 3 already illustrated that there are significantly more server requests for internal hosts than for external hosts. Next, we are interested in the distribution of generated requests among different server types. The next two figures show the server DNS requests separated by server type for external (Figure 4) and internal (Figure 5) host names.

Both figures reveal that the university mail servers generate the vast majority of server requests, both for external (1811 req./min) and internal (8069 req./min) host names with approximately 82% for internal hosts. No other servers generate any significant amount of DNS requests.

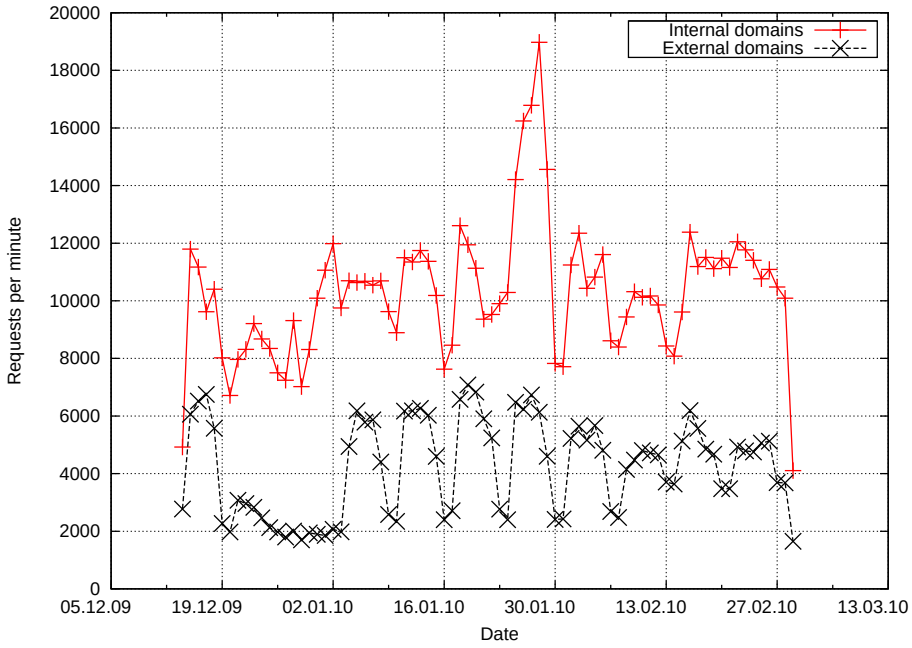

Figure 2: Uni: All requests, *filtered*.

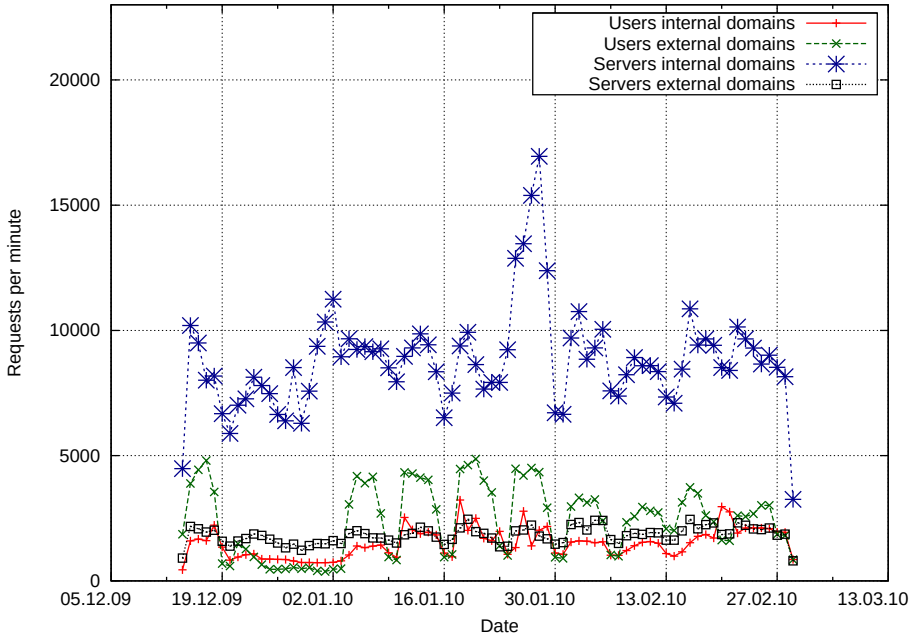

Figure 3: Uni: DNS requests by user terminals and servers.

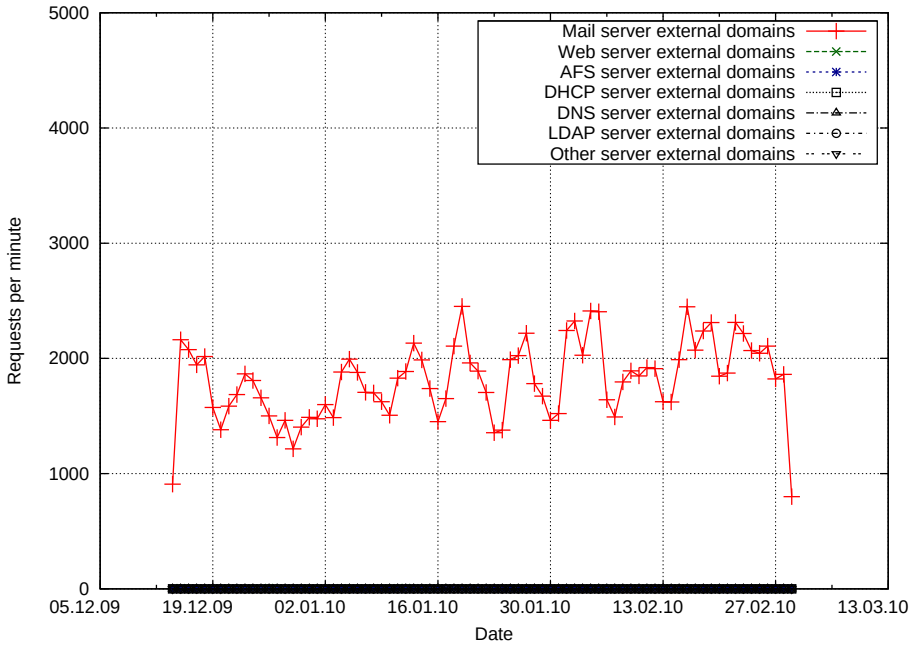

Figure 4: Uni: DNS requests by servers for external host names.

