

Understanding Hybrid CDN-P2P: Why Limelight Needs its Own Red Swoosh*

Cheng Huang
Microsoft Research
Redmond, WA 98052

Angela Wang
Polytechnic University
Brooklyn, NY 11201

Jin Li
Microsoft Research
Redmond, WA 98052

Keith W. Ross
Polytechnic University
Brooklyn, NY 11201

ABSTRACT

In this paper, we quantify the potential gains of hybrid CDN-P2P for two of the leading CDN companies, Akamai and Limelight. We first develop novel measurement methodology for mapping the topologies of CDN networks. We then consider ISP-friendly P2P distribution schemes which work in conjunction with the CDNs to localize traffic within regions of ISPs. To evaluate these schemes, we use two recent, real-world traces: a video-on-demand trace and a large-scale software update trace. We find that hybrid CDN-P2P can significantly reduce the cost of content distribution, even when peer sharing is localized within ISPs and further localized within regions of ISPs. We conclude that hybrid CDN-P2P distribution can economically satisfy the exponential growth of Internet video content without placing an unacceptable burden on regional ISPs.

1. INTRODUCTION

With servers deployed throughout the Internet, Content Distribution Network (CDN) companies are currently aggressively marketing themselves as video distribution companies. CDN nodes are deployed in multiple locations, often over multiple backbones and ISPs, and often in multiple POPs within different ISPs. By providing a shared distribution infrastructure for content companies, CDNs can provide reliable delivery and cost-effective scaling. There are as many as 28 commercial Content Distribution Networks (CDN) [1] and a number of non-commercial ones (e.g. [2,3]). The competition among them for the lucrative video distribution market is fierce.

The design of CDNs broadly follows two different philosophies. One philosophy is to *enter deep into ISPs*, by deploying content distribution servers inside ISP POPs. The idea is to get close to end users, so as to improve user-perceived performance in terms of both delay and throughput. Such a design results in a large number of server clusters scattered around the globe. Because of this highly distributed design,

the tasks of maintaining and managing the networks become very challenging. It also involves sophisticated algorithms to shuffle data among the servers across the public Internet. A leading representative commercial CDN of this type is the Akamai network.

The other design philosophy is to *bring ISPs to home*, by building large content distribution centers at only a few *key* locations and connecting these centers using private high speed connections. Instead of getting into the ISP's POPs, these CDNs typically place each distribution center at a location that is near POPs for multiple large ISPs (for example, within a few miles of both AT&T and Verizon POPs in a major city). Compared to the first philosophy, such a design typically results in lower maintenance and management overhead, *possibly* at the expense of higher delay to end users. A leading representative commercial CDN of this type is the Limelight network.

In addition, Peer-to-Peer (P2P) has recently become a popular alternative to CDN to cope with the growing demand of the end users. For example, PPLive, a P2P-based video startup in China, has shown that it can use 10 Mbps server distribution bandwidth to simultaneously serve 1.48 million users at a total consumption rate of 592 Gbps [4]. In such P2P video streaming systems, the users watching the video bring with them bandwidth, CPU capacity, memory, and disk cache resources. Therefore, the resources available in the system scale with the number of users.

Given the two different distribution approaches - CDNs and P2P - a natural question is whether they can be combined to obtain the scalability advantage of P2P, as well as the reliability and manageability advantages of CDNs. Indeed, with recent rapid growth of P2P applications and CDNs, many initiatives have been taken to combine the two to get the "best of both worlds". Akamai, with its acquisition of Red Swoosh P2P technology, is expected to combine P2P file distribution software with its back-end control system and global network of edge servers. VeriSign, CacheLogic, Grid Networks, Internap, and Joost have all announced their own CDN-P2P services as well. Using P2P to extend CDNs has been suggested by many researchers, e.g., [5-9]. However, all the previous work simply treat CDNs as *virtual* resource pools, instead of *concrete* distributed systems. Hence, they fail to consider the implications of hybrid CDN-P2P solutions on the inner networks of ISPs. Furthermore, to our knowledge, there is no published study quantifying the potential performance and economic gains of hybrid CDN-P2P solutions.

