NETWORKWORLD FROM IDG

Kollective
Previously Kontiki

ASSESSING THE ENTERPRISE VIDEO DELIVERY LANDSCAPE

THERE HAS TO BE A BETTER WAY

Introduction

As video-capable devices and use cases proliferate, so does the variety of delivery technologies and protocols. This guide provides an assessment of current and emerging trends in video, as well as insight into software-defined enterprise content delivery networks (SD ECDNs), which are poised to bring order while improving performance.

Increased adoption of advanced video communications in the enterprise will cause business IP traffic to double between 2014 and 2019, according to Cisco[1]. Nearly 80 percent of respondents to a 2015 survey said they watched or created video as part of their work, spending between 30 minutes and five hours a month doing so[2].

These statistics likely come as no surprise to enterprise IT professionals who have been dealing with the growing use of video for some time. They know employees consume video from a variety of sources, and internal constituents generate video that travels across corporate networks as it finds its way to its intended audience, whether internal or external. Video is now considered just another form of content, like files and documents, as well as a routine communications medium, no different from a phone call or email. As such, users expect it to work flawlessly, all the time.

Unfortunately for IT, making good on that expectation is not easy. While there's no shortage of delivery technologies intended to handle bandwidth-intensive traffic like video, what works for one application may not work well for another. The global nature of many enterprises compounds the problem, because the delivery approaches that work well for users in or near corporate headquarters may be fraught with problems for remote employees in far-flung countries or even in-country branch offices, where bandwidth availability may be significantly constrained.

To help enterprises find their way through these largely uncharted video waters, the pages that follow offer our assessment of the current video delivery technologies and their pros and cons in an overall video delivery strategy. We also provide insight into a solution that takes advantage of software-defined networking (SDN) to protect investments in legacy video delivery technology while improving performance today and positioning the enterprise to handle future video requirements.

[1] Cisco Visual Networking Index: Forecast and Methodology, 2014–2019, May 2015
[2] "The State of Video in the Enterprise 2015," Kaltura, Inc., 2015.

Intertwined video use cases

Before diving into the video delivery technologies, it's helpful to look at the various ways in which companies use video – and how an application that starts by using one delivery mode may morph into something quite different.

Popular use cases include:

- Town hall meetings: These live, one-to-many meetings are useful for communicating to a broad audience, such as a CEO "all hands" event or a sales manager outlining objectives for the year to far-flung teams.
- Online training and education: These videos could be live or on-demand and run the gamut from human resources communicating company policies and benefits for new employees to an educational institution offering an online course.
- Videoconferencing: Typically one-to-one or few-to-few, videoconferencing has become easier with the proliferation of quality, video-capable devices. While videoconferencing is important, this paper focuses on few-to-many broadcast use cases, where delivery is a significant challenge. Videoconferencing will be referenced occasionally, as it relates to the various technologies discussed in this paper.
- Digital signage: Digital signage involves the use of fixed digital displays in public or common areas to present various kinds of multimedia content, including video. Four Winds Interactive, for example, has solutions that enable centralized control over messaging intended for use by groups ranging from human resources, facilities managers, executives and marketers. The tools are intended to foster multimedia communications with everyone from employees in factory break rooms to customers in retail stores.

While each of these individual cases may seem straightforward, the reality in practice is not so simple.

An IT manager can prepare for a CEO town hall meeting by ensuring adequate bandwidth is available at the appointed time and perhaps by sending instructions to employees on how to view it on their chosen device. But even if the event goes off without a hitch, the delivery task is far from over. Employees who missed the live event will want to view it later; others may wish to review it again. This means the video has to be stored somewhere and may be accessed many times over the network by employees who may be located anywhere around the globe. So, what started as a one-time exercise in delivering live video has morphed into an ongoing video-on-demand service that has to be delivered over the corporate network, the Internet and possibly public cellular networks.

Educational content can be equally diverse, as it may include live, on-demand and one-to-one instruction, each of which places different demands on the network.