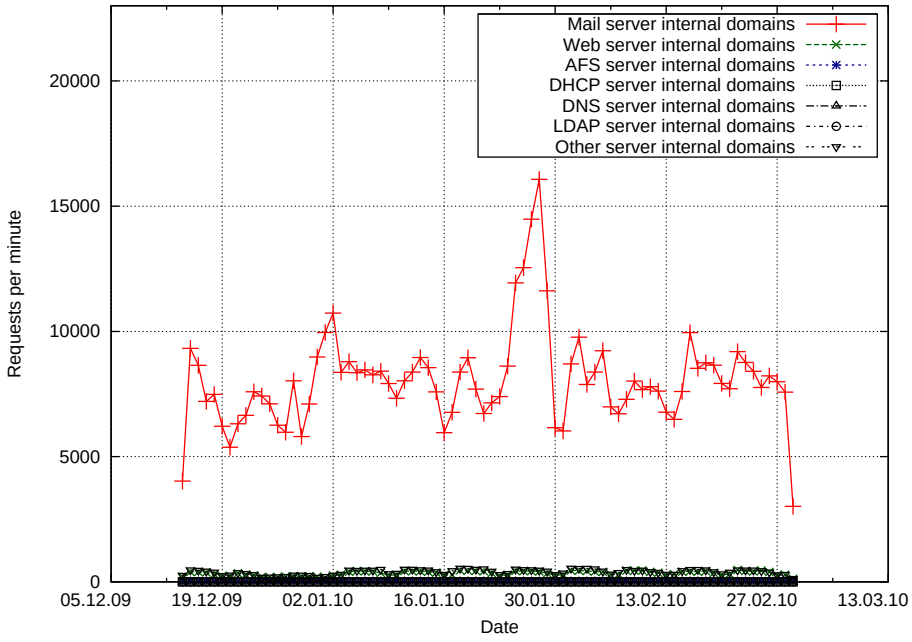

Figure 5: Uni: DNS requests by servers for internal host names.

Our investigation revealed that the major portion of mail server DNS requests is caused by the specific configuration of the mail servers. The mail servers are configured to start a new process for each incoming connection. This process first loads its configuration file that contains several host names that each have to be resolved, causing multiple DNS requests for each connection. These requests are mainly for internal hosts as the configuration file contains information such as the internal Lightweight Directory Access Protocol (LDAP) server, etc. This effect illustrates that the DNS traffic can highly depend on specific configuration settings and the choice of additional services. DNS traffic analysis by Zdrnja et al. [2] have shown related effects. They analyzed DNS responses with a focus on security aspects like botnets and spam. As a result, they identified that a large amount of their DNS responses was caused by the university-internal anti-spam software. For each received email, the anti-spam software queries several real-time black lists (using address record DNS requests) in order to determine if an email sender is likely a spammer.

Another part of the mail server requests is generated by the mail transfer agent (more precisely, the client part of the Simple Mail Transfer Protocol (SMTP)) that checks DNS mail exchanger (MX) records to figure out the destination mail server. The emails producing these requests are generated exclusively by university users (as the SMTP servers require authentication). Note that while the majority of university mail users resides within the university network, some users also use the SMTP servers from outside the university network.

3.2.3 User Requests

Next, we have a look at the requests generated by user devices. Figure 6 shows all user requests, again separated into requests for internal (1493 req./min) and external (2410 req./min) host names. Approximately 38% of the requests are for internal host names.

Figure 6 shows an interesting weekly pattern: The overall requests are significantly higher during weekdays as would be expected. However, it is interesting to note that the number of requests for external domains is much higher than for internal domains during *weekdays*, resulting in a ratio of roughly 1:2 between requests for internal and external host names. During *weekends*, both types of requests are roughly equivalent. Note that the weekly pattern does not exist between December 23rd and January 3rd. This is the Christmas holiday season where the request pattern roughly equals the pattern during weekends.

3.2.4 Influence of Filtering

Next, we analyze if the data preprocessing as described previously has some major influence on our results. Figure 7 shows the same graphs as Figure 2; however, Figure 7 includes all requests by university-internal requesters without filtering, i.e., duplicated requests, requests from misconfigured clients, reverse lookups, etc. are included (but not requests by requesters outside the university). There are more requests for both

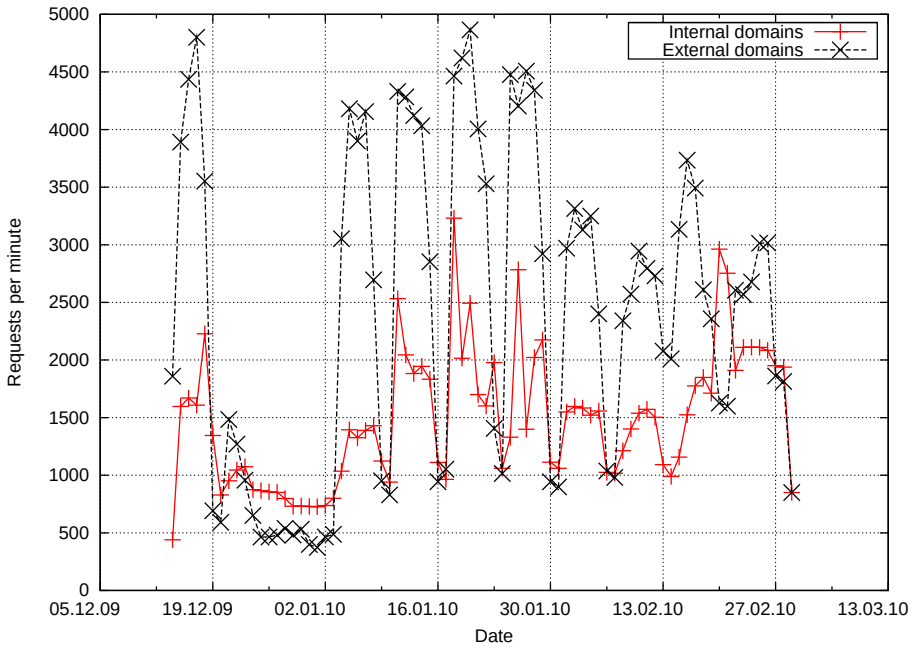

Figure 6: Uni: DNS requests by user devices.

internal (13546 req./min) and external (6594 req./min) hosts. However, the overall structure as well as the ratio between internal and external hosts remains roughly the same with 67% requests for internal hosts.

