

Wowza Video REST API release notes

Updated on 06/15/2023 11:34 am PDT

Versions

At this time, API versions have the following statuses:

| API version | Status |
|-------------|-------------|
| 1.11 | Beta |
| 1.10 | Current |
| 1.9 | Supported 1 |
| 1.8 | Supported 2 |

See [Wowza Video REST API lifecycle management](#) for more information about API versions and statuses.

Changelog

June 15, 2023

We've updated all our live streams to use Wowza Flowplayer as the default player. If you use an embed code that was built using Wowza Video Player, you'll need to update your site to use the new one we've generated for you. Call GET /live_streams/{id} to get the player's embed_code.

June 5, 2023

Beta (v1.11)

- Added GET /usage/real_time_streams/summary endpoint to provide you with the [usage](#) data for real-time streams.
- Added a new endpoint GET /analytics/ingest/live_streams/{id} to provide you with [ingest analytics](#) for a live stream.
 - With this change, we've sunset the following endpoints and the associated operations are no longer available.
 - GET /live_streams/{id}/stats
 - GET /transcoders/{id}/stats
 - GET /transcoders/{id}/uptimes/{id}/metrics/current
 - GET /transcoders/{id}/uptimes/{id}/metrics/historic

May 30, 2023

- Changed uptime_id to transcoder_uptime_id in the recording deleted event for webhooks. While uptime_id is included in the payload currently, it is deprecated and will eventually be removed. Use transcoder_uptime_id instead.

Beta (v1.11), Current (v1.10), Supported 1 (v1.9)

- Real-Time Streaming at Scale is improved to make real-time streaming to your global audiences even more powerful! When you create a real-time stream, select the region your video source is closest to in order to further minimize latency and improve performance. Available regions are phoenix, amsterdam, singapore, bangalore, and auto. If you send auto for the region parameter, Wowza Video will select the region based

on the publisher's location.

Existing streams that use the deprecated values of americas and emea are automatically re-routed. americas will go to phoenix and emea will go to amsterdam.

Use the updated region parameter in [POST /real_time](#).

May 18, 2023

All versions (v1.8 through 1.11)

- Fixed a bug where PATCH /stream_targets/custom/{id} did not remove the password for RTMP authentication of a custom stream target that was created using POST /stream_targets/custom endpoint. To clear both username and password for RTMP authentication, send empty strings for both username and password fields when calling PATCH /stream_targets/custom/{id}.

May 17, 2023

All versions (v1.8 through 1.11)

- We've changed the rendition naming in /usage for stream targets and VOD streams and rendition data in /analytics/viewers (version 1.11). A rendition name is now the smaller number of the resolution in which an asset was streamed. A resolution is [width]x[height]. For more information, see [GET /usage/stream_targets/{id}/renditions](#).

Beta (v 1.11)

- Added renditions and device data for the live analytics/viewers endpoints.

May 11, 2023

Beta (v 1.11)

- We've added four new endpoints to provide you with live and historic viewer data for an account, a live stream, a VOD stream, and a stream target. The data returned includes viewers by device, rendition, and country, as well as viewer trend data.
 - GET /analytics/viewers/account
 - GET /analytics/viewers/live_streams/{id}
 - GET /analytics/viewers/vod_streams/{id}
 - GET /analytics/viewers/stream_targets/{id}

All versions (v1.8 through 1.11)

- Fixed an issue where some boolean attributes in the REST API were set to true when the request had invalid string values. Now, the only acceptable values for boolean attributes are true, false, "true", "false", 1, and 0.
- Fixed an issue where trying to update an existing asset name to longer than 255 characters didn't return the correct error. If you exceed the 255 character limit, you'll now receive a 422 error with the following message: The request couldn't be processed. Validation failed: Title is too long (maximum is 255 characters).

May 8, 2023

We've published developer.wowza.com! Soon to be your one-stop site for all things Wowza integration. During this first iteration, we've added Wowza Flowplayer developer documentation and the Wowza Video REST API reference to the developer portal. We'll continue to add developer documentation for all of our products, as well as development tools, resources, support, and so much more!

April 27, 2023

All versions (v1.7 through 1.11)

- Fixed an issue where enabling the countdown clock for a hosted page didn't set the correct countdown time.

April 26, 2023

- Fixed an issue where stream health data, which you can access through the stream health endpoints in `/transcoders`, wasn't returned for some streams.

April 24, 2023

Beta (v1.11)

Added reason to `GET /vod_streams/ID` and `GET /vod_streams` to show why a VOD stream failed the processing phase.

April 17, 2023

Stream health data, which you can access through the stream health endpoints in `/transcoders`, is retained for 90 days now rather than 1 year.

Example: `/transcoders/{transcoder_id}/uptimes/{id}/metrics/historic`

April 6, 2023

Beta (v1.11)

Player customizations and plugins

Added customizations and plugins for the following endpoints which allow you to customize the player appearance and enable casting to Airplay and Chromecast devices.

- `POST /live_streams`
- `GET /live_streams/{id}`
- `PATCH /live_streams/{id}`
- `GET /players/{id}`
- `PATCH /players/{id}`

Improvements to Asset Management

- Changed the name of `playback_enabled` to `published` for the following endpoints:
 - `GET /assets`
 - `GET /assets/{id}`
 - `PATCH /assets/{id}`
- Added reason to `GET /asset/ID` and `GET /assets` to show why an asset failed the processing phase.
- You can now delete an asset in a processing state. When you delete an asset, any associated VOD streams and recordings are deleted.

Other updates

- The Wowza Flowplayer instance used in the hosted page and in the embed code for your live stream now recognizes and displays closed captions.
- Fixed an issue where custom stream targets didn't honor special characters, like `?`, in the RTMP URL.

March 15, 2023

Beta (v1.10)

Live streams

We've changed the request and response format for configuring the hosted page and the player. This change groups the configuration options for each object together in a more structured way.

- [POST /live_streams](#)
- [PATCH /live_streams/{ID}](#)
- [GET /live_streams/{ID}](#)

| v1.9 | v1.10 |
|------|--|
| | <pre> POST /live_streams { "live_stream": { "name": "MyLiveStream" ... "hosted_page": { "enabled": true, "page_title": "MyLiveHostedPage", "description": "This is my hosted page.", "logo_image": "[Base64-encoded string representation of GIF, JPEG, or PNG file]", "sharing_icons": true } ... "player": { "countdown": true, "countdown_at": "2020-02-01T17:00:00.000Z.", "logo_image": "[Base64-encoded string representation of GIF, JPEG, or PNG file]", "logo_position": "top-left", "responsive": true, "type": "wowza_flowplayer", "video_poster_image": "[Base64-encoded string representation of GIF, JPEG, or PNG file]", "width": 640 } ... } } Response { "live_stream": { "name": "MyLiveStream" ... "hosted_page": { "enabled": true, "title": "MyLiveHostedPage", "description": "This is my hosted page.", "logo_image_url": "https://prod.s3.amazonaws.com/uploads/player/hosted_page_logo_image/23424/5bad28.jpg", "sharing_icons": true, "page_url": "https://player.video.wowza.com/hosted/e8dk5bf6/player.html" } ... "player": { "countdown": true, "countdown_at": "2020-02-01T17:00:00.000Z.", "embed_code": "null", "hls_playback_url": "https://wowzasubdomain.wowza.com/1/TWhoL3BiZnJXFhmNzZVN3JrZDAwUT09/ZmYxSXRrTERrUlk9/hls/live/playlist.m3u8", "id": "kwb6pwnv", "logo_image_url": "https://prod.s3.amazonaws.com/uploads/player/logo_image/23424/5bad28.jpg" } } } </pre> |

```

POST /live_streams
{
  "live_stream": {
    "name": "MyLiveStream"
    ...
    "hosted_page": true,
    "hosted_page_title": "MyLiveHostedPage",
    "hosted_page_description": "This is my hosted page.",
    "hosted_page_logo_image": "[Base64-encoded string r
representation of GIF, JPEG, or PNG file]",
    "hosted_page_sharing_icons": true,
    ...
    "player_countdown": true,
    "player_countdown_at": "2020-02-01T17:00:00.000Z.",
    "player_logo_image": "[Base64-encoded string represe
ntation of GIF, JPEG, or PNG file]",
    "player_logo_position": "top-left",
    "player_responsive": true,
    "player_type": "wowza_flowplayer",
    "player_video_poster_image": "[Base64-encoded string
representation of GIF, JPEG, or PNG file]",
    "player_width": 640,
    ...
  }
}

```

Response

```

{
  "live_stream": {
    "name": "MyLiveStream"
    ...
    "hosted_page": true,
    "hosted_page_title": "MyLiveHostedPage",
    "hosted_page_description": "This is my hosted page.",
    "hosted_page_logo_image_url": "https://prod.s3.amazo
naws.com/uploads/player/hosted_page_logo_image/234
24/5bad28.jpg",
    "hosted_page_sharing_icons": true,
    "hosted_page_url": "https://player.video.wowza.com/
hosted/e8dk5bf6/player.html",
    ...

```

```

P "player_countdown": true,
W "player_countdown_at": "2020-02-01T17:00:00.000Z.",
C "player_embed_code": "null",
owza.com/1/TWhoL3BiZnJXFhmNzZVN3JrZDAwUT09/Z
mYxSXRrTERrUlk9/hls/live/playlist.m3u8",
"player_id": "kwb6pwnv ",
"player_logo_image_url": "https://prod.s3.amazonaws.c
om/uploads/player/logo_image/23424/5bad28.jpg",
"player_logo_position": "top-left",
"player_responsive": true,
"player_type": "wowza_flowplayer",
"player_video_poster_image_url": "https://prod.s3.ama
zonaws.com/uploads/player/video_poster_image/23424/
5bad28.jpg",
"player_width": 640,
...
}
}

```

```

"logo_position": "top-left",
"responsive": true,
"type": "wowza_flowplayer",
"video_poster_image_url": "https://prod.s3.amazonaw
s.com/uploads/player/video_poster_image/23424/5bad2
8.jpg",
"width": 640
}
...
}
}

```

onfiguring the hosted page. This change groups the ay.

v1.10

| v1.9 | v1.10 |
|---|--|
| <pre>PATCH /players { "player": { "hosted_page": true, "hosted_page_title": "MyLiveHostedPage", "hosted_page_description": "This is my hosted page.", "hosted_page_logo_image": "[Base64-encoded string r representation of GIF, JPEG, or PNG file]", "hosted_page_sharing_icons": true, ... "responsive": true, } } Response { "player": { "id": "e8dk5bf6 " ... "hosted_page": true, "hosted_page_title": "MyLiveHostedPage", "hosted_page_description": "This is my hosted page.", "hosted_page_logo_image_url": "https://prod.s3.amazo naws.com/uploads/player/hosted_page_logo_image/234 24/5bad28.jpg", "hosted_page_sharing_icons": true, "hosted_page_url": " "https://player.video.wowza.com/ hosted/e8dk5bf6/player.html", ... "player_responsive": true, ... } }</pre> | <pre>PATCH /players { "player": { "hosted_page": { "enabled": true, "page_title": "MyLiveHostedPage", "description": "This is my hosted page.", "logo_image": "[Base64-encoded string representatio n of GIF, JPEG, or PNG file]", "sharing_icons": true } ... "responsive": true, ... } } Response { "player": { "id": "e8dk5bf6 " ... "hosted_page": { "enabled":true, "title": "MyLiveHostedPage", "description": "This is my hosted page.", "logo_image_url": "https://prod.s3.amazonaws.com/u ploads/player/hosted_page_logo_image/23424/5bad28.j pg", "sharing_icons": true, "page_url": " "https://player.video.wowza.com/hosted/ e8dk5bf6/player.html" } ... "responsive": true, ... } }</pre> |

API Versioned

The REST API has also been versioned:

- **Beta (v1.11)**
 - Established v1.11 as the base path for the beta version of the API.
- **Current (v1.10)**
 - Established v1.10 as the current version of the API. For a list of new features, improvements, and breaking changes, see the [Wowza Video REST API migration guide](#).
- **Supported 1 (v1.9)**
 - Established v1.9 as a supported version of the API.
- **Supported 2 (v1.8)**
 - Established v1.8 as an older supported version of the API. The second supported version is no longer part of our active code base and the functionality is frozen. This version is offered as a convenience as you plan and execute your migration to the current version.
- **Deprecated (v1.7)**
 - Established v1.7 as deprecated and it is no longer guaranteed to work in production environments.
- **Discontinued (v1.6)**
 - Established v1.6 as discontinued. A discontinued version is no longer accessible. This stage was formerly

known as **Sunset**.