Video delivery technologies

To support these varying applications, multiple video delivery technologies have emerged over the years, including:

- Hardware caches and stream repeaters
- Multicast
- WebRTC
- Peer-assisted delivery

We discuss each of these in turn.

Hardware caches and stream repeaters

This approach involves deploying multiple hardware devices at judicious locations around the corporation to cache on-demand video or repeat live video streams for end users near the devices. The idea is to avoid sending the same content multiple times across the wide-area network (WAN) to reach each end user; instead, content is delivered once from the source to each caching or repeater device from where it is delivered to individual end users.

The technology is typically implemented as purpose-built devices from vendors in the WAN optimization business. These devices were initially intended to improve performance for web pages and lightweight transactions, not video. Caches suitable for on-demand video require substantial amounts of fast local storage, and repeaters must be capable of receiving and repeating the many kinds of live video streaming protocols, which may create challenges for global, heterogeneous corporations. Supporting both live and on-demand video requires devices capable of both, or the installation of multiple devices at each location.

Given the significant capital and operational expenses of deploying and coordinating geographically dispersed equipment, this approach is suited for companies with concentrated footprints – a relatively small number of locations where the end-user populations are clustered. It scales poorly in cases where the company has many locations of varying sizes and connectivity, or where user populations are changing, since the devices are sized and placed to match the needs of a location and are not readily changed or redeployed.

Multicast

Whereas hardware caching relies on deploying additional, purpose-built devices, multicast makes use of devices that are already spread throughout the enterprise network: switches, routers and gateways. Most of these have the built-in ability to forward a single incoming data stream to many downstream devices at once, somewhat similar to the way radio is broadcast, so that as with the stream repeaters mentioned above, you can avoid sending the same stream from the source individually to each end user and thus conserve bandwidth on shared links.

Multicast performs that job well, making it an effective way to minimize bandwidth use when delivering high-quality video to a large audience; it can significantly reduce both server and network load. Multicast also enables nearly simultaneous video transmission to large numbers of recipients, which is important in vertical industries such as financial services.

From a practical perspective, however, multicast does have limitations, beginning with operational complexity. It turns out that there are many ways to configure multicast, such as: selecting the protocols used to decide on and optimize the routes over which the stream should pass, and setting the channels (known as multicast groups) on which routers, switches and end-user machines should listen for the multicast. All of this configuration has to be correct and synchronized at every involved server, device and end-user machine for the multicast to succeed. In large global corporations this can prove challenging, to say the least.

Further, the video playback technology on each end-user machine must be capable of receiving multicast streams. This usually takes the form of vendor-specific applications such as Microsoft's Silverlight, Adobe Flash, Java Applets or browser plugins that may not be compatible or even available across all of the end-user devices the company wishes to reach. Worse, many of these client-side multicast technologies are now being deprecated: Silverlight will be end-of-lifed by Microsoft in the near future, Adobe Flash is now deemed insecure by a growing number of experts, and Java Applets are blocked by most corporations for security reasons.

As a real-time stream transport, multicast provides its best benefit for one-to-many live streaming applications, but by that very nature, provides almost no help for on-demand video. It has no inherent caching or storage, and to achieve efficiency it depends on many users watching exactly the same video at the same point in the video, which is not the video-on-demand use case.

A final complication for multicast arises from the growing numbers of wireless-connected end-user machines and devices. Unlike wired switches and routers, WiFi access points do not support multicast well, usually running it at the lowest possible bitrates to ensure reasonable reliability and often converting it to unicast by default. Since WiFi is a shared-carrier transport, low-bitrate multicast streams contend with other unicast traffic for radio time, impacting overall network performance.

For all multicast's promise as a fast, ubiquitous, efficient video transport for the enterprise, very few companies succeed in employing it as a video delivery technology of choice.

