Public Review for

Pushing CDN-ISP Collaboration to the Limit

Benjamin Frank, Ingmar Poese, Yin Lin, Georgios Smaragdakis, Anja Feldmann, Bruce M. Maggs, Jannis Rake, Steve Uhlig, and Rick Weber

Content delivery has become the Internet's primary purpose and its main source of traffic. Current statistics are staggering, with Netflix alone being responsible for 30% of the peak traffic in North America. Not surprisingly, the variety of architectural models for distributing this content is rapidly expanding with approaches that involve, to different degrees, all parties in the content delivery ecosystem - CDNs, content providers, ISPs and end users.

This paper presents the design and evaluation of a system – NetPaaS – to enable the collaboration of two key stakeholders – ISPs and CDNs. NetPaaS builds on the authors' previous work on CaTE*, expanding the forms of collaboration to the placement of content servers in a network.

Reviewers have a number of comments on the paper's early draft, including the apparent simplicity of the system, the potential for information leakage (from CDN to ISP and vice-versa) and a lack of novelty when compared with the authors' own and other's related efforts. In the final version, the authors addressed most of the comments – pointing out, for instance, to the challenge of incorporating in a scalable manner all network updates and clarifying that NetPaaS assumes that the information exchange is between trusted parties that have already formed strategic alliances. Reviewers uniformly agreed that even if the ideas put forward are not particularly new, the impressive empirical evaluation of the proposed ideas, leveraging traces from the largest commercial CDN and a large tier-1 ISP, is a unique and interesting contribution in itself.

Public review written by
Fabián E. Bustamante
Northwestern University, USA

**I. Poese et al., SIGCOMM CCR, Oct. 2012.*

Pushing CDN-ISP Collaboration to the Limit

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

Ingmar Poese
TU Berlin
ingmar@net.t-labs.tu-berlin.de

Yin Lin
Duke
linyin@cs.duke.edu

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

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

Bruce M. Maggs
Duke/Akamai
bmm@cs.duke.edu

Jannis Rake
T-Labs
Jannis.Rake-Revelant@telekom.de

Steve Uhlig
Queen Mary, U. London
steve@eecs.qmul.ac.uk

Rick Weber
Akamai
riweber@akamai.com

Abstract

Today a spectrum of solutions are available for distributing content over the Internet, ranging from commercial CDNs to ISP-operated CDNs to content-provider-operated CDNs to peer-to-peer CDNs. Some deploy servers in just a few large data centers while others deploy in thousands of locations or even on millions of desktops. Recently, major CDNs have formed strategic alliances with large ISPs to provide content delivery network solutions. Such alliances show the natural evolution of content delivery today driven by the need to address scalability issues and to take advantage of new technology and business opportunities.

In this paper we revisit the design and operating space of CDN-ISP collaboration in light of recent ISP and CDN alliances. We identify two key enablers for supporting collaboration and improving content delivery performance: informed end-user to server assignment and in-network server allocation. We report on the design and evaluation of a prototype system, NetPaaS, that materializes them. Relying on traces from the largest commercial CDN and a large tier-1 ISP, we show that NetPaaS is able to increase CDN capacity on-demand, enable coordination, reduce download time, and achieve multiple traffic engineering goals leading to a win-win situation for both ISP and CDN.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design; C.2.3 [**Network Operations**]: Network Management; C.2.4 [**Distributed Systems**]: Client/Server

General Terms

Performance, Measurement.

Keywords

Content Delivery, Network Optimization, CDN-ISP Collaboration.

1. INTRODUCTION

Recently, Akamai formed content delivery strategic alliances with major ISPs, including AT&T [1], Orange [7], Swisscom [8], and KT [5]. The formation of CDN-ISP alliances is a paradigm shift in how content delivery networks will be deployed in the future and opens new directions for innovative solutions for CDN-ISP collaboration. It is also the natural evolution of innovative approaches for content delivery that have been deployed for more than a decade to address scalability, performance, and cost issues as well as to take advantage of business opportunities.

Today's Internet traffic is dominated by content distribution [13, 32, 40, 47] delivered by a variety of CDNs. Gerber and Doverspike [32] and Poese et al. [53] report that a few commercial CDNs account for more than half the traffic in a North American and a European tier-1 carrier, respectively. More than 10% of the total Internet inter-domain traffic originates from Google [40], and Akamai claims to deliver more than 20% of the total Internet Web traffic [50]. Netflix, which uses multiple CDNs, is responsible for around 30% of the traffic in North America during peak hours [33].

To cope with continuously increasing demand for content, a massively distributed infrastructure has been deployed by CDNs [43, 15]. Some CDNs as well as CDN-accelerated cloud and service providers rely on a number of datacenters in strategic locations on the Internet, e.g., Limelight is present in more than 70 locations, Google operates tens of data centers [61], Microsoft Azure uses 24 locations, and Amazon AWS relies on 6 large datacenters and operates caches in more than 22 locations. Others deploy highly distributed infrastructures in a large number of networks, e.g., Akamai operates more than $100,000$ servers in more than $1,800$ locations across nearly $1,000$ networks [50].

The existing content delivery platforms, however, do not always have servers in locations that can satisfy the growing demand and provide good performance. One reason is limited agility in server deployment, as it takes time to find the locations in the right places with the required capacities, make the necessary business arrangements, and install the servers [50]. Moreover, the content delivery market is very competitive, leading CDNs to investigate ways to reduce capital and operating costs [57].

To address these two challenges, a variety of designs have appeared over the last decade. These solutions expand the CDN footprint by dynamically deploying servers as needed or leveraging the resources of end-users. An overview of the spectrum of the various solutions and the level of involvement of content delivery stakeholders is shown in Figure 1. Commercial CDNs [11] as well as ISPs [41] operate hybrid delivery systems where end-users download content from the servers as well as other end-users to reduce the bandwidth and energy cost respectively at the server side. Commercial CDNs also license content delivery software to ISPs that maintain servers. In some cases these licensed CDNs are able to coordinate with the CDN-operated servers or with other CDNs enabling CDN federations, see e.g., the CDNI IETF group. Meta-CDNs have also been proposed to optimize for cost and performance by acting as brokers for CDN selection [45, 29]. P2P systems are also successful in utilizing the aggregate capacity of end-users that are interested in downloading the same content [24].

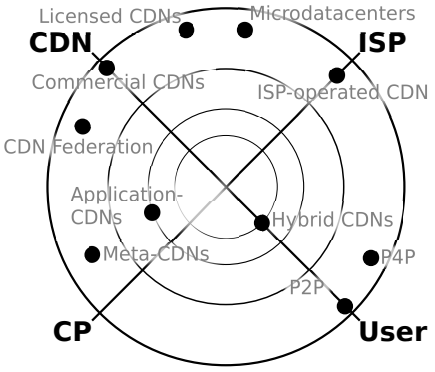


Figure 1: Spectrum of content delivery solutions and involvement of stakeholders.

P4P [66] has been proposed as an ISP-P2P collaboration mechanism to better localize traffic. Content providers (CPs) are also moving to deploy application-specific CDNs with direct peering with or inside ISPs, e.g., Netflix Open Connect for video stream delivery [6] or Google Global Cache, primarily for YouTube [4, 19]. The advantage of such specialized CDNs is that they can be optimized for the application.

Another recent trend is to marry cloud resources (processing and storage) with networking resources to meet the high performance requirements of certain applications, such as high definition video streaming or online gaming on demand [58]. Moreover, many ISPs support the migration from solutions that rely on proprietary hardware to those that rely on generic appliances and take advantage of virtualization to reduce complexity and avoid vendor lock-in [10]. Large ISPs, including AT&T, Deutsche Telekom, and Telefonica, have already deployed generic appliances in relatively small datacenters, also referred to as *microdatacenters*, co-located with their major network aggregation locations. Initially, such deployments were to support their own services such as ISP-operated CDNs, IPTV, carrier-grade NAT, deep packet inspection, etc., but they now offer full virtualization services [9]. These new capabilities allow ISPs to offer network and server resources to CDNs, applications, and services, close to their end users. Recent studies [59] also show that enterprises can outsource part of their infrastructure in the cloud and take advantage of the new virtualization market.