March 1, 2023

Beta (v1.10)

- Updated DELETE /assets/{id} to allow for deleting assets in the processing state. Previously, you could only delete an asset in the uploading, failed, or completed state. When an asset is deleted, any related recordings and VOD streams are removed.

February 28, 2023

- We've improved our encoding process to better support 7.1 multi-channel audio for the delivery of superior audio quality to viewers. With these changes, multi-channel input sources produce matching output renditions that are encoded at higher bitrates. Properly mixed-down stereo renditions are also created to avoid audio distortions.
- Fixed an issue that prevented audio-only MP4 files from being processed correctly when uploaded to Asset Management.

February 7, 2023

Beta (v1.10), Current (v1.9), Supported 1 (v1.8)

- **Wowza Video subscribers:** When you record a real-time stream, an MP4 file and an HLS stream for video-on-demand (VOD) replay are created. VOD streams can take a bit of processing time as multiple renditions are created. If you only need the MP4, you can save processing time by setting `disable_vod_encoder` to true on POST /real_time. See [POST /real_time](#).
 - Beta (v1.10) Added `disable_vod_encoder` to the following endpoints:
 - POST /real_time
 - PATCH /real_time/ID and GET /real_time/ID responses
 - Beta (v1.10), Current (v1.9), Supported 1 (v1.8) Added `real_time_stream_id` to the following endpoints:
 - GET /recordings
 - GET /recordings/ID

Beta (v1.10), Current (v1.9)

- Removed region from the response of GET /real_time.

January 18, 2023

Beta (v1.10), Current (v1.9), Supported 1 (v1.8), Supported 2 (v1.7)

- For Wowza Video subscribers, we added Wowza Flowplayer, an easy-to-use, commercial-grade video player designed for builders and developers. It provides HTML5, HLS, and MPEG-DASH playback and plays streams on most browsers and devices through either a Wowza Video hosted page or your own site. Highly [customizable](#) and lightweight, Wowza Flowplayer enables innovative and scalable video playback for many use cases.

This player is bundled into Wowza Video with no need for a third-party player. It offers broad support with SDKs for iOS, tvOS, and Android. When you embed the player in your own site, you can customize Wowza Flowplayer with 30+ available plugins providing enhanced capabilities, including video playback through Airplay and Chromecast.

You're a subscriber if you have access to Asset Management and Historic and Live Analytics in the user interface. Original HTML5 and Wowza Player remain the only valid options for other subscribers.

- Added default value `wowza_flowplayer` (Wowza Flowplayer) to the `player_type` property for the following

Live stream endpoints:

- POST /live_streams
- GET /live_streams/{id}
- PATCH /live_streams/{id}
- Added default value wowza_flowplayer (Wowza Flowplayer) to the type property for the following Players endpoints:
 - GET /players
 - GET /players/{id}
 - PATCH /players/{id}
- If you're a Wowza Video subscriber, you can update an existing live stream's player to wowza_flowplayer, but won't be able to change it later because Wowza Flowplayer is the only option if you are a subscriber.

December 15, 2022

All versions (v1.6 through 1.10)

- You can now add up to around 40 IP addresses to a block or allow list on a Fastly stream target. See [Geo-block Wowza CDN on Fastly stream targets with the Wowza Video REST API](#) for more information.
- Changed the API server response cache time from 5 seconds to 10 seconds for the following endpoints:
 - live_stream/ID/stats
 - transcoders/ID/stats

See [Wowza Video REST API server response caching](#) for more information.

- Fixed an issue where sending an emoji in the file_name of POST for /assets and in the name and description of PATCH for /assets/{id} returns a 500 error. A 422 error is returned now instead
- Transcoders and live streams created in a trial account can stream for 20 minutes now, up from 15 minutes.

Beta (v1.10)

- You can now use either the filter or the the query parameter to search and filter responses from GET /assets. Examples:
 - **Filter** — [https://api.video.wowza.com/api/beta/assets?filter\[0\]\[field\]=created_at&filter\[0\]\[gte\]=2021-07-14T17:47:45.000Z](https://api.video.wowza.com/api/beta/assets?filter[0][field]=created_at&filter[0][gte]=2021-07-14T17:47:45.000Z)
 - **Query** — https://api.video.wowza.com/api/beta/assets?sort_direction=desc&sort_column=created_at&per_page=30&state=completed&query=sample

See [Get filtered query results with the Wowza Video REST API](#) for more information about using the query parameter. Note that this topic reflects only the functionality for v1.9 of the API; however, it does contain helpful information for using the query parameter with the /assets endpoint. The topic will be updated when v1.10 moves to current stage.

December 12, 2022

- Set a limit for the number of total API requests that you can make within a certain time period, to protect resources and stability. A 429 "Too many requests" error is returned when the limit is exceeded.

December 5, 2022

- Posted new installers for **OBS for Real-Time Streaming at Scale**. If you have Real-Time Streaming at Scale enabled on your account, you can use these installers to send WebRTC source from OBS. You can find the download locations in [Deliver real-time streams with the Wowza Video Rest API](#).

December 1, 2022

Beta (v1.10)

- Fixed an issue where the CMS endpoints using v1.10 in all the requests return `:"message": "API version is no longer supported"`. All the requests now return correct status and payload for an endpoint.

November 21, 2022

Special characters in asset and recording file names

We fixed some issues encountered with special characters in asset and recording file names. Some special characters prevented real-time streams from being recorded while others prevented some recordings from being deleted. As a result, Wowza Video now removes or replaces special characters in file names. Consider using only the following characters when uploading an MP4 in Asset Management or naming a real-time stream, because real-time streams use the stream name as the file name.

- Letters: A-Z, a-z
- Numbers: 0-9
- Special characters: `_ - . ()`

November 16, 2022

- We've enhanced our [Wowza Video free trial](#) experience to get you up and streaming quickly!
 - Simply submit your email to start a trial. No credit card is needed.
 - Complete an interactive tutorial within Wowza Video to get started integrating with our REST API through our Postman live stream tutorial.
 - Our revamped welcome email gives you what you need to explore Wowza Video.

November 15, 2022

- Fixed an issue where successful upload and processing of an asset resulted in an incorrectly displayed **Failed** state.

November 3, 2022

Beta (v1.10), Current (v1.9)

- Real-Time Streaming at Scale now lets you record streams to share or replay later! When you create a real-time stream, configure it to record and store the stream as an MP4 file that you can download for sharing offline. If you're a [Wowza Video subscriber](#), recordings are saved in Asset Management which also gives you an HLS stream for video-on-demand (VOD) replay.

Use the new recording parameter in [POST /real_time](#).

Beta (v1.10), Current (v1.9), Supported 1 (v1.8)

- When you create an asset, it is uploaded to Oracle storage instead of Google. The upload url, download url, and thumbnail url are updated to reflect the Oracle storage URL in the following endpoints.
 - `POST /assets`
 - `GET /assets`
 - `GET /assets/{id}`
 - `PATCH /assets/{id}`
 - `PATCH /assets/{id}/upload_completed`

October 12, 2022

All versions (v1.6 through 1.10)

A value for video_codec returned in the following endpoints is changing from avc1 to h264:

- GET /transcoders/{transcoder_id}/uptimes/{id}/metrics/current
- GET /transcoders/{transcoder_id}/stats

The change is rolling out on a region-by-region basis and will be complete as soon as possible, likely within the next week.

October 6, 2022

All versions (v1.6 through 1.10)

- Fixed an issue where making parallel requests to POST /real_time might cause the token returned in the response to be a duplicate or to be missing.
- Fixed an issue where attempts to update the broadcast_location with PATCH /live_streams/ID resulted in a 200 status rather than a 422 status with the following message: The request couldn't be processed. Broadcast Location cannot be modified. You can't modify the broadcast location after you set it when you create a live stream.
- Fixed an issue where an uploaded asset that contained a space in the file name resulted in an error when re-streaming.

September 29, 2022

v1.9

- Real-Time Streaming at Scale is improved to make real-time streaming to your global audiences even more powerful!
 - **Fine-tune global performance** — When you create a real-time stream, select the region your video source is closest to in order to further minimize latency and improve performance. Available regions are the Americas and Europe/Middle East/Africa (EMEA).

Use the new region parameter in [POST /real_time](#).

- **Access usage data within Wowza Video** — You can now track per stream and account-level usage data for real-time streams within Wowza Video. Access this data from the **Usage** menu in the **Actions** column of the Real-Time Streams page or the **Real-Time Streams** tab of the Account Usage page.

Use the new [GET /usage/real_time_streams](#) and [GET /usage/real_time_streams/ID](#) endpoints.

- The following updates have been made to the /assets resource:
 - Removed null values from the following asset endpoints:
 - GET /assets/ID - The fields returned vary depending on when you send this request. For example, if a resource hasn't been created yet during the processing state, you won't see **available_renditions** in the response.
 - PATCH /assets/ID - The fields returned vary depending on when you send this request. For example, if a stream doesn't have unique viewer data, you won't see **unique_viewers** in the response
 - PATCH /assets/ID/upload_completed
 - PATCH /assets/ID/upload_failed
 - Added upload_url to the response of the GET /assets/ID request when the state of the asset is uploading. If your original upload URL expired and you need a new one for an asset, send this request and the response will contain a new upload URL.
 - Changed DELETE /assets/ID so you can now delete an asset in the uploading state in addition to the complete and failed states. You still can't delete an asset in the processing state.

v1.8

- Fixed an issue where POST /assets returned a 200 OK status instead of a 201 Created status.

