



THE
COMPLETE
GUIDE TO
LIVE
STREAMING

What it is, how it works, and why you need it.

TABLE OF CONTENTS

03 Overview

05 Capture

07 Video Encoding and Codecs

10 Packaging and Protocols

12 Ingest and Transcoding

16 Delivery

18 Playback

20 Use Cases



The Complete Guide to Live Streaming

What it is, how it works, and why you need it.

Overview

Live video streaming is skyrocketing in popularity. It will account for approximately 13 percent of all internet traffic by 2022, [representing a fifteen-fold growth from 2017](#).

While streaming technologies have evolved drastically over the years, the basic definitions still apply. In a nutshell, live streaming involves broadcasting video and audio content across the internet to allow for near-simultaneous capture and playback.

But between capturing a live video feed and broadcasting it, quite a bit occurs. The data must be encoded, packaged, and often transcoded for delivery to virtually any screen on the planet.

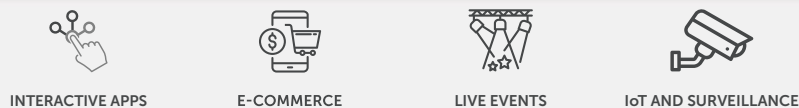
In this guide, we'll take a deep dive into the end-to-end workflow.

LIVE VIDEO DELIVERY FROM CAPTURE TO PLAYBACK



WHY DO YOU NEED LIVE STREAMING?

USE CASES AROUND



Video on Demand vs. Live Streaming

Video streaming can take the form of both live and recorded content. With live streaming, the content plays as it's being captured. Examples of this range from video chats and interactive games to endoscopy cameras and streaming drones.

[Video on demand \(VOD\)](#), on the other hand, describes prerecorded content that internet-connected users can stream by request. Some top players in this space include Netflix, Amazon Prime, Hulu, and Sling. YouTube's [David After Dentist](#) and Netflix's [Stranger Things](#) are both examples of VOD content.

For the purposes of this guide, we'll be focusing on live video. Let's start by summarizing the live streaming workflow and then take a closer look at the individual steps.

VIDEO ON DEMAND VS. LIVE STREAMING

Live Streaming Workflow

Live video streaming starts with compressing a massive video file for delivery. Content distributors use an encoder to digitally convert the raw video with a codec. These two-part compression tools shrink gigabytes of data into megabytes.

The encoder itself might be built into the camera, but it can also be a hardware appliance, a computer software like [OBS Studio](#), or a mobile app.

Once the video stream is compressed, the encoder packages it for delivery across the internet. This involves putting the components of the stream into a file format. These container formats travel according to a [protocol](#), or standardized delivery method. Common protocols include [RTMP](#), [HLS](#), and [MPEG-DASH](#).

The packaged stream is then transported to a media server located either on premises or in the cloud. This is where the magic happens. As the media server ingests the stream, it can transcode the data into a more common codec, transize the video into a different resolution, transrate the file into a lower bitrate, or transmux it into a more scalable format.

This conversion process is critical when streaming to a variety of devices. Without transcoding the original stream, reaching viewers across an array of devices wouldn't be possible. A [streaming server software](#) or [cloud streaming service](#) can be employed to accomplish this and more.

The single stream that first entered the media server will likely depart as multiple renditions that accommodate varying bandwidths and devices for large-scale viewing. But distance is also an issue.

The farther viewers are from the media server, the longer it will take to distribute the stream. That's where a [content delivery network \(CDN\)](#) comes in handy. CDNs use a large network of servers placed strategically around the globe to distribute content quickly.

If done right, the live stream will find its way from the CDN to viewers across the world — in a matter of just seconds. The live stream will play back with minimal buffering and in the highest quality possible for spectators across a range of devices and internet speeds.

It all starts with employing the right tools along the way. Read on to get the skinny.

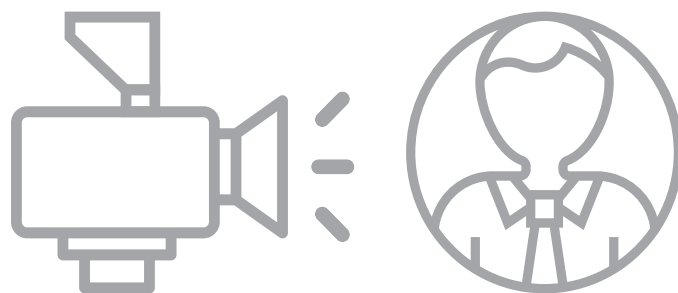


Capture

Three, Two, One... Action!

Live streaming starts at the camera. Most [cameras](#) are digital and can capture images at a stunning 4K resolution (2160p). This resolution requires a very high bitrate to support the raw digital video signal coming out of the camera, so the cables used to transfer this signal must be capable of handling large amounts of data. HDMI or Ethernet cables can be used in some cases. But most often, a 4K signal transferred over long distances requires an SDI cable that can manage the bandwidth requirements.

Wireless cameras can also be used, with [portable broadcasting platforms](#) finding their way into the industry. Today's smartphones are designed for streaming, outperforming digital cameras from ten years prior. The iPhone 12 Pro Max, for example, records HDR video at 60 frames per second.



Multi-Camera Video Production

Some live streaming is done with a smartphone, but more serious live productions employ additional cameras. These multi-camera studio setups, and other video sources, are connected to a switcher that transitions between them. The audio is transferred to a mixer via XLR cables. Generally, the switcher adds the audio from the mixer into the final output signal. The switcher could be hardware, software, or a little of both when capture cards are required.



IP Cameras

When production isn't a priority but speed matters, IP cameras come into play. [IP cameras](#) can send live streams directly over Ethernet cables, making them easy to put wherever you want. Most IP cameras use the [RTSP protocol](#), which supports [low-latency live streaming](#). RTSP is pulled to the media server rather than pushed. For that reason, the camera must be on an open, static IP address for the media server to locate it.

From surveillance to conferencing, IP cameras work great when you want to live stream from one location without getting too fancy. These user-friendly streaming devices don't require a separate encoder, and you can aggregate the content for delivery to any device with a live transcoding solution.

