

Kaltura Media Transcoding Services and Technology

Last Modified on 05/22/2024 12:06 pm IDT

 This article is designated for administrators.

About

Kaltura Media Transcoding Services offers powerful cloud-based transcoding microservices for seamless encoding workflows across web, broadcast, studio, and enterprise applications. Designed to handle any input type and file size, it supports large volumes of jobs, from simple web delivery to complex production workflows, converting various input formats into multiple output renditions.

With Kaltura's scalable Video Platform as a Service, you can avoid the costs and complexities of maintaining transcoding hardware. Our service optimizes performance for any input, delivery network, or playback device, ensuring the best video experience with support for the latest formats, codecs, CDNs, and encryption standards.

Kaltura Media Transcoding Services, built on an open, API-driven architecture, offers robust microservices and API Client Libraries and SDKs. This simplifies creating complex workflows while providing full control and extensibility through backend plugins.

Our technology is deployable on any infrastructure, whether on-premises or public cloud, and comes with provisioning, monitoring, diagnostic tools, and backend management APIs, allowing cloud providers to run their own media transcoding services.

Ingest from any source - transcode for any target

With new devices and formats emerging constantly, keeping up with industry shifts can be challenging. Kaltura Media Transcoding Services handles it all, from studio gear to webcams, conferencing apps to satellite feeds. Our always up-to-date transcoding services ensure the best quality viewing experience, optimized for the latest formats, codecs, and standards. Let us handle the complexities so you don't have to.

Integrated Encoding Engines

The following lists the various tools and encoding engines currently integrated and orchestrated (see [Kaltura Decision Layer](#)) in the Kaltura Media Transcoding Services:

- Common Video Encoding Engines: FFMPEG, Mencoder, VLC
- Proprietary Video Encoding/Encryption Engines: Microsoft Expression Encoder, QuickTimeTools, WebexNbrplayer, Widevine

- Video Segmentation and Delivery Optimization Engines: Mp4box, FastStart, Segmenter, ISM Index, ISM Manifest, SmilManifest, SmoothProtect
- Image Conversion: ImageMagick
- Document Conversion: PPT 2 Image, PDF 2 SWF, PDF Creator
- Experimental Video Features: Third party encoders
- Deprecated or Legacy Video Encoders

Each encoding engine is optimized for quality, speed, reliability, and cost-efficiency. This ensures minimal file sizes without compromising quality, reducing storage and bandwidth costs while delivering a superior viewing experience. Engines are selected based on source media, target specs, and delivery standards, with fallback options available according to transcoding profile settings.

Commonly used video formats and codecs

For a list of commonly used media formats and codecs that were tested and optimized for transcoding and delivery by Kaltura, see [Supported Transcoding Formats](#).

Additional codecs and formats are supported by the underlying encoder engines and can be ingested and transcoded by Kaltura. For further details and discovery regarding a specific codec or format that are not listed, please [contact us](#).

The Kaltura Decision Layer

Kaltura transcoding workflows take the source file and transcoding profile (representing the list of the required flavors) as an input. The output is the transcoded videos as outlined in the transcoding profile. A flavor represents the set of parameters (format, codecs, bitrates, and other parameters such as video manipulation like trimming or scaling). The assets are generated to match the settings specified for each flavor in the transcoding profile.

The transcoding workflow is orchestrated by the **Kaltura Decision Layer** - a mechanism that optimizes the choice of encoder, parameters used and fail-over encoders for a given source media and desired output flavors.

These are the five stages of the transcoding process:

- **Analysis** - Synopsis of the source media file
- **Intermediate-Source Processing** - Pre-processing for non-standard media and handling of proprietary formats and codecs
- **Optimization** - Decision tree that optimizes for quality and efficiency
- **Asset Generation** - Distributed batch transcoding of the source into each desired flavor

- **Validation** - Testing and verification of generated flavors against source specifications and expected standards

Analysis

The transcoding process begins by analyzing the source media's characteristics. Based on this analysis, the Kaltura Decision Layer determines if intermediate-source processing is needed. The system then matches the best encoding settings for the source and specified transcoding profile. Typically, the bitrate and frame size of the output should not exceed those of the source file, though these settings can be customized in the [transcoding profile settings](#).

