AN ADAPTIVE HYBRID CDN/P2P SOLUTION FOR CONTENT DELIVERY NETWORKS

*F. Bronzino, R. Gaeta, M. Grangetto**

Dipartimento di Informatica
Università di Torino
Corso Svizzera 185, 10149, Italy

G. Pau

Computer Science Department
University of California Los Angeles
Boelter Hall, Los Angeles, 90095 CA, USA

ABSTRACT

Streaming services have grown rapidly in the last few years and providers of video on-demand, such as Netflix or YouTube, are increasing the number of users even more quickly. The majority of these companies implement their services using huge Content Delivery Networks that are as much powerful as expensive, e.g. Amazon and Akamai. In this paper we propose a hybrid CDN/P2P solution that aims at reducing the infrastructural costs exploiting local caching and P2P while guaranteeing an optimal quality of service. The proposed architecture uses a classic CDN complemented by a geographically distributed layer where P2P can be activated exploiting network, content awareness and locality. The performance of the proposed solution is evaluated by means of a prototype implementation that has been deployed using the PlanetLab network and the Amazon AWS cloud services. Our findings show that the proposed approach provides adaptive, flexible, scalable and content centric service to the end users while significantly reducing the infrastructural costs.

Index Terms— Content delivery networks, Peer-to-peer, Video delivery

1. INTRODUCTION

Content Delivery Networks (CDNs) have emerged in the last decade as one of the most reliable and utilized solutions to serve contents to end users with high availability and high performance. The main purpose of a CDN is to distribute contents over a set of web servers highly distributed over the world, so as to guarantee a reliable, scalable and efficient delivery of the contents to the end users. In most cases locality is exploited to serve the user from the closest cache.

According to [1] most of the CDN traffic will involve video data. In the short term (2/5 years), it is expected that throughput requirements for a single video event will reach roughly 50 to 100 Tbps (today's record has been set by President Obama's inauguration in 2009, with Akamai serving over 7 million simultaneous streams and an overall traffic level surpassing 2 Tbps). As a consequence CDN providers and many research initiatives have started proposing new approaches to tackle the raising request for bandwidth.

The most promising approaches are trying to bridge the success and reliability of CDNs with the scalability and cost effectiveness of peer-to-peer (P2P) protocols, where the end users contribute their own upload capacity thus reducing the load on the CDN servers.

In this paper we present a hybrid CDN/P2P video content delivery system designed within the COAST European project [2]. In particular we target a Video on Demand (VoD) application on planetary scale. The most important added value of the proposed work with respect to the state of the art is the actual deployment of the prototyped solution in a very realistic scenario exploiting both the Planetlab [3] infrastructure and Amazon cloud services. The proposed approach has allowed us not only to measure the performance of the system in terms of quality of service both also to get an estimate of the running costs based on actual Amazon billing policies.

1.1. Related work

As already mentioned the idea of coupling CDN and P2P is well known in the literature. Nonetheless most of the presented works are either simulative studies or closed commercial solutions. In the following we briefly recall some of the most recent works related to ours.

In [4] and [5] the multi-layered CDN/P2P LiveSky design is proposed. LiveSky tries to offset the shortcomings of P2P systems by leveraging on CDN redirections to make P2P transfers "network-friendly". LiveSky has been now commercially deployed by ChinaCache and during peak number of clients (145,000) for one event the architecture showed promising results, offering users half startup delay compared to pure P2P (15 seconds against 30 seconds) with CDN nodes contributing only the 58.6% of the system upload capacity. In [6] the potential benefit of supplementing a CDN with P2P assistance is estimated using traffic traces collected from MSN Video and Windows Update. In [7] a solution that uses CDN servers to limit the radius of the delivery graph of a P2P video streaming application is proposed. Their ns-2 simulations shown that the playback latency can be reduced by control-

*This work has been partially funded by the European Commission under the project ICT-248036/COAST (http://www.coast-fp7.eu/).

