# Table of Contents

Welcome .................................................................................................................................................. 1  
Resources ................................................................................................................................................ 3  
  Request Account Endpoint .................................................................................................................... 3  
    URI .................................................................................................................................................. 3  
    HTTP Methods ............................................................................................................................... 3  
    Schemas ........................................................................................................................................... 4  
    Properties ....................................................................................................................................... 5  
  Job Templates ...................................................................................................................................... 6  
    URI .................................................................................................................................................. 6  
    HTTP Methods ............................................................................................................................... 6  
    Schemas ........................................................................................................................................... 8  
    Properties ....................................................................................................................................... 38  
  A Specific Job Template ...................................................................................................................... 155  
    URI ................................................................................................................................................ 155  
    HTTP Methods ............................................................................................................................. 155  
    Schemas ....................................................................................................................................... 159  
    Properties ..................................................................................................................................... 188  
  Jobs ................................................................................................................................................... 304  
    URI ................................................................................................................................................ 304  
    HTTP Methods ............................................................................................................................. 304  
    Schemas ....................................................................................................................................... 306  
    Properties ..................................................................................................................................... 337  
  A Specific Job ..................................................................................................................................... 455  
    URI ................................................................................................................................................ 455  
    HTTP Methods ............................................................................................................................. 455  
    Schemas ....................................................................................................................................... 458  
    Properties ..................................................................................................................................... 468  
  Presets .............................................................................................................................................. 585  
    URI ................................................................................................................................................ 585  
    HTTP Methods ............................................................................................................................. 585  
    Schemas ....................................................................................................................................... 588  
    Properties ..................................................................................................................................... 609  
  A Specific Preset ............................................................................................................................... 698  
    URI ................................................................................................................................................ 698  
    HTTP Methods ............................................................................................................................. 699  
    Schemas ....................................................................................................................................... 702  
    Properties ..................................................................................................................................... 724  
  Queues .............................................................................................................................................. 812  
    URI ................................................................................................................................................ 812  
    HTTP Methods ............................................................................................................................. 812  
    Schemas ....................................................................................................................................... 815  
    Properties ..................................................................................................................................... 816  
  A Specific Queue ............................................................................................................................... 819  
    URI ................................................................................................................................................ 819  
    HTTP Methods ............................................................................................................................. 819  
    Schemas ....................................................................................................................................... 823  
    Properties ..................................................................................................................................... 824
Welcome

A Note About Using the CLI

When you send requests via the CLI, you need use PascalCase for all properties. For example, if you used the properties `settings` and `timecodeConfig` in your API call, you would change those to `Settings` and `TimecodeConfig` for your CLI call.

About this Reference

The Resources portion of this guide contains a separate topic for each high level resource: jobs, job templates, presets, and queues. There is also a topic for working with a single specific instance of each of these resources. Within each topic are the available REST methods for that resource and the schema for your request body.

Note: The schemas provided are not working examples. See the User Guide for example job settings.

Getting Started with the REST API

To use the AWS Elemental MediaConvert API, first request your account endpoint. Then send your transcoding service requests to that endpoint. Follow these steps:

Step 1: Set up your permissions and file locations.

For details about setting up IAM permissions and Amazon S3 buckets, see the Getting Started topic of the User Guide. For details about your credentials signature, see Signature Version 4 Signing Process.

Step 2: Request your account endpoint.