API Versioned

The REST API has also been versioned:

Note: Wowza Video transitioned to a JSON Web Token-based authentication scheme, rather than an API key/access key scheme. JWT-based authentication is now the only authentication method in v1.9 and later. See [Authentication](#) for more information.

- **Beta (v1.10)**
 - Established v1.10 as the base path for the beta version of the API.
- **Current (v1.9)**
 - Established v1.9 as the current version of the API. For a list of new features, improvements, and breaking changes, see the [Wowza Video REST API migration guide](#).
- **Supported 1 (v1.8)**
 - Established v1.8 as a supported version of the API.
- **Supported 2 (v1.7)**
 - Established v1.7 as an older supported version of the API. The second supported version is no longer part of our active code base and the functionality is frozen. This version is offered as a convenience as you plan and execute your migration to the current version.
- **Deprecated (v1.6)**
 - Established v1.6 as deprecated and it is no longer guaranteed to work in production environments.
- **Discontinued (v1.5)**
 - Established v1.5 as discontinued. A discontinued version is no longer accessible. This stage was formerly known as **Sunset**.

September 13, 2022

- We'll be replacing the existing IP addresses for Wowza CDN on Fastly stream targets. While the change hasn't gone into effect yet, if you stream directly to Wowza CDN from Wowza Streaming Engine, you should add these IP addresses to your firewall rules so your integration continues to work seamlessly when the change occurs. When we've completed the transition to these new IP addresses, you should remove the old addresses from your rules. We'll add information to these release notes when the transition is complete so you know when to remove the old addresses. We expect the IP addresses below to remain static moving forward.
 - 129.153.100.158
 - 129.153.114.13
 - 129.154.207.194
 - 132.145.253.238
 - 138.3.255.30
 - 141.147.1.186
 - 146.56.156.1
 - 158.101.43.210
 - 193.123.241.42
- See [Send a stream from Wowza Streaming Engine to a Wowza CDN HLS target using Wowza REST APIs](#) to learn more.

August 23, 2022

- Posted new installers for **OBS for Real-Time Streaming at Scale**. If you have Real-Time Streaming at Scale enabled on your account, you can use these installers to send WebRTC source from OBS. You can find the download locations in [Deliver real-time streams with the Wowza Video Rest API](#).

August 11, 2022

- Added region to the POST /real_time endpoint so you can specify a distribution region closest to your broadcast location. The valid values are emea and americas. Previously existing streams default to

americas.

August 10, 2022

- Added response caching to the following endpoints to reduce bandwidth, latency, and the load on servers. See [Wowza Video REST API server response caching](#) for more information.
 - GET /assets - 1 second
 - GET /assets/ID - 1 second
 - GET asset_tags - 1 second
 - GET transcoders/{transcoder_id}/uptimes/{id}/metrics/current - 5 seconds
 - GET transcoders/{transcoder_id}/uptimes/{id}/metrics/historic - 5 seconds

Beta (v1.9), Current (v1.8), Supported 1 (v1.7), Supported 2 (v1.6)

- Changed player embed codes and hosted pages to use video in the URL for new and updated players, instead of cloud. Existing URLs that use cloud will continue to work and are redirected to the Video URL.
- Removed akamai, akamai_rtmp, limelight, and ustream as valid values for custom stream targets. The valid values are akamai_cupertino, rtmp, and rtmps.
- Fixed an issue where updating an asset name using PATCH /assets/ID disabled the video preview on the asset detail page in the UI.

June 28, 2022

- Greatly improved performance for the number of uploaded assets we can process and transcode at a time.

June 23, 2022

- Discontinued use of JapanEast region for stream sources. Stream sources in that area will use the next closest region.

May 25, 2022

Beta (v1.9), Current (v1.8), Supported 1 (v1.7), Supported 2 (v1.6)

- Deprecated the ability to create Wowza CDN on Akamai stream targets, which includes deprecating the following endpoints and parameters/values:
 - POST /stream_targets/akamai
 - delivery_type for the live stream endpoints can only be single-bitrate, which is the default. multi-bitrate is deprecated.

May 17, 2022

Beta (v1.9), Current (v1.8), Supported 1 (v1.7), Supported 2 (v1.6)

For Wowza Video subscribers, we added Wowza Video Player, a robust, industry standard player that provides HTML5, HLS, and MPEG-DASH playback and plays streams on any browser and device. It can be customized when you embed it in your site.

Wowza Video Player is the only player option for Wowza Video subscribers. You're a subscriber if you have access to Asset Management and Historic and Live Analytics in the user interface. Original HTML5 and Wowza Player remain the only valid options for other subscribers.

- Added default value wowza_video_player (Wowza Video Player) to the player_type property for the following Live stream endpoints:
 - POST /live_streams
 - GET /live_streams/{id}

- PATCH /live_streams/{id}
- Added default value wowza_video_player (Wowza Video Player) to the type property for the following Players endpoints:
 - GET /players
 - GET /players/{id}
 - PATCH /players/{id}

If you're a Wowza Video subscriber, you can update an existing live stream's player to wowza_video_player, but won't be able to change it later. Also, you can't update from wowza_player to original_html5 or vice versa if you're a subscriber.

- Added error 422 with a Record Invalid error when a user who does not have a Wowza Video subscription attempts to update or create a livestream using wowza_video_player or update a player using wowza_video_player.

April 19, 2022

Wowza Streaming Cloud REST API is now Wowza Video REST API!

While our URLs transition from cloud.wowza.com to video.wowza.com, you might notice a mix of both domains. Either will work, but we expect to fully transition to video.wowza.com soon.

Example: wss://f5dbf6.entrypoint.cloud.wowza.com/webrtc-session.json

Beta (v1.8)

The following are changes made to the 1.8 beta version before 1.8 was established as the current branch of the API.

- **New endpoints**
 - **Account usage**
 - GET /usage/accounts
 - GET /usage/accounts/{id}/live
 - **VOD stream usage**
 - GET /usage/vod_streams/{id}/live
 - **Transcoder usage**
 - GET /usage/transcoders/countries
 - GET /usage/transcoders/{id}/live
 - **Assets** - Operations related to assets stored in Asset Management.
 - POST /assets
 - GET /assets
 - GET /assets/{id}
 - PATCH /assets/{id}
 - DEL /asset/{id}
 - POST /assets/{id}/live_streams
 - GET /asset_tags
 - PATCH /assets/{id}/upload_completed
 - PATCH /assets/{id}/upload_fail

You must obtain a license Asset Management to add this capability to your new or existing account. [Contact us](#) for more information.

- **Updated endpoints**
 - Added save_asset as an input field for the following endpoints:
 - POST /live_stream
 - PATCH /live_stream/{id}

- Added save_asset as a response field for the following endpoints:
 - POST live_stream
 - PATCH /live_stream/{id}
 - GET /live_stream/{id}
- Added asset_id, the ID of the object in Asset Management, to the following endpoints:
 - GET /transcoder/{id}
 - GET /live_stream/{id}
 - GET /recordings/{id}
 - GET /vod_streams/{id}
- Added countries and viewer_trend to GET /usage/stream_targets/[ID]/live.
- Added egress_seconds_additional_targets the following endpoints:
 - /usage/transcoders
 - /usage/transcoders/{id}
- Added unique_viewers to the following endpoints:
 - /usage/transcoders/{id}
- Renamed seconds to egress_seconds and bytes to egress_bytes for the following endpoints:
 - /usage/transcoders/summary
 - /usage/transcoders
 - /usage/transcoders/{id}
- **New transcoder properties**
 - **Section:** asset_management, **key:** enabled, **value:** true/false
- The REST API has also been versioned:
 - **Beta (v1.9)**
 - Established v1.9 as the base path for the beta version of the API.
 - **Current (v1.8)**
 - Established v1.8 as the current version of the API. For a list of new features, improvements, and breaking changes, see the [Wowza Video REST API migration guide](#).
 - **Supported 1 (v1.7)**
 - Established v1.7 as a supported version of the API.
 - **Supported 2 (v1.6)**
 - Established v1.6 as an older supported version of the API. The second supported version is no longer part of our active code base and the functionality is frozen. This version is offered as a convenience as you plan and execute your migration to the current version.
 - **Deprecated (v1.5)**
 - Established v1.5 as deprecated and it is no longer guaranteed to work in production environments.

April 1, 2022

- Removed all ultra low latency endpoints.

March 23, 2022

Beta (v1.8)

- Added new Analytics endpoints:
 - /usage/accounts/countries - Shows usage details by country for an account.
 - /usage/accounts/trend - Shows viewing trend data for an account.
 - /usage/transcoders/id/trend - Shows viewing trend data for a transcoder.
 - /usage/vod_streams/id/trend - Shows viewing trend data for a VOD stream.
- Added new fields to the /usage/vod_streams/{id} endpoint:
 - egress_bytes
 - egress_seconds

March 14, 2022

Beta (v1.8)

- Added **last_billing_date** as an option for querying /usage endpoints.
- Added filtering of results to the following endpoints:
 - /live_streams
 - /real_time
 - /recordings
 - /schedules
 - /stream_sources
 - /stream_targets
 - /vod_streams
- See [Get filtered query results with the Wowza Video REST API](#) for more information.
- Added added new fields to the following endpoints

| New field | Applicable endpoint |
|---------------------|--|
| avg_viewing_seconds | <ul style="list-style-type: none">◦ /usage/vod_streams◦ /usage/vod_streams/{id}◦ /usage/vod_streams/summary◦ /usage/vod_streams/{id}/countries◦ /usage/vod_streams/{id}/renditions◦ /usage/stream_targets◦ /usage/stream_targets/{id}◦ /usage/stream_targets/summary◦ /usage/stream_targets/{id}/countries◦ /usage/stream_targets/{id}/renditions |
| state | <ul style="list-style-type: none">◦ /transcoders◦ /vod_streams◦ /stream_targets |
| playback_enabled | <ul style="list-style-type: none">◦ /vod_streams |

- Renamed the fields in the responses from the following endpoints

| Old name | New name | Applicable endpoint |
|----------|---------------|--|
| bytes | viewing_bytes | <ul style="list-style-type: none">◦ /usage/vod_streams◦ /usage/vod_streams/{id}◦ /usage/vod_streams/summary◦ /usage/vod_streams/{id}/countries◦ /usage/vod_streams/{id}/renditions◦ /usage/stream_targets◦ /usage/stream_targets/{id}◦ /usage/stream_targets/summary◦ /usage/stream_targets/{id}/countries◦ /usage/stream_targets/{id}/renditions |

| | | |
|--------------|-----------------|---|
| viewing_time | viewing_seconds | <ul style="list-style-type: none"> ◦ /usage/vod_streams ◦ /usage/vod_streams/{id} ◦ /usage/vod_streams/summary ◦ /usage/vod_streams/{id}/countries ◦ /usage/vod_streams/{id}/renditions ◦ /usage/stream_targets ◦ /usage/stream_targets/{id} ◦ /usage/stream_targets/summary ◦ /usage/stream_targets/{id}/countries ◦ /usage/stream_targets/{id}/renditions |
| bytes | egress_bytes | <ul style="list-style-type: none"> ◦ /usage/vod_streams/egress ◦ /usage/vod_streams/egress/{id} ◦ /usage/vod_streams/egress/summary |

March 2, 2022

Beta (v1.8)

Wowza Video is transitioning to a JSON Web Token-based authentication scheme, rather than an API key/access key scheme. JWT-based authentication is now available in the beta version (1.8) of the API. Although you'll be able to authenticate both ways while we transition, we encourage you to update your integrations to use JWT as soon as possible.

