CPE CACHING – USING NETWORK INEFFICIENCY TO DELIVER BETTER INTERNET STREAMING

Scot Loach
Aterlo Networks

Abstract

We are in the midst of a shift in TV viewing behavior from a traditional broadcast-based experience to an on-demand experience delivered over the Internet. At the same time, video bit-rates are increasing as 4K resolution becomes mainstream. These trends are putting pressure on the economics of providing Internet access, since networks are built to peak usage and the majority of video streaming still occurs during "prime time". Caching technologies are being used to respond to this problem, but traditional caching technologies only address the transit edge of the network. Caching does not help the access network, which is often difficult and expensive to upgrade, since it is constrained by physical properties such as limited radio spectrum.

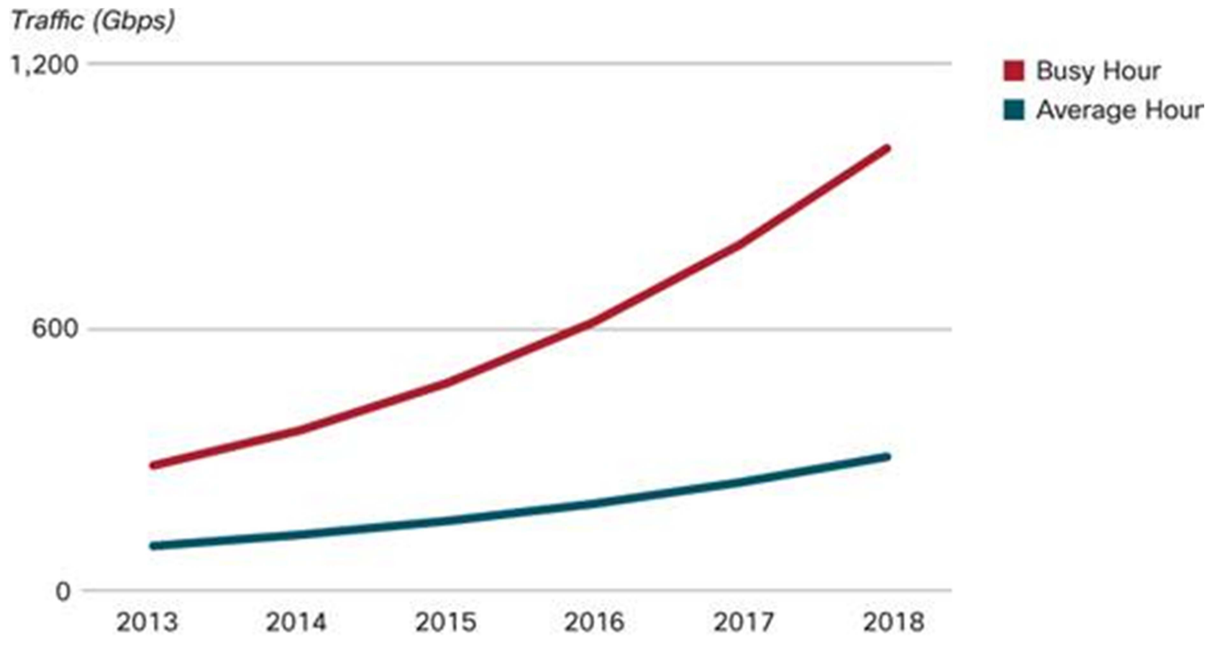
This paper presents the concept of caching within the customer premises. We discuss the opportunities and challenges associated with this approach and illustrate the architecture of such a system. We present a study that models customer premises caching on a consumer broadband network, and show the potential cache efficiency and resulting savings predicted.

INTRODUCTION

Video on Demand (VoD) over the Internet is replacing traditional linear broadcast video, as subscribers increasingly use services such as Netflix and Hulu to watch TV episodes and movies. These services have made great strides in recent years in the amount of content they offer, their ease of use, and their brand awareness with the average consumer. Many TVs natively support these services, so a consumer can simply connect a TV to their home network and use these services without any additional hardware. This is causing an upward trend in the number of subscribers using these services.

Although the technology used to watch TV is changing, the behavior of TV viewers still follows a “prime time” usage pattern, where the majority of users watch TV during a few hours in the evening. At the same time, streaming bitrates are on the rise, as more consumers adopt High Definition (HD) and Ultra High Definition (UHD, 4K) screens. HD video at 1080 lines of resolution is typically encoded at about 5Mbps, and UHD video at 2160 lines of resolution is typically encoded at 16-20Mbps. A standard definition video is encoded at about 1.5 Mbps.

This combination of increased adoption and higher bitrates is causing an exponential increase in the volume of traffic on the Internet during prime time hours. Figure 1 shows a projection by Cisco showing the increase in prime time relative to the average hour of the day. One consequence of this is an increasingly inefficient network, since the cost of building a network is proportional to its peak required capacity. Figure 2 shows a typical average day on a consumer broadband network; in coming years, the peak of this chart will move higher relative to the rest of the day. This represents a high cost of network upgrades and an increasing waste of resources, as much of the network capacity sits unused for most of the day. Consumer broadband networks are designed with oversubscription, assuming that usage is spread out over the day. The trend towards the majority of subscribers using the Internet to watch TV puts pressure on the economics of providing Internet access to consumers.



Source: Cisco VNI, 2014

Figure 1. Busy hour traffic is growing faster than average hour traffic.