In this paper, using both real-world video and software-

*Red Swoosh is a P2P startup acquired by Akamai in 2005.

update traces, we quantify the potential gains of hybrid CDN-P2P for two of the leading CDN companies, Akamai and Limelight. We first determine the locations of the servers of these two companies, and then consider ISP-friendly peer-assisted distribution schemes from the CDN servers. Our contributions and key findings are as follows:

- We present a comprehensive measurement methodology for discovering the IP addresses and locations of all the servers of any given CDN. The methodology exploits thousands of geographically-distributed recursive DNS servers as well as the PlanetLab platform for distributed and parallel execution. In particular, we determine the topologies of the Akamai and Limelight CDNs.
- We then consider ISP-friendly P2P distribution schemes which work in conjunction with the CDNs to localize traffic within regions of ISPs. To evaluate these schemes, we employ two recent, real-world traces: a July 2007 Internet video trace and an August 2007 software update trace.
- We find that hybrid CDN-P2P can significantly reduce the cost of content distribution, even when peer sharing is localized within ISPs and further localized within regions of ISPs.

The rest of the papers is organized as follows. We discover the topology and operation of Akamai CDN network in Sec. 2 and the Limelight CDN network in Sec. 3. The potential of hybrid CDN-P2P is evaluated in Sec. 4 and we conclude in Sec. 5.

2. UNDERSTANDING CDN – THE AKAMAI NETWORK

In this section, we describe a novel method for discovering the topology of CDNs. We focus our study on the two leading CDNs – the Akamai network and the Limelight network.

2.1 How does Akamai work?

Akamai uses “DNS magic” to connect end users to its “edge servers” that are *close* to them. Using the example in [10], say Akamai hosts a content collection for PCWorld (say `images.pcworld.com`). In this case, Akamai provides PCWorld with an *Akamai hostname* – `a1694.g.akamai.net`. When an end user visits PCWorld’s website and downloads an image (say `http://images.pcworld.com/iphone.jpg`), the host name `images.pcworld.com` will be resolved in two stages: 1) through the public DNS infrastructure, authoritative name servers of PCWorld will first be contacted, which return a CNAME (canonical name) `a1694.g.akamai.net` to the query of `images.pcworld.com`; 2) a secondary DNS query will then be issued to resolve `a1694.g.akamai.net`, which is sent to authoritative name servers belonging to Akamai. From that point on, the query enters into Akamai’s private DNS infrastructure. Akamai first resolves `g.akamai.net` to an Akamai POP that has short latency to the end user. Then, the resolution of `a1694.g.akamai.net` returns the IP addresses of a set of edge servers hosting PCWorld’s content. (Akamai normally returns two IP addresses to allow client side load balancing.)

2.2 Charting the Akamai network

Our CDN-charting methodology is based on the following two key observations: 1) Akamai returns IP addresses of its edge servers based on where a query is originated; and 2) Akamai returns different edge servers to the same query over time due to load balancing. To chart the Akamai network, conceptually, we first find out what Akamai hostnames are used by different Akamai customers. Then, we install measurement clients all over the Internet. Finally, we orchestrate these clients to issue, at different times of the day (as well as over different days), DNS queries for all the Akamai hostnames to obtain the IP addresses of all the Akamai servers.

2.2.1 Finding Akamai hostnames

We obtained a collection of web hosts from the Windows Live Search production system, which consists of over 16M unique web hosts. Given an arbitrary web host, a simple DNS query tells us whether it resolves to a CNAME or an IP address, and whether the CNAME resolves to a Akamai hostname. After straightforward DNS resolutions, we found that there are 3,260 unique Akamai hostnames, which can be classified into 3 types:

	akamai.net	akadns.net	akamaiedge.net
# of hostnames	1964	757	539

Interestingly, these three types of hostnames appear to offer very different services, which we will elaborate in a later section. Note that this step involves only the public DNS infrastructure.

2.2.2 Locating distributed vantage points

For each Akamai hostname, we’d like to query the Akamai private DNS infrastructure from different geographic locations, so that each query returns a different set of edge servers. For this, one natural option is to issue such queries from PlanetLab nodes. Using hundreds of PlanetLab nodes, it is possible to obtain a reasonable collection of Akamai edge servers. However, PlanetLab is very US and European centric. Its nodes are also mainly located within education networks. Therefore, using PlanetLab nodes as vantage points might not provide good coverage of the Akamai network. Note that we could compensate for this by repeating experiments over time. Nevertheless, these limitations make PlanetLab nodes as vantage points far from ideal.

