

Improving Content Delivery with PaDIS

Ingmar Poesse
T-Labs/TU Berlin
ingmar@net.t-labs.tu-berlin.de

Benjamin Frank
T-Labs/TU Berlin
bfrank@net.t-labs.tu-berlin.de

Bernhard Ager
T-Labs/TU Berlin
bernhard@net.t-labs.tu-berlin.de

Georgios Smaragdakis
T-Labs/TU Berlin
georgios@net.t-labs.tu-berlin.de

Steve Uhlig
T-Labs/TU Berlin
steve@net.t-labs.tu-berlin.de

Anja Feldmann
T-Labs/TU Berlin
anja@net.t-labs.tu-berlin.de

Abstract

Today, a large fraction of Internet traffic is originated by Content Delivery Networks (CDNs). To cope with the increasing demand for content CDNs, deploy massively distributed infrastructures. Moreover, to minimize their cost, content delivery networks perform their own traffic optimization by assigning end-users to their servers. Such an assignment is at large unaware of the network conditions and based on inaccurate information on the location of the end-user. Thus, users are not always assigned to the CDN servers that provide optimal end-user performance. To improve user assignment especially from a performance perspective we propose and deploy a Provider-aided Distance Information System (PaDIS). PaDIS is a novel system that allows ISPs to utilize their unique knowledge about the network conditions and user locations to better assign users to servers. Our field test results show that significant improvements in download time up to a factor of four for content offered by popular CDNs can be achieved when utilizing PaDIS.

Categories and Subject Descriptors

C.2.4 [Distributed Systems]: Client/Server; C.2.5 [Local and Wide-Area Networks]: Internet

General Terms

Network Architecture, Measurement, Performance

Keywords

Content Distribution, Server Selection, DNS

1. INTRODUCTION

The Internet has evolved into a system where users are able to generate and share large amounts of content with friends and/or other users via applications such as online social networks, video portals, One-click Hosters, Web services, wikis, blogs, or P2P file-sharing applications. Multimedia content, including photos, music, and videos, as well as software downloads and updates, are together responsible for most of the Internet traffic. Recent studies report that HTTP is used to access this information and it accounts for

more than 50 % of the traffic [8, 10, 14, 4]. Among the major causes for the current prevalence of HTTP traffic, we find the increase of streaming content, e.g., offered by `youtube.com`, as well as the popularity of the content offered by One-Click Hosters (OCHs) [2] such as `rapidshare.com`. This popular content is hosted by the new “Hyper Giants” [8] which include large content providers (CPs), such as Google and Yahoo!, as well as Content Distribution Networks (CDNs), such as Akamai and Limelight [6]. To keep the terminology simple, we refer to different types of players in the content delivery landscape, e.g., CPs, CDNs and OCHs, simply as CDNs.

To achieve high levels of performance and scalability, CDNs rely on distributed infrastructures. Some of them even have deployed servers deep inside ISPs in more than 5000 locations throughout the Internet [9]. Others rely on placing a large number of servers in data-centers across strategic locations in the Internet which provide good connectivity to major ISPs [9, 7].

Today, CDNs have full control over the server selection mechanism to direct end-users. Through this, they are able to optimize their traffic flows, minimize their operational costs and bring content closer to the end-user. Two techniques to direct end-users to servers are used by CDNs. These are DNS-based or work by HTTP redirection. Since HTTP redirection yields a higher overhead and is more intrusive into the architecture of the application than DNS-based schemes, the latter is more widely used. The general architecture of the DNS-based approach is presented in Figure 1.

From the viewpoint of the end-users and ISPs, the redirection schemes employed by existing CDNs have three major limitations:

Network Bottlenecks.

Despite the traffic flow optimization performed by CDNs, the assignment of end-user requests to servers by CDNs may still result in sub-optimal content delivery performance for the end-users. This is a consequence of the limited information CDNs have about the network conditions between the end-user and their servers. Tracking the ever changing conditions in networks, i. e., through active measurements and end-user reports, incurs an overhead for the CDN without