Economics and market share are also key drivers. Large CDNs have a strong customer base of content providers and are responsible for delivering content for their customers to end-users around the world. On the other hand, ISPs have a strong end-user base in some regions and also, as mentioned above, have invested significantly in adding infrastructure at the aggregation locations (PoPs) of their networks. The combined "ownership" of content providers and end-users is a major driving force behind recent CDN-ISP alliances [1, 7, 8, 5] as both sides strive to reduce operational cost and at the same time offer better content delivery services.

Despite the clear opportunity for collaboration, the necessary mechanisms and systems to enable joint CDN deployment and operation inside the network are not yet available. Our contributions are summarized as follows:

- We revisit the design and operating space of CDN-ISP collaboration in light of recent announced alliances and we identify two major enablers for collaboration, namely informed user-server assignment and in-network server allocation.
- We design and implement a novel prototype system, called NetPaaS (Network Platform as a Service), that incorporates the two key enablers to address CDN-ISP collaboration system issues towards a joint CDN deployment and operation inside the ISP network.
- We perform the first-of-its-kind evaluation based on traces from the largest commercial CDN and a large tier-1 ISP using NetPaaS. We report on the benefits for CDNs, ISPs, and end-users. Our results show that CDN-ISP collaboration leads to a win-win situation with regards to the deployment and operation of servers within the network, and significantly improves end-user performance.

2. ENABLING CDN-ISP COLLABORATION

CDN-ISP collaboration has to address a set of challenges regardless whether a CDN utilizes traditional or emerging solutions to deliver content. We first highlight these challenges for content delivery today and then propose two key enablers to address them and facilitate CDN-ISP collaboration.

2.1 Challenges in Content Delivery

Economics, especially cost reduction, is a main concern today in content delivery as Internet traffic grows at a annual rate of 30% [49]. Moreover, commercial-grade applications delivered by CDNs often have requirements in terms of end-to-end delay [39]. Faster and more reliable content delivery results in higher revenues for e-commerce and streaming applications [43, 50] as well as user engagement [29]. Despite the significant efforts by CDNs to improve content delivery performance, end-user mis-location, and the limited view of network bottlenecks are major obstacles to improve end-user performance.

Content Delivery Cost: CDNs strive to minimize the overall cost of delivering voluminous content traffic to end-users. To that end, their assignment strategy is mainly driven by economic aspects such as bandwidth or energy cost [45, 57]. While a CDNs will try to assign end-users in such a way that the server can deliver reasonable performance, this does not always result in end-users being assigned to the server able to deliver the best performance. Moreover, the intense competition in the content delivery market has led to diminishing returns of delivering traffic to end-users. Part of the delivery cost is also the maintenance and constant upgrading of hardware and peering capacity in many locations [50].

End-user Mis-location: DNS requests received by the CDN name servers originate from the DNS resolver of the end-user, not from the end-user themselves. The assignment of end-users to servers is therefore based on the assumption that end-users are close to the used DNS resolvers. Recent studies have shown that in many cases this assumption does not hold [45, 57]. As a result, the end-user is mis-located and the server assignment is not optimal. As a response, DNS extensions have been proposed to include the end-user IP information [26, 51].

Network Bottlenecks: Despite their efforts to discover end-to-end characteristics between servers and end-users to predict performance [50, 39], CDNs have limited information about the actual network conditions. Tracking the ever changing network conditions, i.e., through active measurements and end-user reports, incurs an extensive overhead for the CDN without a guarantee of performance improvements for the end-user. Without sufficient information about the characteristics of the network paths between the CDN servers and the end-user, a user assignment performed by the CDN can lead to additional load on existing network bottlenecks, or even create new ones.

2.2 Enablers

Given the trends regarding increasing need of server resources and content demand by end-users, content delivery systems have

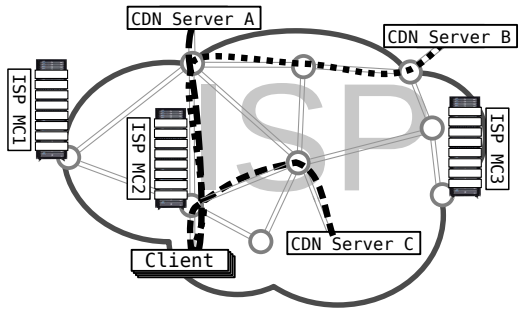


Figure 2: ***Informed User-Server Assignment:*** **Assigning a user to an appropriate CDN server among those available (A, B, C), yields better end-user performance and traffic engineering.** ***In-network Server Allocation:*** **A joint in-network server allocation approach allows the CDN to expand its footprint using additional and more suitable locations (e.g., microdatacenters MC1, MC2, MC3) inside the network to cope with volatile demand. User-server assignment can also be used for redirecting users to already deployed and new servers.**

to address two fundamental problems. The first is the end-user to server assignment problem, i.e., how to assign users to the appropriate servers. The key enabler for addressing this problem is *informed user-server assignment* or in short *user-server assignment*. It allows a CDN to receive recommendations from a network operator, i.e., a server ranking based on performance criteria mutually agreed upon by the ISP and CDN. The CDN can utilize these recommendations when making its final decision regarding end-user to server assignments. This enabler takes full advantage of server and path diversity, which a CDN has difficulty exploring on its own. Moreover, its design allows the coordination of CDNs, content providers and ISPs in near real-time, as we will elaborate in section 3. Any type of CDN can benefit from this enabler including ISP-operated CDNs. The advantage of our enablers in comparison with other CDN-ISP [28, 34] and ISP-P2P [66] cooperation schemes is that no routing changes are needed.

The second is the server allocation problem, i.e., where to place the servers and content. The key enabler is *in-network server allocation*, or in short *server allocation*, where the placement of servers within a network is coordinated between CDNs, ISPs, and content providers. This enabler provides an additional degree of freedom to the CDN to scale-up or shrink the footprint on demand and thus allows it to deliver content from additional locations inside the network. Major improvements in content delivery are also possible due to the fact that the servers are placed in a way that better serve the volatile user demand. The application of this enabler is twofold. One, it helps the CDN in selecting the locations and sizes of server clusters in an ISP when it is shipping its own hardware. The second application is suitable for more agile allocation of servers in cloud environments, such as those mentioned in [10]. Multiple instances of virtual servers running the CDN software are installed on physical servers owned by the ISP. As before, the CDN and the ISP can jointly decide on the locations and the number of servers. A big advantage of using virtual machines is that the time scale of server allocation can be reduced to hours or even minutes depending on the requirements of the application and the availability of physical resources in the network. User-server assignment can also be used for redirecting users to the new servers. We provide the high-level intuition for both enablers in Figure 2.

Until now, both problems have been tackled in a one-sided fashion by CDNs. We believe that to improve content delivery, accurate and up-to-date information should be used during the server selection by the CDN. This also eliminates the need for CDNs to perform cumbersome and sometimes inaccurate measurements to infer the changing conditions within the ISP. We also believe that the final decision must still be made by the CDN. In this paper, we argue that the above enablers (a) are necessary to enable new CDN architectures that take advantage of server virtualization technology, (b) allow fruitful coordination between all involved parties, including CDNs, CPs, and ISPs in light of the new CDN-ISP alliances, (c) enable the launch of new applications jointly by CDNs and ISPs, and (d) can significantly improve content delivery performance. Such performance improvements are crucial as reductions in user transaction time increase revenues by significant margins [35].

3. NetPaaS PROTOTYPE

Today there is no system to support CDN-ISP collaboration and joint CDN server deployment within an ISP network. In this section we design a novel system, NetPaaS (Network Platform as a Service), which incorporates the two key enablers for CDN-ISP collaboration introduced in Section 2. First, we give a overview of NetPaaS and describe its functionalities and the protocols it utilizes to enable collaboration. Next, we give a detailed description of the NetPaaS architecture. Finally we comment on the scalability and privacy preserving properties of NetPaaS.

3.1 NetPaaS Functionalities and Protocols

NetPaaS enables CDNs and ISPs to efficiently coordinate the user to server assignment and allows the CDN to expand or shrink its footprint inside the ISPs network on demand, towards achieving performance targets [39] and traffic engineering goals [55]. Neither of them is a trivial task when dealing with large networks (thousands of routers), highly distributed microdatacenters (in tens of locations and hundreds of machines), and constant network, routing, and traffic updates.