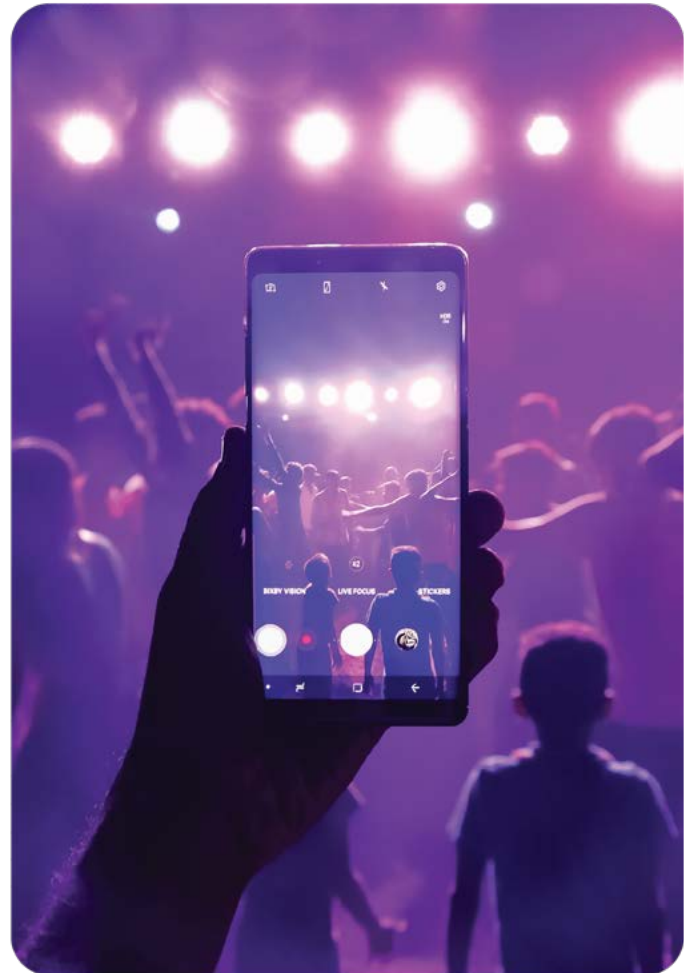


User-Generated Content

User-generated content makes up a significant portion of live streams. In some cases, webcams are used. When it comes to sites like Twitch, users employ a combination of screen-recording software and webcams. But the majority of today's content creators are on their smartphones. In fact, [mobile users account for 58.56 percent of all internet traffic today](#).

Mobile [apps](#) and [social media networks](#) leverage video to drive engagement, but the use cases don't end there. Smartphones can be transformed into everything from [bodycams](#) to [crime-tracking tools](#) with the addition of live streaming.

While users are responsible for supplying their own recording technology (a.k.a. smartphones or webcams), the live streaming app must have encoding functionality built in.



Video Encoding and Codecs

The second step in any live streaming workflow is video encoding. After capturing the video with your camera(s) of choice — be it a production-quality setup, IP camera, or your end-users' mobile devices — live video data must be digitized for efficient transportation across the internet. Video encoding is essential to live streaming, helping to ensure quick delivery and playback.

What Is Video Encoding?

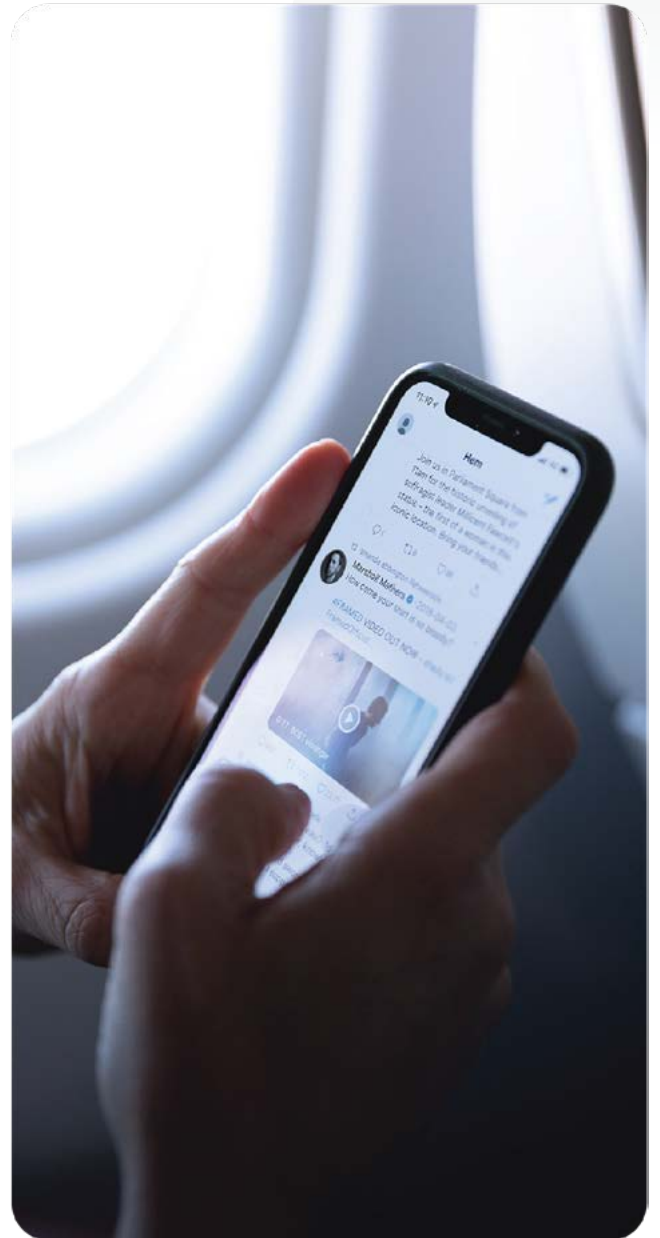
[Video encoding](#) refers to the process of converting raw video into a digital format that's compatible with many devices. Videos are often reduced from gigabytes of data down to megabytes of data. This process involves a two-part compression tool called a codec.

What Is a Codec?

Literally 'coder-decoder' or 'compressor-decompressor,' codecs apply algorithms to tightly compress a bulky video for delivery. The video is shrunk down for storage and transmission, and later decompressed for viewing.

When it comes to streaming, codecs employ lossy compression by discarding unnecessary data to create a smaller file. Two separate compression processes take place: video and audio. Video codecs act upon the visual data, whereas [audio codecs](#) act upon the recorded sound.

[H.264](#), also known as AVC (Advanced Video Coding), is the most common video codec. AAC (Advanced Audio Coding) is the most common audio codec.



What Video Codec Should You Use?