Send a POST request to one of the following endpoints with an empty body:

us-east-1: https://mediaconvert.us-east-1.amazonaws.com/2017-08-29/endpoints
us-west-1: https://mediaconvert.us-west-1.amazonaws.com/2017-08-29/endpoints
us-west-2: https://mediaconvert.us-west-2.amazonaws.com/2017-08-29/endpoints
eu-west-1: https://mediaconvert.eu-west-1.amazonaws.com/2017-08-29/endpoints
eu-central-1: https://mediaconvert.eu-central-1.amazonaws.com/2017-08-29/endpoints
ap-northeast-1: https://mediaconvert.ap-northeast-1.amazonaws.com/2017-08-29/endpoints
ap-southeast-1: https://mediaconvert.ap-southeast-1.amazonaws.com/2017-08-29/endpoints
ap-southeast-2: https://mediaconvert.ap-southeast-2.amazonaws.com/2017-08-29/endpoints

The service returns an endpoint similar to "https://abcd1234.mediaconvert.us-west-1.amazonaws.com", where the first eight digits after “https://” are an alphanumeric combination unique to your account and the region matches the regional endpoint you send the request to.

Step 3: Optional. Use preconfigured Postman collections.

If you are using Postman to send your requests, you can copy and import the preconfigured Postman collections available in the User Guide.

Step 4: Send your transcoding requests.
Use the account endpoint you receive from your post to the regional endpoint to send your requests for working with your transcoding jobs, queues, job templates, and presets. For general information about how these resources work, see the User Guide. For high level information about each resource, including information about each transcoding setting, see the Resources portion of this guide.
Resources

The AWS Elemental MediaConvert REST API includes the following resources.

Topics
• Request Account Endpoint (p. 3)
• Job Templates (p. 6)
• A Specific Job Template (p. 155)
• Jobs (p. 304)
• A Specific Job (p. 455)
• Presets (p. 585)
• A Specific Preset (p. 698)
• Queues (p. 812)
• A Specific Queue (p. 819)

Request Account Endpoint

URI
/endpoints

HTTP Methods
POST

Operation ID: DescribeEndpoints

Send an request with an empty body to the regional API endpoint to get your account API endpoint.

Responses

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>DescribeEndpointsResponse</td>
<td>200: OkResponse</td>
</tr>
<tr>
<td>400</td>
<td>ExceptionBody (p. 5)</td>
<td>400: BadRequestException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The conditional request failed. The service can't process your request because of a problem in the request. Please check your request form and syntax.</td>
</tr>
<tr>
<td>500</td>
<td>ExceptionBody (p. 5)</td>
<td>500: InternalServiceException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The service encountered an unexpected condition and cannot fulfill your request.</td>
</tr>
</tbody>
</table>
### Status Code

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>403</td>
<td>ExceptionBody (p. 5)</td>
<td>403: AccessDeniedException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>You don't have permissions for this action with the credentials you sent. Please check your authorization credentials. You should be sending credentials using the AWS Signature Version 4 signing process.</td>
</tr>
<tr>
<td>404</td>
<td>ExceptionBody (p. 5)</td>
<td>404: ResourceNotFoundException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The resource you requested does not exist.</td>
</tr>
<tr>
<td>429</td>
<td>ExceptionBody (p. 5)</td>
<td>429: LimitExceededException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Too many requests have been sent in too short of a time. The service limits the rate at which it will accept requests. For example, you may be hitting your account limits for preset creation or job submission.</td>
</tr>
<tr>
<td>409</td>
<td>ExceptionBody (p. 5)</td>
<td>409: ResourceInUseException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The service could not complete your request because there is a conflict with the current state of the resource. For example, you may be trying to delete a Queue that has jobs processing.</td>
</tr>
</tbody>
</table>

### Schemas

#### Request Bodies

**Example POST**

```json
{
   "nextToken (p. 5)": "string",
   "maxResults (p. 5)": integer
}
```

### Response Bodies

**Example DescribeEndpointsResponse**

```json
{
   "endpoints (p. 5)": [  
```
Example ExceptionBody

```json
{
    "message (p. 6)": "string"
}
```

Properties

**DescribeEndpointsRequest**

**nextToken**

Use this string, provided with the response to a previous request, to request the next batch of endpoints.