Sizing Up Video Delivery Technologies

	Hardware caching	Multicast	WebRTC	Peer-assisted delivery
Cost effectiveness	6	4	8	9
Ease of deployment	1	5	7	8
Scalability	3	5	2	9
Ease of management	3	3	5	9
VOD support	8	0	0	8
Live streaming support	4	3	0	8
Automated testing	5	0	0	8
Platform type	Hardware	Hardware	Software	Software

Ratings based on a scale of 1-10, with 10 being best.
Source: Based on Kollective Customer Experiences

WebRTC

Another emerging delivery technology is WebRTC, which stands for Web Real-Time Communications. The idea behind WebRTC is to enable real-time, point-to-point communications sessions, including voice and video, from within a web browser natively, without requiring plug-ins.

It's quite early in the game for WebRTC, with few actual applications of the technology in production. Currently, only a limited number of browsers support WebRTC, notably not including the Internet Explorer and Safari browsers that dominate enterprise deployments, and even then there are compatibility concerns as the standard is still maturing.

The promise of WebRTC is in real-time sessions between individual users. In a corporate contact center, for example, an agent may be able to establish an audio or video session with a customer who is visiting its support site for a video chat. But at this stage it is strictly one-to-one, not one-to-many, and only for real-time sessions. There's no support for video on demand or bandwidth management features to deal with performance issues.

Some vendors are exploring attempts at one-to-many applications built on WebRTC that rely on sparsely supported features, but the viability of these is yet to be established.

WebRTC is certainly a technology that bears watching, and may well prove useful in certain one-to-one or one-to-few videoconferencing applications. However, given all this uncertainty it is not yet a viable solution for enterprise video on a large scale.

Peer-assisted delivery

Somewhat like multicast, peer-assisted delivery makes use of existing infrastructure to provide efficient video delivery across the enterprise, recruiting available resources on end-user machines to build a distributed edge cache and stream-repeating server farm. A small software agent is installed on end-user machines, enabling them to participate in an emergent delivery network. They cooperate with a central controller and agents on nearby end-user machines to collectively minimize duplicate traffic across WAN and Internet gateway links, exchanging content with one another over fast LAN links, usually resulting in a single copy of a live stream or on-demand video coming across the shared links.

There are several interesting advantages to this approach. Only end-user software needs to be deployed, no hardware, and this is generally substantially faster and cheaper than geographically dispersed hardware deployments. Enterprises often employ desktop management systems that allow this to be orchestrated simply and centrally. This also allows the system to be easily upgraded to handle new protocols and other changing needs. The central controllers are typically hosted and managed by the vendor, with some vendors offering a private cloud option.

Peer-assisted delivery systems are naturally self-scaling. The more end users wanting to watch a piece of content, the more devices there will be available to serve the content to them. Some implementations take advantage of the fact that there is a smart agent on each user machine and provide additional delivery services such as content subscriptions with background delivery or distribution of other content types.

Streaming protocols

In addition to video delivery technologies, quite an array of protocols have been developed over the years for streaming video across a network. These range from the special-purpose, real-time protocols such as RTSP and RTMP – historically the most common protocols used to stream video – to general-purpose devices, to the current crop of HTTP-tunneled protocols, such as HLS, Smooth Streaming and MPEG-DASH. HTTP-tunneled protocols are designed to allow standard web servers to serve streaming video.

Each of these protocols defines a way to break up a video stream into packets suitable for delivery across IP networks. Often, multiple versions of a source video stream will be available in different bitrates and protocols so that end-user devices with different capabilities and connectivity can choose the best-matching version. A particular manifestation of this is known as Adaptive Bit Rate, or ABR, in which the video player on the end-user machine can switch between bitrates on the fly as it monitors the speed with which it is receiving the stream. This ensures the video will continue playing if bandwidth becomes congested, albeit at a lower quality, and allows the player to upshift to higher quality if extra bandwidth becomes available. Some of the protocols mentioned above provide built-in support for these multistream modalities.

With respect to ABR, an interesting capability of some peer-assisted delivery implementations is to decide on bitrate switches at the level of the whole LAN. The agents on end-user devices are naturally in contact with the other devices nearby in a LAN and so can cooperate collectively to choose a single bitrate for all devices in the LAN, thereby ensuring that just a single stream is coming across the shared WAN link for maximum efficiency.