Intermediate-source processing

With some proprietary formats and codecs, a dedicated hardware, operating system or software may be required to handle the transcoding of such files. In such cases Kaltura will leverage the dedicated technology to convert the ingested proprietary source into an intermediate-source (file of non-proprietary format/codec, usually MP4/H.264 or WMV) and then continue the processing as usual.

- Webex/ARF - [Cisco's WebEx Network Recording Tool](#) will be used to convert the proprietary WebEx format before continuing with the media transcoding process.
- GoToMeeting Codec - Windows Expression Encoder 4 will be used to convert the proprietary GoToMeeting codec before continuing with the media transcoding process.
- QuickTime Video (QT) - MAC machines will be used to convert the proprietary QT format before continuing with the media transcoding process.

In addition to processing 'special source formats', an intermediate-source is also used to handle known video issues, such as handling artifacts in Digital Video sources, adding silent audio tracks for assets that will be ciphered by Widevine DRM, and other cases.

Optimization

After analyzing all the flavors, the transcoding process begins to optimize the following aspects of the video transcoding:

- Bitrate - To prevent generation of assets with similar/close bitrates and avoid reduction in quality and cost-efficiency.
- Frame Size - There should be at least one flavor that will match the source frame size as closely as possible. For example: this is important for "presentations style" videos that are of a low bitrate and large frame sizes.

- Delivery Types - There should be at least one flavor for every delivery type (Web, HLS, ...) to ensure smooth and reliable delivery.

The redundant flavors are marked as Non Applicable (i.e. output flavors that should not be created) for the next step, [Asset Generation](#).

Asset Generation

The [Kaltura batch system](#) executes the asset generation. In this phase, each of the flavors specified in the transcoding profile will be executed as transcoding jobs in parallel and according to priority and resources availability by the designated encoder. In case of a specific transcoding tool failure, the batch worker attempts to use the rest of the fallback transcoders that are available for the specific flavor.

When an asset transcoding completed successfully, the newly transcoded asset goes into the validation phase.

Validation

Based on the source and the flavors, the asset is checked for existence of the required content streams and the correct duration or any errors from the transcoder. If an asset has been found to be invalid or faulty, the next defined fall-back transcoding engine is used to re-try the conversion.

Media encryption and Digital Rights Management (DRM)

Protecting media content from unauthorized access is crucial for both public and corporate videos. DRM (Digital Rights Management) ensures media is used as intended. Kaltura Media Transcoding Services integrates with Google Widevine and Microsoft PlayReady DRM technologies, simplifying multi-DRM management and media encryption as part of the media ingestion and preparation workflow. This ensures your content remains secure and properly managed.

Transcoding prioritization

Whether you're running a private small operation or large public transcoding cloud - Kaltura is built to provide the architecture that serves your needs from a low volume to millions of transcoding jobs daily. With scale, and different deployment architectures, comes the need to control priority queues and even provide reserved instances. Kaltura Media Transcoding Services is built with multi-tenancy and quality of service management in mind.

The weighting of the different factors outlined here may be configured to maximize:

1. **Throughput** - Maximize the speed at which transcoding jobs are completed and


utilization of the available system resources. Here the most dominating factor is the video duration.

2. **Fairness** - Balance the number of running jobs between publisher accounts in the Kaltura deployment and prevent "starvation" of accounts. In this algorithm the dominant factors are the number of currently running jobs per account and the Account Class of Service.

The ratio between these two strategies is configurable per Kaltura deployment (100 = maximum throughput, 0 = maximum fairness).

Video transcoding jobs prioritization factors

The job prioritization mechanism takes into account the following factors:

 Note that each of the factors may be configured and tweaked for influence per Kaltura deployment.