The NetPaaS protocol allows CDNs to express required server specifications and ISPs to communicate available resources and their prices. It is designed to exchange information in very small time scales, e.g., in the order of seconds (similar to the time scale that CDNs can potentially redirect users [53]), enabling fast responses to rapid changes in traffic volumes. Any ISP operating a NetPaaS system offers the following services: ***(1)*** **User-server assignment**: allows to request recommendations for user to server mapping from the ISP. ***(2)*** **Resource discovery**: communicates information about resources, e.g., available locations or number of servers and the conditions for leasing them, e.g., price and reservation times. ***(3)*** **Server allocation**: enables a CDN to allocate server resources within the ISPs network.

The protocol utilized by NetPaaS is designed to be efficient and to minimize delay and communication overhead. The required communication for the different services are explained in more detail in the following Sections 3.1.1 and 3.1.2. For the user-server assignment service NetPaaS also supports BGP as communication protocol as this is already supported by many CDN operators, e.g., Google Global Cache [4], Netflix Open Connect [6], or the Akamai Network [50].

3.1.1 NetPaaS *Protocol for User-Server Assignment*

We first describe the general approach for user-server assignment today and continue with the required additional steps and protocol messages for our collaborative approach, illustrated in the top left of Figure 3 ("CDN: user assign"). When a CDN receives a DNS request, typically by a resolver (i.e., when the answer is not locally available in the local resolver), it utilizes internal information in order to assign a server to satisfy the request. The selection of the server depends on the location of the source of the request, as this is

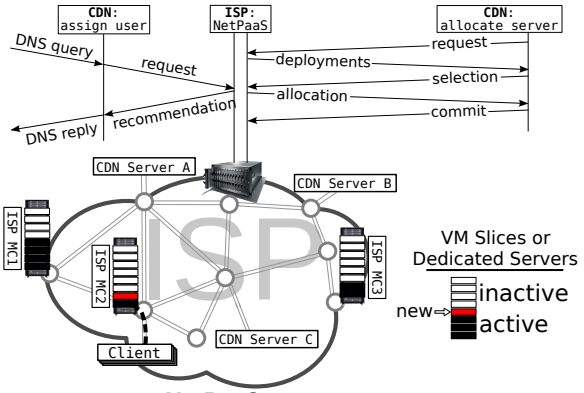


Figure 3: NetPaaS protocols and operation.

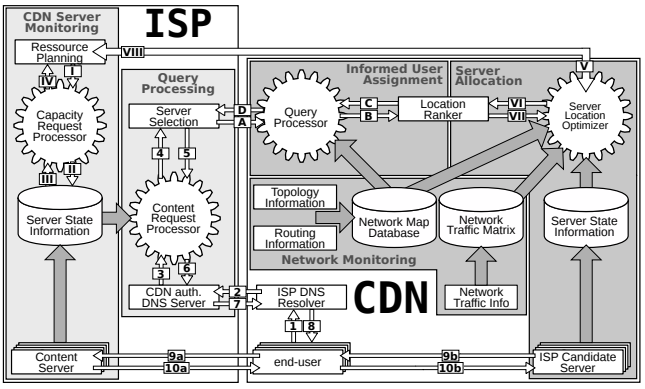


Figure 4: NetPaaS architecture.

inferred from the resolvers that sends it, as well as the availability of close-by servers and cost of delivery [50, 60]. When the CDN selects a set of servers to satisfy the request, it sends a DNS reply back to the resolver that sent the DNS request who then sends it to the source of the request. Notice that for scalability reasons and to deal with flash crowds, large CDNs allow all the available servers to serve the same content [62]. If the content is not locally available, the server fetches the content from other servers or the original server, stores it locally (that yields pull-based replication), and sends it to the end-user [50]. To take advantage of the ISPs NetPaaS user-server assignment service the CDN issues a recommendation request prior to answering the DNS query. The recommendation request contains the source of the DNS request and a list of eligible CDN server IPs which NetPaaS ranks based on ISP-internal information, e.g., link utilization or path delay, and possible traffic engineering goals. If the source of the DNS request is the ISP operated DNS resolver or when the EDNS0 Client Subnet Extension [26] is present, NetPaaS can precisely locate the end-user inside the ISPs network, effectively increasing the recommendations precision of the system. The ISP then returns this preference ordered list in a recommendation message to the CDN which can select the most appropriate servers based on both the ISPs and its own criteria and thus optimizing the user-server assignment while staying in complete control of the final server selection process.

3.1.2 NetPaaS *Protocol for Server Allocation*

We next describe the steps and required protocol messages for collaborative server allocation that are illustrated in the top right of Figure 3 ("CDN: allocate server"). When a CDN decides that additional servers are needed to satisfy the end-user demand or when the CDN and ISP jointly agree to deploy new servers inside the ISP, the CDN submits a request to NetPaaS. The request contains the required hardware resources, a demand forecast (e.g., per region or per subnet) together with a number of optimization criteria and possible constraints. The demand forecast allows NetPaaS to compute an optimal placement for the newly allocated server(s). Optimization criteria include minimizing network distance or deployment cost among others. Possible constraints are the number of locations, minimum resources per server, or reservation time. Based on this information NetPaaS computes a set of deployments, i.e., the server locations and the number of servers, by solving an optimization problem (namely the SiSL or the CFL problem, see Section 3.2.3). The reply contains the possible deployments and their respective prices. The CDN either selects one or more of the offered deployments by sending a selection message to NetPaaS or starts over by submitting a new request. When receiving a selection message, NetPaaS checks if it can offer the selected deployment. If all conditions are met, NetPaaS reserves the requested resources to guarantee their availability and sends an allocation message as confirmation to the CDN. If the conditions cannot be met, the selection by the CDN is denied by NetPaaS. To gain control of the allocated servers, the CDN has to send a commit message to NetPaaS which completes the communication for server allocation.

The ISP may offer physical machines or virtual machines (VMs) to CDNs. In the second case the servers are refereed to as "slices" of hardware servers. To move servers from one to another network position, NetPaaS supports the flexibility of VM migration or consolidation. A possible deployment scenario with VMs can be seen in Figure 3. To improve CDN server startup and cache warm-up times, one option for CDNs is to always keep a small number of active servers in a diverse set of locations to expand or shrink it according to the demand. They can also pre-install an image of their server in a number of locations.

3.2 Architecture

We now provide the detailed architecture of the system, aimed at providing accurate user-server assignments as well in-network server allocations for the CDN. We describe the components and processes both at the ISP as well as the CDN side. In the ISP the main tasks of our system are to: (1) maintain an up-to-date annotated map of the ISP network and its properties as well as the state of the ISP-operated servers within the network, (2) provide recommendation on where servers can be located to better satisfy the demand by the CDN and ISP traffic engineering goals, and (3) to assist the CDN in user-server assignment and server allocation by creating preference rankings based on the current network conditions. The goal of the system is to fully utilize the available server and path diversity as well as ISP-maintained resources within the network, while keeping the overhead for both the CDN and the ISP as small as possible.

NetPaaS comprises three main components: *Network Monitoring*, *Informed User Assignment*, and *Server Allocation Interface*. For an overview of the architecture, see the ISP grey area in Figure 4. Steps 1-10 and I-IV that illustrate the requests and responses and the CDN server selection respectively, as performed in currently deployed CDNs, for more information and details see [50].