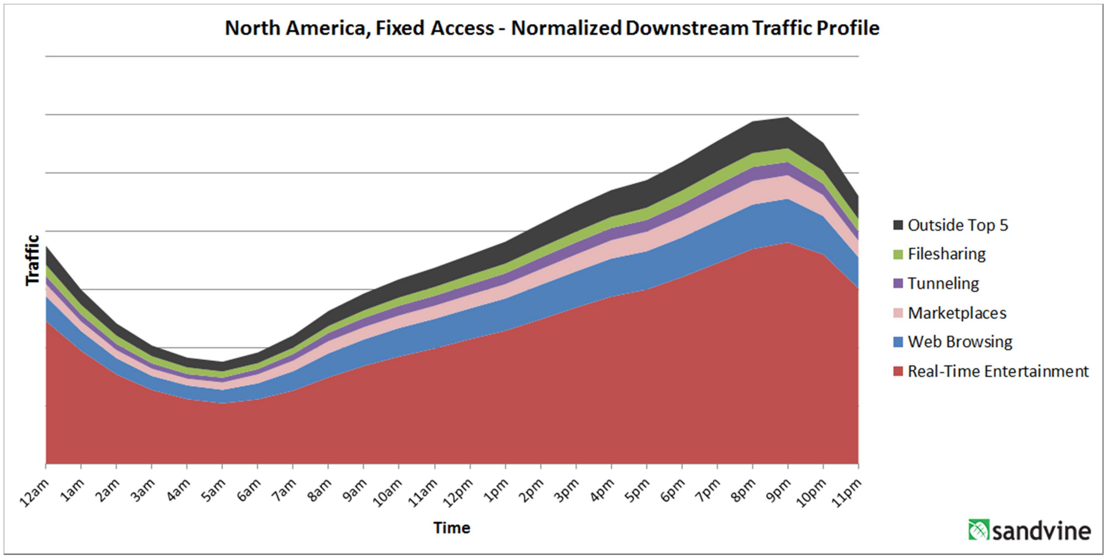


Figure 2. Average day on a consumer broadband network.

Communications Services Providers (CSPs) react to this situation with a variety of strategies. The most common strategies are increasing capacity, reducing demand, and increasing efficiency.

Increasing capacity means upgrading the network. This is a continuous business process for CSPs, and can be extremely expensive, especially on the last mile access network. The access network is typically limited by physical constraints such as wireless spectrum or physical cables in the ground; increasing capacity is a costly endeavor that takes a long lead time to plan. The planning cycle for increasing capacity is usually an annual process that is tied to the budget cycle of a CSP.

Reducing demand is usually done by putting policies in place to incent subscribers to stream less video, or to stream it during off-peak times. Subscribers may be convinced to stream less video or at a lower resolution by enforcing usage quotas on the network, and subscribers may be convinced to stream video at off-peak times by putting an un-capped time period into place, for example during the night. These policies are unpopular with subscribers and may cause churn as subscribers move to competitors with fewer restrictions.

Increasing efficiency can be done by putting optimization systems in place such as network caches. Service-specific network caches are offered by some of the major content providers such as Google and Netflix, and service-agnostic network caches are offered by several network equipment vendors such as PeerApp, as well as open-source implementations such as Squid. Network caches are appliances that sit in the network, load popular content into large storage arrays (passively or during off-peak hours), and serve the content from their storage instead of the Internet. Caches are an effective way to reduce the amount of data volume between the CSP's network and the content provider's servers; however, a cache can only improve efficiency on the part of the network that is upstream from it. Therefore traditional network caches that sit in the core network cannot improve the last mile access network, which is often the most expensive part of the network to upgrade. A cache in the customer premises could improve the efficiency of the entire network, including the access network. This would make it possible to address the bandwidth crunch caused by the increasing usage of VoD over the Internet, while keeping the capacity upgrade budget under control, all without alienating subscribers.

This paper examines the concept of a "CPE Cache" which is a network cache implemented within the customer premises. An overview of caching technology is provided, with a discussion of how this relates to CPE caching. Architectural considerations and tradeoffs for building a CPE cache are discussed. Finally, the results of a case study applying a CPE caching model to real network traffic are presented.

CONCEPT

Network Cache Overview

A network cache can be modeled as a system that transforms one traffic pattern on a network link to a different traffic pattern. The purpose of a network cache is to reduce the peak bitrate of its upstream link, and therefore the entire network upstream of it. Figure 3 shows this model of a network cache. In this paper, we assume a cache does not change the traffic pattern of its downstream link, although in reality it is possible for this to occur if a link is congested, for example by

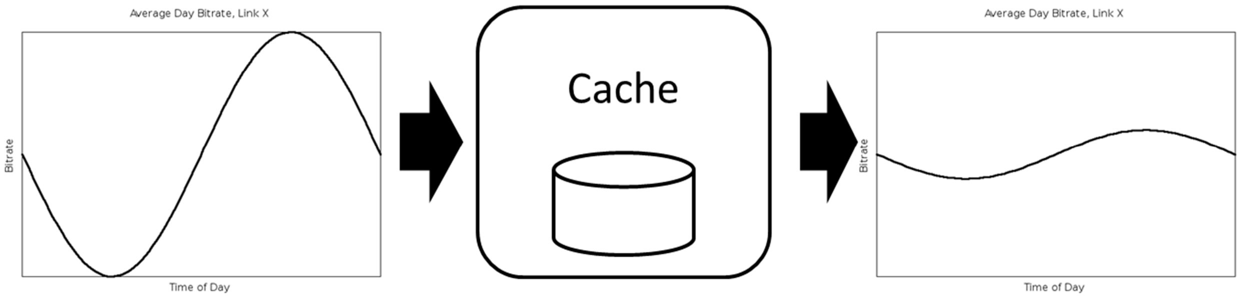


Figure 3. High-level model of a network cache

adaptive video streams shifting to a higher bitrate when congestion is relieved by caching.

The peak bitrate of a link is defined in terms of how it is measured, and the interval it is measured over. This may depend on physical and financial considerations. For example, a transit link that is billed monthly based on 95th percentile downstream bitrate may use the 95th percentile downstream bitrate as the peak, and measure this at 5 minute intervals over a calendar month. For a last-mile network segment such as a DOCSIS channel that is a capital expense, the link efficiency may be monitored quarterly as part of a quarterly reporting package, and again the 95th percentile may be used as a way to eliminate outliers.

The ultimate measure of a cache's effectiveness is the reduction in peak downstream bitrate on its upstream link (the network link that packets take between the cache and the Internet). A cache reduces peak bitrate by serving bytes from its storage; the number of bytes served from cache is noted B_C and the total bytes transmitted downstream from the cache (from the Internet and from cache) is noted B_T. Traditionally, a network cache is evaluated based on cache efficiency or cache hit rate, which is defined as $R_{CH}=B_C / B_T$. However, such a measure only correlates to an improvement in peak bitrate if it is measured exclusively during the peak time of network traffic. During off-peak times, there is idle network capacity that can be used to improve the effectiveness of the cache by speculatively downloading content to the cache. This activity lowers the cache hit rate during those times, since B_T is increased with no increase in B_C; but there is no added cost to doing this since the cost of the network is defined by its peak utilization. For example, the Netflix OpenConnect appliance works by downloading new content to its storage during off-peak hours. This means it can service even the first request for any stored content during peak hours. This strategy increases link efficiency and overall traffic volume in order to reduce peak bitrate.