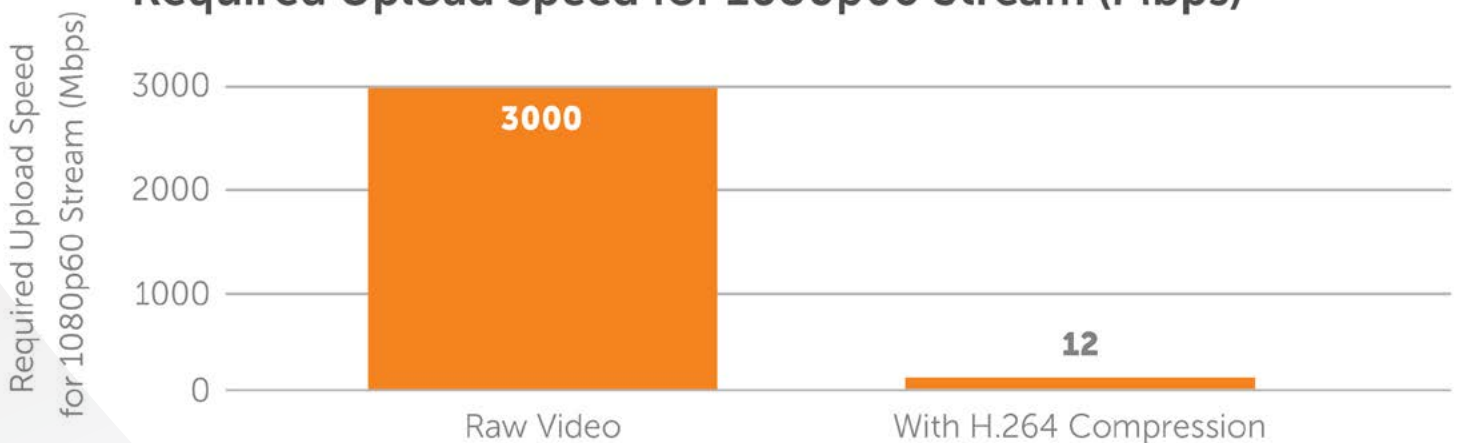
Streaming to a variety of devices starts with supporting a variety of codecs. But to keep the encoding part of the workflow simple and fast, you can always transcode streams later when they're ingested by the media server.

While industry leaders continue to refine and develop the latest compression tools, many content distributors employ older video codecs like H.264/AVC for delivery to legacy devices. H.264 is your best bet for maximizing compatibility, even though other video codecs are more technologically advanced.

Below are some of the most common video codecs in use today.

	Benefits	Limitations
H.264/AVC	Widely supported.	Not the most cutting-edge compression technology.
H.265/HEVC	Supports 8K resolution.	Takes up to 4x longer to encode than H.264.
AV1	Open-source and very advanced.	Not yet supported on a large scale.
VP9	Royalty-free.	An earlier version of AV1.
H.265/VVC	Intended to improve upon H.265.	Same royalty issues as H.265.

Required Upload Speed for 1080p60 Stream (Mbps)



Video Codecs Worth Noting:

H.264/AVC

The majority of encoding output today takes the form of [H.264](#) files, also referred to as AVC (Advanced Video Coding). This widely supported codec is often incorporated with the AAC audio codec and can be packaged into .mp4, .mov, .F4V, .3GP, and .ts containers. H.264 plays on just about any device, delivers quality video streams, and comes with the least concerns surrounding royalties.

VP9

A royalty-free alternative to H.265, [VP9](#) offers even better quality at the same bitrate. A comprehensive list of devices, browsers, and platforms support VP9 — making it second to only H.264 in terms of compatibility.

H.265/HEVC

Developed as the successor to H.264, this codec supports 8K resolution. It also generates smaller files than H.264, thus decreasing the bandwidth required to view high-definition streams.

Uncertainties surrounding royalties initially stifled adoption, but H.265 [eliminated royalty fees](#) in 2018. Even so, the codec only accounts for about 10 percent of encoded live streaming files today.

What Audio Codec Should You Use?

AAC takes the cake when balancing quality with compatibility across audio codecs. While open-source alternatives like [Opus](#) far outperform AAC, they lack support across as many platforms and devices.



	Benefits	Limitations
AAC	Most common audio codec.	Higher-quality alternatives exist.
MP3	Also widely supported.	Less advanced than AAC.
Opus	Highest-quality lossy audio format.	Yet to be widely adopted.
Vorbis	Non-proprietary alternative to AAC.	Less advanced than Opus.
Speex	Patent-free speech codec.	Also obsolete by Opus.

Audio Codecs Worth Noting:

AAC

Defined by MPEG-4, this widely supported standard is used by YouTube, Android, iOS, and iTunes. Two extensions of AAC exist: HE-AAC for low bitrates and AAC-LC for low delay. HE-AAC works best whenever bandwidth is a concern, while AAC-LC lends itself well to two-way communication.

Opus

Developed by the Xiph.Org Foundation, Opus provides higher-quality audio than any other lossy audio format. It's open-source and royalty-free, but has yet to be widely adopted.

Encoding Best Practices

[Encoding best practices](#) go way beyond what codec you select. You'll also want to consider frame rate, keyframe interval, and bitrate.

Luckily, a live stream can always be transcoded into another format once it reaches the server. This can be done using a [software](#) and your own servers, or in the cloud for professionally managed delivery.

In addition to the codecs used, it's important to examine how your stream is packaged and which protocols enter the picture.

MP3

Nearly every audio-supported digital device in the world can play back the MP3 format, making it a viable option for live streaming. But because AAC offers superior compression, we'd recommend going with that.



Packaging and Protocols

Once compressed, video streams must be packaged for delivery. The distinction between compression and packaging is subtle but relevant. For the sake of explanation, let's picture the process in terms of residential garbage removal.

We start with the raw video data, which must be compressed down for delivery across the internet. An encoder allows us to do so by compressing gigabytes into megabytes. Think of the encoder as a household trash compactor and the codecs as the bags of compressed trash.

In order to actually transport the compressed trash (audio and video codecs) to the dump (viewer), another step is required. It's crucial to place the bag of trash, as well as any other odds and ends (such as metadata), into a curbside trash receptacle. File container formats can be thought of in terms of these receptacles. They act as a wrapper for all streaming data so that it's primed for delivery.

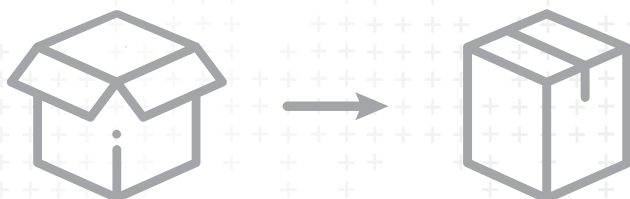
Finally, the contents of the trash bin are transported to the dump via an established route. Think of protocols as the established routes that garbage trucks take.