Instead, we explore the public DNS infrastructure. It is known that there are a large number of DNS servers on the Internet, which will respond to DNS queries from remote clients. For instance, say a client A in Seattle sends a DNS query to a DNS server B in New York. If B is configured to enable *open recursive* DNS queries, it will resolve the query on behalf of A and send results back, even if client A is not in server B’s authority at all. These DNS servers, called *open recursive DNS servers*, have been investigated in King [11] and related work for Internet delay measurement. Here, we explore open recursive DNS servers differently, in a way in fact even closer to the original intention of these servers – for DNS queries. Now, imagine that A sends to B not a regular DNS query, but a query for an Akamai hostname (again, say `a1694.g.akamai.net`). When B resolves this Akamai hostname, it will get results back from Akamai’s private DNS infrastructure. From Akamai’s perspective, the request comes from B, so it will return edge servers close to B (that is, New York here). By now, it should become

clear that, given geographically distributed open recursive name servers, we can issue queries from any single place and discover Akamai edge servers close to arbitrary locations.

Indeed, using a large collection of open recursive DNS servers as vantage points, we thus can have very good coverage of the Akamai network. Finding open recursive DNS servers is a well-understood process. Here, we start with two sets of source data. The first set consists of clients extracted from MSN Video traces. Using one month’s worth of trace, we obtain over 7M unique client IP addresses. Through reverse DNS mapping, we find the authoritative name servers for these clients. Then, we send trial recursive DNS queries to these authoritative name servers and record all those that respond. The second set is the 16M+ web hosts, used in the previous step. Again, we find the authoritative name servers for these hostnames and record those responding to remote DNS queries. The results are summarized as follows:

	clients (7M+)	web hosts (16M+)
# of authoritative NS	83,002	1,161,439
# of open recursive NS	26,453	440,054

2.2.3 Large-scale discovery – the PlanetLab way

The approach just described seems straightforward at first. All we need is to query all the Akamai hostnames (found in the first step) against all the recursive DNS servers (found in the second step). However, we want to issue approximately 1.5 billion ($\sim(26453 + 440054) \times 3260$) DNS requests, and measurements at such scale are very likely to be classified as DNS attacks. Even if we only choose 25,000 ($\sim 5\%$) of the vantage points, there are still more than 81 million DNS queries, which is definitely too much to be completed in a reasonable amount of time from any single place. To this end, we have developed a distributed execution platform, which splits the complete giant task into many smaller jobs, spreads these jobs onto PlanetLab nodes and executes on each node. A rough calculation shows that if we use 300 PlanetLab nodes and send out 3 DNS queries per second from each node (a very low frequency, thereby consuming very modest resources), the entire task takes slightly more than one day to complete. Of course, in practice, it takes some extra time, as jobs on a PL node can fail or be terminated unexpectedly, and PL nodes themselves can be rebooted. Nevertheless, using PlanetLab greatly speeds up the task and also makes it manageable. Moreover, most of the time, we were able to use more than 300 nodes in parallel.

2.3 Our discoveries of the Akamai network

In this section, we report our findings.

2.3.1 Service types

It appears that the 3 types of Akamai hostnames are for 3 different services.

- type (a) – *.akamai.net (1964 in total). This type matches the conventional understanding of Akamai, as a content distribution network: 1) Each DNS resolution returns 2 or more IP addresses; 2) When resolved from different locations, the obtained IP addresses are different; 3) For each Akamai hostname, we obtain hundreds (or even thousands) of unique IP addresses by aggregating from all the vantage points. Altogether, we’ve discovered over 11,500 unique IP addresses belonging to this type.