The effectiveness of a network cache depends on its ability to have requested content in its storage. Assuming infinite storage, and technical feasibility of filling it, a cache could remove all static content from its upstream link. However, there are both physical and financial limits to storage, so caches have methods for filling the storage to optimize their effectiveness. This paper defines storage efficiency as the ratio of bytes served from the storage on the cache to the total amount of bytes stored on the cache. $R_S=B_C/B_S$. The storage capacity B_S is the number of bytes available for storing cached content.

Another consideration related to network caching is the concept of Quality of Experience (QoE). Subscribers judge quality of experience by factors such as the quality of the video encoding, the resolution of the video, and any visual impairments in the video due to network congestion or hardware capability such as dropped frames, buffering "wait" messages, and transitions between different resolutions when adaptive algorithms are operating in changing network conditions. This paper does not undertake a detailed discussion of Quality of Experience, but we assume that the quality of experience for video delivered from a cache is equal to or better than the quality of experience delivered from a remote CDN, since there is less probability of experiencing network congestion the closer the content source is to the user.

CPE Caching Technology

A traditional network cache covers some sample of the subscriber population, and is popularity-based, meaning it fills its storage with content that is most likely to be consumed by the largest number of subscribers. This works well, since most Internet content follows a power-law distribution where a small subset of the total content will be accessed by many users within a time period. For example, although a video streaming service may have 50,000 videos, in general 80% of the videos streamed in a day may be within the top 20% of the total content. Therefore a network cache could reduce the upstream bitrate of that service by 80% if it is able to store the most popular 10,000 videos.

In contrast, a customer premises equipment cache (CPE Cache) only serves a small number of subscribers - in many cases only a single subscriber. A naive approach to achieving a result equal to a traditional network cache would be to store the same 20% of popular content on every CPE Cache. Although this would achieve the same overall savings as a traditional cache, it would reduce the storage efficiency by a factor of the number of CPE Caches deployed. It would be prohibitively expensive to do this; for example, a typical traditional cache may have several terabytes of storage capacity, so replicating this into each CPE would have a high cost both in hardware and also in power and cooling. So an approach is needed that can achieve a high storage efficiency without replicating all popular content.

This paper considers two techniques for maximizing both link efficiency and storage efficiency within the customer premises: predictive and participative caching. Participative caching is when a user explicitly flags some content to be watched later, and a user interface tells the user what is available in the cache to be consumed. This approach is similar in concept to a PVR, and it can achieve a very high link efficiency and storage efficiency. The downside of this approach is that the user has to take extra steps and wait to consume content. Predictive caching uses historical data to pre-position content into the cache that is likely to be consumed. The advantages of this approach are that there is no user interaction required, and the user does not have to wait for content to be loaded into the cache before they can consume it. The disadvantages of predictive caching are that the content needs to be predictable, and since the user is not in control, the link efficiency and storage efficiency are lower than with participative caching. A hybrid approach of these techniques can be a good tradeoff between user experience and link efficiency; for example, predictive caching could be used to keep the cache filled with content that is likely to be consumed, and a user could choose to pre-select content they know they will want to consume later.

The considerations discussed above lead to a relationship between storage efficiency, storage capacity, and user participation. Increasing any of these increases cache effectiveness, and decreasing any of these reduces cache effectiveness.

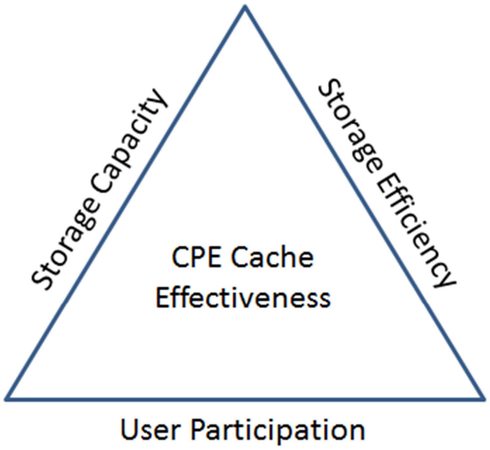


Figure 4. Storage capacity, storage efficiency and user participation.

A traditional network cache improves QoE since the video is delivered from the core network instead of going over transit or peering links to the Internet. A CPE Cache can further improve QoE, since delivery of cached content is not affected by access network congestion.

ARCHITECTURE

The CPE cache system has two major components: the CPE Cache Controller ("Controller") and the CPE Cache. Each CPE Cache is deployed in a customer premises, and is responsible for storing content and serving it to client devices on the LAN. Each CPE Cache is managed by a Controller. The Controller is responsible for providing the CPE Cache with operating instructions such as what content to store and when to download it. The Controller may also provide an interface (API) for other systems such as content providers.

Figure 5 shows how the CPE Cache system fits into the network, and how the parts of the system are interconnected. The client device is a device running on the LAN in the customer premises, such as a laptop, phone, tablet, or TV. The client device communicates with the content provider to authenticate and interact with the content provider's application, and it downloads content from a Content Provider CDN if no CPE Cache is present or if the content is not cached. If the content is cached, the client device downloads content from the CPE Cache. The CPE Cache downloads content from the Content Provider CDN to fill its storage when there is idle network capacity. Each CPE Cache also talks to its Controller. Finally, content providers may communicate with the Controller to pre-position specific content to a CPE Cache.