OK. Enough trash talk. Time to define each term for real.



What Is a Video Container Format?

Video container formats, also called wrappers, hold all the components of a compressed stream. This could include the audio codec, video codec, closed captioning, and any associated metadata such as subtitles or preview images. Common containers include .mp4, .mov, .ts, and .wmv.



What Is a Streaming Protocol?

A protocol is a set of rules governing how data travels from one device to another. For instance, the Hypertext Transfer Protocol (HTTP) deals with hypertext documents and webpages.

Online video delivery uses both streaming protocols and HTTP-based protocols. Streaming protocols like Real-Time Messaging Protocol (RTMP) offer fast video delivery, whereas HTTP-based protocols can help optimize the viewing experience.

 **The protocol used can increase streaming latency by up to 45 seconds.**

Traditional Stateful Streaming Protocols

In the early days, traditional protocols such as [RTSP \(Real-Time Streaming Protocol\)](#) and [RTMP \(Real-Time Messaging Protocol\)](#) were the go-to methods for streaming video over the internet and playing it back on home devices. These protocols are stateful, which means they require a dedicated streaming server.

While RTSP and RTMP support lightning-fast video delivery, they aren't optimized for great viewing experiences at scale.

Additionally, fewer players support these protocols than ever before. Many broadcasters choose to transport live streams to their media server using a stateful protocol like RTMP and then transcode it for multi-device delivery.



RTMP and RTSP keep latency at around 5 seconds or less.

HTTP-Based Adaptive Streaming Protocols

The industry eventually shifted in favor of HTTP-based technologies. Streams deployed over HTTP [are not technically "streams."](#) Rather, they're progressive downloads sent via regular web servers.

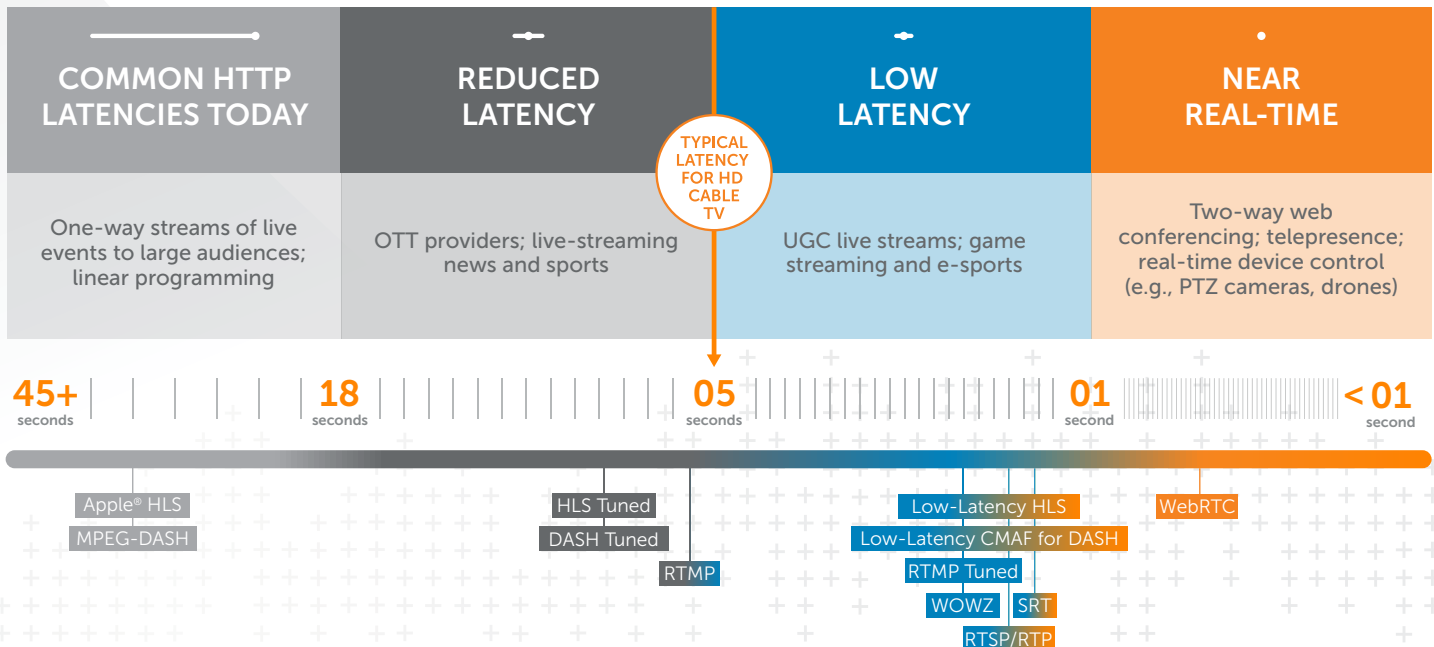
Using [adaptive bitrate streaming](#), HTTP-based protocols deliver the best video quality and viewer experience possible — no matter the connection, software, or device. Some of the most common HTTP-based protocols include [MPEG-DASH](#) and [Apple's HLS](#).

HTTP-based protocols are stateless, meaning they can be delivered using a regular old web server. That said, they fall on the high end of the latency spectrum.



HTTP-based protocols can cause 10-45 seconds in latency.

STREAMING LATENCY AND INTERACTIVITY CONTINUUM



Emerging Protocols for Near-Real-Time Delivery

With an increasing number of videos being delivered live, industry leaders continue to [improve streaming technology](#). Emerging standards like [WebRTC](#), [SRT](#), [Low-Latency HLS](#), and [low-latency CMAF for DASH](#) (which is a format rather than a protocol) support near-real-time delivery — even over poor connections.

These new technology stacks promise to reduce latency to 3 seconds or less!

	Benefits	Limitations
WebRTC	Real-time interactivity without a plugin.	Only scalable with a video platform like Wowza.
SRT	Smooth playback with minimal lag.	Playback capabilities are still being developed.
Low-Latency HLS	Sub-three-second delivery at scale.	Vendors are adding support for this new spec.
Low-Latency CMAF	Streamlined workflows and decreased latency.	Many organizations are prioritizing other technologies at this time.

Video Packaging and Protocols for Every Workflow

Depending on how you set up your streaming workflow, you're not limited to one protocol from capture to playback. Many broadcasters use RTMP to get from the encoder to server, and then transcode the stream into an adaptive HTTP-based format. The best protocol for your live stream depends entirely on your use case. Let's take a closer look at how transcoding works.