Figure 8 shows only reverse lookup requests, illustrating the impact of reverse lookups on the overall internal-to-external ratio. As can be seen, requests for internal hosts (3214 req./min) also dominate the overall requests (4540 req./min) with 71% for internal hosts. Hence, including reverse lookups would not change the overall ratio between requests for internal and external host names. Adding the number of reverse lookups for internal hosts (3214 req./min) to the number of requests for internal hosts in the *filtered* Figure 7 (10312 req./min) adds up to 13526 req./min which almost equals the number of *unfiltered* requests for internal hosts in Figure 7 (13546 req./min). Hence, almost all requests for internal hosts eliminated by filtering are reverse lookups. The requests for external hosts eliminated by filtering are also dominated by reverse lookups but include some additional effects as described in Section 3.1.

Figure 9 shows all reverse lookups separated by requester type. Most reverse lookups are generated by user devices, followed by mail server reverse lookups. All other servers do not generate a significant amount of reverse lookups.

Finally, Figure 10 shows all requests (also excluding reverse lookups, erroneous requests, etc.) by requesters outside the university network. They contribute only 11% of the overall requests. Their requests for university-internal hosts constitute 44% compared to requests for external hosts (56%).

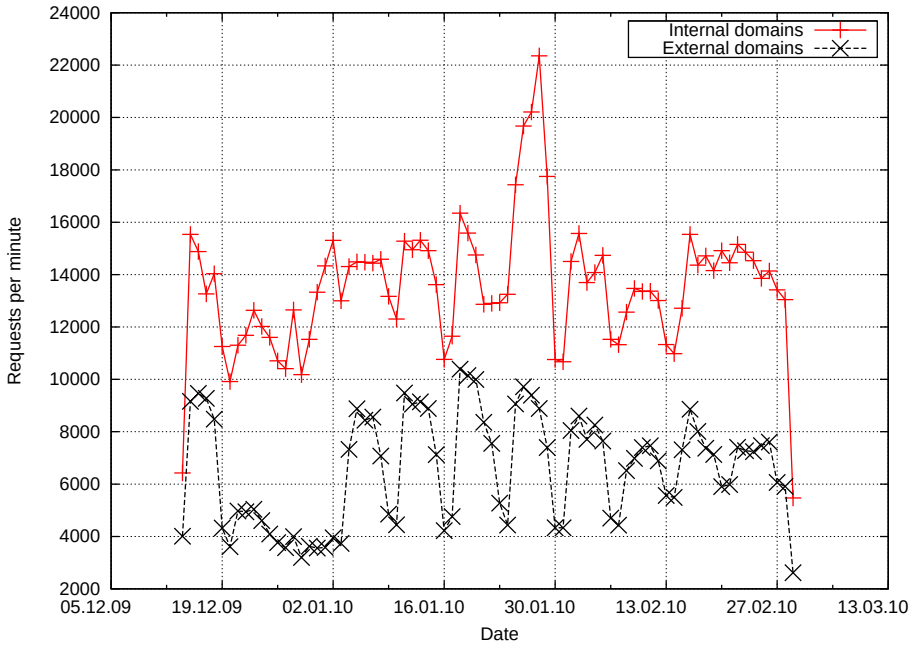

Figure 7: Uni: All requests, *unfiltered*.

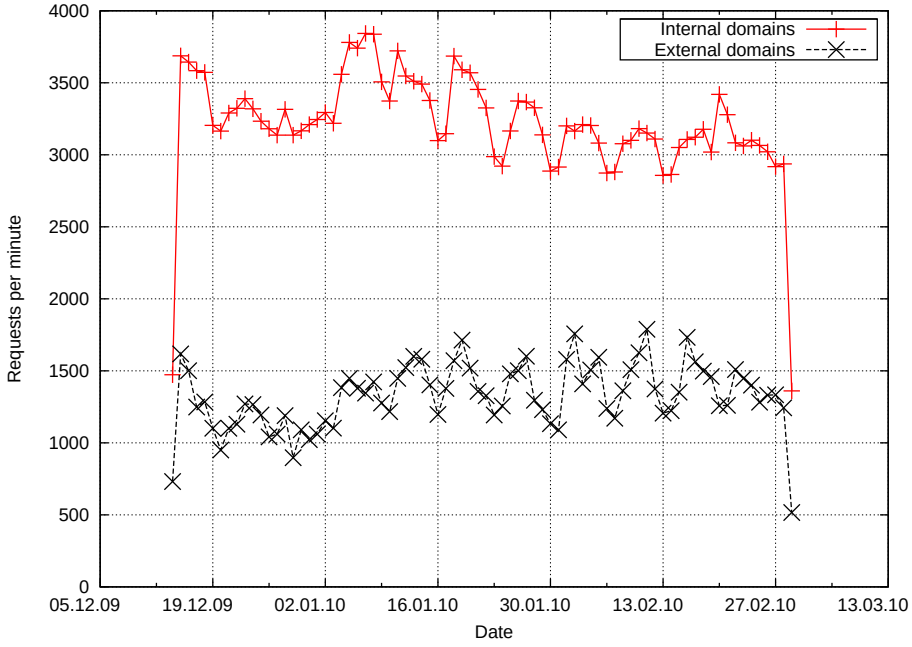

Figure 8: Uni: Only reverse lookup requests.