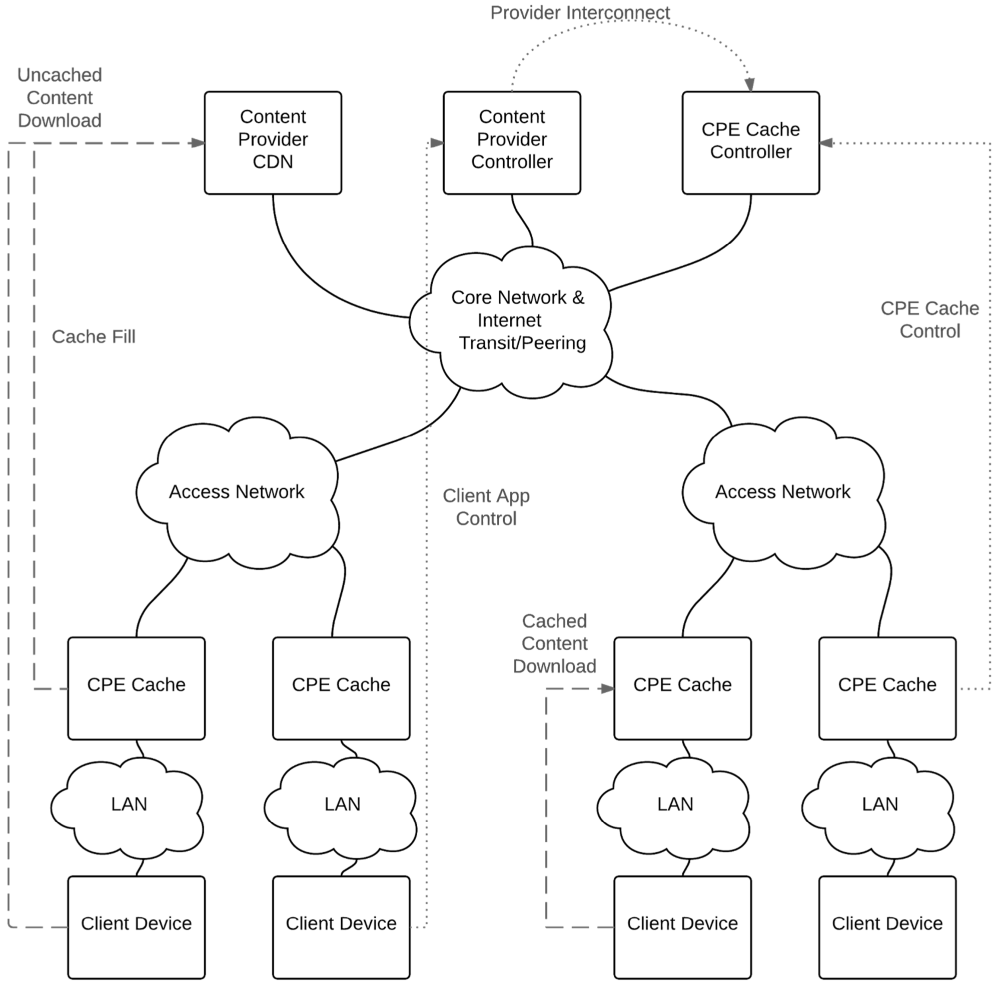


Figure 5. Network Architecture

The following sections discuss several considerations and constraints for the design of the system, where the overall goal of the system is to optimize the peak bitrate of network links on the access network.

Transparent Caching vs Provider Interconnect

Transparent caching works by inserting a transparent proxy into the path of traffic between the client device and the content provider's CDN that serves the content. The transparent proxy analyzes network protocols used to download content, such as HTTP, and may also understand higher-layer protocols such as adaptive streaming protocols that are used to tie several downloads into a single media stream. For example, a transparent cache that works with HTTP and the HTTP Live Streaming (HLS) protocol can intercept the HLS manifest exchange and read the metadata. When chunks of video are subsequently downloaded, the cache can act as the server, and serve the chunks from its storage instead of the Internet.

Provider Interconnect works by the cache system interoperating with the content provider. The Controller exposes an API to the content provider. This API can be queried to expose information about which users have a CPE Cache running in their premises. It exposes a control API for the content provider to push pre-positioning requests that will be delegated to the CPE Cache.

A CPE Cache may use a combination of transparent caching and provider interconnect to achieve the goal of reducing peak bitrate. The key difference between these is that with provider interconnect, the content provider directly participates in the system. Transparent caching can work without any participation from the content provider. Predictive caching, participative caching, or both may be used in either mode. The following table summarizes how content is selected and pre-positioned in each combination.

	Transparent Caching	Provider Interconnect
Predictive Caching	The Controller selects the content to pre-position.	The content provider selects the content to pre-position and programs the Controller.
Partitipative Caching	The Controller is programmed with what files to load by a separate web portal or app.	The user selects the content to pre-position using the content provider's app or web portal. The content provider programs the Controller.

To implement transparent caching, the Controller must be able to select what content to pre-position. This is different from a traditional transparent cache, such as Squid, which saves content as it is consumed and serves the same content on repeated requests. The Controller builds a schedule of content to pre-position and sends this to the CPE Cache. The CPE Cache must be in the path of traffic; for example, it may be implemented as software running on the CPE router, or as a separate device that bridges traffic between the CPE router and an Internet gateway such as a modem. The transparent proxy accepts connections (such as HTTP connections) from a client device, and analyzes the request to see if it is for content that is stored locally. If the content is in cache, it is transparently served from local storage; the client device does not know that it is being served from a cache. If the content is not in cache, the transparent proxy forwards the request to its intended destination and forwards any resulting content back to the client device.

To implement provider interconnect, an API is exposed on the cache controller. The IETF has defined such an API under its Content Delivery Networks Interconnection (Cdni) working group. In the context of this standard, the CPE Cache Controller is the downstream CDN, and the content provider is the upstream CDN. A CPE Cache doing only provider interconnect can be much simpler than the transparent caching case; it needs to have access to storage and it needs to be always on and addressable on the LAN, but there is no requirement for it to be in the direct path of traffic, since the content provider will instruct the client device to connect directly to CPE Cache.

Content Selection

The CPE Cache works by pre-positioning content that has a high probability of being consumed by a downstream client device in the near future. This can be done using participative content selection, predictive content selection, or a combination of the two. The goal of content selection is to maximize the storage efficiency of the CPE Cache, with the minimum amount of user interaction. A perfect content selection algorithm would be able to automatically predict with perfect accuracy what content any downstream client device will consume during the peak hours of the following day, so that the content could be loaded into the cache overnight when the network is idle. This section discusses some possible approaches and heuristics for automatically predicting what content is likely to be consumed in the future.

Storage efficiency can be defined by two factors: content accuracy and format accuracy. *Content accuracy* is a measure of how well the system predicts what content will be consumed in the future. For example, if the system predicts that a user will watch a particular movie from a streaming service, and the user watches that movie, that prediction was successful from the point of view of content accuracy. A specific piece of content may be encoded many times for different devices, screen sizes, and network conditions. *Format accuracy* is a measure of how well the system predicts how content will be consumed. For example, if the system pre-positions every available resolution for a predicted movie into the cache storage, it may have a poor storage efficiency even though the content accuracy was good. A user may only have a device capable of watching standard definition video, so any high-definition video loaded into the cache would have wasted significant storage that could have been used to improve content accuracy by loading a larger selection of content.