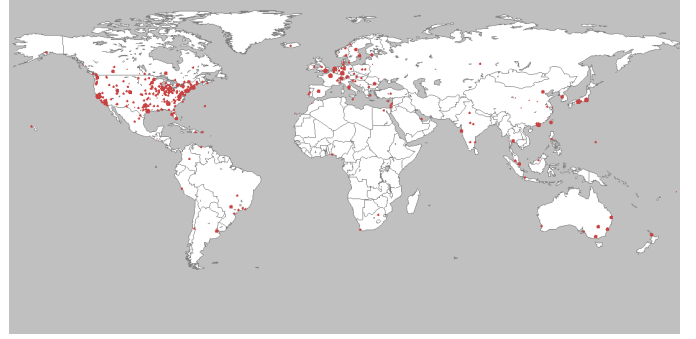


Figure 1: Worldwide Edge Server Locations of the Akamai Network

- type (b) – *.akadns.net (757 in total). This type appears to be using Akamai only for global load balancing [12], not for content distribution. As a typical example, disasteraid.fema.gov.akadns.net maps only to 3 geographically distributed IP addresses (combined from all the vantage points).
- type (c) – *.akamaiedge.net (539 in total). This is a type that appears to be mysterious, based on following observations. First, every query only returns one IP address (no matter resolved from which vantage point). Second, picking a typical Akamai hostname (say e128.b.akamaiedge.net), we obtain only 74 unique IP addresses in total. This is quite different from hundreds (or thousands) of IP addresses seen in type (a), and also different from only a few in type (b). Third, this type has more than 36,000 unique IP addresses (many more than the total 25,000 servers Akamai claims to have deployed!). Fourth, these 36,000+ IP addresses are almost completely disjoint from those for type (a) (or akamai.net). Finally, these IP addresses only appear in less than 30 autonomous systems (ASes), compared to about 650+ for akamai.net.

Combining additional findings beyond the scope of this paper, we conjecture that Akamai is using virtualization technology behind all the IP addresses for akamaiedge.net. Furthermore, Akamai is providing virtualization environments to customers for dynamic content distribution. It appears that virtualization is available only in limited locations among all the Akamai deployments.

To this end, we’ve discovered over 11,500 unique Akamai edge servers belonging to type (a), which Akamai dedicates to provide conventional content distribution services. We’ve also discovered more than 36,000 (*conjectured*) virtual instances, which Akamai dedicates to type (c) service. In this paper, we focus on the Akamai network providing the conventional content distribution service, i.e., all the edge servers belonging to type (a).

2.3.2 Geographic and ISP distributions

For all the edge servers in type (a), we obtain their geographic information (through a commercial IP to geolocation database), including city, state, country, latitude, longitude, as well as autonomous system (AS) number (through ASFinder [13]). It turns out the 11,500+ edge servers cover 69 countries, while more than half of which are located in

US. All the locations are shown in the map – Figure 1.

We also show the distribution in the top 10 countries containing the most servers, as in Figure 2. For the edge servers in US, we report those in the top 10 ISPs [14]. It turns out that only about 12% are in the top 10 ISPs, and there are a large number of edge servers in other ISPs (Figure 2). This should not be surprising, as Akamai *does* claim presence in a large number of ISPs.

Country	# of IP	Percent (%)
United States	6,661	57.7
Japan	865	7.5
United Kingdom	704	6.1
Germany	545	4.7
Netherlands	384	3.3
France	364	3.2
Canada	284	2.5
Australia	164	1.4
Hong Kong SAR	158	1.4
South Korea	124	1.1
Others	1285	11.1

ISP	# of IP	Percent (%)
Qwest	408	6.13
AT&T	157	2.36
Verizon	126	1.89
Road Runner	53	0.80
America Online	15	0.23
Charter	3	0.05
Cablevision	2	0.03
Others	5897	88.53

Figure 2: Geographic (Global) and ISP (US) Distributions of the Akamai Network.

3. UNDERSTANDING CDN – THE LIMELIGHT NETWORK

3.1 How does Limelight work?

We observe that Limelight uses a similar DNS redirection technique as Akamai to map web hosts to Limelight’s servers in one of its data centers. For instance, a DNS query of `www.shrek3.com` is first resolved to a Limelight hostname – `drmwks.vo.llnwd.net` through the public DNS infrastructure, and then resolved to specific IP addresses through Limelight’s private DNS infrastructure.