The HTTP-tunneled streaming protocols are rapidly becoming the favored forms for all streaming video delivery. As mentioned, they all are designed to allow standard HTTP-capable network components such as web servers, proxies, caches, CDNs, etc., to stream video as though it were a sequence of small files, so-called segment files. This greatly improves compatibility with existing infrastructure and traffic policies,

and will likely be the dominant class of video-streaming protocols into the future.

The protocols that use this approach have been developed by a number of tech companies and standards bodies and have mixed support across the current variety of end-user devices and video-player technologies. Given the heterogeneity of most large enterprises, all of these may be in use at some point across the organization, and so any video delivery technology must be able to support all of them. Here's a brief description of the main protocols in this class:

HLS

HLS, for HTTP Live Streaming, is a protocol developed by Apple for its QuickTime video player, Safari browser and OS X and iOS operating systems. It has since become very popular and is now in use by a wide array of vendors.

Smooth Streaming

Smooth Streaming is a Microsoft protocol developed as an extension to its IIS Media Services platform. Smooth is used to stream video from IIS-based web servers. It requires the Silverlight plug-in in order to work with browsers other than Microsoft Internet Explorer and Edge.

MPEG-Dash

Dynamic Adaptive Streaming over HTTP (DASH), also known as MPEG-DASH, is the first standard ABR streaming protocol, ratified in 2012. As an international standard, it is touted as "codec agnostic," meaning it can be used with any audio or video codec. In practice, however, the consortium behind the protocol – the Dash Industry Forum (DASH-IF) – has issued a reference implementation that defines specific codecs and other requirements. Vendors including Microsoft, Google, Adobe, Samsung, Qualcomm, Sony, Akamai, Ericsson and more than 60 others have said they will support the recommendation, according to the DASH-IF.

Emerging trends that may affect video delivery

Enterprise networks will likely see substantial changes in the years to come, many of which will have direct impact on video delivery services. We discuss here several such trends that are imminent, if not already becoming popular.

WiFi

An ever-increasing number of end-user devices in the enterprise connect into the company's network over WiFi. This follows naturally from the increased power of mobile devices and laptops and the increased mobility of the workforce, not to mention the reduced facilities expenses that come from not having to pull wire everywhere.

However, this trend brings complications, notably the way in which bandwidth is provisioned and shared on wireless networks. A typical wired LAN today sees independent gigabit point-to-point capacities with a relatively static load, at least in terms of the number of endpoints. A typical wireless LAN effectively shares a single connection with a bandwidth in the 100 mbit/s range and the number of endpoints can vary substantially throughout the day and over time, with no easy way for IT to control that load. : This means that not only are WAN and gateway links a congested resource for bandwidth-intensive applications like video, so too is the last mile in wireless LANs.

Further, technologies like multicast that may help with applications such as video can be severely constrained in typical WiFi deployments, as mentioned previously in the section on multicast.

Some software-defined peer-assisted approaches can mitigate these concerns, largely because of their distributed adaptivity; for example, ensuring that wireless devices are segregated from wired ones and have delivery policies and controls suited to WiFi connectivity. Some vendors are even exploring ways to opportunistically employ multicast over WiFi, reducing point-to-point traffic over the shared carrier when usable multicast is available, but ensuring robust delivery with adaptive unicast when it is not.

Cloud-hosted services

More and more solutions are being offered to the enterprise from services that are hosted on external clouds, such as Amazon's AWS and Microsoft's Azure. This is a natural consequence of the astounding growth of cloud services, offering solution providers economical, largely frictionless and yet highly scalable hosting infrastructures for their services, along with the attractiveness to their customers of not having to deploy and manage on-premises solutions.

Of particular interest is the growing suite of Azure-hosted services that Microsoft is now aggressively marketing to enterprises, such as Office 365 and Skype for Business. These product lines include video applications spanning the gamut from on-demand video portals in Office 365 Video through conferencing in Skype for Business, to one-to-many webcasting in Skype Meeting Broadcast.