1. **The duration of the source video** - For the following reasons, transcoding jobs of shorter duration generally have a higher priority.
 - The users expectations usually correlate to the video duration: a user who uploads a 5 min video generally expects the video to be ready in a matter of minutes, while a user who uploads an hour long video will expect it to take more time.
 - The duration of a video provides a rough estimate for the execution time required to process the transcoding of that video.
2. **Impact on "Entry Readiness"** - Transcoding of flavors that are marked as "required" as for the entry status to be READY, are assigned higher priority than flavors marked as "optional". If all flavors are marked as optional, the lowest bitrate flavor will implicitly be treated as required. (Read more on [Entry Readiness settings](#)).
3. **The ingestion / upload method** - Videos that were singularly ingested using the upload API (e.g. via browser or mobile app upload) will be assigned a higher priority than videos ingested as part of a bulk upload ([using CSV / XML bulk upload](#)).
4. **The ingesting account overhead** - As more transcoding jobs will be executed under a single account, the pending jobs in the account's growing queue will be assigned a lower priority to avoid clogging the queue with a single account's queue.
5. **Account Class of Service** - Accounts can be assigned classes of service to indicate priority treatment. A high-priority account that has 20 running jobs may get a higher priority than a low-priority account that has 10 running jobs.
6. **User assigned priority** - Priority can be set per flavor in a specific transcoding

profile to indicate the importance of flavors (for example, when the mobile app has more views than your website, publishers can configure the iPhone and iPad flavors to have a higher priority than the web flavors). It is also possible to configure different priorities between an accounts transcoding profiles, to enable the content uploader to choose the priority of their media entry. Transcoding priority can be set via API when manually queuing conversion jobs.

Other priority controls and transcoding management tools

1. **Job Quota** - A quota can be configured to limit the number of concurrently running jobs in the Kaltura deployment. There is a system-wide default as well as defaults for specific job types. It is also possible to configure the quota differently per each account.
2. **Express Queue** - There are several batch workers that are limited to small transcoding jobs (i.e. configured to transcode short duration videos exclusively). This ensures a short wait time for shorter videos, even when many long videos are ingested to the system.
3. **Dedicated (Reserved) Transcoders** - It is possible to configure workers that process only jobs of certain publisher accounts. This allows publishers to reserve dedicated transcoders for their jobs (for example, 8 dedicated transcoders guarantees immediate processing for 8 transcoding jobs at any given time, and any additional jobs will be handled by the shared transcoding queue). This ensures that time-sensitive publishing workflows always have reserved resources in deployments where system resources may be constrained.
4. **Priority Boost** - Kaltura platform operators, can boost the priority of specific transcoding jobs from within the Admin Console, to manually increase the priority of a waiting job.
5. **Moving Jobs** - Transcoding jobs can be balanced between Kaltura Data Centers. As such, if one DC has a longer-than-usual queue of transcoding jobs, it is possible to move jobs to another Kaltura DC after the source video file was synchronized to the other DC.
6. **Remote Transcoding** - Kaltura's batch framework and distributed API-based architecture allows for running transcoding jobs remotely as an extension of the Kaltura deployment. For example, a load on a private local Kaltura deployment gets higher than usual priority, Kaltura operators can provision machines on a public cloud (such as IBM Bluemix, Amazon Web Services, Google Cloud, or Rackspace) and easily configure these remote Kaltura batch workers to pull the source files from the local Kaltura deployment, perform the transcoding job, and push the transcoded assets back to the local deployment. Batch workers use the Kaltura API

(and never access the data base directly), as such remote transcoding is easy to setup, secure and automate, and there is no requirement to setup VPN tunnels or mess with complex network security configurations.

Transcoding times and quality of service

For applications and workflows where fast turnaround is essential, such as sports, breaking news, surveillance, Ad Tech, and social media, fast video queueing time and processing times are highly important to satisfy user expectations and support critical operations.

To ensure best quality and smallest file size optimization the transcoding time of a given source file depends highly on the type of content in the video; For example, a highly dynamic scene, with changing lighting and high level of detail (e.g. nature walk) will achieve different results than a static scene (e.g. talking head) or a scene with low level of details (e.g. text slides). On a single transcoding worker (without segmented parallel transcoding) typical transcoding times are as follows:

- For 360p video, transcoding will usually be up to real time (i.e. equal to the duration of the source video).
- For 720p video, transcoding will usually be up to between 4x and 8x the video duration.
- For 1080p video, transcoding will usually be up to between 10x to 15x of the video duration.

These values represent the Kaltura wide upper level performance values. Longer transcoding times should not be treated as errors or as failures. The transcoding time depends on many parameters, for example, content complexity, traffic load, processing load, resources availability and others, and therefore the times may vary.