Migration timeline

The following table shows what authentication schemes upcoming API versions will support when they move to the **Current** stage.

| API Version | Authentication Support |
|-------------|---------------------------------|
| 1.7 | API key + access key, HMAC |
| 1.8 | API key + access key, HMAC, JWT |
| 1.9 | JWT |

See the [Lifecycle schedule](#) for anticipated delivery dates.

Authenticating with a JWT

To authenticate with a JWT, you'll need to do the following:

1. Create an access token, which is the JWT, in the [Token Management](#) portal. For convenience, you can also access the portal from the **Access Key** page in Wowza Video.
 - **Personal access tokens** — Available to all users with a Wowza Video license. These tokens are specific to a user and can be used in API calls to do quick tests or manual actions that don't need to persist if the user leaves the organization.
 - **System access tokens** — Available to organization owners. These tokens are specific to the organization and transcend individual users who come into and leave an organization. For this reason, system access tokens are best used for integrations that go into production environments and need to persist after specific users leave an organization.
2. Use the JWT in your REST API requests as a bearer token in an Authentication header.

Sample request

```
curl -X POST \
-H "Content-Type: application/json" \
-H "Authorization: Bearer <JWT>" \
-d '{
  "transcoder": {
    "billing_mode": "pay_as_you_go",
    "broadcast_location": "us_west_california",
    "buffer_size": 4000,
    "delivery_method": "push",
    "low_latency": true,
    "name": "MyABRtranscoder",
    "protocol": "rtsp",
    "transcoder_type": "transcoded"
  }
}' "${WSC_HOST}/api/beta/transcoders"
```

Note: Code samples throughout the Wowza Video documentation will be updated to reflect JWT-based authentication when v1.8 moves to the **Current** stage.

December 20, 2021

All versions (v1.5 through 1.8)

- Fixed an issue with webhook integrations not resetting after failures correctly.

December 9, 2021

All versions (v1.5 through 1.8)

- Fixed an issue with webhooks being sent for deleted transcoders and recordings.

November 1, 2021

Beta (v1.8)

- Added the ability to add a security token that must be passed by viewers for playback to the following endpoints:
 - GET /real_time
 - PATCH /real_time/ID

October 19, 2021

Beta (v1.8)

- Established v1.8 as the base path for the beta version of the API.

Current (v1.7)

- Established v1.7 as the current version of the API. For a list of new features, improvements, and breaking changes, see the [Wowza Video REST API migration guide](#).

Supported 1 (v1.6)

- Established v1.6 as a supported version of the API.

Supported 2 (v1.5)

- Established v1.5 as an older supported version of the API. The second supported version is no longer part of

our active code base and the functionality is frozen. This version is offered as a convenience as you plan and execute your migration to the current version.

Additional changes in Beta (v1.8) and Current (v1.7)

- Added the ability to fetch VOD egress usage from the following new endpoints:
 - GET /usage/vod_streams/egress
 - GET /usage/vod_streams/egress/{id}
 - GET /usage/vod_streams/egress/summary
- Removed the zones property from the responses of the following endpoints:
 - GET /usage/vod_streams/{id}
 - GET /usage/vod_streams/summary
 - GET /usage/vod_streams/{id}/countries
 - GET /usage/vod_streams/{id}/renditions

June 1, 2021

- Discontinued the API sandbox environment.

May 10, 2021

- Fixed an issue where stream targets with a primary and backup URL weren't streaming to the backup URL. Additionally, if a stream target has a backup URL, when you add the stream target to an output rendition, both the primary and backup versions of the target are now added. Remove the backup URL entry if you don't want to use it.

April 20, 2021

- MPEG-DASH live streams now use the playlistSeconds property to set the amount of time viewers are able to rewind the live stream.

April 16, 2021

- Fixed an issue where an encoder appears to be connected to an ultra low latency stream target, but there's no stream for playback.
- Fixed an issue where creating a live stream didn't always generate the expected player.

April 12, 2021

- Added reason_code and reason to the payload of the failed recording webhook. See [Wowza Video webhook event reference documentation](#) for more information.

Beta (v1.7)

Added reference_id to the following endpoints:

- POST /transcoders
- PATCH /transcoders/[ID]
- GET /transcoders/[ID]
- POST /live_streams
- PATCH /live_streams/[ID]
- GET /live_streams/[ID]

Use this to set a unique ID returned in object_data of transcoder webhook payloads. This is useful if you have an ID in your system or application you want to associate with transcoder events.

The default value for reference_id in the webhook payload is null.

Example request:

```
curl -X POST \
-H "Content-Type: application/json" \
-H "wsc-api-key: ${WSC_API_KEY}" \
-H "wsc-access-key: ${WSC_ACCESS_KEY}" \
-d '{
  "transcoder": {
    "billing_mode": "pay_as_you_go",
    "broadcast_location": "us_west_california",
    "buffer_size": 4000,
    "delivery_method": "push",
    "low_latency": true,
    "name": "MyABRtranscoder",
    "protocol": "rtsp",
    "transcoder_type": "transcoded",
    "reference_id": "mySystemID_01"
  }
}' "${WSC_HOST}/api/v1.7/transcoders"
```

Example transcoder webhook payload:

```
{
  "version": "1.0",
  "event": "stop.complete",
  "event_id": "67360faf-b709-4e2c-a846-1d130f643449",
  "event_time": 1617908933.0713735,
  "object_type": "transcoder",
  "object_id": "y5mfswdy",
  "object_data": {
    "uptime_id": "jbf6npg",
    "reference_id": "mySystemID_01"
  }
}
```

See the REST API documentation for [transcoders](#) and [live streams](#) for more information.

April 5, 2021

Beta (v1.7)

Added the ability to set an array of properties from the following endpoints:

- POST /transcoders/*[[ID]]*/properties
- POST /stream_targets/fastly
- PATCH /stream_targets/fastly/*[[ID]]*
- POST /stream_targets/akamai
- PATCH /stream_targets/akamai/*[[ID]]*
- POST /stream_targets/*[[ID]]*/properties

The following endpoints already allowed setting arrays of properties and the API reference documentation is updated to reflect this:

- POST /transcoders
- PATCH /transcoders/*[[ID]]*

All versions (v1.7 through 1.4)

- Fixed an issue where PATCH /transcoders returned a 500 error if you tried to set transcoder properties that were set previously.

April 1, 2021

Sunset (v1.3)

Version 1.3 of the Wowza Video REST API is sunset and no longer available. For assistance migrating to the current version of the API (v1.6), see [Wowza Video REST API migration guide](#).

March 29, 2021

Beta (v1.7)

- Added functionality to the Wowza Video REST API that allows you to set a referer policy and require a Referer header in order to access a stream through a Wowza CDN on Fastly stream target. Use this functionality to prevent unauthorized websites from hosting a stream you own. The following parameters are added to the /stream_targets/fastly endpoints:

- referer_enabled
- referer_allow_empty
- referer_policy
- referer_domains

- **Example request:**

```
curl -X POST \  
-H "Content-Type: application/json" \  
-H "wsc-api-key: ${WSC_API_KEY}" \  
-H "wsc-access-key: ${WSC_ACCESS_KEY}" \  
-d '{  
  "stream_target_fastly": {  
    "name": "MyHLSTarget",  
    "referer_enabled": true,  
    "referer_policy": "allow",  
    "referer_allow_empty": false,  
    "referer_domains": "example.com, example2.com"  
  }  
' "${WSC_HOST}/api/v1.7/stream_targets/fastly"
```

See the [REST API reference documentation](#) for Fastly stream targets for more information.

- When geoblock_enabled is **false**, we no longer return the other geoblock parameters in /stream_targets/fastly responses. geoblock_enabled: false continues to be returned.

March 9, 2021

- The API sandbox, , will be sunset and no longer available for use starting June 1, 2021. Contact wsc-sandbox@wowza.com for questions, comments, or concerns.

February 23, 2021

- Added Wowza Video REST API integration with EZDRM, a third-party digital rights management (DRM) service you can use to protect content from unauthorized viewing. You can use EZDRM with the REST API to protect streams on Apple/iOS, Google, and Microsoft devices. See [About digital rights management in Wowza Video](#) for more information.

Beta (v1.7), Current (v1.6), Supported (v1.5)

- Added the following new transcoder properties, in the new ezdrm section, to configure your stream for DRM protection with Wowza Video and EZDRM:
 - username
 - password
 - fairPlayAssetId
 - wideVineContentId
- Added a new transcoder property, allowDirectPlayback, that you can use to block direct playback of RTMP

streams. Use with DRM to ensure only devices and platforms that can decrypt your stream can access it.

Beta (v1.7)

- Added `delivery_protocols` to the request for the following endpoints:
 - `POST /stream_targets/fastly`
 - `PATCH /stream_targets/fastly/[ID]`

The default value is `hls`. The valid values are **hls** and **dash**. Enable **dash** to send an MPEG-DASH stream to the target for digital rights management (DRM) use on Google and Microsoft devices.

February 22, 2021

All versions (v1.7 through v1.3)

- Fixed an issue where player logo images were limited to 1 MB in size instead of the documented 2.5 MB.

February 9, 2021

All versions (v1.7 through v1.3)

- Fixed an issue where calling `DELETE /recordings/[ID]` twice in quick succession might result in a race condition that left the recording stuck in the removing state.

February 1, 2021

- The REST API server response caching time changed from 30 seconds to 1 second for the following endpoints:
 - `/live_streams`
 - `/recordings`
 - `/schedules`
 - `/stream_sources`
 - `/stream_targets`
 - `/transcoders`
- See [Wowza Video REST API server response caching](#) for a list of all endpoints and cache times.
- Fixed an issue where `hosted_page_logo_image_url` was returned as a boolean (`false`) instead of a string when `hosted_page` was set to `false`. `hosted_page_logo_image_url` is no longer returned in the responses for `POST /live_streams` and `PATCH /live_streams/[ID]` when there is no hosted page.

January 11, 2021

- Added functionality that allows you to use webhooks to integrate Wowza Video with other applications to streamline or automate your workflows. See [Receive event notifications with webhooks](#) to learn how to enable webhooks for your account and receive event notifications.

January 5, 2021

Deprecated (1.3, 1.4)

- Deprecated versions 1.3 and 1.4.