3.2.1 Network Monitoring Component

The Network Monitoring component gathers information about the topology and the state of the network to maintain an up-to-date view of the network. The Topology Information component gathers detailed information about the network topology, i.e., routers and links, annotations such as link utilization, router load as well as topological changes. An Interior Gateway Protocol (IGP) listener provides up-to-date information about routers and links. Additional

information, e.g., link utilization and other metrics can be retrieved via SNMP from the routers or an SNMP aggregator. The Routing Information uses routing information to calculate the paths that traffic takes through the network. Finding the path of egress traffic can be done by using a Border Gateway Protocol (BGP) listener. Ingress points of traffic into the ISP network can be found by utilizing Netflow data. This allows for complete forward and reverse path mapping inside the ISP. In total, this allows for a complete path map between any two points in the ISP network. The Network Map Database processes the information collected by the Topology and Routing Information components to build an annotated map of the ISP network. While it builds one map of the network, it keeps the information acquired from the other two components in separate data structures. The Topology Information is stored as a weighted directed graph, while the prefix information is stored in a Patricia trie [48]. This separation ensures that changes in prefix assignment learned via BGP do not directly affect the routing in the annotated network map. To further improve performance, the path properties for all paths are pre-calculated. This allows for constant lookup speed independent of path length and network topology. Having ISP-centric information ready for fast access in a database ensures timely responses and high query throughput.

3.2.2 Informed User-Server Assignment Component

When the CDN sends a request for user-server assignment to NetPaaS, the request is handled by the Query Processor (steps A to D in Figure 4). The request from the CDN specifies the end-user and a list of candidate CDN servers. First, the Query Processor maps each source-destination (server to end-user) pair to a path in the network. Note that the end-user is seen through its DNS resolver, often the ISPs DNS resolver [14], unless both ISP and CDN support the EDNS0 Client Subnet Extension [26, 51]. The properties of the path are then retrieved from the Network Map Database. Next, the pairs are run individually through the Location Ranker subcomponent (see below) to get a preference value. Finally, the list is sorted by preference values, the values stripped from the list, and the list is sent back to the CDN. The ISP Location Ranker computes the preference value for individual source-destination pairs based on the path properties and an appropriate function (see steps B, C). The function depends on the goal specified by the CDN, such as a performance goal, as well as an operational one, such as a traffic engineering objective. Note that NetPaaS is not limited to a single optimization function per CDN.

3.2.3 In-network Server Allocation Component

When the CDN Resource Planner sends a server allocation request to NetPaaS asking for available servers within the ISP (steps V to VIII), the request is handled by the ISP Server Location Optimizer. It uses the Network Monitoring component to get up-to-date information about the ISPs network and the current and historic network traffic matrices and the Server State Information database, which collects up-to-date state information regarding the ISP's servers (e.g., server load and connectivity).

The problem that the ISP Server Location Optimizer has to solve can be modeled as an instance of either the Simultaneous Source Location problem (SiSL) [16], or the Capacitated Facility Location problem (CFL) [36]. The locations at which facilities can be opened correspond to the locations at which servers can be placed, and there is a constraint on the amount of bandwidth available at each location or on each network link.

In SiSL, the goal is to determine where the servers should be placed so as to satisfy demand while respecting the capacity constraints, and also possibly minimizing the distance between servers and users. Given the specification of a server, if the capacity of a location allows multiple servers to be allocated then the solution may allocate more than one server per location. The ISP has a detailed view of the network activity (e.g., traffic matrices over a period of time), the annotated network topology, and the candidate locations to install servers, along with the available resources, including the network capacity at these locations. The CDN can also express the demand that needs to be satisfied with additional servers as well as the server requirements.

In the CFL solution, to prevent the creation of hot-spots, the distance of users to servers is proportional to the utilization of the most congested link (given the background traffic) along the path from the server to the end-user. We also assume that the user-server assignment enabler is in place. In our setting users can be assigned to different servers for each request to a server. Thus, the demand is splittable. This allows for fast and accurate server allocations using standard local search heuristics for CFL [18].

The outcome of joint server allocation is the number and location of additional servers. The result is communicated to the two parties that have to agree on the calculated setting.

Joint Hardware Server Allocation: In this case the collaboration of the ISP and CDN is in large time scales (weeks) and the servers are physical machines installed and maintained by the ISP and operated by the CDN. In the setting of the ISP-operated CDN, the server allocation is an optimized way of deploying the CDN footprint inside the network. The forecast of the demand by analyzing CDN logs can also be incorporated. This joint operation also allows the launch of new and demanding applications such as video streaming and interactive online gaming.

Joint Software Server Allocation: As mentioned before, servers can be either physical machines owned by the CDN, virtual machines offered by the ISP, or both. With virtualization, the above solution can be utilized whenever software servers are allocated. This allows for flexible server allocation using a mature technology. Virtualization has been used to allocate heterogeneous resources [64, 25], computation (e.g., VMWare, Xen, and Linux VServer), storage, and network [58], in datacenters [17], as well as distributed clouds inside the network [22, 10]. Recent measurement studies have shown significant performance and cost variations across different virtualization solutions [44]. In response, a number of proposals have addressed the specific requirements of applications [20, 37, 46] and the scalability to demand [56, 65]. To capitalize on the flexibility and elasticity offered by virtualization, a number of systems have been built to automate data and server placement [12, 27, 63] and server migration [21, 42] even between geographically distributed datacenters. Other approaches have focused on the selection of locations for service mirrors and caches inside a network, to minimize the network utilization [38, 41]. In the joint server allocation setting the decision and installation time can be reduced to hours or even minutes. This is feasible as an ISP can collect near real-time data for both the network activity and availability of resources in datacenters operated within its network or in microdatacenters collocated with ISP network aggregation points [22].

3.3 Scalability

User-Server Assignment: To improve scalability and responsiveness, we do not rely on HTTP embedded JSON as proposed in by ALTO IETF group, but on light protocols that are similar to DNS. A single instance of our system is able to reply to more than $90,000$ queries/sec when serving requests with 50 candidate CDN servers. At this level, the performance of our system is comparable to popular DNS servers, e.g., BIND. The computational response time is below 1 ms for a 50 candidate server list. By placing the

service inside ISP networks at well connected points, the additional overhead is small compared to the DNS resolution time [14]. This performance was achieved on a commodity dual-quad core server with 32 GB of RAM and 1Gbps Ethernet interfaces. Furthermore, running additional servers does not require any synchronization between them since each instance is acquiring the information directly from the network. Thus, multiple servers can be located in different places inside the network to improve scalability.

Server Allocation: Today, a number of off-the-shelf solutions are available to spin a virtual server based on detailed requirements [46], and are already available from vendors such as NetApp and Dell. To test the scalability of in-network server allocation we used an appliance collocated with a network aggregation point of ADSL users which consists of 8 CPUs (16 cores), 24 GByte RAM, Terabytes of solid state disks, and a 10 Gbps network interface. A management tool that follows the VMware, Cisco, and EMC (VCE) consortium industrial standard [25] is also installed. We tested different server configurations and our results show that VM boot up times are on the order of tens of seconds while virtualization overhead during runtime is negligible. To that end we confirm that it is possible to even fully saturate a 10 Gbps link. It was also possible to add, remove, and migrate live servers on demand in less than a minute. To reduce the cache warm-up time when allocating a new server, the requests to an already operational cache are duplicated and fed to the new one for around ten minutes.

3.4 Privacy

During the exchange of messages, none of the parties is revealing sensitive operational information. In user-server assignment, CDNs only reveal the candidate servers that can respond to a given request without any additional operational information (e.g., CDN server load, cost of delivery). On the other side, the ISP does not reveal any operational information or the preference weights it uses for the ranking. In fact, the ISP only re-orders a list of candidate servers provided by the CDN. This approach differs from [66], where partial or complete ISP network information, routing weights, or ranking scores are publicly available. During the server allocation a CDN can decide either to request a total demand or demand in a region (e.g., city, country), thus it does not unveil the demand of an end-user.

4. DATASETS

To evaluate the NetPaaS system, we use traces from the largest commercial CDN and a large European tier-1 ISP.

Commercial CDN Dataset: The CDN dataset covers a two-week period from 7th to 21st March 2011. All entries in the log we use relate to the tier-1 ISP. This means that either the server or the end-user is using an IP address that belongs to the address space of the tier-1 ISP. The CDN operates a number of server clusters located inside the ISP and uses IPs in the IP address space of the ISP (see Section 5.1). The log contains detailed records of about 62 million sampled (uniformly at random) valid TCP connections between the CDN's servers and end-users. For each reported connection, it contains the time it was recorded, the server IP address, the cluster the server belongs to, the anonymized client IP address, and various connection statistics such as bytes sent/received, duration, packet count and RTT. The CDN operates a number of services, utilizing the same infrastructure, such as dynamic and static web pages delivery, cloud acceleration, and video streaming.