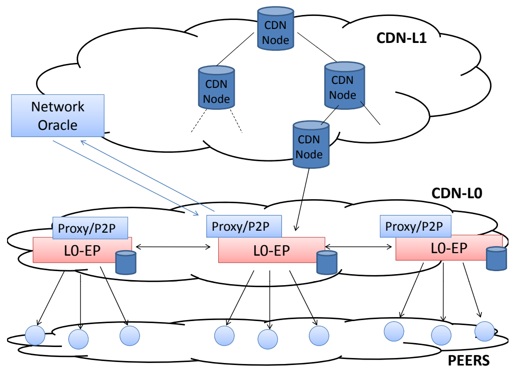

Fig. 1. Proposed hybrid P2P-CDN architecture.

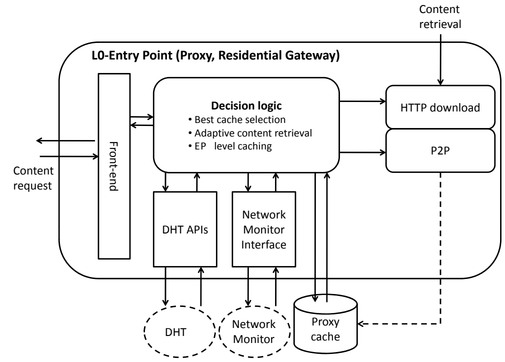

Fig. 2. L0-EP service functionalities in the proposed architecture.

ling the distance from the CDN servers. An idea for managing the playing buffer for video on demand has been proposed in [8]. In this simulative study the playout buffer is split into two sections: the CDN priority part and the P2P priority part. Missing packets closest to the playout deadline tend to be downloaded from CDN servers, whereas the remaining ones are retrieved by P2P. In [9] a novel framework for cloud-based peer-assisted CDN solutions is described. The content server (inside the cloud) is able to adjust the off-cloud bandwidth it contributes to the swarm of P2P (the set of clients outside the cloud) so as to achieve a specific objective based on a feedback signal related to the state of the swarm.

2. SYSTEM ARCHITECTURE

The platform analyzed in this work has been integrated with the architectural design proposed with the COAST EU project [2] targeting seamless, user centric, context and network aware content search and delivery.

2.1. Two-tier topology

A two-tier hybrid infrastructure is proposed, targeting the delivery of high quality and popular video contents. In such architecture, two levels of CDN overlays have been defined. The first level (CDN-L1) is composed by high capacity servers strategically placed in order to increase network backbone capacity. A second level of specific network nodes (CDN-L0) are intended to be managed by ISP operators, equipped with some caching functionalities, and placed geographically closer to end users, as depicted in Fig. 1. Such nodes are named Level-0 Entry Point (L0-EP) and can be co-located with Residential Gateways, LAN proxies or cellular base stations. The goal of the L0-EP is to react to the content requests issued by users so as to locate and fetch the content in an optimal way exploiting all the caching resources, namely its local cache, the CDN-L1 caches and other L0-EP caches in the CDN-L0 layer. The different caching levels are exploited to offer an adaptive, flexible, scalable and content centric service to the users: the L0-EP local cache is used to serve multiple local requests for the same content; the CDN-L0 is used when the content is not yet cached locally but is available in the neighborhood of L0-EP nodes, e.g. it resides in the same autonomous system of the L0-EP; finally, standard content retrieval from CDN-L1 nodes, which are responsible for geographic distribution at the global level, applies in all the other cases.

The L0-EP component with its most important interactions is graphically depicted in Fig. 2. The L0-EP includes a standard HTTP proxy frontend used to optimally redirect the HTTP GET requests of the users to the best content location in CDN-L0 or CDN-L1. The smartness of the L0-EP resides in its decision logic module that is able to aggregate all the available information and select the best cache and content retrieval strategy. The final decision is taken based on the information retrieved from:

1. the availability of the content in the L0-EP local cache;
2. the list of cache nodes (CDN-L0) that can contribute to retrieve the content. This information can be stored in a Distributed Hash Table (DHT), such as Kademlia [10];
3. the locality awareness provided by the interaction with a Network Oracle entity, offering monitoring functionalities such as those available with ALTO initiative [11].