December 10, 2020

Beta (v1.7)

- Added `playback_urls` to the following endpoints:
 - `POST /stream_targets/akamai`
 - `PATCH /stream_targets/akamai/[ID]`

- GET /stream_targets/akamai/[ID]
- POST /stream_targets/custom
- PATCH /stream_targets/custom/[ID]
- GET /stream_targets/custom/[ID]

With this change, the following parameters are sunset in the previous endpoints:

- hds_playback_url
- hls_playback_url
- rtmp_playback_url
- Added delivery_protocols to the response for the following endpoints:
 - POST /stream_targets/akamai
 - PATCH /stream_targets/akamai/[ID]
 - GET /stream_targets/akamai/[ID]
 - POST /stream_targets/custom
 - PATCH /stream_targets/custom/[ID]
 - GET /stream_targets/custom/[ID]
 - POST /stream_targets/fastly
 - PATCH /stream_targets/fastly/[ID]
 - GET /stream_targets/fastly/[ID]
 - POST /stream_targets/ull
 - PATCH /stream_targets/ull/[ID]
 - GET /stream_targets/ull/[ID]

December 7, 2020

Beta (v1.7), Current (v1.6), Supported (v1.5)

- Added a new transcoder property, live2vod, that replaces the live playlist with the VOD playlist after a stream ends. Use this property if you want your viewers to access the VOD stream using the same playback URL they used to view the original stream. See [Create a VOD stream with the Wowza Video REST API](#) for more information about working with VOD streams.

All versions (v1.7 through v1.3)

- Fixed an issue where playback URLs didn't change between secure or unsecure HTTP given selected stream target properties and security settings.

December 1, 2020

Beta (v1.7), Current (v1.6), Supported (v1.5)

- **Beta** - Established v1.7 as the base path for the beta version of the API.
- **Current** - Established v1.6 as the current version of the API. For a list of new features, improvements, and breaking changes, see the [Wowza Video REST API migration guide](#).
- **Supported** - Established v1.5 as a supported version of the API.

November 18, 2020

Beta (v1.6)

- Renamed the ingest_ip_whitelist parameter to ingest_ip_allowlist for the following endpoints:
 - POST /stream_targets/ull
 - GET /stream_targets/ull/[ID]
 - PATCH /stream_targets/ull/[ID]

Beta (v1.6)

- Fixed an issue where default values for conditionally required parameters weren't being used for the POST `/transcoders/[ID]/outputs` endpoint. For example, in v1.6, `audio_codec` is a required parameter and when specified as `aac`, `bitrate_audio` is also required. Because `bitrate_audio` has a default value for `aac`, 128, that default value is now used when the parameter is not explicitly specified.

November 12, 2020

All versions (v1.6 through v1.3)

- Added the `acoa` property for Wowza CDN on Fastly stream targets. While most streams won't need the origin explicitly set, you might use this property for older players that don't send the Referer header in their stream request. See [Set advanced properties with the Wowza Video REST API](#) for more information.
- Adjusted transcoder functionality to allow only 10 output renditions per transcoder. Existing transcoders with more than 10 outputs retain those outputs, but you can't add more. For the best playback performance, we recommend limiting the outputs on a single transcoder to 8 or fewer.
- Fixed an issue where the idle timeout counter on a transcoder didn't reset between multiple starts in a short period of time, resulting in the idle transcoder timing out prematurely on a subsequent start.

October 29, 2020

All versions (v1.6 through v1.3)

- As of August 31, 2021, the ultra low latency (ULL) feature will be sunset, and existing ULL targets will no longer function. See the [Wowza Player](#), [Wowza GoCoder SDK](#), and [ULL notification](#) to learn more about Wowza's future plans for low latency streaming.

October 16, 2020

All versions (v1.6 through v1.3)

- Fixed an issue where GET `/transcoders/[ID]/uptimes/[ID]/metrics/historic` returns an error if the transcoder you're requesting metrics for is active.
- Fixed an issue where a stream target can't be deleted if it has a VOD stream that contains no video.
- Fixed an issue where deleting a transcoder that shares a stream target with other transcoders removes the shared target from all transcoders, silently.

October 12, 2020

All versions (v1.6 through v1.3)

- Added functionality to the Wowza Video REST API that allows you to deliver WebRTC streams for playback directly from the transcoder. You can do direct playback using the `/live_streams` or `/transcoders` endpoints. See [WebRTC workflows in Wowza Video](#) to learn more about WebRTC and the workflows you can choose from.
- Changed the maximum value for the `play_maximum_connections` parameter from 100 to the maximum number of connections the specific transcoder can support. Generally, adaptive bitrate transcoders are larger and support more viewer connections (up to 300) while passthrough transcoders are smaller and support fewer viewer connections.
- Changed the format of the name object in `direct_playback_urls` to display the video and audio bitrate values for the output rendition.

Beta (v1.6)

- Replaced the `stream_format`, `passthrough_audio`, and `passthrough_video` parameters with `audio_codec` and `video_codec` for the following endpoints:
 - POST `/transcoders/[transcoder_id]/outputs`
 - GET `/transcoders/[transcoder_id]/outputs`

- GET /transcoders/[*transcoder_id*]/outputs/[*id*]
- PATCH /transcoders/[*transcoder_id*]/outputs/[*id*]

October 5, 2020

All versions (v1.6 through v1.3)

- Fixed an issue where the cross-origin resource sharing (CORS) Access-Control-Allow-Headers header didn't allow wsc-timestamp and wsc-signature for HMAC authentication.

October 1, 2020

Beta (v1.6), Current (v1.5), Supported (v1.4)

- Added usage data for Wowza CDN on Akamai stream targets to the following endpoints:
 - /usage/stream_targets
 - /usage/stream_targets/summary
 - /usage/stream_targets/[*id*]
 - /usage/stream_targets/[*id*]/countries
 - /usage/stream_targets/[*id*]/renditions

September 25, 2020

All versions (v1.6 through v1.3)

- Fixed an issue where changing a live stream's encoder didn't update source information as expected, resulting in an error message about an invalid source delivery method.

September 8, 2020

Beta (v1.6)

- Added the *playback_stream_name* parameter to the responses for the following endpoints:
 - POST /live_streams
 - GET /live_streams/[*id*]
 - PATCH /live_streams/[*id*]
 - POST /transcoders
 - GET /transcoders/[*id*]
 - PATCH /transcoders/[*id*]
- Added the *file_name* parameter to the response for the GET /recordings endpoint.

August 31, 2020

All versions (v1.6 through v1.3)

- Removed **passthrough** as a *transcoder_type* for WebRTC streams created through the /live_streams endpoint.

Beta (v1.6)

- Replaced the *playback_url* parameter with *playback_urls* in the response for the following endpoints:
 - POST /stream_targets/fastly
 - GET /stream_targets/fastly/[*target_id*]
 - PATCH /stream_targets/fastly/[*target_id*]

August 4, 2020

Beta (v1.6), Current (v1.5)

- Added hostname validation when specifying a *primary_url* for custom RTMP stream targets.
- Updated the transcoder *application_name* property to default to names that are eight characters in length instead of four.
- Improved audio and video synchronization by changing default behavior for WebRTC to automatically transcode audio and video from the source stream.
- Added IPv6 support for Fastly stream targets.

July 29, 2020

Beta (v1.6), Current (v1.5)

- Removed the **file** value from the *delivery_method* parameter for the following endpoints:
 - POST /live_streams
 - GET /live_streams/[*live_stream_id*]
 - PATCH /live_streams/[*live_stream_id*]
 - POST /transcoders
 - GET /transcoders/[*transcoder_id*]
 - PATCH /transcoders/[*transcoder_id*]

To create a live stream that uses a file as the source, you will need to set the following values:

- **Live stream workflow:** Set *encoder* to **file**, *delivery_method* to **pull**, and *source_url* to the URL where the file is located.
- **Transcoder workflow:** Set *protocol* to **file**, *delivery_method* to **pull**, and *source_url* to the URL where the file is located.

For more information, see [Stream a file with the Wowza Video REST API](#).

Note: Existing live streams that use a file as the source will automatically be updated to the new configuration. No further action is required.

June 25, 2020

Beta (v1.6), Current (v1.5), Supported (v1.4)

- *unique_views* would occasionally show the number of current viewers as 0. To address this issue, we've changed how often current viewer data is updated from 90 seconds to 5 minutes.

June 10, 2020

All versions (v1.6 through v1.3)

- Added functionality that allows you to record a live stream while simultaneously allowing viewers to play or pause the live stream, rewind it to a previously recorded point, or resume viewing at the current live point. For more information, see [Configure nDVR using the Wowza Video REST API](#).

June 4, 2020

All versions (v1.6 through v1.3)

- Passthrough audio isn't supported in recordings for WebRTC. To improve user experience, audio from the incoming source stream is now automatically transcoded by default when recording a WebRTC stream created through the live stream workflow. For more information, see [Connect a WebRTC stream to Wowza Video using the Wowza Video REST API](#).

Beta (v1.6), Current (v1.5), Supported (v1.4)

- Fastly playlists now return a 404 when no new media segments are received.

May 27, 2020

Beta (v1.6)

- Established v1.6 as the base path for the beta version of the API.

Current (v1.5)

- Established v1.5 as the current version of the API. For a list of new features, improvements, and breaking changes, see the [Wowza Video REST API migration guide](#).

Supported (v1.4)

- Established v1.4 as a supported version of the API.

Beta (v1.6), Current (v1.5)

- Added functionality that allows you to create and distribute video on demand (VOD) streams. For more information, see [Create a VOD stream with the Wowza Video REST API](#).
- When creating a live stream, you now have the option to set the *encoder* parameter to **file**. Previously, file streaming was only available in the transcoder workflow. For more information, see [Stream a file with the Wowza Video REST API](#).

All versions (v1.6 through v1.3)

- Reference documentation for all versions of the REST API is now available at api.docs.cloud.wowza.com.
- Fixed an issue with the *stream_extension* parameter that allowed you to enter values other than a period (.) and alphanumeric characters.

May 14, 2020

Beta (v1.5)

- Removed *peak* and *recordings* from the response received when querying GET /usage/storage/peak. The response now only returns *bytes*.

May 6, 2020

Beta (v1.5)

- Added new transcoder properties that allow you to play a file on a continuous loop and schedule a file to begin streaming at a specific date and time. For more information, see [Stream a file with the Wowza Video REST API](#).

April 20, 2020

Beta (v1.5)

- Removed the *page* query parameter from the usage/stream_targets endpoint. To specify which page of results should be displayed, use the new *next_page_key* parameter instead.

April 14, 2020

Beta (v1.5)

- Added the ability to set the *play_maximum_connections* parameter for all transcoder types. Previously, this

parameter was only available for 24x7 and passthrough transcoders. For more information, see [Play streams directly from a transcoder with the Wowza Video REST API](#).

April 8, 2020

Beta (v1.5)

- Added functionality that allows you to stream a file (.mp4, .mp3, .flv) in a live stream broadcast. This feature is only available in the transcoder workflow. When creating a transcoder, set the *delivery_method* to **file** and *source_url* to the location where the file is stored. For more information, see [Stream a file with the Wowza Video REST API](#).

March 26, 2020

Beta (v1.5)

- The Wowza Video REST API documentation for v1.5 has moved. Click [here](#) to go to the new site.