PROTOCOLS FOR EVERY WORKFLOW

Ingest and Transcode

The fourth step in a live streaming workflow is [transcoding](#) the stream into a variety of different codecs, bitrates, resolutions, and file containers. While this step can be skipped, it's essential to most streaming scenarios.

A media server located either on premises or in the cloud ingests the packaged stream, and a powerful conversion process called transcoding ensues.

This allows broadcasters to reach almost any device — regardless of the viewer's connection or hardware. Once transcoding is complete, multiple renditions of the original stream depart for delivery.

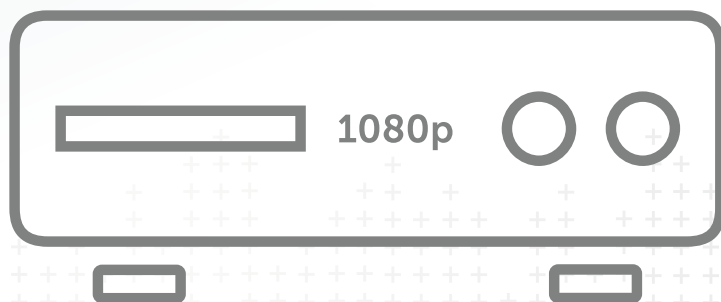


Transmuxing, Transcoding, Transizing, Transrating

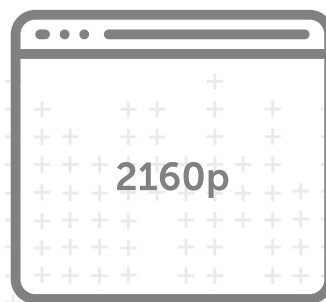
To optimize the viewing experience across a variety of devices and connection speeds, broadcasters often elect to transmux, transcode, transrate, and transize streams as they pass through the media server.

- **Transmuxing:** Taking the compressed audio and video and repackaging it into a different container format. This allows delivery across different protocols without manipulating the actual file. Think of transmuxing like converting a word doc into a pdf and vice versa.
- **Transcoding:** An umbrella term for taking a compressed/encoded file and decompressing/decoding it to alter in some way. The manipulated file is then recompressed for delivery. Transrating and transizing are both subcategories of transcoding.
 - **Transrating:** Changing the bitrate of the decompressed file to accommodate different connection speeds. This can include changing the frame rate or changing the resolution.
 - **Transizing:** Resizing the video frame — or resolution — to accommodate different screens.

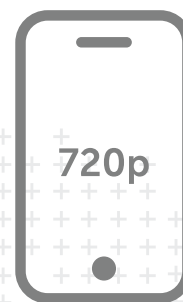
Rather than creating one live stream at one bitrate, transcoding allows you to create multiple streams at different bitrates and resolutions. That way, your live streams can dynamically adapt to fit the screen sizes and internet speeds of all your viewers. This is known as [adaptive bitrate \(ABR\) streaming](#).



MPEG-DASH



RTMP

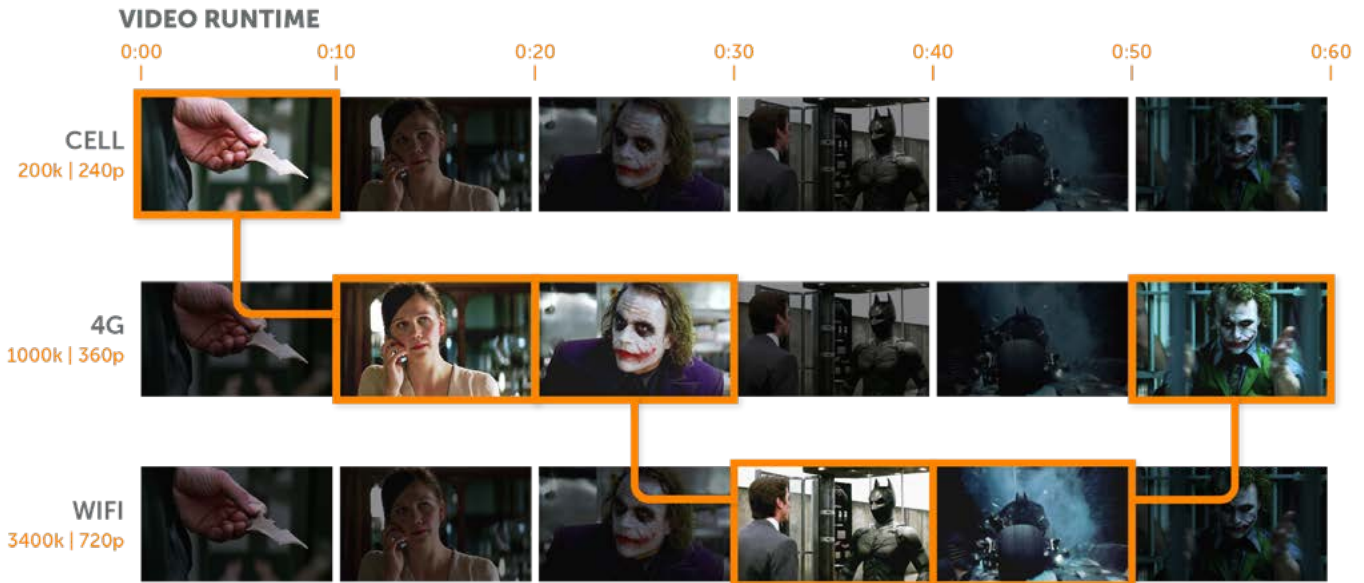


HLS

What Is Adaptive Bitrate Streaming?

Adaptive bitrate (ABR) streaming involves outputting multiple renditions of the original video stream to enable playback on a variety of devices and connection speeds. Content distributors use adaptive bitrate streaming to deliver high-quality streams to users with outstanding bandwidth and processing power, while also accommodating those lacking in the speed and power departments.

The result? No buffering or stream interruptions. Plus, as a viewer's signal strength goes from two bars to three, the stream automatically adjusts to deliver a superior rendition.



How Does Adaptive Bitrate Streaming Work?

The first step to adaptive bitrate streaming is creating multiple renditions of the original stream to provide a variety of resolution and bitrate options. These transcoded files fall on an encoding ladder. At the top, a high-bitrate, high-frame-rate, high-resolution stream can be output for viewers with the most high-tech setups. At the bottom of the ladder, the same video in low quality is available for viewers with small screens and poor service.

During the process of transcoding, these renditions are broken into segments that are 2-10 seconds in length. The video player can then use whichever rendition is best suited for its display, processing power, and connectivity. If power and connectivity change mid-stream, the video will automatically switch to another step on the ladder.