All of these externally hosted services come to end users inside the enterprise across the company's Internet gateways, in contrast to typical on-premises services that come exclusively across the company's WAN infrastructure. These alternatives can present wildly different traffic patterns for IT to manage and accommodate, and now promote the Internet gateways to be yet another business-critical congested resource, particularly if video traffic is part of the external service.

This presents a distinct challenge for some video delivery approaches, notably multicast, increasing the complexity of establishing multicast networks across these gateways to almost unmanageable proportions.

Fortunately, some of the delivery approaches mitigate this traffic restructuring, either by physically deploying more hardware caches and stream repeaters at the gateways, or automatically, as in the case of the peer-assisted approaches. Many peering systems will treat Internet gateways as just another shared link in the delivery network topology and will automatically minimize traffic and build cooperative end-user machine delivery meshes downstream of the gateways.

Another twist presents itself when corporations concerned about security choose to connect to these externally hosted services via private connections rather than the public Internet, say through an MPLS provider. While this does address the security issue, the number of connection points and transit bandwidth in such setups is typically quite limited, weighing heavily against high-bandwidth applications such as video. Caching and peering solutions may be the only way to accommodate video applications in such scenarios.

360-degree video

A recent trend in conferencing and event webcasting is the use of 360-degree cameras and panoramic players to enable immersive and compelling experiences for users, allowing them to independently pan around a camera location –

to focus on different speakers, follow a presenter around a stage or watch questioners from the audience. This is typically implemented using stitching techniques that map the different viewing directions onto a spherical mapping within a single, standard video stream that is then transformed at playback into a panoramic, VR-style viewing experience by special panoramic video players.

Because only a small portion of the spherical video is viewed at any one time by a viewer, it is common to record the entire viewing sphere at a much higher resolution, at least 1080P HD, and often 4K or even 8K resolutions. The essential concern for enterprises is not 360-degree video as such, since the schemes use a standard video stream for delivery. It is the much higher resolutions and consequent higher bitrates at which these must be delivered, typically between 3 mbit/s and 8 mbit/s or even higher in the case of 4K or 8K streams.

The higher resolutions and bitrates clearly just compound the bandwidth problem for the enterprise and may even rule out this kind of video in some cases. Again, the approach here is to employ one of the video-specific delivery technologies discussed to mitigate and minimize duplicated streaming traffic across contested links, and peer-assisted solutions are best placed here given their adaptability.

SDN

If you follow the IT world at all, you will have heard about software-defined networking. SDN is hailed by many as a revolutionary approach to the way large networks are deployed and managed within an enterprise, aiming at the prime goals of reducing capex and opex while increasing network manageability, flexibility and adaptability.

SDN is a sensible move toward making all of the management and control of network infrastructure fully software-based. The idea is to provide a centralized control architecture that abstracts over all the differences in all the variations of networking hardware across an organization, and so enables unified control and monitoring. Further, SDN architectures provide a standard set of APIs, making them fully programmable and enabling generic orchestration, management and monitoring tools to be developed that work consistently across the diverse array of equipment that corporations accumulate. SDN is thus meant to promote vendor neutrality, reducing switching costs and increasing vendor competition.

The SDN APIs also allow applications that make heavy use of the network to influence its behavior directly, or access various kinds of network status and information services to help the application work more effectively. A commonly cited example of this happens to be in the area of video, videoconferencing in particular. An SDN-enabled videoconferencing application can inform the SDN of upcoming point-to-point connections for a conference, and the SDN system would then programmatically optimize the routes or improve quality of service (QoS) on those routes for the duration of the call.

Although it is still early, as SDN adoption increases, expect to see network-intensive enterprise applications such as video incorporate the ability to interact with SDN systems and generally improve network usage. Peer-assisted systems will be able to make particular use of this, both informing the SDN of its expected use of links, gateways, leader machines and so on, as well as retrieving topology and congestion status from the SDN to inform the peering network's optimization.

Building a better video delivery model

As the drive to support truly capable and comprehensive video delivery within the enterprise increases, a model is emerging that leverages the strengths of many of the approaches mentioned above. It also provides an adaptive framework within which these approaches are optimally applied to the diverse contexts and use-cases discussed, and within which they can evolve to meet emerging challenges.