All versions (v1.5 through v1.3)

- When creating a live stream, you now have the option to set the *encoder* parameter to **other_udp** or **other_srt**. For more information, see [Connect a UDP encoder to Wowza Video using the Wowza Video REST API](#) and [Connect an SRT encoder to Wowza Video using the Wowza Video REST API](#).
- Added new transcoder properties for enabling AES-128 encryption for playback of HLS streams. The new properties are *aes128Host* and *aes128Secret*. For more information, see [Secure HLS streams with AES-128 external encryption using the Wowza Video REST API](#).

March 4, 2020

Beta (v1.5)

- Removed the *recording* parameter from the */transcoders* endpoint. To enable stream recording when creating a transcoder, use the new **mp4** transcoder property instead.

February 28, 2020

Beta (v1.5), Current (v1.4)

- Fixed an issue where live streams with *low_latency* set to **true** and *target_delivery_protocol* set to **hls-https** were assigned Wowza CDN on Akamai - HLS stream targets instead of Wowza CDN on Fastly targets.

February 25, 2020

Beta (v1.5)

- Added the ability to configure transcoder properties when creating a transcoder using the POST */transcoders* endpoint.

February 19, 2020

All versions (v1.5 through v1.3)

- When creating a live stream, you can now set the *encoder* parameter to **other_webrtc**. WebRTC is available for push connections only. For more information, see [Connect a WebRTC stream to Wowza Video using the Wowza Video REST API](#).

February 10, 2020

All versions (v1.5 through v1.3)

- Removed the following *region_override* parameter values for ultra low latency stream targets. Ultra low latency streams that use these regions will automatically be routed to the next closest region.
 - **origin-au-east1.cdn.wowza.com**
 - **origin-br-south1.cdn.wowza.com**
 - **origin-in-west1.cdn.wowza.com**

February 3, 2020

Beta (v1.5), Current (v1.4)

- Moved Wowza CDN on Fastly functionality from private preview to public release. For more information, see [Get started streaming to Wowza CDN on Fastly with the Wowza Video REST API](#).

January 31, 2020

Beta (v1.5)

- Established v1.5 as the base path for the beta version of the API.

Current (v1.4)

- Established v1.4 as the current version of the API. For assistance migrating from v1.3 to v1.4, see [Wowza Video REST API migration guide](#).

Supported (v1.3)

- Established v1.3 as a supported version of the API.

January 28, 2020

Beta (v1.4)

- Added new functionality and enhancements for Wowza CDN on Fastly stream targets, a beta feature that's available in private preview.
 - Updated the *hls_playback_url* parameter in the response received when creating, updating, and fetching a specific Fastly stream target.
 - Enabled the *relativePlaylists* property for Fastly stream targets. For more information, see [Manage HLS playback over SSL for Wowza CDN on Fastly with the Wowza Video REST API](#).
 - Added geo-block functionality. For more information, see [Geo-block Wowza CDN on Fastly stream targets with the Wowza Video REST API](#).

January 16, 2020

Beta (v1.4)

- Removed the Akamai stream sources feature.
 - The */stream_sources/akamai* endpoint and its associated operations is sunset and no longer available.
- Added support for token authentication and secure HLS playback on Wowza CDN on Fastly stream targets, a beta feature that's available in private preview. For more information, see [Protect a Wowza CDN on Fastly stream target with token authentication using the Wowza Video REST API](#) and [Manage HLS playback over SSL for Wowza CDN on Fastly with the Wowza Video REST API](#).

Current (v1.3)

- Removed the following Akamai stream sources operations:
 - POST */stream_sources/akamai*
 - PATCH */stream_sources/akamai/[stream_source_id]*

January 15, 2020

Beta (v1.4)

- Updated the peak recording storage endpoint:
 - To retrieve peak recording storage for your account, use `/usage/storage/peak` instead of `/usage/storage/peak_recording`.
 - Added `from` and `to` parameters to the response to indicate the start and end range of time specified in the request.
- Removed and reorganized Wowza CDN on Akamai stream targets endpoints:
 - To create a Wowza CDN on Akamai stream target, use POST `/stream_targets/akamai`. The POST `/stream_targets/wowza` operation is sunset and no longer available.
 - To retrieve a single Wowza CDN on Akamai stream target, use GET `/stream_targets/akamai/[stream_target_id]`. The GET `/stream_targets/wowza/[stream_target_id]` is sunset and no longer available.
 - To retrieve Wowza CDN on Akamai stream targets, use GET `/stream_targets/akamai`. The GET `/stream_targets/wowza` operation is sunset and no longer available.
 - To update a Wowza CDN on Akamai stream target, use PATCH `/stream_targets/akamai/[stream_target_id]`. The PATCH `/stream_targets/wowza/[stream_target_id]` operation is sunset and no longer available.
 - To delete a Wowza CDN on Akamai stream target, use DELETE `/stream_targets/akamai/[stream_target_id]`. The DELETE `/stream_targets/wowza/[stream_target_id]` operation is sunset and no longer available.

January 8, 2020

Beta (v1.4)

- Added the `archived` and `name` parameters to the response for the following stream targets usage endpoints:
 - GET `/usage/stream_targets`
 - GET `/usage/stream_targets/[stream_target_id]`
- Added new transcoder usage endpoints:
 - To retrieve usage data for a specific transcoder, use GET `/usage/transcoders/[transcoder_id]`.
 - To retrieve a summary of usage for all transcoders on your account, use GET `/usage/transcoders/summary`.
- Removed and reorganized existing transcoder usage endpoints:
 - To retrieve stream processing times for transcoders, use GET `/usage/transcoders`. The GET `usage/time/transcoders` endpoint is sunset and no longer available.
- Removed the stream sources usage feature.
 - The `/usage/stream_sources` endpoint and its associated operations are sunset and no longer available.
- We've replaced Akamai stream sources with Wowza stream sources in the live stream workflow. To create an Akamai stream source, use POST `stream_sources/akamai`.

December 19, 2019

Beta (v1.4)

- Updated the following stream targets and transcoders usage endpoints to limit the maximum number of days that can be queried in a single request to 90 days.
 - GET `/usage/stream_targets`
 - GET `/usage/stream_targets/summary`
 - GET `/usage/stream_targets/[stream_target_id]`
 - GET `/usage/stream_targets/[stream_target_id]/countries`
 - GET `/usage/stream_targets/[stream_target_id]/renditions`

- GET /usage/transcoders
- Added the *from* and *to* parameters to the response for the following stream targets usage endpoints to indicate the start and end range of time for which usage data was returned.
 - GET /usage/stream_targets
 - GET /usage/stream_targets/summary
 - GET /usage/stream_targets/[*stream_target_id*]
 - GET /usage/stream_targets/[*stream_target_id*]/countries
 - GET /usage/stream_targets/[*stream_target_id*]/renditions

December 12, 2019

Sunset (v1.0, 1.1, 1.2)

- API versions 1.0, 1.1, and 1.2 are sunset and no longer available. For assistance migrating to the current version of the API (v1.3), see [Wowza Video REST API migration guide](#).

December 5, 2019

All versions (v1.4 through v1.0)

- Added support for WebRTC ingest. When creating a transcoder, you can now set the *protocol* to **webrtc**. WebRTC is available for push connections only and does not support source authentication. See [Connect a WebRTC stream to Wowza Video using the Wowza Video REST API](#) for more information.
- Added support for UDP ingest. When creating a transcoder, you can now set the *protocol* to **udp**. UDP is available for push connections only and does not support source authentication. See [Connect a UDP encoder to Wowza Video using the Wowza Video REST API](#) for more information.

Beta (v1.4)

- Updated the response to the following player queries to remove *hds_playback_url*.
 - GET players/[*player_id*]
 - PATCH players/[*player_id*]
- Removed the geo-blocking and token authorization features for Wowza stream targets.
 - The */stream_targets/[stream_target_id]/geoblock* endpoint and its associated operations is sunset and no longer available.
 - The */stream_targets/[stream_target_id]/token_auth* endpoint and its associated operations is sunset and no longer available.

December 3, 2019

Beta (v1.4)

- Wowza CDN on Fastly stream targets, an alpha feature in private preview, is now available as a beta feature. See [Stream to Wowza CDN on Fastly with the Wowza Video REST API](#) for more information.

November 25, 2019

All versions (v1.4 through v1.0)

- Added *request_id* and *request_timestamp* fields to error responses. If you need some help from our Support team, reference those values and the failed API request in your ticket. See [Accessing Support](#) for more information.

October 31, 2019

Beta (v1.4)

- Removed and reorganized network usage endpoints.

- To retrieve viewer data for a stream target, use GET /usage/stream_targets/[*stream_target_id*]. The GET /usage/viewer_data/stream_targets/[*stream_target_id*] operation is sunset and no longer available.
- To retrieve usage data for stream targets, use GET /usage/stream_targets. The GET /usage/network/stream_targets operation is sunset and no longer available.
- To retrieve usage data for a transcoder, use GET /usage/transcoders. The GET /usage/network/transcoders operation is sunset and no longer available.
- To retrieve usage data for a stream source, use GET /usage/stream_sources. The GET /usage/network/stream_sources operation is sunset and no longer available.
- Removed these stream health metrics related to CPU and GPU:
 - **cpu**
 - **cpu_idle** (historic metrics only)
 - **gpu_decoder_usage**
 - **gpu_driver_version**
 - **gpu_encoder_usage**
 - **gpu_memory_usage**
 - **gpu_usage**
- From these requests:
 - GET /transcoders/[*transcoder_id*]/stats
 - GET /live_streams/[*live_stream_id*]/stats
 - GET /transcoders/[*transcoder_id*]/uptimes/[*uptime_id*]/metrics/current
 - GET /transcoders/[*transcoder_id*]/uptimes/[*uptime_id*]/metrics/historic
- For efficiency, multiple customers may share a single CPU and GPU instance. When this occurs, the statistics returned for CPU and GPU utilization don't provide accurate data.
- Removed the **hls-hds target_delivery_protocol** option to create an HDS live stream. When creating or updating a live stream, the only valid *target_delivery_protocol* value is now **hls-https**. To send a stream using HDS, create a transcoder with a Wowza stream target that has a *provider* of **akamai**.
- Updated the response to the following usage queries to remove *bytes_billed*. Due to timing differences between usage and billing systems, (when this information was available at GET /usage/network/stream_sources and GET /usage/network/transcoders) the *bytes_billed* value didn't consistently reflect the number of bytes used in a billing cycle.
 - GET /usage/stream_sources
 - GET /usage/transcoders
- Updated the response to the following usage query to remove *seconds_billed*. Due to timing differences between usage and billing systems, the *seconds_billed* value didn't consistently reflect the seconds billed for in a billing cycle.
 - GET /usage/time/transcoders

Current (v1.3) and Deprecated (v1.0, v1.1, v1.2)

- Updated these stream health metrics related to CPU and GPU so that their *value* returns **null** and their *status* returns **no_data**:
 - **cpu**
 - **cpu_idle** (historic metrics only, returns a *status* of **ok**)
 - **gpu_decoder_usage**
 - **gpu_driver_version**
 - **gpu_encoder_usage**
 - **gpu_memory_usage**
 - **gpu_usage**
- In these requests:
 - GET /transcoders/[*transcoder_id*]/stats
 - GET /live_streams/[*live_stream_id*]/stats
 - GET /transcoders/[*transcoder_id*]/uptimes/[*uptime_id*]/metrics/current
 - GET /transcoders/[*transcoder_id*]/uptimes/[*uptime_id*]/metrics/historic