ISP Dataset: The ISP dataset contains two parts. First, detailed network information about the tier-1 ISP, including the backbone topology, with interfaces and link annotations such as routing weights, as well as nominal bandwidth and delay. It also contains

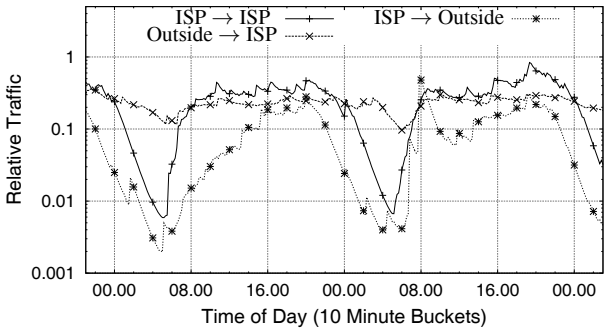


Figure 5: Activity of CDN in two days.

the full internal routing table which includes all subnets propagated inside the ISP either from internal routers or learned from peerings. The ISP operates more than 650 routers in about 500 locations (PoPs), and 30 peering points worldwide. We analyzed more than 5 million routing entries to derive a detailed ISP network view.

The second part of the ISP dataset is an anonymized packet-level trace of residential DSL connections. Our monitor, using Endace monitoring cards [23], observes the traffic of around 20,000 DSL lines to the Internet. We capture HTTP and DNS traffic using the Bro IDS [52]. We observe 720 million DNS messages and more than 1 billion HTTP requests involving about 1.4 million unique hostnames. Analyzing the HTTP traffic in detail reveals that a large fraction it is due to a small number of CDNs, including the considered CDN, hyper-giants and one-click-hosters [40, 32, 47] and that more than 65% of the traffic volume is due to HTTP.

To derive the needed traffic matrices, on an origin-destination flow granularity, we compute from the DSL traces (on a 10-minute time bin granularity) the demands for the captured location in the ISP network. This demand is then scaled according to the load imposed by users of the CDN to the other locations in the ISP network. For CDNs without available connection logs, we first identify their infrastructure locations using the infrastructure aggregation approach as proposed by Poese et al. [53] and then scale the traffic demands according to the available CDN connection logs.

5. EVALUATION

In this section we quantify the benefits of using NetPaaS. For our evaluation we rely on traces from the largest commercial CDN and the tier-1 ISP described in Section 4. We start by presenting the traffic characteristics of the CDN inside the ISP and discuss the rationale for NetPaaS. We then evaluate the benefits of NetPaaS in the emulation environment described in [54].

5.1 Collaboration Potential

We first describe our observations on the traffic and deployment of the large commercial CDN inside the tier-1 ISP and analyze the potential benefits of CDN-ISP collaboration. In Figure 5, we plot the normalized traffic (in log scale) from CDN clusters over time. We classify the traffic into three categories: *a)* from CDN servers inside the ISP to end-users inside the ISP (annotated ISP → ISP), *b)* from servers outside the ISP to end-users inside the ISP (annotated outside → ISP), and *c)* from CDN servers inside the ISP to end-users outside the ISP (annotated ISP → outside).

We observe the typical diurnal traffic pattern and a daily stability of the traffic pattern. Over the two week measurement period, 45.6% of the traffic belongs to the ISP → ISP category. 16.8% of the traffic belongs to the outside → ISP category. During peak hours, outside → ISP traffic can grow up to 40%. Finally, 37.6%

Figure 6: Potential hop reduction by using NetPaaS.

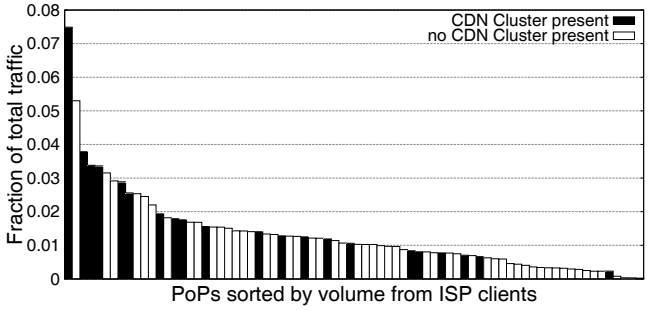


Figure 7: Traffic demand by ISP network position.

of the traffic is served by inside clusters to outside end-users. Our first important observation is that a significant fraction of the CDN traffic is served from servers outside the ISP despite the presence of many servers inside the ISP that would be able to serve this traffic.

Figure 6 shows the re-allocation of traffic that would be possible using user-server assignment. Each full bar shows the fraction of traffic currently traversing a given number of router hops within the ISP network. In this evaluation, we only consider the end-users inside the ISP. The bar labeled "N/A" is the traffic of the outside $\rightarrow$ ISP category. The different shaded regions in each bar correspond to the different router hop distances after re-allocation of the traffic. Almost half of the traffic currently experiencing 3 hops can be served from a closer-by server. Overall, a significant fraction of the traffic can be mapped to closer servers inside the ISP. Note that the tiny amount of traffic for router hop count 0 and 1 is due to the topology design of the ISP network: either the traffic stays within a PoP or it has to traverse at least two links to reach another PoP.

In Figure 7, we show the traffic demand towards the CDN generated by each PoP. We observe that some PoPs originate high demand while others have limited demand, if any. Manual inspection reveals that some of the PoPs with high demand cannot be served by a close-by CDN server, while other low demand PoPs have a cluster near by. Variations in the demand over time exhibit even more significant mismatches between demand and CDN locations. With such a time-varying demand and the timescales at which CDN deployments take place today, such mismatches should be expected.

We conclude that there are ample opportunities for CDNs to benefit from collaboration with ISPs to re-arrange or expand their footprint. Also, these observations support the use of NetPaaS to improve the operation of both the CDN and the ISP in light of the new CDN-ISP strategic alliances [1, 7, 8, 5].

5.2 Improvements with NetPaaS

In this section we quantify the benefit of NetPaaS for the large commercial CDN inside the tier-1 ISP. First we show the benefits of user-server assignment for the existing CDN infrastructure and continue with the additional benefit of server allocation. In our evaluation we ensure that NetPaaS respects the available CDN server capacities and specifications in different locations. In the rest of the section, unless otherwise mentioned, we optimize the delay between end-user and CDN server [50]. Moreover, as we will show in our evaluation, by optimizing the delay between end-user and CDN server other traffic engineering goals are achieved.

5.2.1 *Informed End-user to Server Assignment*

We first evaluate the benefits NetPaaS can offer when using user-server assignment only for the already deployed infrastructure of the large commercial CDN. In Figure 8(a) we show the current path delay between end-user and CDN servers, annotated as "Base". When using user-server assignment, annotated as "User assign", the delay is reduced by 2–6 msecs for most of the CDN traffic and another 12% of all traffic can be fetched from nearby CDN servers, a significant performance gain. To achieve similar gains CDNs have to rely on complicated routing tweaks [39].

When utilizing NetPaaS for user-server assignment the traffic traverses a shorter path within the network. This yields an overall traffic reduction in the network. In Figure 8(b) we plot the reductions in the overall traffic within the network, labeled "User-assign". The reduction can be as high as 7% during the peak hour. This is a significant traffic volume that is on the scale of tens to hundreds of Terabytes per day in large ISPs [2, 3]. As a consequence, the most congested paths are circumvented, as the full server and path diversity is utilized [55]. Our evaluation shows that user-server assignment significantly improves CDN operation with the already deployed infrastructure and capacity. Moreover, the ISP does not need to change its routing, thus reducing the possibility of introducing oscillations [30].