The following sections discuss some methods for predicting future videos and other forms of content.

Historical Content Analysis. One way to predict future content is by analyzing historical content. A CPE Cache that is protocol-aware is able to analyze content requests and responses from downstream client devices, and send relevant metadata to the Controller. The Controller can then do an analysis of the total sum of data from the CPE Caches to build a prediction model of content and format for each user.

For example, a popular video streaming service works by sending MP4 files over HTTP. This service is popular for “binge viewing” of TV shows. The service uses a URL format that includes a unique hash for a show as part of the URL. It encodes videos at several different bitrates to adapt to different device capabilities and different network conditions, and each of these encodings is a separate MP4 file. A protocol-aware CPE Cache sends information about the client device, the identifier of the show, and the bitrates played to the Controller every time a client device plays a show. The Controller analyzes this data and for every show, it finds the show that is most frequently played next.

This way, it can find the series order of shows without knowing what the show actually is. It can also analyze the device and bitrate data to predict what video bitrates are likely to be played. With this information, a prediction can be made for what content and formats are likely to be played in the future.

Content Awareness. If the system is content-aware, it can predict future content by comparing historical content to a categorized index of available content. For example, the historical content analysis described in the previous section may be improved on by partnering with the content provider and using their API to obtain information about each TV series, the order of episodes in the series, and the identifier of each episode. Then the Controller can predict the next show without doing an analysis of all historical data.

Premises Awareness. If the system is premises-aware, it is able to collect data about the environment within the customer premises that is relevant to predicting future content consumption. For example, a protocol-aware CPE Cache can analyze the user agent sent in requests it intercepts and determine that there is a device running Apple iOS within the premises. A CPE Cache that optimizes OS updates could use this information by pre-positioning new iOS upgrades to homes that have devices that will consume the upgrade.

Trending Content. The system can use broad trending information across a large population to predict content to pre-position. For example, if the CPE Cache network is deployed across several time zones, and a high percentage of users are consuming a new viral video during peak hours in the leading time zone, the trailing time zones can predict that this will be watched during peak time and pre-position the content in advance. As another example, if new content is being released that is known to be popular with some demographic of the population, and the Controller has this demographic knowledge about its CPE Cache users, it can pre-position this content to selected users in advance.

Deep Learning. The previous paragraphs discuss methods and heuristics for predicting content. A deep learning algorithm may be able to do similar predictions if it is trained with attributes of the historical data seen by the CPE Caches.

Storage

Storage is an important consideration for a CPE Cache. Traditional caches contain expensive, high-capacity storage that is centralized in the core network. A CPE Cache is highly distributed and usually placed in customers' homes, so a variety of different factors must be considered.

Availability and performance are important considerations. Storage must be "always on" and available at any time, and it must be able to serve data at high speeds, since a cache may be serving several concurrent sessions at high bitrates. Storage that is directly connected to the device where the CPE Cache runs will have a better and more consistent performance than network-attached storage.

Another important factor is cost. This includes both the cost of the storage device, and the cost of operating it, such as power and cooling. The CPE Cache system needs to have a good storage efficiency to optimize the storage size required, but there is a base storage capacity needed to serve a typical household. At this time, a 64GB USB drive is a good tradeoff between cost and capacity; these cost about $30 for a high-end USB drive and are powered directly by the USB port, so they draw up to 2 watts of power when fully active, and much less when idle. Since no separate power source is required, this makes them a simple option to install and operate, and in the case of failure they can be replaced without the expense of replacing a device with internal storage.

Reliability is another important factor, since many storage devices will be distributed and a high failure rate would be expensive and inconvenient. The quality of USB drives is extremely variable; the more expensive devices based on SLC or MLC memory are much more reliable and long-lasting than low-end drives based on TLC memory.

Storage Management

Storage on a CPE Cache is a limited and valuable resource. Storage management may need to trade off between the technical goals of maximizing storage efficiency, and commercial goals of different content providers wanting to maximize their content. Therefore one of the roles of the CPE Cache system is to arbitrate the storage space between different content providers, while maximizing storage efficiency.

Privacy and Encryption

One consideration for a CPE Cache is the privacy of the content individuals are watching. In the absence of any cache, a user's viewing history is stored in the systems of the content provider, and not anywhere else without the explicit permission of the user. Media files transferred over the network usually do not contain any information that can be used to map them to specific content metadata such as the title of a show. A CPE Cache system should ensure that this privacy is maintained. Any APIs to content providers should be based on obfuscated data, such as opaque IDs for shows that cannot be mapped back to a title. If a cache is running in transparent mode without a provider interconnect, predictions should be based on analysis of content without knowledge of the content metadata such as titles. If a transparent cache is interacting with a user, for example to do participative caching, then there may be a requirement to present video title information to the user in the form of a web site or app. In this case, the system must ensure that user privacy is secured by keeping user viewing history as isolated and secure as possible, by purging historical data after some period of time, and by not sharing personal data with anyone except for the user.

Much of the content that is consumed over the Internet is copyrighted content owned by some third party. This content is frequently protected by digital rights management (DRM) technology. When a client device wants to play some content, it may perform a negotiation with a DRM server to obtain a key to decrypt the protected content, use this key to play the content. This negotiation may be done over an encrypted channel. A CPE Cache that stores copyrighted content must keep the content stored in its DRM protected form, so that a user cannot play locally stored content without being authorized to play it by the content provider. This is the natural implementation of a CPE Cache, since the files served from the Internet are already DRM protected. An implication of storing DRM protected files is that a CPE Cache does not allow offline viewing of content.

Most content providers encrypt user sessions which carry authentication information, usage information such as viewing history and recommended content, and user control such as the ability to initiate streaming sessions. Most content providers deliver the DRM protected content over plain HTTP, rather than encrypting it inside a HTTPS session. This is because the content being transferred is already encrypted, there is usually a high volume of content which would use significant server CPU to encrypt, and some client devices may not be able to easily handle the added load of decrypting content in addition to playing it. Another reason not to encrypt the content sessions is to allow transparent caching, which has historically been used to improve network efficiency and reduce transit cost for CSPs.

A minority of content providers, most notably YouTube, encrypt their streaming sessions. With continued improvement in CPUs and dedicated encryption hardware, it is possible that more content will be delivered over encrypted sessions in the future.