Limelight differs from Akamai in that it also lets customers directly use Limelight hostnames on their websites. For example, downloading movies from Amazon Unbox reveals that Amazon directly embeds Limelight *sub-hostnames* (e.g., `amazon-936.vo.llnwd.net`) in their web pages.

Moreover, we observe that Limelight adopts a very regular naming convention for their sub-hostnames. In the case of Amazon Unbox, it appears that most Limelight sub-hostnames in the format of `amazon-xxx.vo.llnwd.net` are valid. Based on this observation, we do *not* need to crawl websites extensively to obtain most Limelight sub-hostnames. Additionally, Limelight’s business model also appears to be different from Akamai. It does *not* serve as many customers, but instead serves a few major customers with huge volume of traffic [15]. Therefore, besides the Limelight hostnames found in resolving the 16M+ web hosts, we also manually checked out all of Limelight’s major customers (only a few dozens, based on its press releases between 2005 and 2007). It turns out that all Limelight sub-hostnames are discovered in this way (not through web host resolutions).

3.2 Our discoveries of the Limelight network

We again deploy our distributed execution platform onto PlanetLab nodes to discover the Limelight network. In summary, we discovered 3,168 unique Limelight servers, which are located in 17 locations over 7 countries worldwide (again, using the same commercial IP to geolocation database). Figure 3 shows the details of our findings.

4. HOW MUCH CAN P2P HELP?

City	# of Svrs	City	Country	# of Svrs
Washington, D.C. (DC)	418	All Cities	United States	2213
Los Angeles (CA-LA)	381	Amsterdam	Netherlands	372
New York (NY)	373	Frankfurt	Germany	158
Chicago (IL)	314	Tokyo	Japan	129
San Jose (CA-SJ)	311	London	United Kingdom	118
Dallas (TX)	165	Paris	France	91
Atlanta (GA)	100	Hong Kong	China	47
Miami (FL)	98	Sydney	Australia	39
Seattle (WA)	52	Total		3618
Phoenix (AZ)	1			
Total	2213			

Figure 3: Geographic Distributions of the Limelight Network (US and Global).

In this section, using traces collected from MSN Video and Windows Update, we demonstrate the potential benefit of supplementing a CDN with peer assistance. In doing so, we assume that such a solution will be highly ISP-friendly, because win-win relationships with ISPs are critical to all CDNs. In particular, we believe that the a viable hybrid CDN-P2P solutions will confine the P2P traffic within ISP boundaries so that peers only share with other peers within the same ISP. In addition, when there are multiple points of presences (POPs) within any single ISP, the hybrid CDN-P2P will further localize the P2P traffic within the network dictated by the POP.

We choose to study the benefit of hybrid CDN-P2P within two of the top 10 US ISPs – AT&T and Verizon. Using our CDN-charting methodology, we have discovered that Akamai is present in 7 locations in AT&T and in 4 locations in Verizon. We then classified all US clients into geographic regions centered at these Akamai POP locations based on the geographic distance between the client and the Akamai POP. We acknowledge that this classification is an approximation, and that a more accurate classification should follow the topology of the physical AT&T and Verizon networks, and use network hop counts or network latency as the distance measure. But given the fact that there are not too many regional centers and the layout of the physical networks within each ISP should reflect the physical geography, we believe this is a reasonable approximation. For the Limelight network, it has 10 locations in US. Each of the location is shown in both POP maps of AT&T and Verizon [17]. Therefore, we assume Limelight is peering with both networks at each of the locations, and all clients are classified to one of the 10 regions.

4.1 CDN-P2P: Internet video-on-demand

AT&T	download	768	1500	3000	6000
	upload	128	384	512	768
	percent (%)	15.76	34.78	31.58	17.89

Table 1: Bandwidth Breakdown of AT&T Users (U.S., in Kbps, down – measured, up – inferred)

We report our study on the service of Internet video-on-demand. We use traces of MSN Video collected during the entire month of July, 2007. For pure CDN solutions, we assume all the clients in one region get data solely from Akamai’s edge servers present at that region inside AT&T. For CDN-P2P solutions, we assume clients can get data from both Akamai’s edge servers, as well as other clients, with the latter more preferable. We assume that the peers only share with the others in the same CDN service region. In such way,

the peers assist the CDN, but do not create any additional cross region traffic compared with a pure-CDN solution. We infer the clients' upload bandwidths from their download bandwidths, which were measured by serving Windows Media Servers when they requested videos. Based on the high speed Internet packages offered by AT&T, the inferred upload bandwidths are shown in Table 1. We carry out simulations for both Akamai and Limelight.