With adaptive bitrate streaming, mobile viewers with poor connections don't have to wait for the stream to load. And for those plugged into high-speed internet, a higher-resolution alternative can play.

Bitrate (kbps)	Resolution
235	320x240
375	512x384
750	512x382
1050	640x480
1750	720x480
2350	1280x720
3000	1920x720
4300	1920x1080
5800	1920x1080

Netflix's Traditional Bitrate Ladder

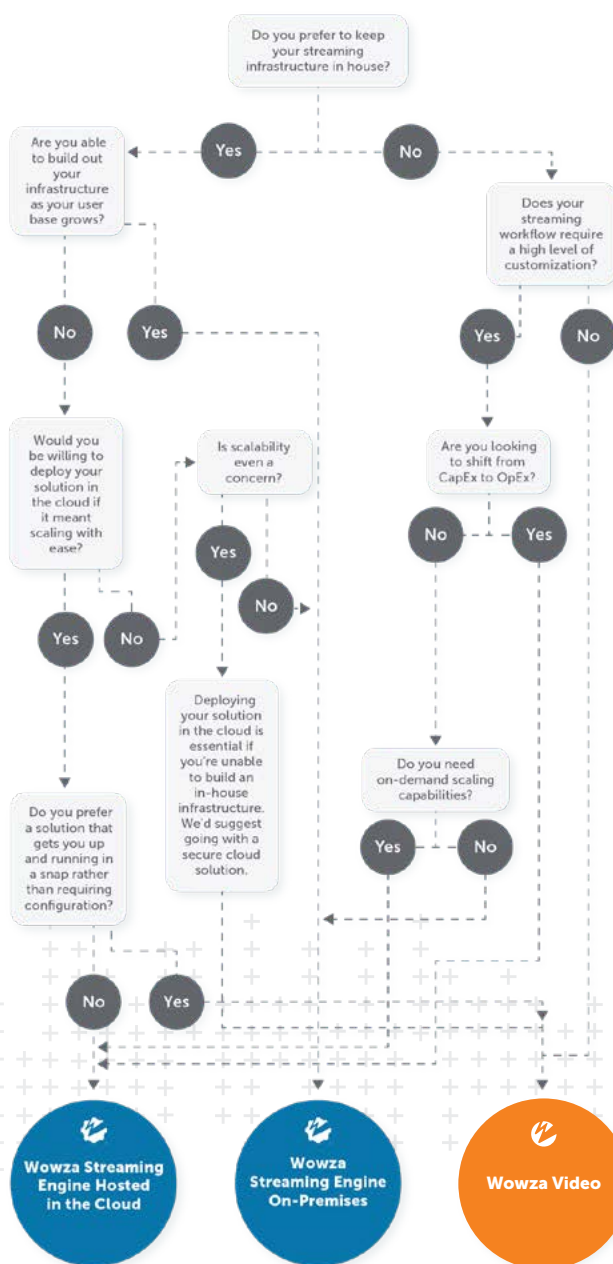
Cloud-Hosted vs. On-Premises Streaming Platforms

Adaptive bitrate streaming requires a trusted video transcoding solution. Some content distributors install a [media server software](#) on their own servers — both on premises and in the cloud — while those who want to get up and running quickly without any hassles often prefer a [managed cloud service](#).

But how do you make the choice between on-premises and cloud-based transcoding? Let's take a look.

Decision Tree:

Cloud vs. On-Premises Streaming Platform



Delivery

Back to the viewers. Because we don't know where they're located, there's still the issue of distance. The farther your audience is from the media server, the longer it will take to deliver the live stream. This can cause [latency](#) and [buffering](#).

To resolve the latency inherent to global delivery, many broadcasters employ a content delivery network (CDN).

What Is a CDN?

As the name suggests, a [CDN](#) is a system of geographically distributed servers used to transport media files. This removes the bottleneck of traffic that can result when delivering streams with a single server, as CDNs only require a single stream for each rendition of an outbound video.

These large networks help truncate the time it takes to deliver video streams from origin to end users. Sharing the workload across a network of servers also improves scalability should viewership increase.

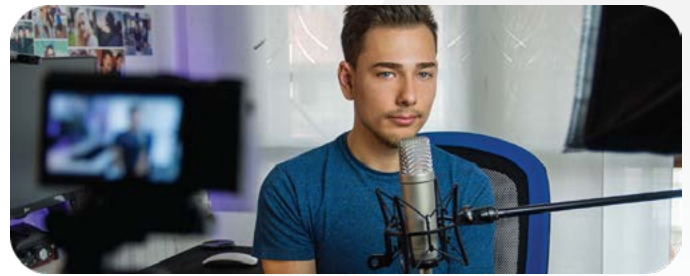
Benefits of Using a CDN for Live Streaming

- **Scalability:** Employing a CDN is the fastest, most reliable way to get your content in front of numerous viewers — even with viewership spikes.
- **Quality:** Streaming through a CDN allows you to achieve the highest sound quality and video resolution possible, while minimizing buffering and delays.
- **Speed:** CDNs use speedy superhighways to deliver content to vast audiences across the globe.
- **Security:** A CDN can help prevent distributed denial of service (DDoS) attacks, which occur when a site or resource is flooded by multiple, simultaneous attempts to breach it.

Reasons Not to Use a CDN

- **Small-scale streaming:** If you have a small number of viewers and/or your geographic scale is limited, you probably don't need a CDN. The more elements you introduce into your live streaming workflow, the more opportunities for failure — so why do so unless you have to?
- **Limited budget:** We recommend that you compare your own egress cost to the cost of a CDN, as this can vary based on deployment. There are both paid and free CDN options available.

Even if you forgo a CDN, other strategies can be employed to reach a broad audience. Many content distributors leverage social media platforms to ensure scalability, speed, and quality without breaking the bank. Simulcasting makes its easy.



What Is Simulcasting?

Simulcasting is the ability to take one video stream and broadcast it to multiple destinations at the same time — thereby maximizing your impact. This allows you to reach a broader audience, no matter which platform or service your viewers prefer.

Live streaming social platforms are everywhere, with both wide-ranging and niche applications. While Facebook connects you with the largest general audience, Twitter is a top destinations for news coverage and events. Meanwhile, Twitch is dedicated to gaming. And let's not forget about YouTube — which has become a search engine in its own right.

While 'the more the merrier' might be an intuitive approach, you should only stream to the destinations that make sense for your audience. Broadcasting in the wrong context can result in negative feedback and wasted resources.