Please contact your Kaltura Representative to get information of the average processing times for your account.

Beyond transcoding time and quality, the following considerations are built into Kaltura's architecture to ensure optimal quality of service and SLA.

- **Queuing Time:** For 90% of all videos uploaded to Kaltura SaaS, video transcoding begins within 3 seconds after upload complete. For 10% of videos, transcoding generally begins within 3 seconds to 3 minutes.
- **Playback Availability:** With Kaltura, all video is transcoded in parallel, utilizing as many system resources and workers available to the Kaltura deployment. Video is generally available as soon as the first flavor is available and for high-priority

immediate / live requirements Kaltura utilizes dedicated transcoders, and priority settings.

- **Scale:** With Kaltura, publishers can define unlimited number of transcoding profiles, and execute unlimited number of transcoding jobs.
- **Control:** With Kaltura, publishers can set a wide variety of configurations to customize and optimize the transcoding workflows for their business needs and requirements.

Video file encryption (i.e. data-at-rest or Widevine) is encrypted at a much higher rate than video transcoding, and generally is faster than real-time.

Programmatic media authoring API and tools

Format and codec transcoding is merely where your journey with video begins. To support the many video experiences and workflows required for video operations, Kaltura Media Transcoding Services feature a wide range of authoring and editing tools, ranging from thumbnail generation through Image watermarks and overlays, sub-clip creation and trimming. Use the UI tools for quick tweaking of your videos, or the Kaltura APIs to build creative video manipulations at scale.

The following lists a few of the common video authoring scenarios achievable with the Kaltura Media Transcoding Services:

Real-time thumbnails creation

The Kaltura Thumbnail API provides a simple web interface to dynamically generate image snapshots of Kaltura video entries on the fly. Generated images are generated upon demand, and caching is managed on disk and CDN levels. The result of the thumbnail API is a JPEG, PNG or GIF image, and provides many image manipulation tools, including;

- Image re-sizing and frame cropping.
- Extracting specific frame from a video clip on.
- Thumbnail versions caching.
- Various compression quality controls.
- Generation of JPG sprites for optimal delivery and use as CSS sprites.
- Applying overlays and templates.

Watermarking

When defining a new video flavor, publishers can include an image overlay for watermarking setting using PNG or JPG image formats as the watermark. The watermark image will be burnt on the video flavor at the specified coordinates and

dimensions (relatively to video frame borders). Scaling, opacity and transparency are also supported. The watermark image can be another Kaltura entry of type image, or a URL pointing to the watermark image.

Sub-clipping and trimming

Publishers can leverage the clipping and trimming tool to define an offset and duration, that in turn either creates a new clip of the source video, or trims the beginning and end of the video.

Audio enhancements

- **Multiple audio streams** - Upon detection of a source containing multiple-audio streams, the transcoding logic attempts to match it with known audio layouts (for example, [5.1](#), [7.1](#)). If such a match is detected, the following can be performed:
 - The multi-channel audio can be down-mixed to stereo.
 - The first channel can be chosen as the audio source.
 - The flavor can be customized to force a specific audio channels mapping.
- **Noise reduction and normalizing volume level** - through custom settings on the flavor level, noise reduction filters as well as volume level settings can be applied to the transcoded assets.

Frame rotation

Rotation is often an important setting with videos recorded on mobile device. To ensure correct orientation of uploaded video, Kaltura automatically detects camera rotation and rotates the video frame accordingly to ensure mobile captured videos are always viewable on the correct rotation.

Advanced video coding

Depending on the chosen video transcoding engine, Kaltura exposes a wide range of capabilities including controlling the gopsize, frame rate, bitrate, 2-pass as well as automatic filters such as [deinterlacing](#) and more.

Batch architecture and backend plugins

Kaltura Media Transcoding Services operate as Kaltura batch workers. For more details, see [Introduction to Kaltura Batch Processes](#). Our services are built on the Kaltura Backend Plugins Framework, ensuring easy extensibility for adding new transcoding engines and capabilities.

Transcoding server plugins are automatically configured and registered with Kaltura batch managers, which activate the plugin worker for assigned job types. These plugins

extend encoding engine APIs, using command-line or web interfaces, to perform and monitor transcoding tasks. Engines range from command-line tools like ffmpeg to API-based encoders like Harmonic. Once a job is completed, the plugin reports the status and outputs to the batch manager.