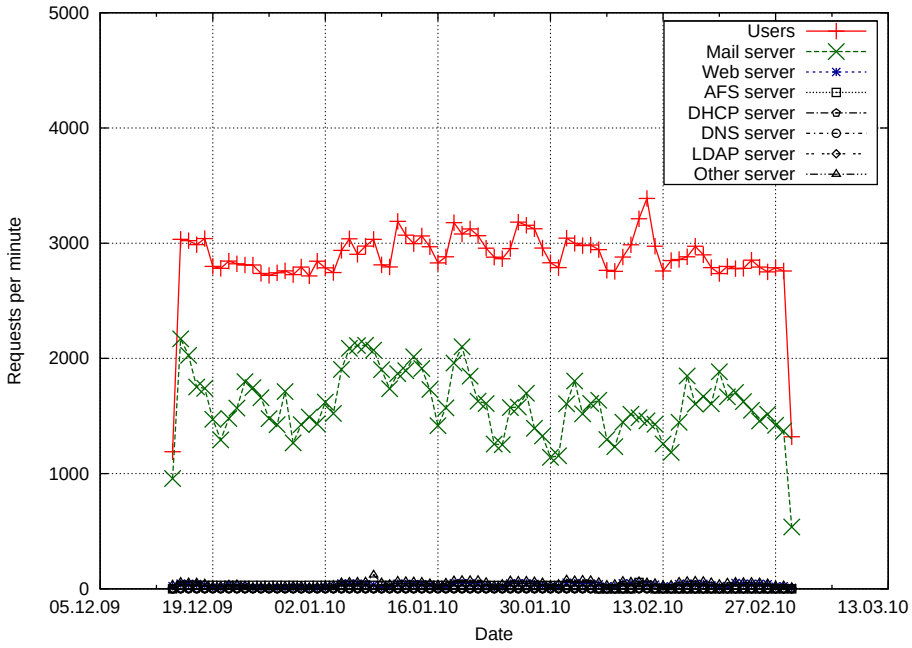

Figure 9: Uni: Reverse lookups by requester type.

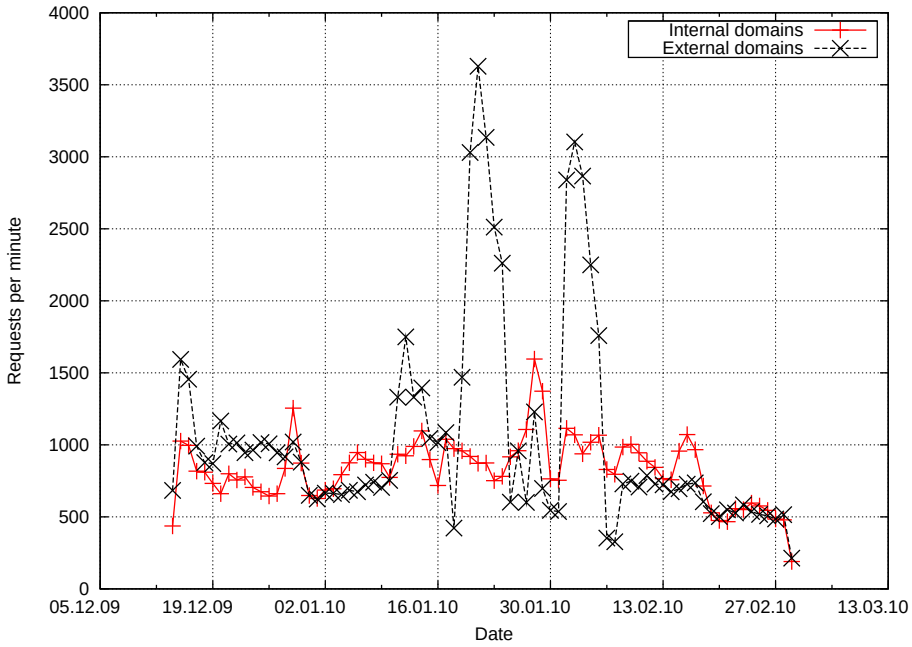

Figure 10: Uni: DNS requests by university-external requesters.

3.2.5 Summary of University-wide Results

In summary, we can conclude that about 70% of the overall DNS requests at the university level are for internal host names. The number of requests by servers outweighs the number of client requests with the mail servers generating most of the server requests. For both the user and server requests, the number of requests for internal host names is significant with approximately 40% for user devices and even 80% for servers.

Reverse lookups dominate the overall requests that we filtered out. We filtered reverse lookups as they are not relevant for our focus, a future ICN architecture. In any case, the reverse lookups are also dominated by lookups for internal hosts. Hence, including reverse lookups in our results would not change the characteristics of the results (Figure 7 and 8).

Client requests show a distinct weekly request pattern with significantly more overall requests during weekdays. Not surprisingly, a similar but less pronounced pattern can be observed in the server requests. More interestingly, the *type* of the user-requested content seems to change between weekdays and weekends as observed in the user's request patterns.

3.3 Computer Science Department DNS Zone

In the following, we evaluate the DNS requests within the computer science department (*IRB*). We take the sub-zone relationship between university-wide DNS servers and IRB department into account by separating the DNS requests into three subgroups: *IRB-internal domains* (all requests for hosts of the computer science sub-zone), *UPB-internal domains* (all requests for hosts of the university-wide zone, *excluding* the IRB sub-zone), and *external domains* (all requests for hosts outside the university zone).

3.3.1 All Requests (Filtered)

Figure 11 shows all DNS requests received at the authoritative IRB DNS servers (filtered as described in Section 3.1). In total, the IRB DNS servers get much fewer requests compared to the university-wide DNS servers. This is not surprising as the computer science department is just a single sub-zone of the overall university and many IRB users use the university-wide DNS servers instead of the department-internal DNS servers. Figure 11 shows that approximately 48% of the requests at the IRB DNS servers are for university-external hosts (337 req./min), 40% for IRB-internal hosts (280 req./min), and 12% for university-internal hosts (86 req./min).

3.3.2 Influence of Filtering

Figure 12 shows the same graphs as Figure 11. However, it contains all additional requests that are filtered in Figure 11, but still excludes requests by university-external requesters. The unfiltered results are very similar to the filtered results. The main