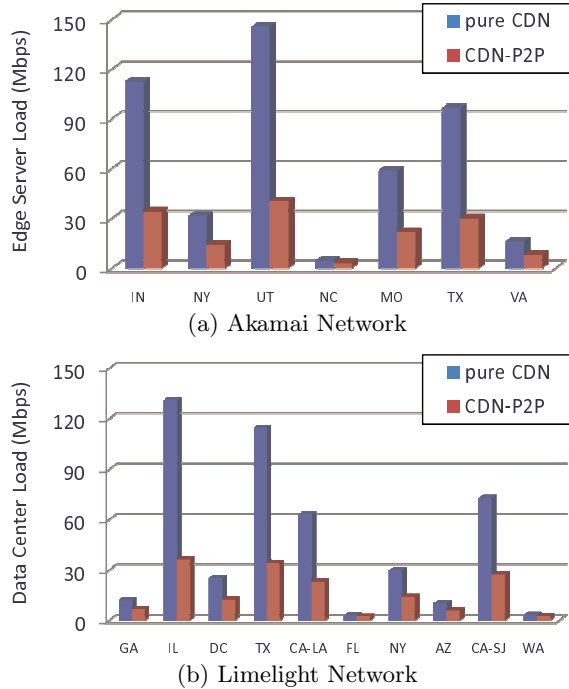


Figure 4: CDN-P2P for Internet Video Distribution.

There are in total 1.25 million unique clients from AT&T, watching over 17,000 videos that month. The load of the pure CDN approach is the total traffic incurred on the Akamai POPs (or Limelight data centers) for all clients served. To calculate the load of the hybrid CDN-P2P approach, we apply the model developed in [18]. In particular, we conservatively assume that a peer only uploads a video when it is watching the same video. The results from each location are shown in Figure 4 and summarized in Table 2.

In short, we observe that the hybrid CDN-P2P solutions can potentially reduce the load on both the Akamai edge servers and the Limelight data centers by about 2/3. Although these gains are quite significant, they would be even higher if we allowed peers to share previously-viewed videos from local caches. It should be no surprise that the benefit of P2P to the Limelight network appears to be slightly lower, which is due to the fact that the same number of clients are classified into more (10 vs. 7) separate sharing regions, and thus the P2P efficiency is affected slightly. Note that the impact of P2P on the network resource utilization is different for the two CDNs, which we didn't compare here. Furthermore, due to the fact that the clients are segmented into service regions differently, the results here should *not* be interpreted as a *complete* head-to-head comparison between the two CDNs.

	pure CDN	CDN-P2P	savings (%)
Akamai	459.5	147.6	66.53
Limelight	459.5	158.3	65.55

Table 2: Benefits of P2P to Akamai and Limelight in Internet Video-on-Demand (unit in Mbps)

4.2 CDN-P2P: Large-scale software distribution

Using a Windows Update trace, we now examine the benefits of hybrid CDN-P2P for a large-scale software distribution. We collected traces from a complete circle of Windows Update between Monday, Aug. 13th and Wednesday Aug. 15th, 2007. Due to the gigantic volume of Windows Update, we only sampled about 0.012% of the entire client populations, which already had more than 34GB of raw log data. The clients sampled belong predominantly to Verizon, which we focused on in the study. Note that the sampling skips a large fraction of the clients, which would otherwise be classified into the same sharing region, and thus is expected to result in a very conservative estimation of the benefit of P2P. Nevertheless, it still sheds light on the potentials of P2P to aid CDNs in large-scale software distributions.