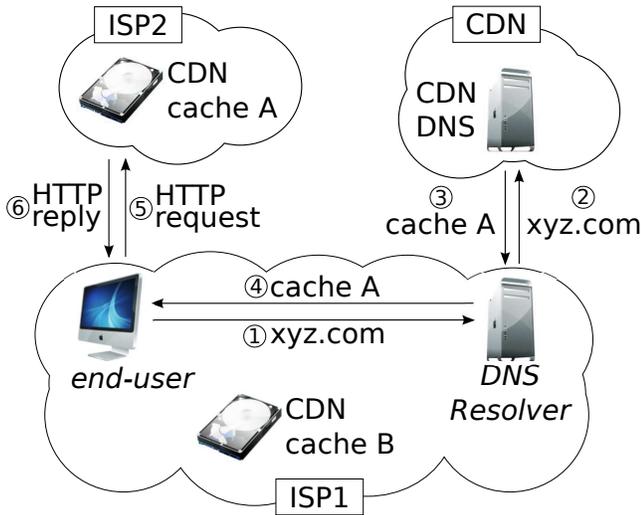


Figure 1: A DNS-based server selection mechanism assigns end-users to servers whenever a request for content hosted by the CDN is submitted. To retrieve the content from the CDN, the end-user first resolves the hostname associated with the content. Once the hostname is resolved to an IP address, a HTTP connection to the supplied IP address is opened and the content is retrieved.

In detail, the following steps are performed:

- (1) The end-user issues a DNS request to its configured DNS resolver for the hostname associated with the content it wishes to retrieve.
- (2) The DNS resolver asks the CDN DNS server for an IP addresses resolving to the requested hostname.
- (3) After the CDN has selected the servers based on the IP address of the DNS resolver, it returns them to the DNS resolver.
- (4) The DNS resolver forwards the reply to the end-user.
- (5,6) With the IP address the end-user retrieves the content via HTTP.

a guarantee of performance improvements for the end-user. Without sufficient information about the network paths between the CDN servers and the end-user, any assignment performed by the CDN may lead to additional load on existing network bottlenecks, or to the creation of new bottlenecks.

User Mis-location.

DNS requests received by the CDN DNS servers originate from the DNS resolver of the end-user, not from the end-user itself. The assignment is therefore based on the assumption that end-users are close to their DNS resolvers. Recent studies have shown that in many cases this assumption does not hold [11, 1]. As a result, the end-user is mis-located and the server assignment is not optimal. As a response to this issue, DNS extensions have been proposed to include the end-user IP information [5].

Content Delivery Cost.

Finally, CDNs strive to minimize the overall cost of delivering huge amounts of content to end-users. To that end, their assignment strategy is mainly driven by economic aspects. While a CDN will always try to assign users in such a way that the server can deliver reasonable performance, this can again result in end-users not being directed to the server able to deliver best performance.

To overcome the limitations of current CDNs, we design and evaluate a Provider-aided Distance Information System (PaDIS). PaDIS is a novel system operated by an ISP that allows ISPs to improve end-user experience by utilizing information about the network bottlenecks and end-user location. We believe that a system like PaDIS fills a gap in the content delivery landscape, especially by taking into account ISP constraints and end-user performance. Although PaDIS is at the moment deployed as a platform for ISPs to improve content delivery, on the long-term it is meant as a generic platform to help both ISPs and CDNs to deliver content to end-users. Therefore, we see PaDIS as an opportunity to think globally about content delivery, by including the network and its end-users in the picture.

In the remainder of the paper we describe the architecture and deployment of PaDIS in Section 2. In Section 3, we quantify the content delivery performance while using PaDIS in the wild. We summarize the paper and discuss the outlook of PaDIS in Section 4.

2. PaDIS

Today’s content delivery landscape is mostly unaware of the information ISPs have about dynamic network conditions and end-user location in the network. Through a Provider-aided Distance Information System (PaDIS), we propose to let the ISP influence the server selection by extending the ISP DNS infrastructure. To improve end-user performance, PaDIS leverages the server diversity of CDNs available through the multiple locations from which a given content can be obtained as well as network information only available to the ISP.

We show that there is significant server diversity in the currently deployed CDNs [13]. We find that more than 70 % of the HTTP traffic in a large European ISP can be obtained from at least three different network locations. We also find that only seven CDNs are responsible for more than 50 % of the HTTP traffic. Moreover, large-scale studies show that popular CDNs allow end-users to download the content they host from any server [15]. To leverage the available server diversity, PaDIS uses network information only available to the ISP to give recommendations on which server is best chosen from a network perspective. Our evaluation of PaDIS provides evidence that the knowledge of the ISP can improve the CDN server assignment process by avoiding the current limitations of DNS-based server selection.