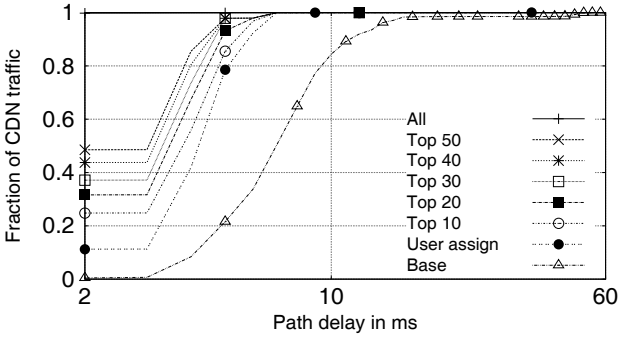
In Figure 8(c) we plot the reduction in utilization for the most congested link at any point of time. We observe that during the peak time the utilization of the most congested link can be reduced by up to 60%. This is possible as traffic is better balanced and the link is utilized to serve mainly the local demand. Such a reduction in utilization can postpone link capacity upgrades.

5.2.2 *In-network Server Allocation*

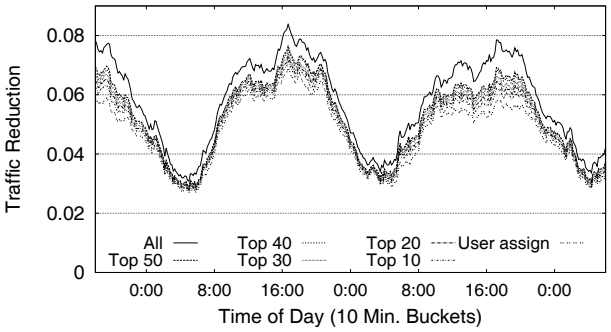
We next evaluate the benefits of NetPaaS when server allocation is used in addition to user-server assignment. For short term CDN server deployments virtualized servers offer flexibility. For long term deployments, especially in light of the CDN-ISP alliances [1, 7, 8, 5], bare metal servers offer better performance. As our evaluation shows, the optimized placement of servers improves end-user performance as well as server and path diversity in the network, and enables ISPs to achieve traffic engineering goals.

To estimate the locations for installing new servers, we use the local search heuristic to approximate the solution of CFL (see Section 3.2.3). Figure 9 shows the accuracy of server allocation in terms of delay reduction when deploying 30 and 50 additional servers, labeled "Top 30" and "Top 50" respectively (similar observations are made for other numbers of servers). Notice that these 30 or 50 servers are not necessarily in the same PoP. It can be the case that more than one server is in the same PoP. For the optimal cases we pre-computed the best server locations based on the full knowledge of our 14-days dataset, while NetPaaS calculates the placement by utilizing past traffic demands and the current network activity during runtime. Our results show that NetPaaS achieves gains close to those of the optimal placement.

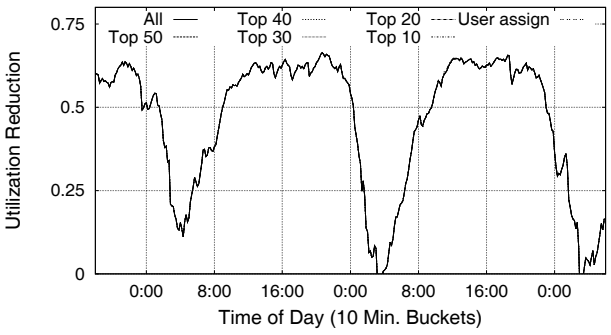
In Figure 8(a) we show the delay improvements of NetPaaS when less than 10% of the servers are utilized, thus we range the number of servers between 10 to 50 servers that are allocated in any



(a) Improvements in user to server delay.



(b) Total traffic reduction within the network.



(c) Maximum link utilization reduction.

Figure 8: Utilizing NetPaaS for user-server assignment and server allocation.

of the about 500 locations within the ISP, labeled "Top 10" to "Top 50". We also include a case where servers are allocated in all possible locations, labelled "All". As expected, in this case, nearly all traffic can be served from the same PoP as the end-user. Yet, with only 10 additional servers around 25% of the CDN demand can be satisfied in the same PoP. With 50 additional servers it is possible to satisfy more than 48% of the CDN demand by a server located in the same PoP as the end-users. This shows that a relatively small number of servers can reduce the user to server delay significantly. It also shows the impact that the placement of a server plays in reducing the delay between end-user and content server. Note, that we report on the reduction of the backbone delay, the reduction of the end-to-end delay is expected to be even higher as the server is now located in the same network.

We next turn our attention to the possible traffic reduction in the network when NetPaaS is used. In Figure 8(b) we show the possible network wide traffic reduction with server allocation when 10 to 50 servers can be allocated by the CDN. The traffic reduction especially during the peak hour ranges from 7% with 10 additional

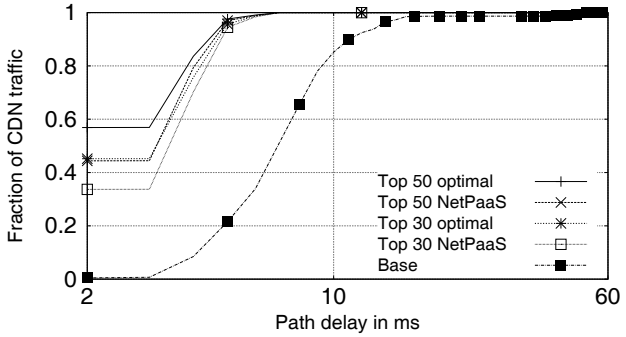


Figure 9: NetPaaS accuracy in selecting server location.

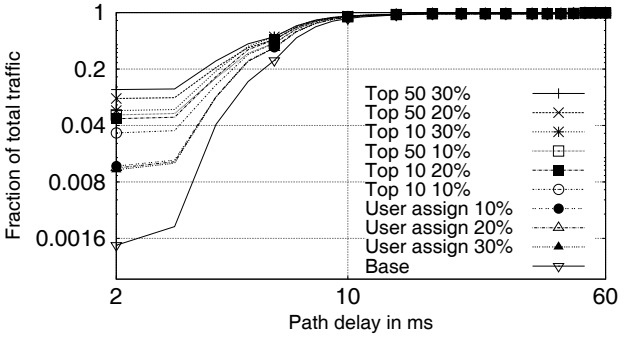
servers and reaches up to 7.5% when 50 additional servers can be utilized. Again, this is a significant traffic volume that is on the scale of tens to hundreds of Terabytes per day in large ISPs. Note that the primary goal of NetPaaS was to reduce the end-user to server delay, not network traffic. If all available locations (about 500) are utilized by the CDN, then the total traffic reduction is around 8% during peak time. This shows that a small number of additional servers significantly reduces the total traffic inside the network. We also notice that our algorithm places servers in a way that the activity of the most congested link is not increased, see Figure 8(c). In our setting, further reduction of the utilization of the most congested link by adding more servers was not possible due to routing configuration.

5.3 Joint Service Deployment with NetPaaS

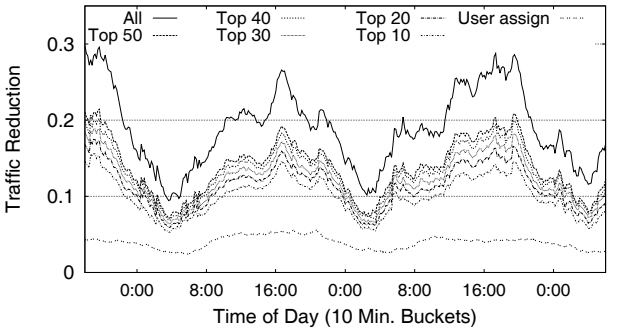
We next consider the case of a CDN or an application that is launched within an ISP by exclusively utilizing NetPaaS. Examples include ISP-operated CDNs, licensed CDNs, or application-based CDNs. The latter is already happening with Google Global Cache [4] and with Netflix Open Connect in North America and North Europe [6]. Today, Netflix is responsible for around 30% of the total traffic in the peak hour in major US-based carriers [33]. In this section, we evaluate the performance of NetPaaS when such a service is launched using a CDN-ISP collaborative deployment scheme. In Figure 10 we show the benefits of a joint CDN-ISP server deployment within the network. For our evaluation, we use the large commercial CDN, for which we know the sources of the demand and the server specifications and locations, and scale its traffic to reach 10%, 20%, or 30% of the total traffic of the ISP. As previously, with NetPaaS and using only user-server assignment, it is possible to satisfy a significant fraction of the total traffic from close-by servers, see Figure 10(a). This can be even increased further when additional locations are available via server allocation. Our results also show that while increasing the traffic demand for the CDN, NetPaaS manages to keep the delay between users and servers low, as well as to reduce the total network traffic.