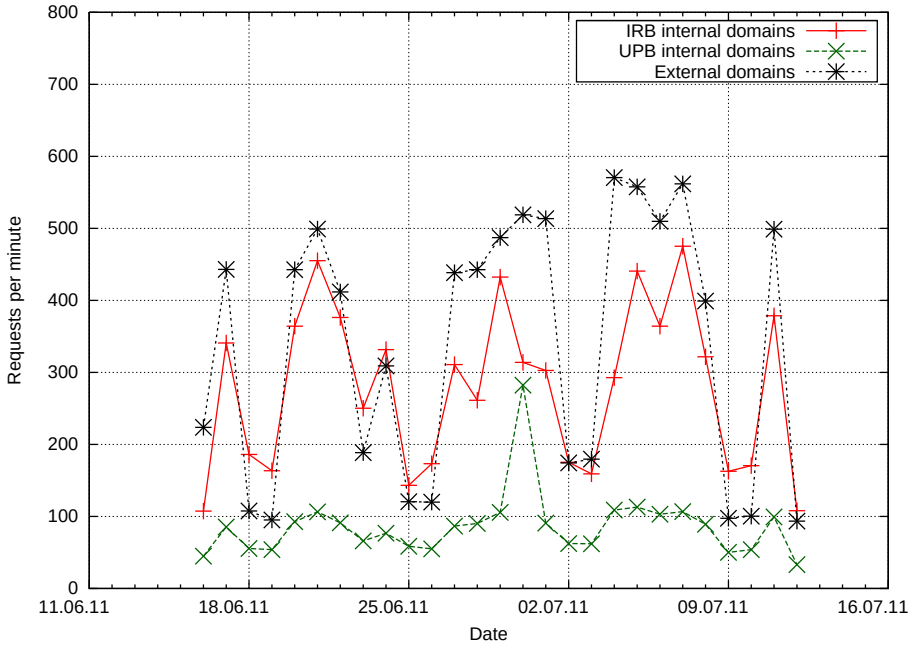

Figure 11: IRB: All DNS requests, *filtered*.

difference is an increase in the number of university-internal requests by 192 req./min, which results primarily from reverse lookups (168 req./min) as shown in Figure 13.

3.3.3 Client and Server Requests

Figure 14 shows all requests by user devices. These are dominated by requests for external hosts (57%). Adding up the IRB-internal requests (32%) and university-internal requests (11%) results in 43% requests for hosts within the overall university network.

Figure 15 shows the same for IRB server requests. The overall number of server requests is much smaller compared to user requests (18% of the overall requests) and is strongly dominated by requests for IRB-internal hosts (75%). The server requests for university-internal and IRB-internal hosts are mainly generated by the IRB LDAP servers as can be seen in Figure 16. The file servers are generating all requests for external hosts (11 req./min).

3.3.4 Summary of Computer Science Zone Results

In summary, the computer science DNS servers show similar results compared to the university-wide results with a large number of DNS requests for internal hosts (40%). However, this number is smaller than the university-wide 71% for internal hosts. The total number of requests is dominated by requests from user devices (82%).

Our evaluation of the sub-zone relationship between the university-wide zone and

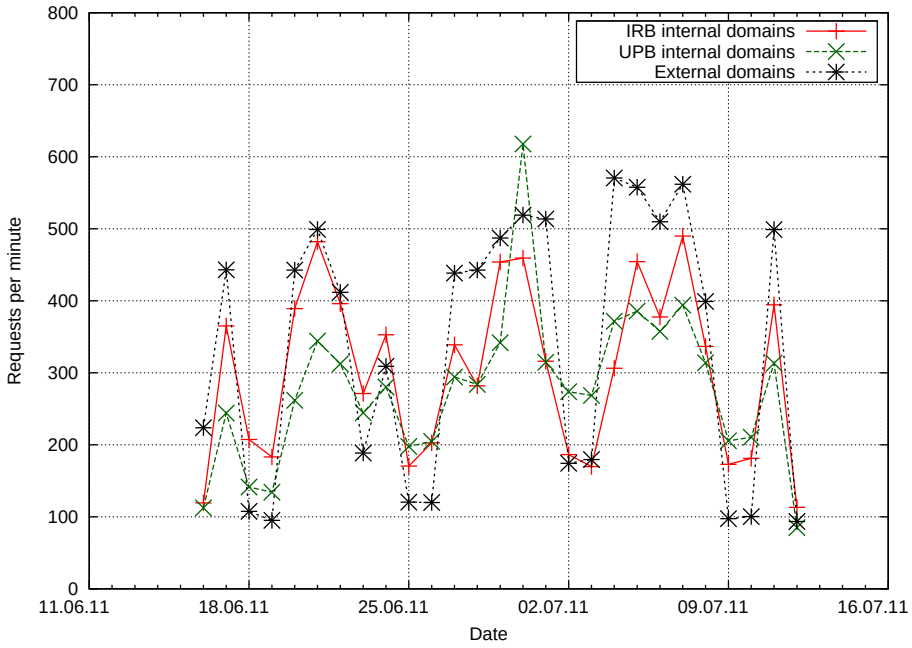

Figure 12: IRB: All DNS requests, *unfiltered*.

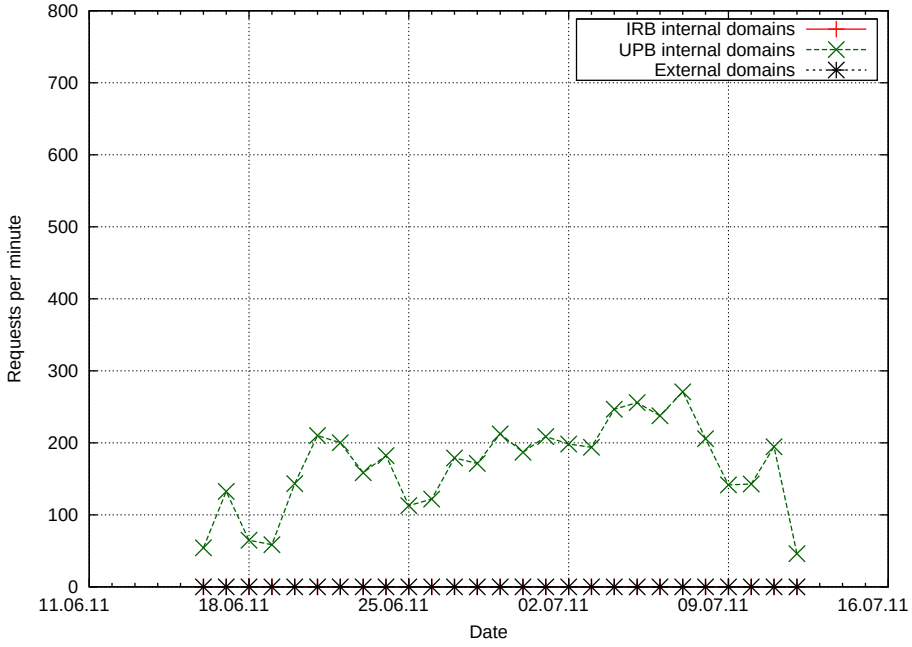

Figure 13: IRB: Reverse lookups.