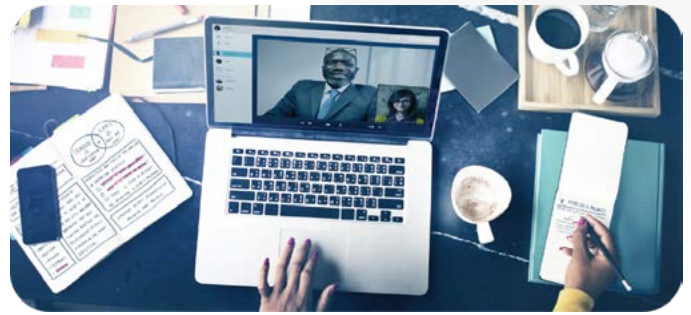
It's also worth noting that the process of simulcasting can be complex. Multi-destination broadcasting requires massive amounts of bandwidth and is prone to errors. That's why it's important to start with the right tools.

Whether via a CDN, simulcasting, or both, the last step to consider is your stream's final destination: playback.



Playback

The end-goal of live streaming is playback. And through the process of encoding, transcoding, and globally distributing the video content, your live video stream should do just that — at high quality, with low latency, and at any scale.



Multi-Device Delivery

If every viewer had a 4K home theater plugged into high-speed internet, delivering live streams would be easy. But that's not the case.

Today's viewers are out and about. A significant portion of your audience is on mobile devices, relying on depleted batteries and LTE connections. Other viewers stream content to their laptops using public Wi-Fi, or perhaps to their iPads via mobile hotspots.

Different screens and varying internet speeds make transcoding essential. And with [adaptive bitrate streaming](#), you can deliver the highest-quality stream possible to internet-connected TVs and mobile users alike.

While the bulk of the work is complete at playback, a few final considerations remain: recording, monetization, and security.

Recording

Live streams disappear in the blink of an eye. But they don't have to.

While recorded videos won't match the immediacy and authenticity of a live stream, it's still useful to have these files on hand for later use. That way, you can edit the video assets; enable immediate pause, rewind, and fast-forward capabilities; and repurpose them for marketing collateral or internal reference.

Sports fans often review game-changing plays during cable broadcasts, so why not provide your end users the same freedom when viewing your live streams? [nDVR technology](#) allows users to record a live stream while simultaneously playing, pausing, resuming, and rewinding the recorded stream. Content distributors can also further extend audience reach by transcoding a single set of recorded source files into multiple formats for streaming to any player and device.

Monetization

A successful stream is a revenue stream. Regardless of whether you're crowdsourcing content from users via a mobile live streaming app or producing it yourself, profitability often revolves around the ability to monetize content.

Live streaming monetization comes in many shapes and sizes.

Whereas surveillance streaming devices like doorbell cameras and service-oriented applications like virtual doctor visits generate revenue from different means altogether, some tactics can be employed across a range of industries. Below are the three most straightforward models for converting content into cash.

Subscriptions

Viewers are moving toward subscription-based content consumption due to the quality, lack of interruptions, and ability to take an à la carte approach to content selection. When an affordable monthly payment means the difference between low-brow, ad-riddled fluff and sought-after, commercial-free content — it's a no-brainer.

Services like Netflix and HBO Go have championed subscription-based monetization.

When using this model, the onus is on the provider to deliver content worthy of the investment. [Adaptive bitrate streaming](#) underpins this requirement, allowing content distributors to compete with industry giants on the basis of video quality and user experience.

Paywalls help restrict access to subscribed viewers, resulting in a recurring revenue for your business.

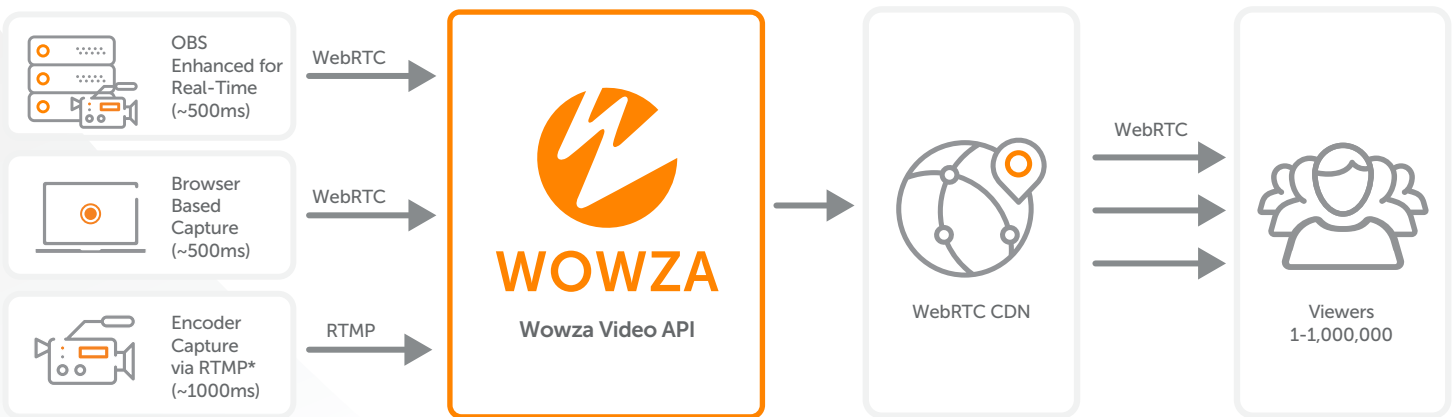
Transactions — Paywalls and Pay-per-View Services

Gone are the days when seeing an event in real time meant you actually had to be there. Paywalls also enable pay-per-view revenue generation for live sporting events, concerts, and more.

Cord-cutting viewers expect the same on-demand live options that traditional satellite and cable services provide, which opens up opportunities for live streaming broadcasts.

Successfully monetizing these live events depends on two things: quick scalability for viewership spikes and minimal latency for near-real-time delivery.

Technologies like Wowza's [Real-Time Streaming at Scale](#) feature help content distributors fulfill both of these requirements, allowing OTT to compete directly with cable in terms of latency and viewership.



*U.S.-based ingest

Advertising

Rather than generating revenue directly from viewers, many video service operators turn to advertising. Unlike traditional TV — where the content and ads come packaged together — internet streaming opens opportunities for user-specific advertisements based on the demographics of an individual viewer.

In addition to [inserting ads into live streams](#), [text and graphic overlays](#) can be used for branding, advertising, or adding a call to action.

Security

Security and monetization go hand in hand. Unauthorized access can compromise the revenue-boosting strategies of subscription and pay-per-view services. And when streaming content contains sensitive or private data, protection is key.