- For efficiency, multiple customers may share a single CPU and GPU instance. When this occurs, the statistics returned for CPU and GPU utilization don't provide accurate data.
- Updated the response to the following usage queries so that *bytes_billed* always returns a value of **0**. Due to timing differences between usage and billing systems, the *bytes_billed* value didn't consistently reflect the number of bytes billed for in a billing cycle.
 - GET /usage/network/stream_sources
 - GET /usage/network/stream_targets
 - GET /usage/network/transcoders
- Updated the response to the following usage query so that *seconds_billed* always returns a value of **0**. Due to timing differences between usage and billing systems, the *seconds_billed* value didn't consistently reflect the seconds billed for in a billing cycle.
 - GET /usage/time/transcoders

October 15, 2019

Beta (v1.4)

- Wowza CDN on Fastly stream targets, which are available in preview release, now provide analytics for near real-time usage data. You can use these new requests in v1.4 to query for a range of Wowza CDN on Fastly stream target usage:
 - GET /usage/stream_targets
 - GET /usage/stream_targets/[*stream_target_id*]
 - GET /usage/stream_targets/[*stream_target_id*]/summary
 - GET /usage/stream_targets/[*stream_target_id*]/countries
 - GET /usage/stream_targets/[*stream_target_id*]/live
 - GET /usage/stream_targets/[*stream_target_id*]/renditions
- For more information, see [Stream to Wowza CDN on Fastly with the Wowza Video REST API](#).

October 10, 2019

All versions (v1.4 through v1.0)

- The production host URL for the Wowza Video REST API is <https://api.cloud.wowza.com>. Although <https://cloud.wowza.com> also worked previously, it no longer functions. All requests in production code should point to [https://api.cloud.wowza.com/api/\[version\]/\[resource\]](https://api.cloud.wowza.com/api/[version]/[resource]).

September 23, 2019

Beta (v1.4) and Current (v1.3)

- The live stream and transcoder broadcast location **asia_pacific_s_korea** now supports a *billing_mode* of **pay_as_you_go** or **twentyfour_seven** and a *transcoder_type* of **transcoded** or **passthrough**.

August 20, 2019

Beta (v1.4) and Current (v1.3)

- Standardized transcoding and billing features for broadcast locations. Most broadcast locations for transcoders and live streams now support a *billing_mode* of **pay_as_you_go** or **twentyfour_seven** and a *transcoder_type* of **transcoded** or **passthrough**.
The only exceptions are **asia_pacific_s_korea** and **eu_ireland**, which support **pay_as_you_go** and **transcoded** only.

August 1, 2019

Beta (v1.4)

- Wowza CDN on Fastly stream targets, which are available in preview release, are now incorporated into live streams. If you're participating in the preview program, when you create a live stream (a POST operation on the `live_stream` endpoint), the only valid `target_delivery_protocol` is now **hls-https**. Wowza Video generates a Wowza CDN on Fastly stream target for the stream.

July 30, 2019

Deprecated (v1.0, 1.1, 1.2)

- Deprecated versions 1.0, 1.1, and 1.2.

July 17, 2019

Beta (v1.4)

- Made Wowza stream sources generally available to all Wowza Video customers. Using a live stream or transcoder with a Wowza stream source allows Wowza Video to detect the broadcast location automatically, start and stop a transcoder automatically, and instantly connect to a source. See [Control a transcoder with a Wowza stream source using the Wowza Video REST API](#) for more information.
- Added support for select advanced properties on Wowza CDN on Fastly stream targets, an alpha feature in private preview. You can now specify the **chunkSize** of HLS media segments, enable **convertAMFdata** to convert AMF metadata to ID3, and define the maximum allowable duration of a playlist by specifying the **playlistSeconds** property. For more information, see [Stream to Wowza CDN on Fastly with the Wowza Video REST API](#).

June 25, 2019

Beta (v1.4)

- Introduced Wowza CDN on Fastly stream targets, an alpha feature in private preview. Use this stream target type to stream from Wowza Video over the Wowza CDN via Fastly. To be considered for preview participation, contact cloud-preview@wowza.com. For more information, including feature limitations in the alpha phase, see [Stream to Wowza CDN on Fastly with the Wowza Video REST API](#).

June 11, 2019

Beta (v1.4), Current (v1.3), and Supported (v1.2 only)

- Added a new region for ultra low latency streams. You can now use servers in the East Australia region for ingest to an origin or delivery from an edge. The network automatically selects the nearest region, but you can manually override this automatic selection to use East Australia.

May 28, 2019

Beta (v1.4), Current (v1.3), and Supported (v1.2, 1.1, 1.0)

- Added a new transcoder output property, **record**, that allows you to record a specific output rendition instead of recording the stream sent from your encoder or camera to Wowza Video. A valid value for the property is an output ID associated with the transcoder you want to record. For details see [Transcoder properties](#) and [Record a specific output rendition](#).

April 18, 2019

Beta (v1.4), Current (v1.3), and Supported (v1.2, 1.1, 1.0)

- Added support for custom stream targets to send a stream using RTMPS. When configuring a custom stream target, you can now set the `provider` to **rtmps** to send a stream over RTMP with a secure TLS/SSL connection.
- Fixed an issue where live streams with `low_latency` set to **true** were unable to regenerate connection codes.

as expected.

April 11, 2019

Beta (v1.4)

- Added operations to start and stop the recording of stream passing through a transcoder. You can now start recording after a transcoder has started or stop recording before a transcoder stops. Existing functionality lets you start and stop recording when the transcoder starts and stops.

New operations:

- PUT /transcoders/[*transcoder_id*]/start_recording
- PUT /transcoders/[*transcoder_id*]/stop_recording
- Introduced transcoder properties to enable the use of a jitter buffer with an RTSP connection. New transcoder properties are **rtpDePacketizerWrapper**, **rtpDePacketizerPacketSorterFlushTime**, and **rtpDePacketizerPacketSorterBufferTime**. To learn more, see [Set advanced properties with the Wowza Video REST API](#).

Current (v1.3) and Supported (v1.2, 1.1, 1.0)

- Introduced transcoder properties to enable the use of a jitter buffer with an RTSP connection. New transcoder properties are **rtpDePacketizerWrapper**, **rtpDePacketizerPacketSorterFlushTime**, and **rtpDePacketizerPacketSorterBufferTime**. To learn more, see [Set advanced properties with the Wowza Video REST API](#).

April 4, 2019

Current (v1.3)

- Publicly released SDKs for the Wowza Video REST API in Java and Ruby. Accelerate your development process with the [Wowza Video Java SDK](#) and the [Wowza Video Ruby SDK](#).

April 3, 2019

Beta (v1.4), Current (v1.3), and Supported (v1.2 only)

- Added a new region for ultra low latency streams. You can now use servers in the South Brazil region for ingest to an origin or delivery from an edge. The network automatically selects the nearest region, but you can manually override this automatic selection to use South Brazil.

March 20, 2019

Beta (v1.4), Current (v1.3), and Supported (v1.2, 1.1, 1.0)

- Added support for SRT ingest. When creating a transcoder, you can now set the *protocol* to **srt**. SRT is available for push connections only and does not support closed captions or source authentication. See [Connect an SRT encoder to Wowza Video using the Wowza Video REST API](#) for more information.

February 28, 2019

Beta (v1.4), Current (v1.3), and Supported (v1.2, 1.1, 1.0)

- Added a new transcoder property, **fitMode**, that adjusts the aspect ratio of all of a transcoder's output renditions according to the specified setting. Valid values include **fit-height** (the default), **fit-width**, **letterbox**, **crop**, **stretch**, and **match-source**.

February 13, 2019

Beta (v1.4)

- Introduced Wowza stream sources in a private preview. Using a live stream or transcoder with a Wowza stream source allows Wowza Video to detect the broadcast location automatically, start and stop a transcoder automatically, and instantly connect to a source. See [Control a transcoder with a Wowza stream source using the Wowza Video REST API](#) for more information. To be considered for preview participation, contact cloud-preview@wowza.com.
- Updated Node.js and curl code examples in Wowza Video REST API documentation. Node.js examples show how to programmatically implement the HMAC method of authentication, while examples in curl now use environment variables so you can easily copy and paste them. To learn how to set up environment variables, see [Using cURL](#).

Current (v1.3)

- Updated Node.js and curl code examples in Wowza Video REST API documentation. Node.js examples show how to programmatically implement the HMAC method of authentication, while examples in curl now use environment variables so you can easily copy and paste them. To learn how to set up environment variables, see [Using cURL](#).

January 31, 2019

Beta (v1.4)

- Established v1.4 as the base path for the beta version of the API.
- Established a new authentication method for the Wowza Video REST API. To learn more about how to format your requests with HMAC, see [Authentication](#).

Current (v1.3)

- Established v1.3 as the current version of the API.
- Established a new authentication method for the Wowza Video REST API. To learn more about how to format your requests with HMAC, see [Authentication](#).
- Stream targets can no longer be created, updated, deleted, and individually retrieved through the `/stream_targets` endpoint. To perform these operations, use the specific endpoint for a Wowza, custom, or ultra low latency stream target. You can, however, still retrieve limited details for all types of stream targets through a GET request to `/stream_targets`.

Supported (v1.2, 1.1, 1.0)

- Established v1.2 as a supported version of the API. Versions 1.2, 1.1, and 1.0 will continue to be supported until their deprecation date, which is targeted for July 30, 2019. To learn more about deprecations, the REST API versioning schedule, and migration support, see the [Wowza Video REST API deprecation policy](#).

January 17, 2019

Beta (v1.3)

- Updated object references for custom stream targets, Wowza stream targets, and ultra low latency stream targets. Updates are for an array and a single reference. They now look like this:
 - `stream_targets_custom` and `stream_target_custom`
 - `stream_targets_wowza` and `stream_target_wowza`
 - `stream_targets_ull` and `stream_target_ull`
- Updated the type parameter values returned when you fetch all stream targets. Types are now `custom`, `wowza`, and `ull`.
- When starting, stopping, and fetching the state of a transcoder, removed the previously deprecated `transcoding_uptime_id` parameter and replaced it with `uptime_id`.
- Removed the previously deprecated `delivery_protocol` parameter from the live streams object.

December 19, 2018

Beta (v1.3)

- Limited the details that are returned when you fetch all ultra low latency stream targets, Wowza stream targets, custom stream targets, stream sources, or Akamai stream sources objects.
 - Fetching all ultra low latency stream targets now returns *id* and *name*.
 - Fetching all Wowza stream targets now returns *id* and *name*.
 - Fetching all custom stream targets now returns *id* and *name*.
 - Fetching all stream sources now returns *id*, *name*, and *type*.
 - Fetching all Akamai stream sources now returns *id* and *name*.
- To get detailed information about an ultra low latency stream target, Wowza stream target, custom stream target, stream source, or Akamai stream source, request a single object.
- Updated object references for an array of Akamai stream sources and a single Akamai stream source. They now look like this:
 - `stream_sources_akamai`
 - `stream_source_akamai`
- Pagination is now enabled by default for requests to fetch all stream targets. A maximum of 1000 records display at one time. To learn more about using pagination parameters, see [Get paginated query results with the Wowza Video REST API](#).