The state of the CDN-L0 and CDN-L1 caches is stored in the DHT. The DHT is used to keep an updated record for each content in a scalable and robust fashion. Each content record is indexed by a proper ID and stores the list of caches that can be used to retrieve such content.

To summarize the L0-EP decision logic module is in charge of exploiting all information available to optimally drive the content download if this latter has not yet been stored in the local cache. The decision logic then activates one of the two available content retrieval paths, i.e. standard progressive download from a single cache or the P2P application.

2.2. DHT information

The information about the status of local caches of each node is stored in the DHT system implemented at CDN-L0. For each content cached, the L0-EP is responsible of storing in the DHT the following tuple:

$$\langle C_{ID}, N\_Addr, S, I_{bits}, A_{bits} \rangle$$

where C_{ID} is the content ID, N_Addr is the address of the node, S is one bit representing the status of the node (seeder or leecher), I_{bits} are 11 bits representing the number of consecutive chunks owned by the node, A_{bits} are 11 bits representing the total number of chunks owned by the node.

2.3. CDN/P2P decision logic

Once a L0-EP node receives a request for content C_{ID} by one end user, it starts the decision process.

If C_{ID} has been already (or is being) cached locally the LO-EP begins the delivery from its cache. Otherwise the DHT is queried for the content and returned tuples are analyzed to infer the status of the CDN. If the number of retrieved tuples $N(C_{ID})$ is inferior to a threshold Th_1, L0-EP starts the retrieval from the CDN-T1, since the content availability in CDN-L0 looks limited. The tuples are then filtered by the network Oracle to select among them only the nodes belonging to the node's neighborhood, e.g. nodes belonging to the same autonomous system. If the size of the returned local tuples $L(C_{ID})$ turns out to be inferior to a threshold Th_2 the node starts the retrieval from the CDN-T1 as in previous case. On the contrary, if $L(C_{ID}) > Th_3$ L0-EP starts the P2P retrieval. In this latter case many nodes in the LO-EP neighborhood are caching and sharing the content, thus we avoid requesting a new copy from L1 servers.

In the remaining cases, i.e. $Th_2 \leq L(C_{ID}) \leq Th_3$, a heuristic is used to predict the efficiency of the P2P protocol. To this end the third quartile $Q_3(C_{ID})$ of the number of cached chunks in the local list of EP-L0 nodes is evaluated using the A_{bits} field. Then, P2P is activated only if $Q_3(C_{ID}) > Th_4$.

3. THE TESTBED

The proposed hybrid CDN/P2P architecture has been validated and its performance evaluated using a very realistic testbed designed on top of two network services, namely the academic PlanetLab [3] platform and the commercial Amazon Web Services [12].

PlanetLab is a global overlay network for developing and accessing broad-coverage network services, with 1118 nodes at 539 sites. Thanks to such highly distributed nature Planetlab has been used to emulate the distribution of the L0-EP machines. It is well known that PlanetLab nodes selection is crucial to achieve reliable and statistically significant results. In all the following experiments the L0-EPs have been selected among the Planetlab nodes that were able to i) download a content from a certain web server, ii) generate a random 20 Megabytes file and store it in memory, iii) upload a content to a web server, within a time limit of 60 s. Adopting these criteria we have selected 121 machines in Europe, 134 in the US and 42 in Asia.