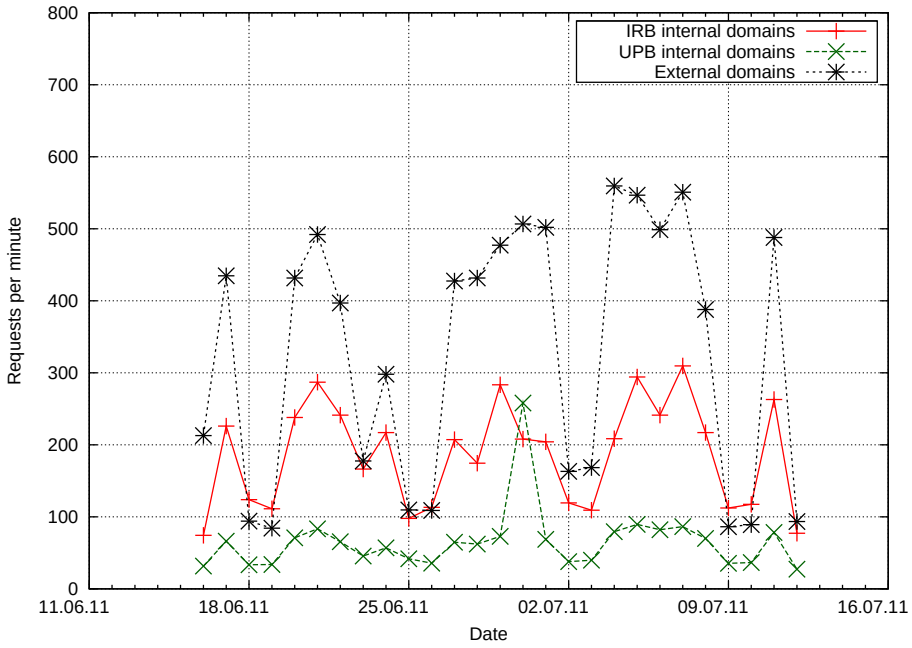

Figure 14: IRB: DNS requests by user devices.

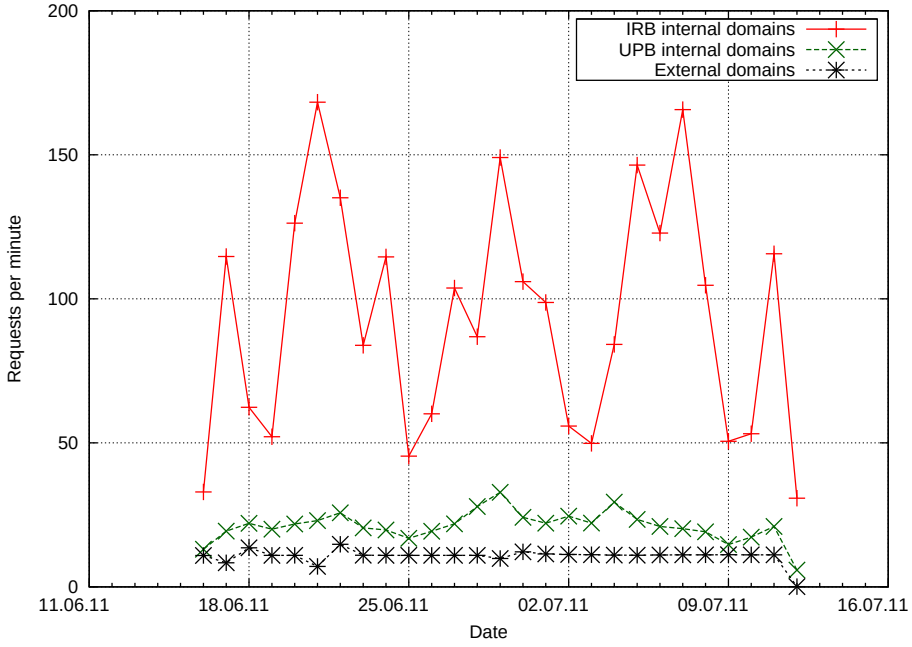

Figure 15: IRB: DNS requests by IRB servers.

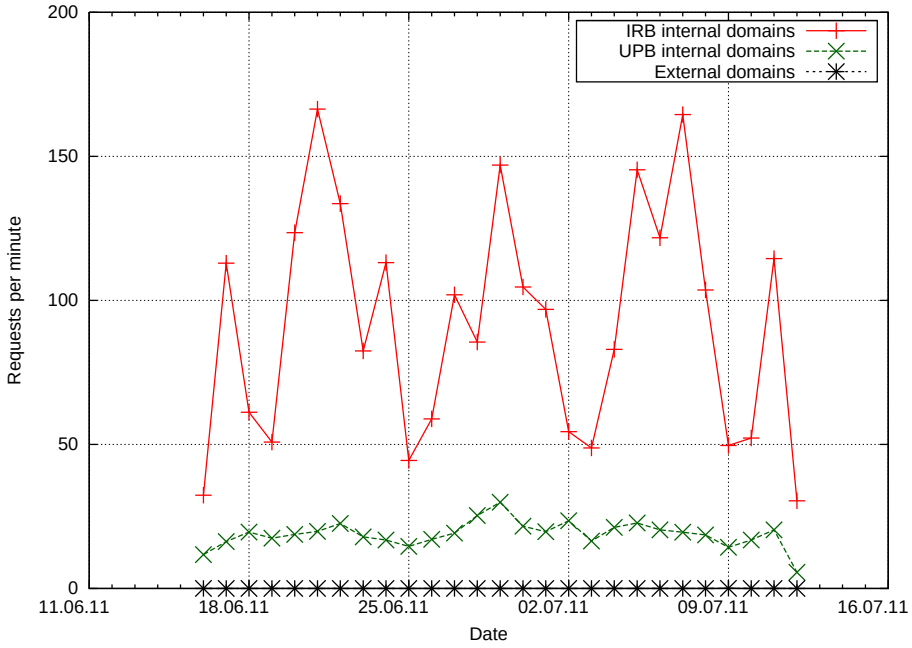

Figure 16: IRB: DNS requests by IRB LDAP servers.

the computer science zone revealed that 12% of the DNS requests at the computer science DNS servers are for hosts within the higher-level zone (i.e., the university-wide zone). Hence, a total of 52% of the requests are for hosts within the overall university zone. This can be relevant from a network perspective as both zones belong to the same physical campus network.