Figure 10(b) shows the total traffic reduction when the CDN traffic accounts for 30% of the total traffic. With user-server assignment only, NetPaaS is able to reduce the total traffic inside the network by up to 5%. When assigning additional servers, NetPaaS is able to reduce the total traffic from 15% with 10 servers to 20% with 50 servers. A traffic reduction of up to 30% is possible when additional servers can be allocated in all ISP PoPs.

We also tested NetPaaS with multiple CDNs to evaluate the scalability of the system as well as the potential benefit of the system. For this, only user-server assignment was used as no information about the server requirements and the capacity of the other CDNs is available. We consider the top 1, 10, and 100 CDNs by traffic volume in the ISP. The largest CDN accounts for 19% of the

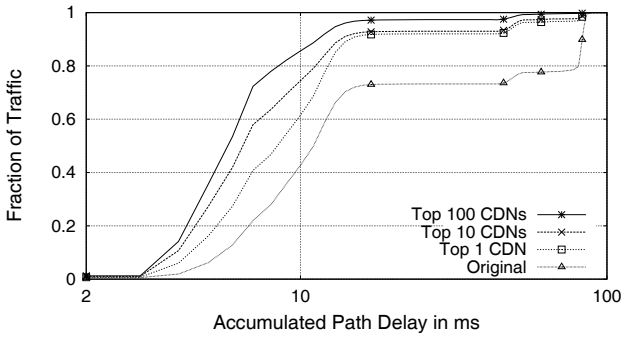


(a) Reduction in user-server delay by NetPaaS.

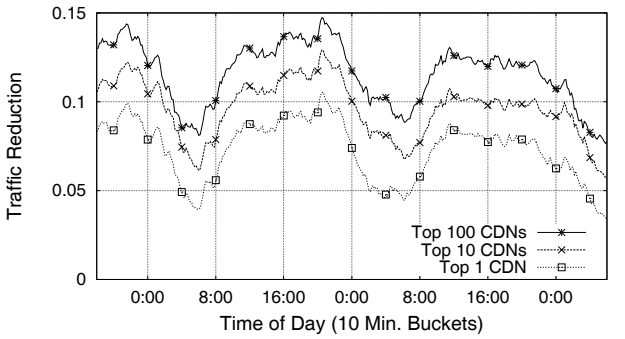


(b) Total traffic reductions by NetPaaS (30% CDN traffic).

Figure 10: Joint service deployment with NetPaaS.



(a) Reduction in user-server delay for top 1, 10, and 100 CDNs.



(b) Total network traffic reduction for top 1, 10, and 100 CDNs.

Figure 11: Improvements with NetPaaS when considering the top 1, 10, and 100 CDNs.

total traffic, the top 10 CDNs are responsible for more than 40% and the top 100 CDNs for more than 67% respectively. Most of the large CDNs have deployed distributed infrastructure, located in a number of networks [55]. Figure 11 shows the improvements in user-server delay as well as the total traffic reduction achieved by NetPaaS. For the largest CDN most of the traffic can be served from close-by servers and as a result the total traffic can be reduced by up to 10%. When turning our attention to the top 10 and top 100 CDNs, we observe that NetPaaS is able to further increase the improvements, but with diminishing returns. With the top 10 CDNs the traffic is reduced by up to 13% and with the top 100 CDNs 15% respectively. We conclude that by utilizing NetPaaS for the top 10 CDNs, it is possible to achieve most of the reduction in user-server delay and total traffic. We present a larger set of results for the top CDNs and an evaluation for a number of optimization goals under various network topologies in [31].

6. CONCLUSION

Motivated by recent CDN and ISP alliances we revisit the problem of CDN-ISP collaboration from a systems perspective. We identify two major enablers, namely informed user-server assignment and in-network server allocation. Today, there is no system to support CDN-ISP collaboration. To that end we design and implement a system for CDN-ISP collaboration, called NetPaaS, that incorporates the above enablers. We perform the first-of-its-kind evaluation of CDN-ISP collaboration based on traces from the largest commercial CDN and a large tier-1 ISP using NetPaaS. We report on the benefits for CDNs, ISPs, and end-users. Our results show that with NetPaaS, CDN-ISP collaboration leads to a win-win situation with regards to the deployment and operation of servers within the network, and significantly improves end-user performance. A key observation is that agile and online placement of servers inside the network closer to the source of demand is the key to improve content delivery and address traffic engineering, while some benefits are also possible with the already deployed server infrastructure. We believe that NetPaaS can be widely used in the new landscape of joint CDN-ISP server deployment inside the network and act as a catalyst for innovative solutions towards improving network operation, reducing content delivery cost and enabling new applications inside the network.

Acknowledgment

This work was supported in part by the EU projects CHANGE (FP7-ICT-257422) and BigFoot (FP7-ICT-317858), an IKY-DAAD award (54718944), and AFRL grant FA8750-11-1-0262.

7. REFERENCES

[1] Akamai and AT&T Forge Global Strategic Alliance to Provide Content Delivery Network Solutions. http://www.akamai.com/html/about/press/releases/2012/press_120612.html.
[2] AT&T Company Information. http://www.att.com/gen/investor-relations?pid=5711.
[3] Deutsche Telekom ICSS. http://ghs-internet.telekom.de/dtag/cms/content/ICSS/en/1222498.
[4] Google Global Cache. http://ggcadmin.google.com/ggc.
[5] KT and Akamai Expand Strategic Partnership. http://www.akamai.com/html/about/press/releases/2013/press_032713.html.
[6] Netflix Open Connect. https://signup.netflix.com/openconnect.
[7] Orange and Akamai form Content Delivery Strategic Alliance. http://www.akamai.com/html/about/press/releases/2012/press_112012_1.html.
[8] Swisscom and Akamai Enter Into a Strategic Partnership. http://www.akamai.com/html/about/press/releases/2013/press_031413.html.
[9] T-Systems to offer customers VMware vCloud Datacenter Services. http://www.telekom.com/media/enterprise-solutions/129772.
[10] Network Functions Virtualisation. SDN and OpenFlow World Congress, October 2012.

[11] P. Aditya, M. Zhao, Y. Lin, A. Haeberlen, P. Druschel, B. Maggs, and B. Wishon. Reliable Client Accounting for Hybrid Content-Distribution Networks. In *NSDI*, 2012.

[12] S. Agarwal, J. Dunagan, N. Jain, S. Saroiu, A. Wolman, and H. Bhogan. Volley: Automated Data Placement for Geo-Distributed Cloud Services. In *NSDI*, 2010.

[13] B. Ager, N. Chatzis, A. Feldmann, N. Sarrar, S. Uhlig, and W. Willinger. Anatomy of a Large European IXP. In *SIGCOMM*, 2012.

[14] B. Ager, W. Mühlbauer, G. Smaragdakis, and S. Uhlig. Comparing DNS Resolvers in the Wild. In *IMC*, 2010.

[15] B. Ager, W. Mühlbauer, G. Smaragdakis, and S. Uhlig. Web Content Cartography. In *IMC*, 2011.

[16] K. Andreev, C. Garrod, B. Maggs, and A. Meyerson. Simultaneous Source Location. *ACM Trans. on Algorithms*, 6(1):1–17, 2009.

[17] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. H.. Katz, A. Konwinski, G. Lee, D. A. Patterson, A. Rabkin, I. Stoica, and M. Zaharia. Above the Clouds: A Berkeley View of Cloud Computing. UC Berkeley Technical Report EECS-2009-28, 2009.

[18] V. Arya, N. Garg, R. Khandekar, A. Meyerson, K. Munagala, and V. Pandit. Local Search Heuristics for k-Median and Facility Location Problems. *SIAM J. on Computing*, 2004.

[19] M. Axelrod. The Value of Content Distribution Networks. AfNOG 2008.

[20] H. Ballani, P. Costa, T. Karagiannis, and A. Rowstron. Towards Predictable Datacenter Networks. In *SIGCOMM*, 2011.