Amazon Web Services offer information technology infrastructures in the form of web services, or as now commonly known, cloud computing. Within our testbed we exploited the Amazon Elastic Compute Cloud (EC2), that is a web service providing scalable compute capacity in the cloud, to implement the CDN-L1 tier. Indeed, EC2 provides a highly reliable and scalable architecture that is used by business companies, e.g. Netflix. EC2 has been selected for three main reasons, namely locality, pricing and reliability. EC2 provides the ability to run linux and windows instances in multiple locations. In particular, for our testbed we have used 4 micro-instances in the US (Northern Virginia and Oregon), 2 in Ireland and 2 in Singapore. Amazon EC2 pricing is defined as pay only for what you use according to the rates reported in Tab. 1. In our context, pricing is based on two main factors: instances usage time and out data transfer. The possibility to calculate the price of the service has been a fundamental factor for the design of our testbed environment allowing us to determine the real running costs of the proposed architecture. While PlanetLab offers a wide range of different machines all over the world, consistency can be easily affected by numerous factors, such as the presence of various running experiments during a test session. On the other hand, Amazon EC2 offers consistent CPU and bandwidth resources with an uptime guaranteed of 99,95% over the entire year. We have evaluated this property fundamental to replicate a real CDN system.

3.1. Tier-1

CDN-L1 has been implemented as a pool of distributed web severs using the Amazon EC2 instances. All video contents available in the system are assumed to be replicated over all the nodes in Tier-1. In other words we are implicitly assuming an optimal replica placement algorithm in CDN-L1. Each L0-EP node knows the hostname and the location of the closest CDN-L1 server; in a real deployed system this simplification would be efficiently substituted by a DNS mechanism as e.g., Akamai.

The video content delivery from CDN-L1 has been implement using HTTP progressive download. CDN-L1 servers offer a HTTP interface to L0-EP nodes. Each video content is

Table 1. Data transfer (outward) costs in US dollars per GB for Amazon EC2 .

	US East	**US West**	**Irland**	**Singapore**
First 1 GB/month	free	free	free	free
Up to 10 TB/month	0.120	0.120	0.120	0.190
Next 40 TB/month	0.090	0.090	0.090	0.150
Next 100 TB/month	0.70	0.070	0.070	0.130
Next 350 TB/month	0.50	0.050	0.050	0.120

divided into a number of chunks. The L0-EP node requesting the video starts downloading the first chunk from the closest server and gets the size (in number of chunks) of the entire content embedded in the HTTP header. Then the remaining chunks are retrieved using ad-hoc HTTP GET requests.

3.2. Tier-2

In CDN-L1 a P2P strategy is activated for content downloading according to the heuristic presented in Sect. 2. To this end a P2P algorithm has been implemented among the L0-EP nodes instantiated on the Planetlab machines. For homogeneity with the CDN-L1 tier each L0-EP exhibits the same progressive HTTP interface enabled to receive HTTP GET requests for a given content and chunk number. Each L0-EP retrieves the identity of the peers in its neighborhood using the information stored in the DHT. Such information is constituted by the list of L0-EPs that have cached (even partially) the content. Every L0-EP in the list is characterized by the following data: IP/port address, geographical distance and the field representing the cached chunks (I_{bits} and A_{bits}). The geographical distance is computed using PlanetLab APIs providing geographical coordinates of each EP-L0. This list is kept ordered for increasing distance values and a CDN-L1 server is also included to guarantee full content availability. The P2P algorithm aims at retrieving chunks sequentially starting from the first one. For each chunk, the L0-EP tries to find the closest peer in the list that has already cached the chunk. Parallel download of multiple chunks is allowed only from distinct peers and the number of chunks downloaded in parallel is determined by the system parameter N_p. To keep the status of the CDN-L0 caches updated each L0-EP periodically sends a HTTP HEAD request to all its neighboring peers; in response to this request the updated values of I_{bits} and A_{bits} are communicated. The update takes place with a period T_b that is another protocol parameter. Finally, we assume unlimited L0-EP cache sizes so that all the downloaded contents can be stored locally for future requests.

As for the DHT implementation we selected Kademlia as its properties have already been tested in various popular Internet application, e.g. Azureous and Vuze. In order to utilize Kademlia in our two tiered system we substituted pairs $\langle key, value \rangle$ with the previously introduced tuples $\langle C_{ID}, N\_Addr, S, I_{bits}, A_{bits} \rangle$ where the C_{ID} is a 160 bit key generated using the SHA-1 hash function. A DHT node has been integrated in each L0-EP to take advantage of their distributed nature.