4 Related Work

Multiple research activities have performed DNS traffic measurements and have focused on different aspects. Several have focused on DNS measurements at the DNS root servers. For example, the Réseaux IP Européens Network Coordination Centre (RIPE NCC) DNS Monitoring Services (DNSMON) provides current monitoring data about several DNS root servers via test requests sent to these servers. The focus is on responsiveness and request latency. Likewise, Castro et al. have evaluated several aspects of the DNS root servers, including traffic growth and usage patterns [3] and general workload at the root servers [4], both based on data collected by the Cooperative Association for Internet Data Analysis (CAIDA). However, traffic analysis at the DNS root servers does not provide us with information about request locality.

To analyze request locality, measurements at lower DNS levels are required. Such measurements have been performed, e.g., by Jung et al. [5] and by Ager et al. [6], however, without analyzing the request locality. Jung et al. have analyzed the performance and prevalence of failures and the effectiveness of DNS caching. Ager et al.

have focused on data cacheability for the protocols HTTP, BitTorrent, eDonkey, and Network News Transfer Protocol (NNTP). In both cases, the similarity and popularity distribution of DNS requests is of main importance. Similar evaluations have been performed in other areas like peer-to-peer (P2P) networking with the goal to reduce inter-domain traffic by exploiting the request popularity distribution for caching [7]. Although the request popularity distribution and the locality of DNS requests can be related, they are two separate things. For example, although all users of an organization might exclusively query the same 100 hosts, these hosts might all be internal or external hosts or a mixture of both. The latter is the main aspect that we focus on.

We are not aware of any DNS analysis that mainly focuses on request locality as we do. However, a study by McDaniel and Jamin [8] that aimed at developing a secure public-key distribution system has generated some interesting results about DNS request locality as a side effect. To gather test data for their architecture, McDaniel and Jamin have performed DNS measurements at five different networks of which four include information about request locality. Although their focus is on other aspects and, therefore, their results do not provide more details about the neighborhood effects, the results do support our main findings in terms of general request locality. Like our results, their results indicate that a large percentage of DNS requests is for local hosts. This effect does not seem to be limited to organizations like universities but is also visible in other networks like the AT&T network. Specifically, they found the following locality results: Their AT&T trace contains about 50% requests for local hosts. The University of Michigan (UoM) trace contains about 65% local requests, the College of Engineering (subdomain of UoM with separate DNS servers) contains 38% local requests, and the Electrical Engineering and Computer Science Department (subdomain of UoM with separate DNS servers) contains 43% local requests.

Likewise, also as a side effect, a study by Ager et al. [9] that focuses on the identification and classification of content hosting and delivery infrastructures generated some interesting results concerning the neighborhood effect at the continent level. Specifically, they performed world-wide DNS measurements for the top 2000 most popular host names according to Alexa[2] and evaluated the percentage of requests that could be answered from within the same continent (including cached content) that the request originated from. Their results show that, e.g., 58.2% of all requests from North America could be served from within North America. Of course, this high number is partly due to the fact that most of the popular content is hosted in North America. For a comparison, "only" 10.1% of requests originating from South America could be served from within South America. However, these numbers are more relevant when compared to the average percent of *world-wide* requests served from a certain continent. For example, on average 50% of all world-wide requests are served from North America whereas only 2.4% are served from South America. Consequently, subtracting the world-wide average share of answered requests from the share of requests originating and served from within the same country shows higher interest in locally available

[2] http://alexa.com/topsites/

content than average, i.e., some neighborhood effect at the continental level. In North America, the interest for content served from within the continent is 8.2 percent points higher that average, for South America, it is 7.7 percent points higher. The study also investigates requests from all other continents with comparable results. All continents (except Africa) show some kind of neighborhood effect. Please note that these results are not quantitatively comparable to the results of our evaluation in this paper as the study by Ager et al. includes locally cached content, e.g., via Content Distribution Networks (CDNs). However, the results indicate that the neighborhood effect also exists at higher levels like the continent level.

5 Discussion and Conclusion

Our DNS traffic analysis has shown a significant percentage of requests for internal hosts. At the university-wide DNS servers, 71% of the requests are for local hosts. At the computer science department, 41% of the requests are for department-internal hosts and 52% for university-internal hosts. These requests contain requests generated by user devices and by servers. When removing the server requests, which are only indirectly generated by users, we still can observe approximately 42% requests for university-internal hosts directly originating from user devices.

We strongly assume that similar neighborhood effects can be observed in other institutional, company, and Internet service provider (ISP) networks. These assumptions are reinforced by results in related work (Section 4). We also assume that similar effects can be observed at higher levels, e.g., at a country-wide level thanks to cultural, linguistic, and political effects. In general, this effect varies depending on the specific context. For example, some ISPs in developing countries might not provide much local content to their customers, resulting in a low neighborhood effect. In contrast, ISPs in countries with strong cultural, linguistic, or political boundaries might observe a much higher neighborhood effect. It is interesting to note that these locality patterns seem to persist for at least 13 years as evidenced by McDaniel and Jamin [8] although the Internet usage patterns have changed dramatically during that time.

In today's Internet, the neighborhood effect has a positive influence on DNS. It reduces the average resolution latency as resolution can often be performed at the local DNS servers. This effect also helps the DNS scalability as most requests do not reach the DNS top level.

In this paper, we are particularly interested in the consequences of the neighborhood effect on future Internet architectures, especially on ICNs. We note that DNS requests are for hosts whereas ICN requests are for named data objects. However, every request for a data object in the ICN context would translate into a request for a host in the DNS context. Hence, the results are also applicable to the ICN context. Our performed data filtering is tailored for the ICN context. Specifically, we have filtered reverse lookups as it is currently unclear if they will have an equivalent in the ICN context. Also, it remains to be seen how servers will work in an ICN context. We assume, however,

that an information-centric server will produce equivalent name resolution requests in an ICN context as in today's DNS context.

We see three main areas in future Internet architectures and ICN architectures specifically where the neighborhood effect will likely have a strong positive influence: *Name resolution*, *name-based routing*, and *caching*.