December 5, 2018

Beta (v1.3)

- Limited the details that are returned when you fetch all transcoders, live streams, players, or recordings objects.
 - Fetching all transcoders now returns *id*, *name*, and *workflow*.
 - Fetching all live streams now returns *id* and *name*.
 - Fetching all players now returns *id*, *type*, and *transcoder_id*.
 - Fetching all recordings now returns *id*, *transcoder_id*, *state*, and *reason*.
 - Fetching all recordings for a transcoder returns *id*, *transcoder_id*, *state*, and *reason*.
- To get detailed information about a transcoder, live stream, player, or recording, request a single object.
- Updated default values for pagination query parameters; *page* defaults to **1**, and *per_page* defaults to **1000**. Pagination is now enabled by default for requests to fetch all live streams, players, recordings, schedules, stream sources, or transcoders. A maximum of 1000 records display at one time. To learn more about using pagination parameters, see [Get paginated query results with the Wowza Video REST API](#).

November 7, 2018

Beta (v1.3), Current (v1.2), and Supported

- API access in the production environment is now available in the Wowza Video 30 day free trial. See [Wowza Video free trial](#) for information on trial limitations.

October 29, 2018

Beta (v1.3), Current (v1.2), and Supported

- Added general performance enhancements to API processes, including faster stream target creation time.

September 5, 2018

Beta (v1.3)

- Ultra low latency stream targets can no longer be created, updated, deleted, and individually retrieved through the `/stream_targets` endpoints. However, you can still retrieve limited details for ultra low latency stream targets through a GET request to `/stream_targets`.
The following endpoints no longer support the type **UltraLowLatencyStreamTarget**. You must create and

manage ultra low latency stream targets through `/stream_targets/ull` endpoints as of v1.3.

- POST `/stream_targets`
- GET `/stream_targets/[stream_target_id]`
- PATCH `/stream_targets/[stream_target_id]`
- DELETE `/stream_targets/[stream_target_id]`

Supported (v1.1, v1.0)

- Ultra low latency stream targets can no longer be created and managed through v1.0 and v1.1 of the REST API. You must use v1.2 or later for ultra low latency streaming.

August 23, 2018

Beta (v1.3)

- Stream sources can no longer be created, updated, and deleted through the `/stream_sources` endpoints. However, you can still retrieve limited details for stream sources with a GET request to `/stream_sources`. Stream source functionality remains the same, but you configure it through the new Akamai stream source endpoints:
 - POST `/stream_sources/akamai`
 - GET `/stream_sources/akamai`
 - GET `/stream_sources/akamai/[akamai_stream_source_id]`
 - PATCH `/stream_sources/akamai/[akamai_stream_source_id]`
 - DELETE `/stream_sources/akamai/[akamai_stream_source_id]`
- To learn how to use Akamai stream sources, see [Address network congestion with stream sources using the Wowza Video REST API](#) and [Build a redundant workflow with stream sources using the Wowza Video REST API](#).
- Default simultaneous viewer limits per ultra low latency stream target are now set according to the subscription plan you select.

Current (v1.2)

- Default simultaneous viewer limits per ultra low latency stream target are now set according to the Wowza Video with Ultra Low Latency subscription plan you select.

August 7, 2018

Beta (v1.3)

- Added the **convertAMFData** stream target property, which determines whether Wowza Video converts AMF data into ID3 tags. ID3 tags allow you to include metadata in your HLS stream. See [Set advanced properties with the Wowza Video REST API](#) for information on how to use this property.
- Updated `region_override` values and functionality for ultra low latency stream targets:
 - When creating an ultra low latency stream target, you can now use updated values to override regions that Wowza Video assigns to your origin and edge servers.
 - When updating an ultra low latency stream target with a *delivery method* of **pull**, you can now update the `region_override` value through a PATCH request to `/stream_targets/ull/[ull_stream_target_id]`.

Current (v1.2)

- Added the **convertAMFData** stream target property, which determines whether Wowza Video converts AMF data into ID3 tags. ID3 tags allow you to include metadata in your HLS stream. See [Set advanced properties with the Wowza Video REST API](#) for information on how to use this property.
- Updated `region_override` values and functionality for ultra low latency stream targets:
 - When creating an ultra low latency stream target, you can now use updated values to override regions that Wowza Video assigns to your origin and edge servers.
 - When updating an ultra low latency stream target with a *delivery method* of **pull**, you can now update

the `region_override` value through a PATCH request to `/stream_targets/ull/[ull_stream_target_id]`.

Supported (v1.1, v1.0)

- Added the **convertAMFData** stream target property, which determines whether Wowza Video converts AMF data into ID3 tags. ID3 tags allow you to include metadata in your HLS stream. See [Set advanced properties with the Wowza Video REST API](#) for information on how to use this property.

June 26, 2018

Beta (v1.3), Current (v1.2), and Supported

- Moved ultra low latency streaming functionality from private preview to public release. Ultra low latency stream targets and stream targets with the type **UltraLowLatency** are now available with a subscription to Wowza Video with Ultra Low Latency.

June 19, 2018

Beta

- Established v1.3 as the base path for the beta version of the API.

Current

- Established v1.2 as the current version of the API.

Supported

- Established v1.1 as a supported version of the API. Version 1.0 continues to be supported.

June 5, 2018

Beta

- Added stream target endpoints specific to custom stream targets. Use `/stream_targets/custom` to create, fetch, update, and remove custom stream targets.

New endpoints:

- POST `/stream_targets/custom`
 - GET `/stream_targets/custom`
 - GET `/stream_targets/custom/[custom_stream_target_id]`
 - PATCH `/stream_targets/custom/[custom_stream_target_id]`
 - DELETE `/stream_targets/custom/[custom_stream_target_id]`
- Added stream target endpoints specific to Wowza stream targets. Use `/stream_targets/wowza` to create, fetch, update, and remove Wowza stream targets.
- New endpoints:
- POST `/stream_targets/wowza`
 - GET `/stream_targets/wowza`
 - GET `/stream_targets/wowza/[wowza_stream_target_id]`
 - PATCH `/stream_targets/wowza/[wowza_stream_target_id]`
 - DELETE `/stream_targets/wowza/[wowza_stream_target_id]`
- Updated the stream targets endpoint so that GET `/stream_targets` returns limited details about custom stream targets and Wowza stream targets. GET requests to the stream targets endpoint now return only `id`, `type`, and `name` parameters for all types of stream targets.

May 3, 2018

Beta, Current, and Supported

- Added the **redundantChunklists** stream target playlist property, which determines whether Wowza Video creates and sends redundant chunklists within the playlist. If playback falters, players that support redundancy can switch to the redundant chunklist. See [Set advanced properties with the Wowza Video REST API](#) for information on how to use this property. Affects the calls and endpoints
 - POST /stream_targets/[*stream_target_id*]/properties,
 - GET /stream_targets/[*stream_target_id*]/properties, and
 - GET /stream_targets/[*stream_target_id*]/properties/[*id*]

April 23, 2018

Beta

- Added stream target endpoints specific to ultra low latency stream targets. Use /stream_targets/ull to create, fetch, update, and remove ultra low latency stream targets.
New endpoints:
 - POST /stream_targets/ull
 - GET /stream_targets/ull
 - GET /stream_targets/ull/[*ull_stream_target_id*]
 - PATCH /stream_targets/ull/[*ull_stream_target_id*]
 - DELETE /stream_targets/ull/[*ull_stream_target_id*]
- Updated the stream targets endpoint so that GET /stream_targets returns limited details about ultra low latency stream targets. GET requests to the stream targets endpoint now return only *id*, *type*, and *name* parameters for ultra low latency stream targets.

April 9, 2018

Beta

- Added the *state* parameter, which indicates the state of a stream target. Only stream targets with the type **UltraLowLatencyStreamTarget** provide this parameter. Valid stream target states are **started**, **stopped**, and **error**. The *state* is returned for GET and POST calls made to /stream_targets as well as GET and PATCH calls made to /stream_targets/[*stream_target_id*].
- Updated the release notes format for ease of use. Content for beta, current, and supported versions is now in a changelog on the same page. Newest information is at the top, and changelog entries include release dates. You can compare updates between release versions and more easily skim the content.

Current

- Updated the release notes format for ease of use. Content for beta, current, and supported versions is now in a changelog on the same page. Newest information is at the top, and changelog entries include release dates. You can compare updates between release versions and more easily skim the content.

Supported

- Updated the release notes format for ease of use. Content for beta, current, and supported versions is now in a changelog on the same page. Newest information is at the top, and changelog entries include release dates. You can compare updates between release versions and more easily skim the content.

March 19, 2018

Beta

- Established v1.2 as the base path for the beta version of the API.
- Deprecated the *video_fallback* parameter, which was in preview release. Video fallback will continue to work on transcoders for which it's enabled. However, you can no longer enable it, and if you turn it off for a transcoder, you can't turn it back on.

Current

- Established v1.1 as the current version of the API.

Supported

- Established v1 as a supported version of the API.

March 8, 2018

Beta

- Added the **playlistSeconds** property, which defines the maximum allowable length of the playlist, to the **playlist** section of the stream target configuration table. Valid values are any integer between **6** and **200**, inclusive. See [Create custom media playlists with the Wowza Video REST API](#) for information on how to use the property to create custom chunklists. Affects these calls and endpoints:
 - POST /stream_targets/[*stream_target_id*]/properties
 - GET /stream_targets/[*stream_target_id*]/properties
 - GET /stream_targets/[*stream_target_id*]/properties/[*id*]

February 26, 2018

Beta

- Extended the allowable length of the *stream_name* parameter for custom stream targets: The protocol, host name, and path components of the stream name URL can be up to 255 characters. The query strings and parameter components of the stream name URL can be up to 1024 characters. Affects POST requests to /stream_targets.
- Added the ability to filter the results of GET queries to the /transcoders endpoint. To learn more about the *filter* parameter, see [How to get filtered query results with the Wowza Video REST API](#).

February 7, 2018

Beta

- Established v1.1 as the base path for the beta version of the API.
- Added reference documentation for the beta version of the API at /api/beta/docs.
- Created release notes for the beta version of the API to /api/beta/notes.
- Added the *ip_address* parameter to the response for GET calls to /transcoders/[*transcoder_id*]/state and /live_streams/[*live_stream_id*]/state.

Current

- Established v1 as the base path for the current version of the API.
- Changed the URL for the documentation of the current version of the API to /api/v1/docs.
- Created release notes for the current version of the API at /api/v1/notes.
- Added the /api/versions endpoint. A GET call to /versions returns the base URI and status (**current**, **supported**, or **deprecated**) of each non-beta version of the API.