[21] R. Bradford, E. Kotsovinos, A. Feldmann, and H. Schiöberg. Live Wide-Area Migration of Virtual Machines Including Local Persistent State. In *VEE*, 2007.

[22] K. Church, A. Greenberg, and J. Hamilton. On Delivering Embarrasingly Distributed Cloud Services. In *HotNets*, 2008.

[23] J. Cleary, S. Donnelly, I. Graham, A. McGregor, and M. Pearson. Design Principles for Accurate Passive Measurement. In *PAM*, 2000.

[24] B. Cohen. Incentives Build Robustness in BitTorrent. In *P2PEcon Workshop*, 2003.

[25] Virtual Computing Environment Consortium. http://www.vce.com.

[26] C. Contavalli, W. van der Gaast, S. Leach, and E. Lewis. Client subnet in DNS requests. draft-vandergaast-edns-client-subnet-01.

[27] E. Cronin, S. Jamin, C. Jin, A. Kurc, D. Raz, and Y. Shavitt. Constraint Mirror Placement on the Internet. *JSAC*, 2002.

[28] D. DiPalantino and R. Johari. Traffic Engineering versus Content Distribution: A Game-theoretic Perspective. In *INFOCOM*, 2009.

[29] F. Dobrian, A. Awan, I. Stoica, V. Sekar, A. Ganjam, D. Joseph, J. Zhan, and H. Zhang. Understanding the Impact of Video Quality on User Engagement. In *SIGCOMM*, 2011.

[30] B. Fortz and M. Thorup. Optimizing OSPF/IS-IS Weights in a Changing World. *IEEE J. Sel. Areas in Commun.*, 2002.

[31] B. Frank, I. Poese, G. Smaragdakis, S. Uhlig, and A. Feldmann. Content-aware Traffic Engineering. *CoRR arXiv*, 1202.1464, 2012.

[32] A. Gerber and R. Doverspike. Traffic Types and Growth in Backbone Networks. In *OFC/NFOEC*, 2011.

[33] Sandvine Inc. Global broadband phenomena. Research Report http://www.sandvine.com/news/global_broadband_trends.asp.

[34] W. Jiang, R. Zhang-Shen, J. Rexford, and M. Chiang. Cooperative Content Distribution and Traffic Engineering in an ISP Network. In *SIGMETRICS*, 2009.

[35] R. Kohavi, R. M. Henne, and D. Sommerfield. Practical Guide to Controlled Experiments on the Web: Listen to Your Customers not to the HiPPO. In *KDD*, 2007.

[36] M. Korupolu, C. Plaxton, and R. Rajaraman. Analysis of a Local Search Heuristic for Facility Location Problems. *J. Algorithms*, 37:146–188, 2000.

[37] M. Korupolu, A. Singh, and B. Bamba. Coupled Placement in Modern Data Centers. In *IPDPS*, 2009.

[38] P. Krishnan, D. Raz, and Y. Shavitt. The Cache Location Problem. *IEEE/ACM Trans. Networking*, 8(5), 2000.

[39] 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 *IMC*, 2009.

[40] C. Labovitz, S. Lekel-Johnson, D. McPherson, J. Oberheide, and F. Jahanian. Internet Inter-Domain Traffic. In *SIGCOMM*, 2010.

[41] N. Laoutaris, P. Rodriguez, and L. Massoulie. ECHOS: Edge Capacity Hosting Overlays of Nano Data Centers. *ACM CCR*, 38(1), 2008.

[42] N. Laoutaris, G. Smaragdakis, K. Oikonomou, I. Stavrakakis, and A. Bestavros. Distributed Placement of Service Facilities in Large-Scale Networks. In *INFOCOM*, 2007.

[43] T. Leighton. Improving Performance on the Internet. *CACM*, 2009.

[44] A. Li, X. Yang, S. Kandula, and M. Zhang. CloudCmp: Comparing Public Cloud Providers. In *IMC*, 2010.

[45] H. H. Liu, Y. Wang, Y. Yang, H. Wang, and C. Tian. Optimizing Cost and Performance for Content Multihoming. In *SIGCOMM*, 2012.

[46] H. Madhyastha, J. C. McCullough, G. Porter, R. Kapoor, S. Savage, A. C. Snoeren, and A. Vahdat. scc: Cluster Storage Provisioning Informed by Application Characteristics and SLAs. In *FAST*, 2012.

[47] G. Maier, A. Feldmann, V. Paxson, and M. Allman. On Dominant Characteristics of Residential Broadband Internet Traffic. In *IMC*, 2009.

[48] D. R. Morrison. Practical algorithm to retrieve information coded in alphanumeric. *J. of the ACM*, 1968.

[49] CISCO Global Visual Networking and Cloud Index. Forecast and Methodology, 2011-2016. http://www.cisco.com.

[50] E. Nygren, R. K. Sitaraman, and J. Sun. The Akamai Network: A Platform for High-performance Internet Applications. *SIGOPS Oper. Syst. Rev.*, 2010.

[51] J. S. Otto, M A. Sánchez, J. P. Rula, and F. E. Bustamante. Content Delivery and the Natural Evolution of DNS - Remote DNS Trends, Performance Issues and Alternative Solutions. In *IMC*, 2012.

[52] V. Paxson. Bro: A System for Detecting Network Intruders in Real-Time. *Com. Networks*, 1999.

[53] I. Poese, B. Frank, B. Ager, G. Smaragdakis, and A. Feldmann. Improving Content Delivery using Provider-Aided Distance Information. In *IMC*, 2010.

[54] I. Poese, B. Frank, S. Knight, N. Semmler, and G. Smaragdakis. PaDIS Emulator: An Emulator to Evaluate CDN-ISP Collaboration. In *SIGCOMM demo*, 2012.

[55] Ingmar Poese, Benjamin Frank, Georgios Smaragdakis, Steve Uhlig, Anja Feldmann, and Bruce Maggs. Enabling Content-aware Traffic Engineering. *ACM SIGCOMM CCR*, 42(5), 2012.

[56] J. Pujol, V. Erramilli, G. Siganos, X. Yang, N. Laoutaris, P. Chhabra, and P. Rodriguez. The Little Engine(s) That Could: Scaling Online Social Networks. In *SIGCOMM*, 2010.

[57] A. Qureshi, R. Weber, H. Balakrishnan, J. Guttag, and B. Maggs. Cutting the Electric Bill for Internet-scale Systems. In *SIGCOMM*, 2009.

[58] G. Schaffrath, C. Werle, P. Papadimitriou, A. Feldmann, R. Bless, A. Greenhalgh, A. Wundsam, M. Kind, O. Maennel, and L. Mathy. Network Virtualization Architecture: Proposal and Initial Prototype. In *SIGCOMM VISA*, 2009.

[59] J. Sherry, S. Hasan, C. Scott, A. Krishnamurthy, S. Ratnasamy, and V. Sekar. Making Middleboxes Someone Else's Problem: Network Processing as a Cloud Service. In *SIGCOMM*, 2012.

[60] A. Su, D. Choffnes, A. Kuzmanovic, and F. Bustamante. Drafting behind Akamai (travelocity-based detouring). In *SIGCOMM*, 2006.

[61] M. Tariq, A. Zeitoun, V. Valancius, N. Feamster, and M. Ammar. Answering What-if Deployment and Configuration Questions with Wise. In *SIGCOMM*, 2009.

[62] S. Triukose, Z. Al-Qudah, and M. Rabinovich. Content Delivery Networks: Protection or Threat? In *ESORICS*, 2009.

[63] Y. A. Wang, C. Huang, J. Li, and K. W. Ross. Estimating the Performance of Hypothetical Cloud Service Deployments: A Measurement-based Approach. In *INFOCOM*, 2011.

[64] J. Whiteaker, F. Schneider, and R. Teixeira. Explaining Packet Delays under Virtualization. *ACM CCR*, 41(1), 2011.

[65] C. Wilson, H. Ballani, T. Karagiannis, and A. Rowstron. Better Never then Late: Meeting Deadlines in Datacenter Networks. In *SIGCOMM*, 2011.

[66] H. Xie, Y. R. Yang, A. Krishnamurthy, Y. G. Liu, and A. Silberschatz. P4P: Provider Portal for Applications. In *SIGCOMM*, 2008.