Verizon	download	768	3000 (and up)
	upload	128	768
	percent (%)	31.85	68.15

Table 3: Bandwidth Breakdown of Verizon Users (U.S., in Kbps)

The Windows Update traces do *not* contain client bandwidth information. Hence, we apply the distribution collected about Verizon clients from the MSN Video trace. Given the large population of the MSN Video clients originated from Verizon, it is reasonable to assume that the bandwidth distribution is representative, as shown in Table 3.

Again, we assign all the clients to one of four Akamai POPs, or one of 10 Limelight POPs in Verizon. To calculate the CDN server load with peer assistance, we assume clients contribute their bandwidths, as long as they are on-line downloading updates or patches. Note that we do *not* explicitly consider the content availability at the clients, because Windows Update is unique in 1) there are a very large amount of clients (1.57 million) downloading a much smaller amount of updates and patches (2617 in total); and 2) every client needs to be updated, so there is no long tail content. Hence, it is reasonable to assume that, at any given time, individual clients do have something others can download and thus can fully utilize their upload bandwidths. We compare pure CDN, CDN with *full* P2P-assistance (clients contribute all their entire upload bandwidths), and CDN with *limited* P2P-assistance (clients only contribute 1/3 of their upload bandwidths).

We show the results aggregated across all the locations in Figure 5 (also summarized in Table 4). It is exciting to see that the CDN bandwidth reduction can be more than 95% when the peer upload resources are *fully* utilized (even when all the P2P traffic is confined to local regions). Also, the benefit is much higher than in the Internet video-on-demand scenario. This should be easy to understand, as

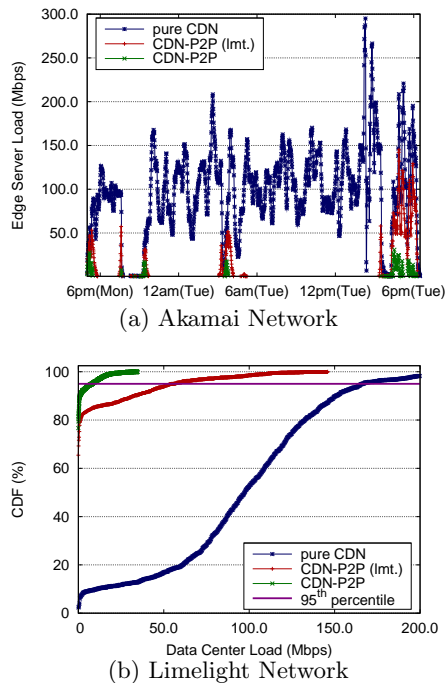


Figure 5: CDN-P2P for Large Scale Software Distribution.

roughly the same amount of clients (1+ million) are activated in a two-day period in Windows Update, compared to an entire month in Internet video-on-demand. Furthermore, even with *limited* peer contributions (again, use only 1/3 of clients' upload bandwidths), the savings can still be as high as 2/3. This implies that hybrid CDN-P2P solutions can be deployed at a level with minimum intrusiveness and still be *very* effective.

Very careful readers might notice that, even for the pure CDN case, the variation of load during the Windows Update cycle (shown in Figure 5(a)) appears rather flat. The absence of significant spikes is because the Windows Update system is well-engineered so that clients automatically back-off to avoid overwhelming the distribution servers. Additionally, in the CDN-P2P case, the eventual load on the edge servers fluctuates much more prominently than the pure CDN case. This is again due to the Windows Update traffic engineering. The contribution from P2P can be much higher when there are more clients in the system and each is downloading at a lower rate, compared to less clients while each is downloading at a relatively higher rate. In other words, even when the total demand is about the same, the size of the client population (i.e., assisting peers) eventually determines the CDN load.

Akamai	pure CDN	CDN-P2P	CDN-P2P (lmt.)
	166.7	7.89	56.3
	savings (%)	95.3	66.2
Limelight	pure CDN	CDN-P2P	CDN-P2P (lmt.)
	166.7	7.84	55.8
	savings (%)	95.3	66.5

Table 4: Benefits of P2P to Akamai and Limelight in Large-Scale Software Distribution (unit in Mbps)

5. CONCLUSION

Using a novel methodology to chart CDNs, we have evaluated the potential savings for hybrid CDN-P2P for two major CDN companies. We observed that hybrid CDN-P2P can be great beneficial to CDNs. In particular, we saw that (1) it can cut the server load for both Akamai and Limelight by more than 2/3 in Internet video-on-demand; and (2) even if peers only upload at limited rate (1/3 of their upload bandwidths), the potential savings can still be as high as 2/3 in large-scale software distribution. We believe that hybrid CDN-P2P will emerge as the dominate distribution architecture for high-quality video in the Internet (both live and on-demand), as well as for large volumes of static content (e.g., software updates, etc.).