For a CPE Cache that uses content provider interconnect, session encryption is not an obstacle, since the content provider instructs the client device to connect directly to its local CPE Cache. However, for a transparent cache, there is no standard way to operate on sessions that are encrypted end-to-end. Organizations such as the Streaming Video Alliance are working on developing standards for open caching and content delivery; these may facilitate transparent caching of such sessions in the future.

Transfer of Content

CPE Caching technology works on the basis of downloading content from a content provider to a device in the customer premises for temporary storage. The CPE Cache downloads content during off-peak hours. The Controller provides instructions to the Cache; these instructions include what content to download, when to download it, and how to download it.

One consideration for pre-positioning content is whether it should be pre-positioned using unicast or multicast technology. Regular video on demand always uses unicast technology, because subscribers are rarely watching the same content at the same time. However, by pre-positioning content, the CPE Cache system has the opportunity to take advantage of multicast and broadcast technology to push popular content to multiple CPE Caches on the same access network segment. This approach can be used to make more efficient use of idle network capacity, but it is more complex than a unicast approach, since issues such as retransmission of dropped packets must be handled by the system.

For unicast pre-positioning, the HTTP protocol is normally used to download content to the Cache. The Controller distributes URLs to the cache that can be used to download the content from a CDN.

The CDN hosting content is often geographically distributed, and for best performance the CPE Cache should download content from a nearby server. If Provider Interconnect is used, the content provider will provision a URL using the API; the content provider knows the location of the subscriber and can target the server. In the case of transparent caching, the Controller is responsible for choosing a server; this means it must have knowledge of the location of the CPE Cache and the locations of potential servers hosting the content.

Another approach is to use a combination of CPE Caching and traditional network caching in the network core, organized into a hierarchy of caches. In this case, each caching layer acts as an “upstream CDN” to the layer below it, so the upstream CDN of each CPE Cache is a network cache. This architecture has several benefits. The CPE Cache system does not need to take the location of a server into account; it obtains content directly from the network cache, and the network cache is responsible for obtaining the content from an upstream CDN. This approach makes more efficient use of transit capacity. The network cache will have standard interfaces to the CPE Cache that are agnostic of the content provider; without it, the CPE Cache may need special logic for some content providers. Furthermore, in a targeted deployment where a CPE Cache is only distributed to some sample of users, the overall population can still take advantage of increased efficiency and quality from the network cache.

STUDY

A simulation was done to evaluate the effectiveness of content selection for a transparent CPE Cache. A monitoring probe was installed for a sample of subscribers on a fixed wireless CSP in the US. The probe logged anonymous information about Netflix usage, which was post-processed by a simulated CPE Cache Controller. Netflix was chosen for this study because it represents a large percentage of traffic on most residential broadband networks, usually between 35-50% of peak traffic, and because it represents a good sample of video on demand traffic with a mix of TV shows and movies, so the results from it may be applicable to the more general case of all Internet streaming video.

Figure 6 shows the traffic distribution of Netflix over an average day. The amount of traffic varies significantly over the day, with its peak between 6-10pm. This 6-10pm period is used for peak time data for the remainder of the study.

Figure 7 shows the relationship between the time of day and the average bitrate of an average Netflix stream. This chart shows that average stream bitrates are lowest during the peak time of day; this suggests that the network may be congested during this time. Also, this network experiences lower speeds than the average US network; according to the Netflix ISP Speed Index, the average Netflix stream in the US is 3Mbps but in the study it is under 1.5Mbps. This may be a result of usage policies and/or technology limitations at this particular CSP, but this indicates that the content provider's end users on this network are getting a less than desired quality of experience. A CPE caching system would alleviate the congestion, which would provide a better experience to subscribers during peak hours. It would also improve the overall average bitrate, since speed limitations in the network can be overcome by pre-positioning files before they are played. This would help both the CSP and the content provider ensure subscriber satisfaction, and both of these could use CPE Caching to realize a competitive advantage.