Dedicated transcoding

What is Dedicated Transcoding?

Under normal circumstances, when an asset is uploaded into Kaltura, the video the client uploads is put into a first-in-first-out transcode queue with other (non-dedicated) Kaltura customers (note: batch actions are given lower priority). Dedicated transcoding aims to ensure the video the client uploads is transcoded immediately, and not queued.

Terminology

1. Job - the work of transcoding a single source video file to a single flavor.
2. Worker - a process that executes a single job at a time, for example, a machine that has 10 workers will have at most 10 running jobs in parallel.

What is a VCore?

- A “core” is a single Central Processing Unit (CPU), basically the part of the server that does the main processing of the software logic.
- A virtual machine (VM) is where the functions of a single machine (or server) are simulated instead of being directly tied to a single physical server.
- Putting both of these concepts together (VM and Core) helps understand what a VCore (or virtual core) is.
- You can have a single physical server with six physical cores. You could then have three VMs running on this one physical server, where each VM has two VCores. Each VM “thinks and acts” like it is a single physical machine with two cores.

Since ffmpeg can work with several threads, on a 20 core server it is not recommended to run 20 ffmpeg's, it may be better to run, for example, only 10. If we use 20 workers, then yes 20 videos are guaranteed to start processing immediately, but they will run slower than if we use only 10 workers.

How many videos can be transcoded at the same time?

- The number of videos (x) that can be transcoded simultaneously is directly proportional to the number of VCores (y).
- The client is therefore only guaranteed that x videos will transcode with zero wait time if they also have y VCores (i.e. where $x=y$).

- If the client uploads more than x videos at once, and only has y VCores (where $y < x$), then some videos go into the normal Kaltura transcode queue with all other client's videos.

Can some videos from a single KMC use the dedicated transcoders and others not?

- Often not all content within a single KMC needs to be treated with the same priority.
- Certain urgent content (such as news or gossip content) can be transcoded immediately via the dedicated transcoders.
- The standard, less urgent content can then go through the normal queue.

How does dedicated transcoding work?

- The single transcode of an asset is known as a job.
- A job (and therefore transcode) can be assigned a priority.
- Kaltura can assign a set number of VCores to only work on jobs that have a given priority (and KMC ID).
- Individual transcode profiles can be associated with individual priorities. Kaltura can also go down to the flavor level if needed. For example, the iPad flavor will use dedicated transcoding and the iPhone flavor will not.
- Since uploaded videos are assigned transcode profiles, by extension therefore, uploaded videos can be assigned to specific jobs (i.e. certain VCores).
- If you want to have some videos trafficked through the dedicated transcoders (such as news) and other less urgent videos added to the normal queue, you need a different transcode profile to be assigned to the specific video.
- The end result: if you set a given transcode profile to a newly created entry, the entry will be prioritised accordingly.

What are the steps to implement dedicated transcoding?

Assigning the priorities to flavors / conversion profiles can only be done by Kaltura at this point (can't be done via API)

1. Create a transcode profile to be used for urgent videos, i.e. news content.
2. Create a transcode profile to be used for non-urgent videos, i.e. normal content.
3. Programmatically, insert the appropriate transcode profile ID into the dropfolder XML based on the type of content.
4. The content is sent to either a queued job or dedicated job, depending on the associated transcode profile ID.

What information does Kaltura require from the client?

1. At which data center do they want the VCores applied? The options are: just the west coast, just the east coast or split between the two (for example, if the customer bought 30 CPUs, we can have 20 in the EC and 10 in the WC. This decision should be based on the DC the client most often gets data from, and the geo location of the most uploads.
2. KMC ID
3. Number of VCores
 - As described, the number of videos (x) that can be transcoded simultaneously is directly proportional to the number of VCores (y).
 - For there to be no wait time, x must equal.
 - For example, if the client wants to be able to transcode 10 videos at the same time they will need 10 VCores.
4. Transcode Profile ID (only relevant if segregating content into queued vs dedicated for a single KMC).

For additional information about dedicated transcoding, please contact your Kaltura Account Manager.
