

Kaltura Media Transcoding Services and Technology

Kaltura Media Transcoding Services provides robust, production grade cloud transcoding microservices and tools to manage seamless encoding workflows at any scale and quality requirements, for the web, broadcast, studio, or secure internal enterprise applications. Architected to handle any input type and file size over large volumes of jobs, Kaltura Media Transcoding Services handle any job from simple web delivery to complex production workflows converting any input format of uploaded video, audio, image and even documents into a variety of flavors (transcoded output renditions).

With Kaltura's infinitely scalable enterprise-grade Video Platform as a Service, our media transcoding services free you of the expense and hassle of maintaining your own transcoding hardware, tuning performance, and doing the guesswork of which architecture, encoder engines or settings work best for your input files, delivery networks or playback devices. We continuously monitor and optimize the transcoding cloud to support the latest formats and codecs, CDNs, encryption and delivery standards, ensuring your viewers always get the best video experience on any device.

True to Kaltura's Open and Interoperable values and API driven architecture, Kaltura Media Transcoding Services are consumable through robust set of microservices and the Kaltura API Client Libraries and SDKs, simplifying the creation of complex workflows alongside full control and extensibility through Kaltura's backend plugins mechanism.

Like the rest of the Kaltura Platform, the underlying technology of the Kaltura Media Transcoding Services was built to be deployed on any infrastructure, on premises or public cloud. It is provided with provisioning, monitoring and detailed diagnostic tools as well as robust set of backend management APIs allowing cloud providers to run its own media transcoding services.

Ingest From Any Source, Transcode For Any Target

New devices, cameras and input sources are introduced every month, from studio gear to webcam through conferencing and webcasting applications, to surveillance cameras and satellite feeds. The ever growing innovations in media bring challenges with new formats, codecs and standards that require constant monitoring of the industry shifts and continuous optimization of hardware, software and practices to ensure your users always get the best quality viewing experience. Kaltura Media Transcoding Services free you from these worries by providing always up to date transcoding services, optimized to the latest formats, codecs and standards.

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: Harmonic Encoders
- Deprecated or Legacy Video Encoders: Inlet Armada

Each encoding engine is designated with the tasks it does best, and is optimized for quality, speed, reliability and cost-efficiency (saving on file size without compromising quality which in turn reduces publisher's costs on storage and bandwidth and delivers a superior viewing experience). Engines are leveraged according to the source media, target media specs and expected delivery standard, some are often used as fall-back engines to each other according to the transcoding profile settings.

Commonly Used Video Formats and Codecs

The following lists commonly used media formats and codecs that were tested and optimized for transcoding and delivery by Kaltura. 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](#).

Ingest (Input) File Formats and Codecs

The following video formats and codecs have been tested and optimized for optimal transcoding input:

Media File Formats: Mpeg-4 and QuickTime Formats (MP4, MOV, QT, and M4V), Flash Video (FLV and F4V), Microsoft Windows Formats (AVI, ASF, WMV and WMA), MPEG-1/2 (MPG, M1V, M2V, MP3), WAV, Matroska (MKV), OGG OGM & OGV, WEBM, 3GP, RM, Webex (ARF), MXF

Video Codecs: DivX (Div3/4/5, DX50), DV, H.263, H.264 and AVC, MPEG-4 Visual, MPEG-1/2, MJPG, MP42/3, IV40/50 (Indeo codecs), RV30/40, RMVB, FLV1/4, VP3/5/6/7/8/9, Sorenson (SVQ1/3), Xvid, Theora, WMV1/2/3, VC1, ProRes 422, ICOD, DVCPRO, PXL, TCSS/TCS2,

GoToMeeting Codec (G2M3/4)

Audio Codecs: MP3, MP1/2, AC3, AAC, Vorbis, AMR, PCM, WMA7/8/9, WMSpeech, FLAC, QDM2, RA, Nellymoser, Cook, GSM, SPEEX