Such a model has many of the same goals and attributes as SDN: increasing network efficiency and flexibility while reducing expenses. It achieves this, not surprisingly, by using the same essential strategy as SDN – making the delivery model *software-defined.*

Enterprise-focused systems that embody this model are known as SD ECDNs, Software-Defined Enterprise Content Delivery Networks, and the best truly *are* CDNs, capable of efficiently delivering all kinds of heavy content, not just streaming video.

Defining the SD ECDN

As suggested above, Software-Defined Enterprise Content Delivery Networks are CDNs designed to work as efficient, economical content delivery systems *within* corporate networks. As such, they feature many attributes aimed at the enterprise, including advanced security, business-oriented delivery modalities, the ability to incorporate multiple delivery technologies and adaptively employ these as needs dictate, and APIs that enable content-delivery services to be integrated into various content-heavy business applications, among others. They are also specifically adapted to the network architectures seen in today's large enterprises.

Software-Defined ECDNs bring a level of control, flexibility and adaptivity that is becoming essential these days, and mirrors similar developments in other aspects of an enterprise's IT infrastructure, as we've seen with SDN. Not surprisingly, they also tend to have architectures similar to SDN, with central controllers orchestrating distributed and diverse delivery assets, providing a common API for full programmability.

Current SD ECDN implementations build themselves around a peer-assisted delivery core, capitalizing on its software-based nature and adaptability as well as its many clear advantages as a video delivery technology.

To afford the required security, the best peer-assisted systems build their delivery network on a multilevel, PKI-based security framework that eliminates attack vulnerabilities; provide point-to-point and at-rest encryption; validate signatures on data from arbitrary nodes; and implement identity-based content access-control. They also maximize their robustness, speed and adaptivity by forming *mesh-based* delivery grids, which make for fast route-finding and failover, and which bond the bandwidth from multiple concurrent serving nodes to accelerate delivery.

Further, having agents on all the end-user machines that are in touch with a central controller provides the basis for deep, centralized control over the delivery network, a signature attribute of SD systems. This allows IT to put in place delivery and resource-utilization policies that match the performance and impact trade-offs they wish to make, and makes it easy to change and adapt these dynamically as needs dictate.

These are intelligent agents and so are able to handle multiple delivery modalities, maximizing performance for live events, spreading the caching load for libraries of on-demand content, and politely soaking up idle bandwidth for background subscription deliveries. They can be asked to perform crucial ancillary functions as well, such as cooperating to silently simulate the load of an upcoming major live event, reporting back on the readiness of the network to handle that load, and allowing tuning and adjustments to ensure optimum end-user experience during the actual event.

Finally, best-of-breed SD ECDN vendors will also offer their own tightly integrated enterprise video applications in addition to a robust API. These applications, including video portals and webcasting systems, take best advantage of the vendor's core SD ECDN solutions. This gives customers a one-stop shop for end-to-end solutions should they desire it, and also provides a closed-loop development environment for the vendors, ensuring their SD ECDNs are indeed best of breed.

In short, with an SD ECDN, enterprises get a video delivery solution that is:

- Economical, since no additional hardware needs to be purchased, deployed, managed or upgraded
- Easy to deploy, because it's based on a simple software agent
- Adaptive, with the ability to dynamically adjust to changes in traffic patterns as well as physical network changes in the WAN or LAN
- Self-scaling, because more content requestors equates to more resources available for content distribution
- Self-healing; if one node stops serving, others take over
- Centrally controlled, including automated workflows governing publishing privileges and the ability to define attributes such as peering policies, latency and bitrate levels
- Extensible; being software-based, it can be readily extended to encompass new and alternate delivery technologies and adaptively employ these as needs dictate

IT has been dealing with a dizzying array of video delivery technologies long enough, grappling with how best to serve a diverse user base. The SD ECDN model brings sanity to the enterprise video environment, along with a new level of performance and control that end users and IT alike will appreciate.

Learn more about SD ECDNs and how they benefit your peers:
http://kollective.com/kollective-software-defined-ecdn/