In the future, it will be worthwhile investigating the additional gains when limited traffic are allowed among ISP POPs and even peering ISPs, which will *not* be a far stretch with the adoption of hybrid CDN-P2P and ISPs embracing such technology. Additionally, it will be important to inspect into ISPs' topology deeper than the POP-level, where clients can potentially share within an much smaller region. Understanding the impact of hybrid CDN-P2P on ISPs' last miles and quantifying the trade-off between additional locality and overall savings should be very valuable.

6. REFERENCES

- [1] D. Rayburn, "CDN Market Getting Crowded: Now Tracking 28 Providers In The Industry," Sep. 24th, 2007. http://blog.streamingmedia.com/the_business_of_online_v/-2007/09/cdn-market-gets.html.
- [2] M. J. Freedman, E. Freudenthal, and D. Mazières, "Democratizing Content Publication with Coral," *USENIX NSDI*, Mar. 2004.
- [3] L. Wang, K. Park, R. Pang, V. S. Pai, and L. Peterson, "Reliability and Security in the CoDeeN Content Distribution Network," *USENIX Annual Technical Conference*, Jun. 2004.
- [4] G. Huang, "Experiences with PPLive," *Keynote at ACM SIGCOMM P2P-TV*, Aug. 2007.
- [5] T. Karagiannis, P. Rodriguez, and K. Papagiannaki, "Should Internet Service Providers Fear Peer-Assisted Content Distribution," *ACM IMC*, 2005.
- [6] D. Pakkala, and J. Latvakoski, "Towards a Peer-to-Peer Extended Content Delivery Network," *14th IST Mobile & Wireless Communications Summit*, Dresden, Jun. 2005.
- [7] P. Rodriguez, S.-M. Tan, and C. Gkantsidis, "On the Feasibility of Commercial, Legal P2P content Distribution," *ACM SIGCOMM CCR*, 36(1), 2006.
- [8] S. Saroiu, K. P. Gummadi, R. J. Dunn, S. D. Gribble, and H. M. Levy, "An Analysis of Internet Content Delivery Systems," *USENIX OSDI*, 2002.
- [9] D. Xu, H. K. Chai, C. Rosenberg, and S. Kulkarni, "Analysis of a Hybrid Architecture for Cost-Effective Streaming Media Distribution," *ACM MMN*, Jan. 2003.
- [10] A.-J. Su, D. Choffnes, A. Kuzmanovic, and F. Bustamante, "Drafting Behind Akamai (Travelocity-Based Detouring)," *ACM SIGCOMM*, Pisa, Italy, Sep. 2006.
- [11] K. P. Gummadi, S. Saroiu, and S. D. Gribble, "King: Estimating Latency between Arbitrary Internet End Hosts," *ACM SIGCOMM IMW*, Marseille, France, Nov. 2002.
- [12] "Akamai Global Traffic Management," <http://www.akamai.com/html/technology/products/gtm.html>.
- [13] CAIDA. CoralReef suite. <http://www.caida.org/tools/measurement/coralreef>.
- [14] "Top 21 U.S. ISPs by Subscriber: Q2 2007," <http://www.isp-planet.com/research/rankings/usa.html>.
- [15] "Limelight SEC Filings," Limelight Networks, Inc. May 2007.
- [16] "Limelight CDN Map," http://www.limelightnetworks.com/-technology_map_cities.html.
- [17] N. Spring, R. Mahajan, D. Wetherall, and T. Anderson, "Measuring ISP Topologies with Rocketfuel," *IEEE/ACM ToN*, 12(1), Feb. 2004.
- [18] C. Huang, J. Li, and K. W. Ross, "Can Internet Video-on-Demand be Profitable?" *ACM SIGCOMM*, Aug. 2007.