PaDIS complements today’s content delivery landscape

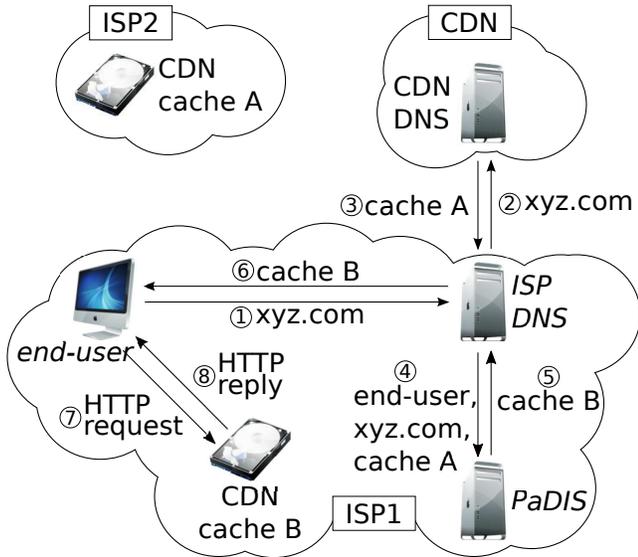


Figure 2: The incremental PaDIS deployment adds two additional steps to the DNS resolution process: the DNS infrastructure of the ISP forwards the authoritative answer to the PaDIS server (4), which incorporates the answer into the server diversity, ranks all available servers based on the network information and then returns the best server to the DNS resolver of the end-host (5) which in turn forwards the answer to the end-host.

by adding an ISP optimization component. The main tasks of a PaDIS server are:

1. Discover server diversity.
2. Maintain an up-to-date annotated map of the ISP network.
3. Rank lists of available servers (discovered in 1) based on the network map (available in 2).

PaDIS acts as an enabler for an ISP to take advantage of the server diversity of deployed CDNs to improve end-user experience.

2.1 Deployment

Deploying PaDIS inside an ISP is a two-step incremental process. First, the DNS infrastructure needs to be modified to be able to communicate with PaDIS. As ISPs often operate multiple DNS resolvers this has to be done step by step to keep interruptions to a minimum. Furthermore, leveraging the DNS process requires the end-users to utilize the ISP DNS resolver instead of a third party DNS resolver, e. g., OpenDNS or Google DNS. To that end, we verify in a large European ISP that more than 97 % of the customers use the DNS resolver supplied by the ISP [13].

Second, PaDIS can be configured to influence the server selection of all CDNs or only the server selection of some of them, e. g., the most popular. Given that CDNs are responsi-

ble to deliver different types of content, e. g., real-time content, websites and/or bulk data, the administrator of PaDIS can set diverse performance metrics for different CDNs.

In Figure 2, we present a possible PaDIS deployment, co-located with the ISP DNS infrastructure. Three aspects are important for a successful PaDIS deployment: transparency, latency and server aggregation. It is challenging to obtain the cooperation from the end-user as well as from the CDN. Unless such cooperation is available, the PaDIS deployment should be completely transparent to both. To achieve this, PaDIS intercepts and modifies part of the DNS messages involved in the server selection, without requiring neither the user nor the CDN to change their current operation. Moreover, the communication delay between ISP DNS resolver and PaDIS should be minimized to not harm end-user performance. The best solution is to physically co-locate PaDIS with the DNS resolvers. This co-location also allows PaDIS to discover the largest possible server diversity of a CDN as unveiled from the authoritative DNS answers. The larger the population of users that utilize the ISP DNS resolver the higher is the chance to discover a large set of candidate servers even in small time scales.

In order to scale PaDIS with the growing requirements of ISPs the architecture of our PaDIS prototype is designed, as we show in the next section, to keep up with the request rate of an ISP DNS resolver [13]. Moreover, the increase of the overall DNS resolution time is negligible [13]. Also, every PaDIS server has its own network feeds, enabling it to work independent of other PaDIS instances, and thus removing any overhead caused by synchronization.

2.2 Architecture

To achieve the tasks of PaDIS we propose an architecture that comprises of a *Diversity Discovery* component, a *Network Monitoring* component and a *Query Processing* engine. In Figure 3 we provide the architecture overview of PaDIS.