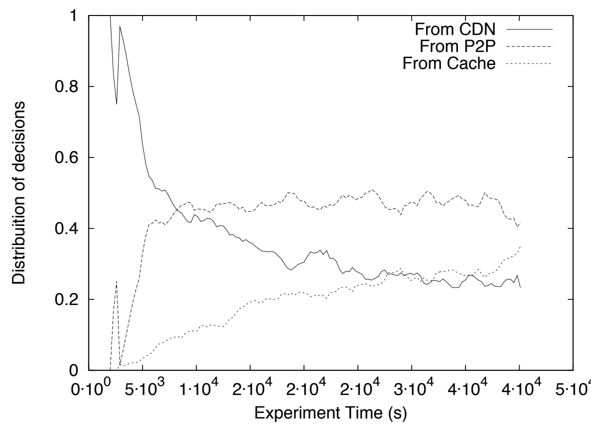

Fig. 3. Decision Logic distribution among CDN-L1, P2P and local cache.

4. EXPERIMENTAL CAMPAIGN

The designed testbed has been deployed and tested emulating the users' inter-arrival times at the L0-EP nodes using the exponential distribution with parameter $\lambda = 0.0016$. The CDN-L1 has been populated with all the video contents replicated in every cache. The total number of contents has been fixed to $N_v = 10000$ and content popularity has been modeled with a Zipf distribution with slope $\alpha = 1$ [13]. All the L0-EP nodes start with their caches empty. The parameters of the decision logic and P2P protocol used in all the following experiments are reported in Tab. 2. Experimental trials with a time duration between six and twelve hours have been worked out.

The testbed has allowed as to simulate and compare three different system configurations, namely the proposed hybrid CDN architecture with and without activation of the P2P protocol and a standard CDN approach where the users download the contents directly from CDN-L1 servers.

The performance has been measured in terms of the average time for the retrieval of the first chunk t_{first}, that is representative of the average start-up delay experienced by the final user. We have measured the average total download time t_{all}, as well. Moreover, the billing costs of the CDN-L1

Table 2. Experimental settings.

Description	Parameter	Value
User inter-arrival (Exp)	λ	$0.016\ s^{-1}$
Num. of contents	N_v	10000
Requests' distribution (Zipf)	α	1
Min. CDN-L0 size for C_{ID}	Th_1	25
Min. local P2P swarm size	Th_2	5
Max. local P2P swarm size	Th_3	25
3^{rd} quartile of C_{ID} availability	Th_4	75
Num. of parallel downloads	N_p	6
Bitmap refresh period	T_b	$180\ s$

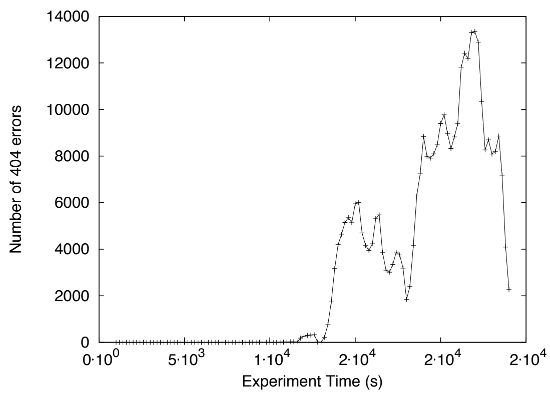

Fig. 4. Evolution of the number of HTTP errors of CDN-L1 servers.

servers have been used as an estimate of the cost due to the management of the CDN infrastructure.

Before comparing costs and performance of the various alternative solutions we have validated the behavior of the decision logic, that is responsible for the activation of P2P. In Fig. 3 the distribution of the choices taken by the decision logic is shown as function of time for a 12 hours experiment considering video contents of 50 MB each. The results show that the proposed heuristic is effective in redirecting the content requests to local nodes exploiting P2P after the transient time taken by the system to populate the L0-EP caches. Indeed after about 3 hours 50% of the content requests are served with P2P. Another hint of the fact that P2P activation is effective in offloading CDN-L1 can be obtained counting the number of HTTP 404 errors, returned by Amazon EC2 servers in case of overload. Fig. 4 shows the temporal evolution of the number of such errors in absence of P2P activation. After a certain amount of time the maximum number of requests in some HTTP servers queue exceeds its maximum limit and error 404 is triggered. On the contrary, activating P2P such errors completely disappear.