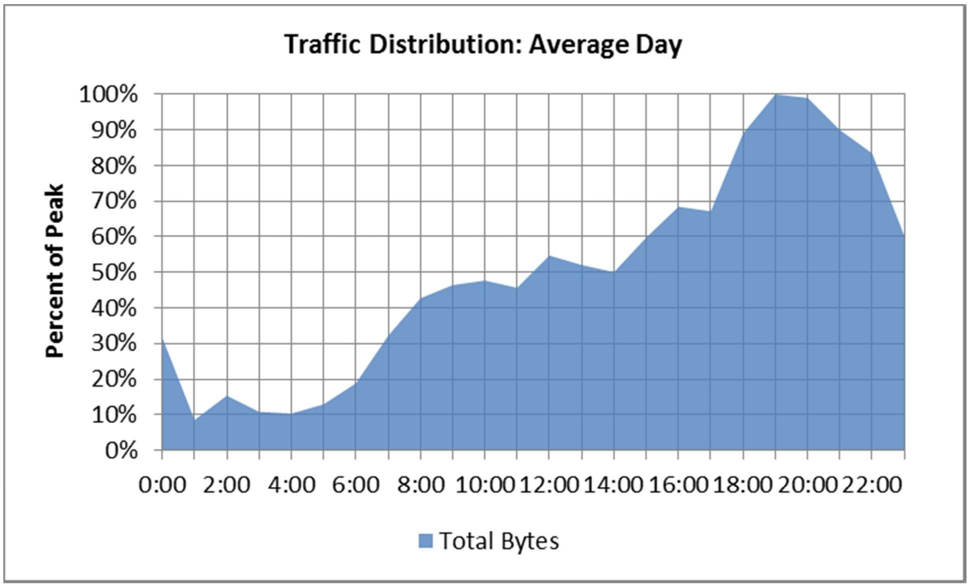


Figure 6. Netflix traffic distribution over an average day of the study

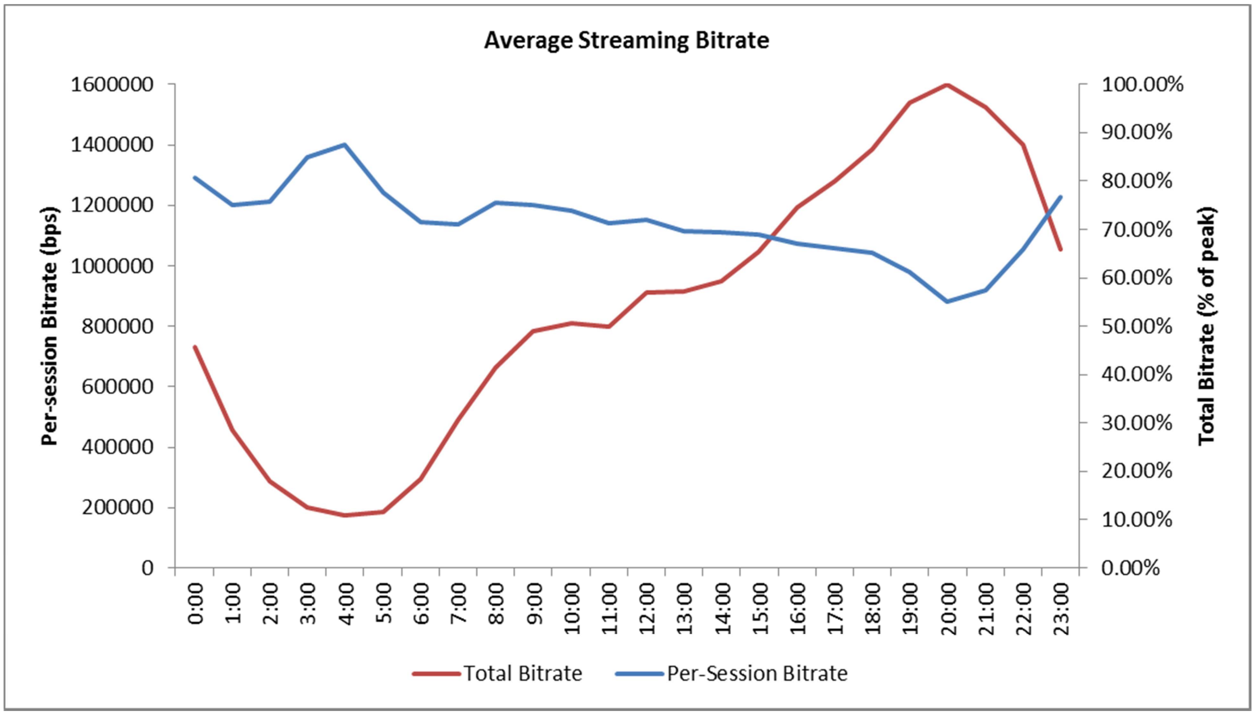


Figure 7. Streaming bitrates over an average day of the study

Figure 8 shows the resulting hit rates during peak time from the simulation, run with four different storage capacity settings. With the content selection algorithms being used, the maximum hit rate achieved is about 47%. The storage capacity has a significant impact on the hit rate until 64GB of capacity; after that, additional storage capacity does not significantly change the cache efficiency. The content selection algorithm used in this study is based on historical content analysis, and uses premises awareness to optimize storage based on the capability of devices in the customer premises. Since this network has a lower bitrate than the US average, this result may not translate well to other networks; a higher storage capacity may be needed to achieve the same savings on a network where higher-bitrate video is the norm.

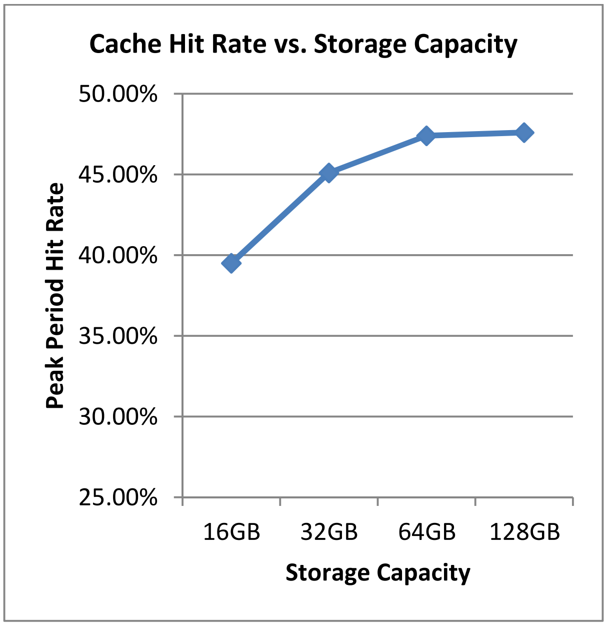


Figure 8. Hit Rate vs. Storage Capacity

Figures 9 and 10 show the results of the simulation for an average day, using a 64GB storage capacity. The peak network utilization by Netflix with CPE caching is 47% lower than the peak network utilization without CPE caching. This study did not examine the effect of downloading content into storage; that would add traffic during the lowest-usage hours of the day, and would make the traffic graph much flatter over the average day.

Since Netflix is about 35% of peak traffic on the average consumer broadband network, implementing CPE Caching for Netflix as shown in this study would save about 16% of peak traffic. According to Sandvine, overall streaming video traffic is 67.53% of peak downstream traffic in North America, so extrapolating the results of this study to all video content implies a potential savings of about 1/3rd of all peak downstream traffic.