Diversity Discovery.

Improving the assignment of end-users to servers requires knowledge about the location diversity of content servers. However, CDNs do not provide a list of the the content servers they operate and their location. The *Diversity Discovery* component extracts CDN server diversity from DNS replies observed by the ISP DNS resolver and applies aggregation rules defined by the administrator of PaDIS. The Aggregation rules filter and build lists of IP addresses of servers able to satisfy a user request for content. With this, they enhance the choice offered to the query processing beyond the scope of the individual DNS reply.

The *Diversity Discovery* component also acts as a protocol converter providing an interface between the ISP DNS resolver and the other components of PaDIS. It speaks the standard DNS protocol with the ISP DNS resolver, reducing the needed changes in ISP DNS resolver.

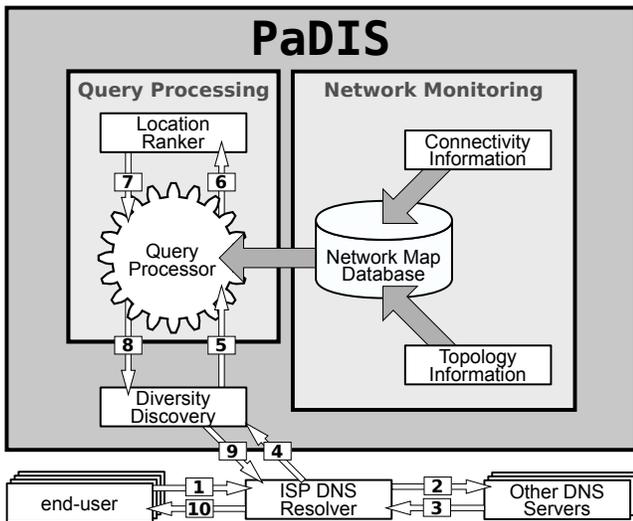


Figure 3: A DNS request from an end-user is processed by the PaDIS enhanced DNS resolution as follows:

- (1) The end-user sends a DNS request for CDNized content that is optimized by PaDIS.
- (2) The ISP DNS resolver forwards the request to the appropriate authoritative DNS servers if the answer is not in the cache.
- (3) The ISP DNS resolver receives an answer of the query, either from the authoritative DNS server or from its own cache.
- (4) The DNS answer is handed to the Diversity Discovery of PaDIS instead of being directly forwarded to the end-user.
- (5) The Diversity Discovery enhances the DNS answer with additional sources capable of satisfying the request from previous, similar DNS answers that have already been received. It then sends the complete list to the Query Processor.
- (6) The Query processor augments each source-destination pair from the received list with information from the Network Monitoring and hands each pair separately to the Location Ranker.
- (7) The Location Ranker returns the preference value for the specific source-destination pair.
- (8) Once all pairs have been given a preference, the list of sources is sorted by the assigned preferences and is sent back to the Diversity Discovery.
- (9) The Diversity Discovery selects the top-ranked sources (servers), generates a DNS reply and sends it back to the ISP DNS resolver.
- (10) The modified answer is handed back to the end-user.

Network Monitoring.

ISPs have very detailed and up-to-date information about their own network infrastructure. Nonetheless, finding out the overall network status involves several systems each one

responsible to monitor a part of the network status. The *Network Monitoring* component gathers information about the state of the network from several sources and maintains an up-to-date view of the network. It also provides an interface for network status queries. The Network Monitoring component consists of three sub-components:

- The *Topology Information* component is responsible to gather detailed information about the network topology as well as link utilization, router load and topological changes.
- The *Connectivity Information* component uses routing information from within the network and from other ISP (i. e., through analyzing BGP messages) to allow a mapping from the topology to paths in the network.
- The *Network Map Database* component processes the information of the other two components to allow fast access to network information. It also abstracts the topology from setup specific setting into an annotated graph representation and pre-caches network paths along with their properties.

Having ISP-centric information ready for fast access in a database ensures timely response if a decision based on the state of the network is required.

Query Processing.

The *Query Processing* component combines the information about the server diversity as well as the up-to-date network status to improve content delivery. Its function is to calculate the preference of a given set of candidate sources (servers) to reach a destination (end-user) while taking into account the network conditions between the sources and the destination. It comprises of two sub-components. The *Query Processor* and the *Location Ranker*.