Now, let us briefly discuss the performance achieved by the proposed architecture. In Fig. 5 the average download time for the first chunk t_{first} is shown as a function of time for the proposed system (left) and the counterpart that does not use P2P. The reported results show that the hybrid CDN/P2P permits to guarantee a start-up delay inferior to 45 s whereas, without using P2P, peaks around 70 s show up. In Fig. 6 (left) the overall average download time t_{all} is reported as function of time; again it turns out that the selective activation of P2P produces a higher quality of service with a reduction of t_{all} of about 50% with respect to the architecture without P2P. The same conclusion can be drawn observing the cumulative distributions of t_{all} in Fig. 6 (right).

Table 3. Costs in USD for pure CDN, CDN with local caches and proposed hybrid CDN-P2P solution for video content of 5 MB (a) and 50 MB (b).

Exp.	Hybrid CDN-P2P	CDN w. caching	CDN only
(a)	5.9$	12.3$	14.8$
(b)	14.6$	29$	36.5$

Finally, we move to the most interesting part of our experimental campaign, i.e. the comparison among the CDN management costs yielded by the three simulated architectures: pure CDN, CDN with L0-EP caching, CDN with L0-EP caching and selective P2P activation. Tab. 3 shows the Amazon billing cost for two sets of experiments where the size of the video content is fixed to 5 MB (a) and 50 MB (b), respectively. It can be noted that the proposed hybrid architecture yields savings of about 50% with respect to the case with L0 caches only and of about 60% with respect to a classical CDN setting. This results, combined with the good performance guaranteed by the selective activation of P2P demonstrate that the proposed system can be configured to achieve both higher quality of service and much lower operational costs.

5. CONCLUSION

We have proposed a novel hybrid CDN/P2P design that aims at reducing the infrastructural costs of VoD applications exploiting local caching and P2P while guaranteeing optimal quality of service in terms of start-up delay and download time. Our findings, obtained in a accurate experimental setup based on Amamzon cloud services and Planetlab, confirm that the CDN and P2P technologies complement each other allowing a cost reduction of about 60%.

6. REFERENCES

[1] E. Nygren, R.K. Sitaraman, and J. Sun, "The akamai network: A platform for high-performance internet applications," *ACM SIGOPS Operating Systems Review*, vol. 44, no. 3, pp. 2–19, 2010.

[2] "Coast EU funded project," http://www.coast-fp7.eu.

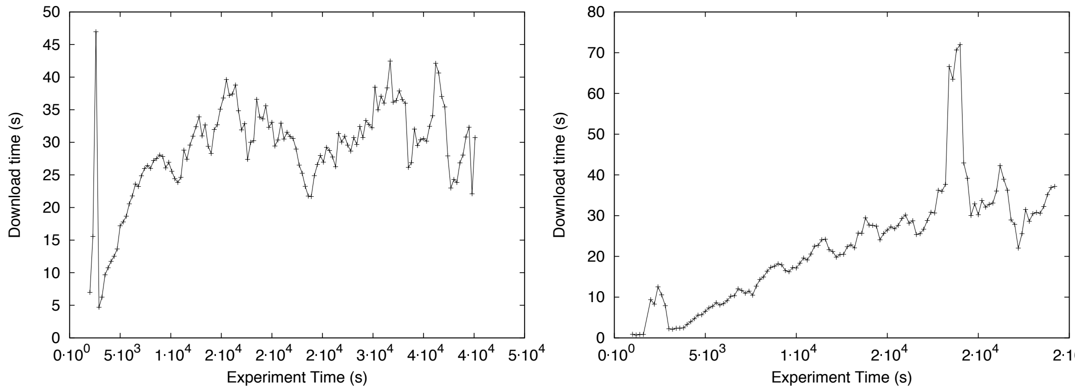

Fig. 5. t_{first} for contents of 50 Megabytes with (left) and without (right) P2P.