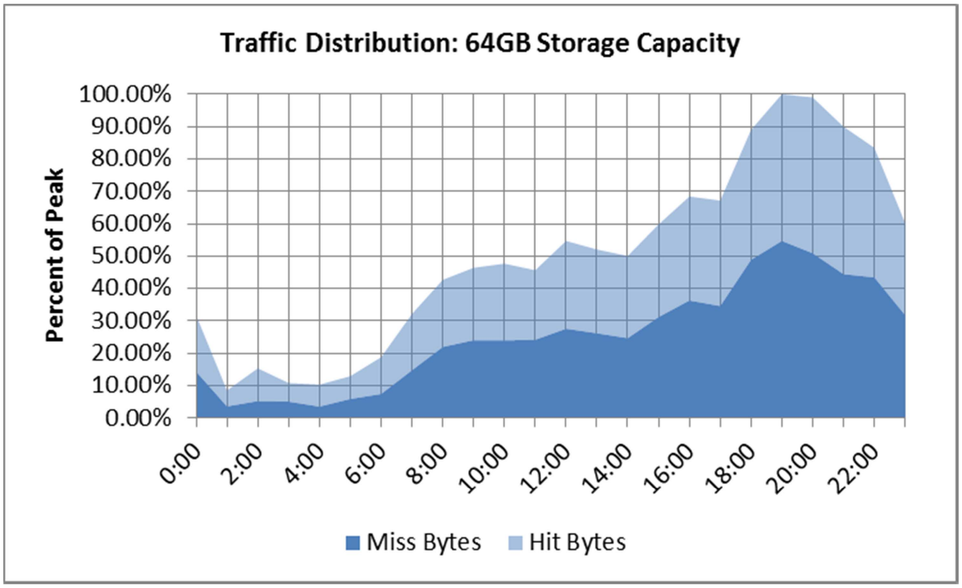


Figure 9. Average Traffic Distribution with 64GB Storage Capacity

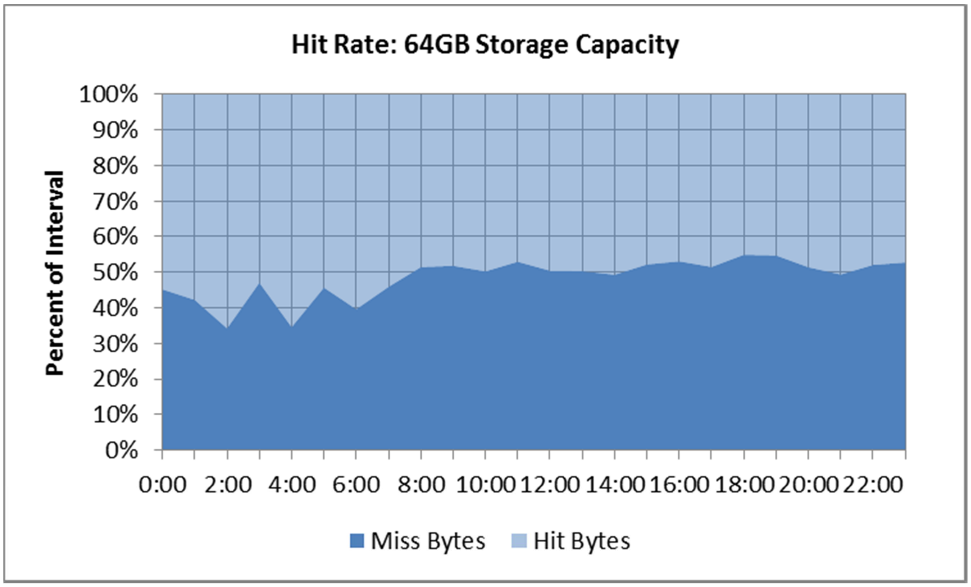


Figure 10. Average Hit Rate with 64GB Storage Capacity

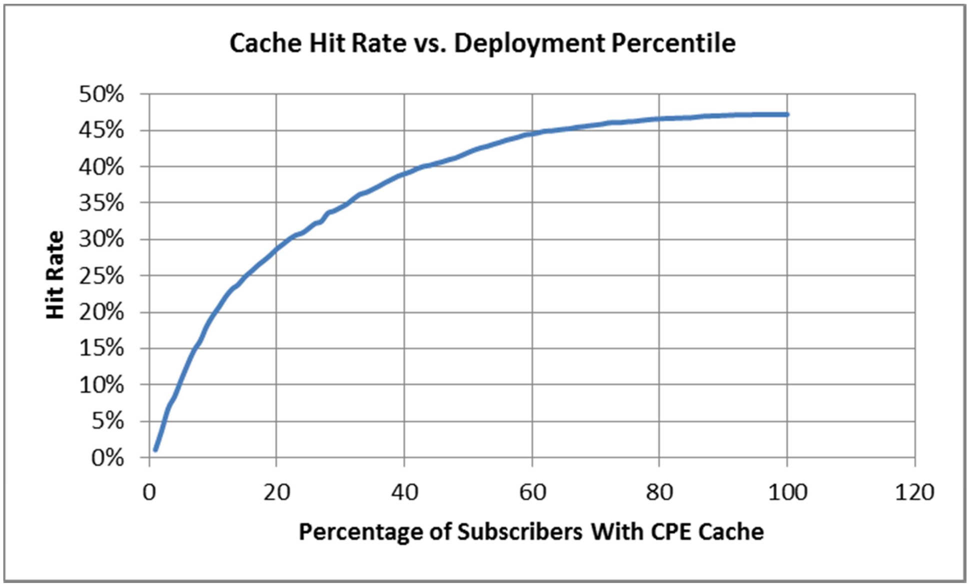


Figure 11. Cache Hit Rate vs. Deployment Percentile

Finally, Figure 11 shows a comparison of the percentage of Netflix users that have a CPE Cache to the benefit overall traffic reduction during peak hours. This shows that about 90% of the benefit can be achieved by deploying a CPE Cache to the top 50% of peak Netflix consuming subscribers. So a targeted deployment can be an economical way to improve network efficiency while minimizing the overall storage capacity across the network.

CONCLUSIONS

The Internet is replacing traditional TV viewing and video bitrates are increasing. Both of these trends are contributing to an exponential increase in peak time Internet traffic, which challenges the economics of providing consumer broadband Internet access. However, as the peak hour usage outgrows the average hour usage, the resulting inefficiency creates an opportunity for optimization using CPE Caching technology.

This paper introduced the key metrics for evaluating a CPE Cache system, and discussed several architectural considerations. A successful CPE Caching system achieves a reduction in peak network bitrate by maximizing storage efficiency while mimimizing required storage capacity and user participation. A study was done evaluating one content selection algorithm that showed a savings of nearly 50% of peak bitrate, for included network traffic. Most of this benefit can be realized by deploying CPE Caches to 50% of the subscriber population.

Future study should focus on what impact the CPE Cache has on the network during off-peak hours; this paper did not examine that aspect of the system.

CPE Caching is a technology that is beneficial to all stakeholders in the Internet video delivery ecosystem. It helps subscribers by providing better quality video even if the network is congested. It helps CSPs by using the network more efficiently, which makes providing consumer broadband access more economical. Finally, it helps content providers by reducing their video delivery costs and improving the quality of their service to their subscribers.

REFERENCES

1. Cisco Visual Networking Index (VNI)

2. Sandvine Global Internet Phenomena Report 2H 2014

3. Application of Policy Based Indexes and Unified Caching for Content Delivery. 2014. Andrey Kisel, Alcatel-Lucent.

4. Netflix ISP Speed Index, February 2015.