- The *Query Processor* augments each source-destination pair in the query with network conditions. Then, it hands each pair individually to the *Location Ranker* to get its preference value. Finally, the list of source-destinations pairs is sorted by its preference values and handed back to the Diversity Discovery.
- The *Location Ranker* receives exactly one source-destination pair along with the network conditions between them. It applies a specific metric to be optimized, i. e., delay or link utilization, to calculate a preference value.

PaDIS is not only a platform for content delivery optimization. The modular architecture of PaDIS can be adapted to arbitrary protocols and network environments. For example, PaDIS currently gathers its information about server diversity from DNS replies. Future versions of the diversity discovery can also obtain it directly from the CDN or a third party mapping system like ALTO [3]. Furthermore, PaDIS currently aims at optimizing the HTTP traffic from

CDNs. However, the actual HTTP connection is influenced indirectly by the preceding server selection process through DNS. Thus, PaDIS leverages the decoupling of the server selection and the content transfer. Thus, it improves content delivery independent of the protocol used for content transfer. Moreover, PaDIS can be an enabler for collaboration between different parties involved in the content delivery, namely CDNs or other ISPs, to jointly improve their operation and end-user experience. It is also worth noticing that PaDIS does not unveil any sensitive information about the ISP operation to other parties.

3. EXPERIENCE WITH PADIS

To quantify the effect of PaDIS, we consider two different CDNs. Our choice is driven by the large traffic volume they carry in the major European ISP we consider. With the help of anonymized packet level traces from the large European ISP, we select a large CDN, responsible for more than 20 % of the HTTP traffic in the studied ISP, with highly distributed caches that typically deliver small to average size files. The second is a OCH, responsible for more than 15% of the HTTP traffic in the studied ISP, relying on a multi-homed data-center to deliver large size files to end-users. Once the CDNs were chosen, we deployed vantage points at residential locations with DSL connectivity in the large European ISP that operates PaDIS and performed extensive active measurements to evaluate PaDIS in the wild. The results presented here are representative of a much larger set of experiments. More details can be found in [13].

For the CDN, we utilize the Diversity Discovery component of PaDIS to extract the server diversity. The aggregated rule we are using is based on the DNS redirection signature of the CDN. We were able to identify more than 3500 unique servers operated by the large CDN spread in over 124 different network locations. A large fraction of these caches are located within the studied ISP. Next, we randomly selected one cache from each location in order to run our experiments and confirmed that all the chosen caches all serve the content we are downloading for our measurement. Note that most CDNs server all content from all their caches [15]. As most of the files delivered by this CDN have small to average size, the download performance is dominated by the end-to-end delay [12]. Thus, we use end-to-end delay as the metric to rank caches in PaDIS. During the experiment, we downloaded several files from all 124 chosen servers, as well as from the server chosen by the CDN and the server chosen by PaDIS. Figure 4 (top) shows the download time of an average size object. We also plot the median download time across all the 124 pre-selected caches. The reduction in download time was up to a factor of four when using PaDIS. When considering larger files, e. g., more than 10 Mbytes, the improvement in download time is smaller as the download performance is restricted by the network bandwidth on the bottleneck link and the end-to-end delay becomes less significant.

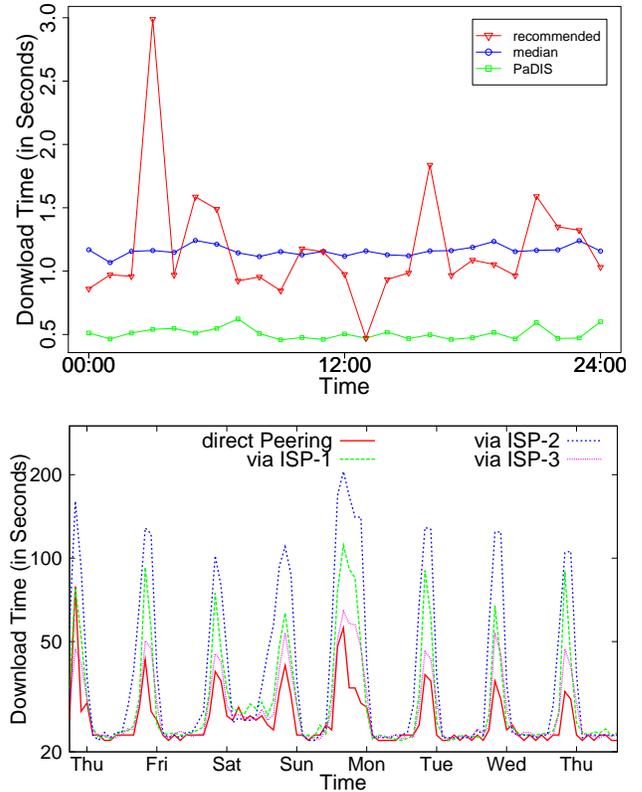


Figure 4:

Top: Downloading a 510 Kbytes file distributed by a CDN:

- (recommended) download time when the cache recommendation from the CDN is used,**
- (median) the download time of all available CDN caches as discovered by PaDIS,**
- (PaDIS) download time when using PaDIS.**

Bottom: Downloading a 50 Mbytes file from a OCH: PaDIS utilizes the direct peering with OCH improving the download time by a factor of four during the peak hour.

We now turn our attention to the OCH. Using the same approach as with the CDN, we were able to uncover that the OCH servers are located in a single data-center. The OCH is well connected to four large ISPs, including the large European ISP that operates PaDIS. A closer look at the server selection strategy of the OCH reveals that it assigns 60% of the requests originated by the end-users of the studied ISP to servers behind the direct peering with it regardless of the time of day or the day of week. The remaining requests are randomly assigned to the servers behind the peerings with the other three ISPs. Knowing this, we set the ranking function of PaDIS such that OCH servers that are behind the direct peering with the large European ISP are ranked higher than the others. This way, PaDIS always prefers the direct

peering. We performed an experiment where we repeatedly downloaded content served using the different peerings of the OCH. In Figure 4 (bottom) we plot the download time of a 50 Mbytes file during the period of one week. The improvement of the download time was up to a factor of four during the peak hour when the network resources are limited.

4. SUMMARY AND OUTLOOK

Content delivery networks serve a major portion of today's Internet traffic. To cope with the large traffic volume they deploy large distributed infrastructures. Moreover, to minimize their costs, CDNs perform their own traffic optimization based on DNS redirection. CDNs can, to some extent, consider the user's performance within their optimization criteria. However, given the limited amount of information they have about ISP networks and the location of network bottlenecks, the ability of CDNs to consider ISP constraints is limited. This constrains the CDNs ability to optimally assign end-users to servers and thus offer the best performance to them.

To improve end-user performance, we propose a solution where the ISP operates a Provider-aided Distance Information System (PaDIS). PaDIS uses information that is only available to the ISP, in order to rank any user-server pair based on network conditions and user location. As a result, PaDIS utilizes the deployed CDNs infrastructure to improve end-user performance. We evaluate PaDIS with different providers and show that improvements in download times of up to a factor of four are possible.

The expected gains from deploying PaDIS depend on several aspects, such as the number of locations from which a given content is available, the properties of the paths available between the servers and the end-users, as well as the type and volume of the requested content. CDNs continue to expand their infrastructures to keep up with the increasing demand of traffic. Thus, the gain of deploying PaDIS is expected to grow in the future as it can take advantage of the increased server diversity exposed by the CDNs.

PaDIS leverages the decoupling of the server selection and the content transfer, thus it is not restricted to DNS-based server selection and "CDNized" traffic, but can be utilized by any protocol. As part of our future work we would like to investigate how PaDIS can be utilized to improve the content delivery performance of applications that do not depend on DNS as well as evaluating PaDIS in other ISPs.

Bios of authors

Ingmar Poesse is a Ph.D. candidate at Deutsche Telekom Laboratories and the Technische Universität Berlin. He received his Diplom in Computer Science from Technische Universität Berlin in 2009. His research interest includes network architectures, network measurements and content distribution.

Benjamin Frank is a Ph. D. candidate at Deutsche Telekom Laboratories and the Technische Universität Berlin. He received his M.Sc. in Computer Science from Technical University of Munich in 2009. His research interest includes the measurement, analysis and optimization of content distribution systems and architectures as well as cloud systems.

Bernhard Ager is a research scientist at at Deutsche Telekom Laboratories and Technische Universität Berlin, from which he obtained a doctoral degree in 2011. His research interests comprise passive and active Internet measurement, behavioral characterization, and content distribution.