Secure streaming success with [content protection at every step of the workflow](#). The primary ways to safeguard live streams are authentication, encryption, and digital rights management (DRM).

- **Authentication:** User authentication ensures that only authorized users — like those who've paid or registered — have access to content.

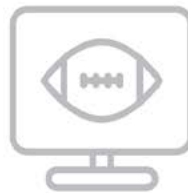
- **Encryption:** In online banking and email access, encryption is used to scramble the messages exchanged between your browser and the servers. This same process of ciphering and deciphering helps ensure that streaming content remains secure while travelling across the public internet.
- **Digital Rights Management (DRM):** Digital rights management (DRM) technologies can be used to encrypt premium content. Unlike simple encryption, DRM workflows rely on both DRM vendors and third-party providers for more sophisticated protection.

During encoding, each video file is encrypted with media keys from one or more DRM providers. Once the user attempts to access the live video, the player client communicates directly with the DRM server to confirm that the license is valid. Upon validation, the player gains access to the decryption key for playback.

Use Cases



82% of internet traffic will be video by 2022ⁱ



5.7m viewers per minute live streamed Super Bowl LVIIIⁱⁱ



404k+ hours of content are streamed on Netflix every minuteⁱⁱⁱ

Technological advancements, improved connectivity, and video maturity make live streaming more accessible than ever. Capitalizing on this trend means incorporating the technology into your brand's entire strategy.

Examples of successful companies doing just that abound. Drive-through banking, virtual doctor visits, and other forms of customer support use live video streaming to better engage customers and extend the physical reach of their employees.

Whether video is an integral part of the services your organization offers or a supplementary component aimed at increasing engagement, it's delivering measurable impact for today's most innovative companies.

Marketing requires it. Customer engagement depends on it. And innovation starts with it. Here are some interesting ways that brands are using live streaming today.

ⁱ <https://www.fiercevideo.com/cable/live-video-usage-will-increase-15-fold-by-2021-cisco-predicts>

ⁱⁱ <https://www.nfl.com/news/super-bowl-lv-draws-audience-of-96-4-million-viewers>

ⁱⁱⁱ <https://www.statista.com/statistics/195140/new-user-generated-content-uploaded-by-users-per-minute/>

Interactive Software and Apps: Engaging Remote Users

[Integrating live video into online games and interactive events](#) helps improve audience engagement. This results in more opportunities to maximize revenue through paid subscribers, sponsorship, and advertising.

From live trivia to real-time auctions, interactive video-based mobile apps have mushroomed in popularity. These broadcasts tend to be shorter in length at approximately 10-20 minutes, while inciting more active participation from users than any other format.

Rather than “breaking the fourth wall,” interactive video experiences presume that there are no walls to begin with. So, keeping the conversation going requires a continuous feedback loop between the content and the end users.

Adding these interactive overlays often involves synchronizing timed metadata, chat functionality, polling, and two-way live video. This integration is key to luring in users.



Across all types of interactive live streaming, one thing remains true: minimizing latency is a must. That’s where solutions like [WebRTC](#), [low-latency CMAF](#), and [SRT](#) help up the ante.

E-Commerce: Bridging the Gap Between Digital and In-Store Engagement

When the founder of Zappos first pitched the idea of selling shoes online, the logic of ‘try before you buy’ made investors skeptical. Fast forward twenty years, and Zappos’ parent company Amazon is every bit as popular as any brick-and-mortar chain.

But what if you could replicate the tactile aspects of in-store shopping via a digital format? By fusing online retail with live streaming, you can.

Dubbed live [commerce](#), the trend holds a strong foothold in China. Taobao — the world’s biggest e-commerce website — empowers farmers, business owners, and self-employed entrepreneurs with consumer-to-consumer live streaming. Successful influencers on the site ascend to celebrity-like status, helping elevate the shopping experience from mundane to swanky. And just like at a Nordstrom beauty counter, consumers can ask questions and learn application techniques.

Although certainly late to the party, Amazon launched [Amazon Live](#) in 2019, where models try on different accessories and outfits while hosts vividly describe the material and fit. Think home shopping network goes digital, with the addition of icons fluttering on the screen each time a purchase is made.

No, these aren’t exact replicas of the in-store shopping experience. But for now, they’re the closest possible alternative. And with technologies like [5G](#), virtual [reality](#), and artificial intelligence, the possibilities are endless.



Live Events: The Next Closest Thing to Being There

Whether it's an instantly sold-out reunion concert, the sporting event of the year, or a breakthrough in space exploration, [live events](#) are one of the most obvious reasons to use live streaming technology.

Just think of the time [Felix Baumgartner broke the record for the highest freefall](#) — while simultaneously breaking YouTube's record for the largest number of concurrent live streams.

The COVID-19 pandemic transformed live events forever, bringing virtual and hybrid events into living rooms across the world. And virtual reality promises to transport viewers even closer to these events by enabling immersive 360-degree experiences.



IoT and Surveillance: Security Cameras Go Mainstream

IoT video streaming dominates the market. Coastguards use drones for search-and-rescue missions, whereas military-grade bodycams help connect frontline responders with their commander.

Recently, though, IoT streaming has moved beyond these niche applications and into the consumer world. Examples of IoT and video surveillance include pet monitors, security systems, doorbell, cameras,

[IP cameras](#), and more. Just take [Nest's](#) word for it, "For everything you need, there's a Nest Cam."

Live streaming surveillance is also going mobile. [Carbyne](#) designed a next-generation 911 platform that combines live video chat with dynamic location information for users making emergency calls from mobile devices. For COVID-19 response, traffic accidents, and everything in between, real-time surveillance technologies are transforming the way we monitor and respond to emergencies.



Our Tech = Your Superpower

In the content-flooded landscape — where digital interfaces have largely replaced in-person interactions — live streaming is one of the most authentic ways to connect with your audience. While a lot goes into getting that live stream from capture to playback, we designed our [software](#), [services](#), and [hardware](#) to help you along the way.

Wowza Streaming Engine™ Software:

Stream on your terms and to any device. Our downloadable media server software powers live and on-demand streaming, on-premises or in the cloud, with fully customizable software.

Wowza Video:

Quickly build live streaming applications with our fully managed cloud service. The flexible platform gets you up and running in a snap, and can be used from capture to playback or as part of a custom solution.

Wowza CDN:

Reach global audiences of any size with our streaming delivery network. Automatically included with Wowza Video, the Wowza CDN can also be configured as a stream target for Wowza Streaming Engine.

Push the limits of what's possible with live streaming technology. Streamline your streaming workflow with Wowza.