Name resolution: Some ICN architectures rely on a (globally) scalable name resolution service (NRS) (e.g. Network of Information (NetInf) [10]). Scalability is extremely important because the number of data objects that will be registered in an ICN NRS is expected to be much larger that today's number of DNS domains. In the following, we propose some design principles that help exploiting the neighborhood effect to improve the scalability of the NRS and decrease the network load and latency.

Hierarchical structure: A hierarchical structure with multiple levels mapping the underlying local zones helps to make use of the neighborhood effect at multiple levels.

Registration scheme: Local data should be registered in the local resolution domain (i.e. local NRS) to benefit from the high local popularity of the local content. To make content available for a wider audience and exploit locality at higher levels, also register the content in the higher resolution domains.

Resolution scheme: In accordance with the registration scheme, users should first query the local resolution domain before iteratively querying higher-level domains. Thereby, they will retrieve the closest available copy, reducing latency, global network traffic, and costs for inter-domain traffic. This scheme also prevents resolution requests from propagating farther than necessary. Consequently, most requests will be answered at lower levels, reducing the load at the critical global level, hence, improving scalability.

We have developed an NRS for ICN networks that follows there design principles and illustrates the highlighted advantages [1]. In reference [1], we also present quantitative analysis and simulation results of the neighborhood effect influence on NRS latency and load distribution among NRS nodes.

Name-based routing: Several ICN architectures like Content-Centric Networking (CCN) [11] rely on a name-based routing scheme. To exploit the neighborhood effect with name-based routing, the routing scheme has to ensure that traffic for local content stays within the local network. Given this requirement, a name-based routing scheme will benefit from the neighborhood effect via reduced global network traffic and reduced costs for inter-domain traffic.

Caching: Thanks to the high popularity of local content, caching of local content will be efficient, i.e., high cache hit rates can be expected, which can be used to reduce the load on the original server. In addition, we expect that the extensive use of caching in ICN architectures will even further increase the magnitude of the neighborhood effect compared to today's host-centric Internet. Thanks to in-network caches placed in local provider networks, copies of popular, non-local content will be cached in the local provider networks. As these locally cached copies will also be registered at the local NRS in architectures like NetInf, requests for these cached copies will further increase the share of requests for locally registered content. To reduce the *data traffic* in all preceding cases, it is required to host a copy of the local content in the local network,

either the original source or a cached copy.

This evaluation can just be a first step to evaluate the neighborhood effect. Although related work seems to confirm our results, more evaluation results are needed to validate these results. For example, the neighborhood effect should be validated at different levels, e.g., at the regional, country, and continent level, and in other types of networks, including enterprise networks and ISP networks. Also, the influence of mobile users on the neighborhood effect, i.e., the intensity of the neighborhood effect in mobile carrier networks seems interesting. Finally, the results could be validated and detailed via web traffic analysis that takes the size of requested objects into account. Web traffic analysis would also take additional client requests into account that might be missing in the DNS request patterns due to DNS lookup result caching at the client side.

To conclude, we believe that the neighborhood effect as evaluated in this paper can have a significant positive influence on future Internet architecture in general and ICN architectures particularly, increasing scalability of the ICN architectures, reducing latency, and reducing overall network traffic.

6 Acknowledgments

The authors would like to thank Michael Schwarz for his help in performing the DNS measurements and the SAIL[3] EU project colleagues for many fruitful discussions.

[3]http://www.sail-project.eu/

References

[1] Matteo D'Ambrosio, Christian Dannewitz, Holger Karl, and Vinicio Vercellone, "MDHT: A hierarchical name resolution service for information-centric networks," in *Proc. ACM SIGCOMM Workshop on Information-centric Networking*, New York, NY, USA, August 2011, pp. 7–12, ACM.

[2] B. Zdrnja, N. Brownlee, and D. Wessels, "Passive monitoring of DNS anomalies," in *Proc. Detection of Intrusions, Malware, and Vulnerability Assessment (DIMVA)*, Lucerne, Switzerland, July 2007, pp. 129–139.

[3] S. Castro, M. Zhang, W. John, D. Wessels, and k. claffy, "Understanding and preparing for DNS evolution ," in *Traffic Monitoring and Analysis Workshop (TMA)*, Zurich, Switzerland, April 2010, pp. 1–6.

[4] S. Castro, D. Wessels, M. Fomenkov, and k. claffy, "A Day at the Root of the Internet," *ACM SIGCOMM Computer Communication Review (CCR)*, , no. 5, pp. 41–46, Oct 2008.

[5] Jaeyeon Jung, Emil Sit, Hari Balakrishnan, and Robert Morris, "DNS performance and the effectiveness of caching," in *Proc. 1st ACM SIGCOMM Workshop on Internet Measurement (IMW)*, New York, NY, USA, 2001, pp. 153–167, ACM Press.

[6] Bernhard Ager, Fabian Schneider, Juhoon Kim, and Anja Feldmann, "Revisiting cacheability in times of user generated content," in *Proc. INFOCOM IEEE Conference on Computer Communications Workshops*, New York, NY, USA, March 2010, pp. 1–6, IEEE.

[7] Krishna P. Gummadi, Richard J. Dunn, Stefan Saroiu, Steven D. Gribble, Henry M. Levy, and John Zahorjan, "Measurement, modeling, and analysis of a peer-to-peer file-sharing workload," *ACM SIGOPS Operating Systems Review*, vol. 37, no. 5, pp. 314–329, December 2003.

[8] P. McDaniel and S. Jamin, "A scalable key distribution hierarchy," Tech. Rep., University of Michigan. Department of Electrical Engineering and Computer Science, 1998.

[9] Bernhard Ager, Wolfgang Mühlbauer, Georgios Smaragdakis, and Steve Uhlig, "Web content cartography," in *Internet Measurement Conference (IMC '11)*. November 2011, ACM.

[10] Christian Dannewitz, "NetInf: An information-centric design for the future Internet," in *Proc. 3rd GI/ITG KuVS Workshop on The Future Internet*, Munich, Germany, May 2009.

[11] Van Jacobson, Diana K. Smetters, James D. Thornton, Michael Plass, Nick Briggs, and Rebecca Braynard, "Networking named content," *Comm. ACM*, vol. 55, pp. 117–124, Jan. 2012.