Georgios Smaragdakis is a Senior Research Scientist at Deutsche Telekom Laboratories and the Technical University of Berlin. He received the Ph.D. in Computer Science from Boston University in 2008. His research interest includes the measurement, performance evaluation and optimization of content distribution systems and overlay networks.

Steve Uhlig is a Senior Research Scientist at Deutsche Telekom Laboratories (T-labs) and Technische Universität Berlin, Germany. Prior to joining T-labs he was a faculty member at Delft University of Technology, the Netherlands, and a Post-doctoral Fellow of the Belgian National Fund for Scientific Research (F.N.R.S.), as well as a visiting scientist at Intel research Cambridge, UK, and at the University of Adelaide, Australia. He obtained the Ph. D. degree in Applied Sciences from the University of Louvain, Louvain-la-neuve, Belgium (2004). He received the Annual IBM Belgium/F.N.R.S. Computer Science Prize in 2005 for his Ph.D. thesis. His research interests focus on the large-scale behavior of the Internet, especially through network measurements, including traffic engineering, routing, topology and traffic.

Anja Feldmann is a Professor at Technische Universität Berlin and Deutsche Telekom Laboratories. She is heading the research group on Intelligent Networks and is dean of the department of electrical engineering and computer science at the Technische Universität Berlin. Before joining Technische Universität Berlin she was a Professor at Technical University of Munich and Saarland University, and a researcher at AT&T Labs. She was awarded with the Gottfried Wilhelm Leibniz Prize in 2011 and was elected to the German Academy of Sciences Leopoldina in 2010. She received the Ph.D. in computer science from Carnegie Mellon University (1995). Her research interests include Internet measurements, programmable networks and network architecture.

5. REFERENCES

- [1] B. Ager, W. Mühlbauer, G. Smaragdakis, and S. Uhlig. Comparing DNS Resolvers in the Wild. In *ACM IMC*, 2010.
- [2] D. Antoniadis, E. P. Markatos, and C. Dovrolis.

- One-click Hosting Services: a File-sharing Hideout. In *ACM IMC*, 2009.
- [3] Application-Layer Traffic Optimization (ALTO) IETF working group. <https://datatracker.ietf.org/wg/alto/charter/>.
- [4] Cisco Visual Networking Index: Forecast and Methodology, 2009–2014. http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white_paper_c11-481360.pdf, 2010.
- [5] C. Contavalli, W. van der Gaast, S. Leach, and D. Rodden. Client subnet in DNS requests. IETF draft, work in progress, draft-vandergaast-edns-client-subnet-00, 2011.
- [6] C. Huang, A. Wang, J. Li, and K. Ross. Measuring and Evaluating Large-scale CDNs. In *ACM IMC*, 2008.
- [7] R. Krishnan, H. Madhyastha, S. Srinivasan, S. Jain, A. Krishnamurthy, T. Anderson, and J. Gao. Moving Beyond End-to-end Path Information to Optimize CDN Performance. In *ACM IMC*, 2009.
- [8] C. Labovitz, S. Lekel-Johnson, D. McPherson, J. Oberheide, and F. Jahanian. Internet Inter-Domain Traffic. In *ACM SIGCOMM*, 2010.
- [9] T. Leighton. Improving Performance on the Internet. *Commun. of the ACM*, 52(2), 2009.
- [10] G. Maier, A. Feldmann, V. Paxson, and M. Allman. On Dominant Characteristics of Residential Broadband Internet Traffic. In *ACM IMC*, 2009.
- [11] Z. Mao, C. Cranor, F. Douglis, M. Rabinovich, O. Spatscheck, and J. Wang. A Precise and Efficient Evaluation of the Proximity Between Web Clients and Their Local DNS Servers. In *Usenix ATC*, 2002.
- [12] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose. Modeling TCP Reno performance: A Simple Model and its Empirical Validation. *IEEE/ACM Trans. Netw.*, 8(2), 2000.
- [13] I. Poese, B. Frank, B. Ager, G. Smaragdakis, and A. Feldmann. Improving Content Delivery using Provider-aided Distance Information. In *ACM IMC*, 2010.
- [14] Sandvine Inc. Global Internet Phenomena (2009 and 2010 reports). http://www.sandvine.com/news/global_broadband_trends.asp.
- [15] S. Triukose, Z. Al-Qudah, and M. Rabinovich. Content Delivery Networks: Protection or Threat? In *ESORICS*, 2009.