Target (Output) Video and Audio Codecs

The following video formats and codecs have been tested and optimized as optimal transcoding output and delivery:

Media File Formats: MP4, FLV, AVI, 3GP, OGG, MKV, WMV, WMA, Microsoft Silverlight Smooth Streaming, Apple HTTP Live Streaming (HLS), WEBM, MPEG, MPEG TS, WAV, M4V, MXF

Video Codecs: H.265, H.264, H.263, MPEG-4 Visual, VP9, VP8, VP6, FLV1, Theora, WMV1/2, VC1, MPEG-2, APCN/S/O/H, DNXHD, DV

Audio Codecs: MP3, AAC LC, AAC HE, Vorbis, WMA/Pro, AMR NB, MPEG2, AC3, PCM

Delivery-specific Formats

Kaltura supports various streaming standards and protocols, see the [Video Delivery Settings](#) article for more information about the types of video delivery formats support and supported delivery settings in Kaltura.

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 first step in the transcoding process is to analyze the source media characteristics and parameters. Based on the source parameters, the Kaltura Decision Layer decides whether there is a need for an intermediate-source processing phase. The transcoding logic then attempts to match the best encoding settings for that source and for the specific flavor in the transcoding profile specified. For example, in most cases, the bitrate and the frame size of the generated flavors should not exceed the source file bitrate and frame size. These decision settings can be overridden 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/codecs, usually MP4/H.264 or WMV) and then continue the processing as usual.

For example (read: [Best Practices For Uploading Content Created Using Screenshot Tools](#)):

- 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 viewing and copying is often critical to video operations. Whether for public facing paid content or sensitive corporate videos, DRM (Digital Rights Management) offers controls needed to ensure media is used only as intended. Integrated and certified with Google Widevine (Setup guide) and Microsoft PlayReady DRM technologies for content protection, Kaltura Media Transcoding Services simplifies the management of multi-DRM solution and media encryption process as an integrated step of the Kaltura media ingestion and preparation workflow.

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

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

- **Queueing 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 (e.g. [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

The Kaltura Media Transcoding Services are implemented as a Kaltura batch workers. [See Introduction to Kaltura Batch Processes](#) to learn more about batch processes and the Kaltura batch architecture.

To ensure ease of extensibility and allow for adding new transcoding engines and capabilities, the Kaltura Media Transcoding Services are built on the [Kaltura Backend Plugins framework](#). Transcoding server plugins are configured and registered automatically with the Kaltura batch managers, who activate the plugin worker for all job types that are assigned to that backend plugin.

When activated, the plugin implementation extends the specific encoding engine API (often command line interface, or web interface) to perform, manage and monitor the required transcoding tasks it needs to carry. Transcoding server plugins run engines that vary from command line binary tools (for example ffmpeg) to API based encoder integrations (for example using Harmonic Encoders' proprietary API). Upon completion of the transcoding job, the plugin informs the batch manager of completion, any error states and the output asset files.

Parallel Segmented Transcoding - 2015 Roadmap Highlight

Transcoding of long-form HD quality and higher video source may take very long processing times, often around 10x the duration of the video. To better utilize system resources and enhance performance, Kaltura is introducing "Parallel Segmented Transcoding" - by splitting the source video into small segments (1-5min duration each) and distributing the transcoding jobs across many workers in parallel. Moreover, as each segment will be processed independently of the other segments, transcoding long files can be paused and resumed without the need to restart the transcoding of the entire source file, providing more reliable and fault-tolerant transcoding for large files. Upon completion of all required segments, the segments are merged into a single file. The merging process is significantly shorter, compared to transcoding times.

Among the key benefits of Parallel Segmented Transcoding are:

- Optimal resource utilization - by taking advantage of idle workers.
- Better priority and queue management - no more lengthy jobs that hold the queue back.
- Shorter Time-to-Ready - real-time and faster-than-real-time transcoding times.
- Shorter Time-to-Playback - ability to start playback of partial media files with on-the-fly segments packaging and delivery.