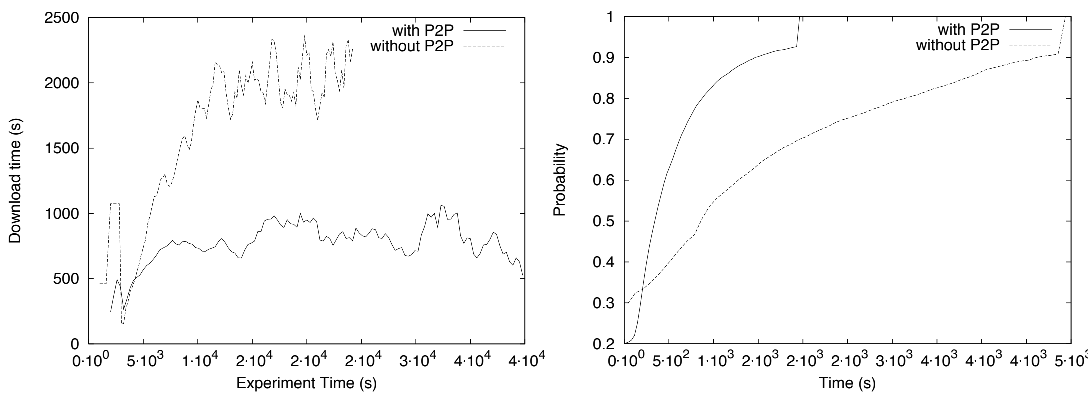

Fig. 6. t_{all} for contents of 50 Megabytes with and without P2P as a function of time (left) and cumulative distributions (right).

[3] "Planetlab europe," http://planet-lab.eu.

[4] S. Kang and H. Yin, "A hybrid cdn-p2p system for video-on-demand," in *Future Networks, 2010. ICFN'10. Second International Conference on*. IEEE, 2010, pp. 309–313.

[5] H. Yin, X. Liu, T. Zhan, V. Sekar, F. Qiu, C. Lin, H. Zhang, and B. Li, "Design and deployment of a hybrid cdn-p2p system for live video streaming: experiences with livesky," in *Proceedings of the 17th ACM international conference on Multimedia*. ACM, 2009, pp. 25–34.

[6] C. Huang, A. Wang, J. Li, and K.W. Ross, "Understanding hybrid cdn-p2p: why limelight needs its own red swoosh," in *Proceedings of the 18th International Workshop on Network and Operating Systems Support for Digital Audio and Video*. ACM, 2008, pp. 75–80.

[7] S. Cho, J. Cho, and S.J. Shin, "Playback latency reduction for internet live video services in cdn-p2p hybrid architecture," in *Communications (ICC), 2010 IEEE International Conference on*. IEEE, 2010, pp. 1–5.

[8] D.H. Ha, T. Silverton, and O. Fourmaux, "A novel hybrid cdn-p2p mechanism for effective real-time media streaming," M.S. thesis, Universite Pierre et Marie Curie, 2008.

[9] F. Albanese, D. Carra, P. Michiardi, and A. Bestavros, "Cloud-based content distribution on a budget," 2010.

[10] P. Maymounkov and D. Mazieres, "Kademlia: A Peer-to-Peer Information System Based on the XOR Metric," in *Proceedings of the 1st International Workshop on Peer-to-Peer Systems*, 2002.

[11] R. Alimi, R. Penno, and Y. Yang, "ALTO protocol," in *Internet Engineering Task Force, Internet Draft draft-ietf-alto-protocol-03*, 2010.

[12] "Amazon web services," http://aws.amazon.com.

[13] R. Bolla, R. Gaeta, A. Magnetto, M. Sciuto, and M. Sereno, "A measurement study supporting p2p file-sharing community models," *Computer Networks*, vol. 53, no. 4, pp. 485–500, 2009.