*Type: string*

*Required: False*

**maxResults**

*Type: integer*

*Required: False*

*Format: int32*

**DescribeEndpointsResponse**

**endpoints**

List of endpoints

*Type: Array of type Endpoint (p. 5)*

*Required: False*

**nextToken**

Use this string to request the next batch of endpoints.

*Type: string*

*Required: False*

**Endpoint**

**url**

URL of your account-specific endpoint.

*Type: string*

*Required: False*
ExceptionBody

message

Type: string
Required: False

Job Templates

URI
/jobTemplates

HTTP Methods
GET

Operation ID: ListJobTemplates

Retrieve a JSON array of up to twenty of your job templates. This will return the templates themselves, not just a list of them. To retrieve the next twenty templates, use the nextToken string returned with the array

Query Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>listBy</td>
<td>String</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>nextToken</td>
<td>String</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>maxResults</td>
<td>String</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>order</td>
<td>String</td>
<td>False</td>
<td></td>
</tr>
</tbody>
</table>

Responses

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>ListJobTemplatesResponse</td>
<td></td>
</tr>
</tbody>
</table>
| 400         | ExceptionBody (p. 38)   | 400: BadRequestException
The conditional request failed. The service can't process your request because of a problem in the request. Please check your request form and syntax. |
| 500         | ExceptionBody (p. 38)   | 500: InternalServiceException
The service encountered an unexpected condition and cannot fulfill your request. |
### Status Code | Response Model | Description
--- | --- | ---
403 | ExceptionBody (p. 38) | 403: AccessDeniedException
You don't have permissions for this action with the credentials you sent. Please check your authorization credentials. You should be sending credentials using the AWS Signature Version 4 signing process.

404 | ExceptionBody (p. 38) | 404: ResourceNotFoundException
The resource you requested does not exist.

429 | ExceptionBody (p. 38) | 429: LimitExceededException
Too many requests have been sent in too short of a time. The service limits the rate at which it will accept requests. For example, you may be hitting your account limits for preset creation or job submission.

409 | ExceptionBody (p. 38) | 409: ResourceInUseException
The service could not complete your request because there is a conflict with the current state of the resource. For example, you may be trying to delete a Queue that has jobs processing.

### POST

**Operation ID: CreateJobTemplate**

Create a new job template. For information about job templates see the User Guide at http://docs.aws.amazon.com/mediaconvert/latest/ug/what-is.html

**Responses**

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
</table>
| 201 | CreateJobTemplateResponse | 201: CreatedResponse
Your resource has been successfully created. |
| 400 | ExceptionBody (p. 38) | 400: BadRequestException
The conditional request failed. The service can't process your |
### Status Code

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>500</td>
<td>ExceptionBody (p. 38)</td>
<td>An unexpected condition has occurred and the service cannot fulfill your request. Please check your request form and syntax.</td>
</tr>
<tr>
<td>403</td>
<td>ExceptionBody (p. 38)</td>
<td>You don't have permissions for this action with the credentials you sent. Please check your authorization credentials. You should be sending credentials using the AWS Signature Version 4 signing process.</td>
</tr>
<tr>
<td>404</td>
<td>ExceptionBody (p. 38)</td>
<td>The resource you requested does not exist.</td>
</tr>
<tr>
<td>429</td>
<td>ExceptionBody (p. 38)</td>
<td>Too many requests have been sent in too short a time. The service limits the rate at which it will accept requests. For example, you may be hitting your account limits for preset creation or job submission.</td>
</tr>
<tr>
<td>409</td>
<td>ExceptionBody (p. 38)</td>
<td>The service could not complete your request because there is a conflict with the current state of the resource. For example, you may be trying to delete a Queue that has jobs processing.</td>
</tr>
</tbody>
</table>

### Schemas

#### Request Bodies

**Example GET**

```json
{
}
```
Example POST

{
  "settings (p. 59)": {
    "timecodeConfig (p. 110)": {
      "timestampOffset (p. 148)": "string",
      "anchor (p. 148)": "string",
      "start (p. 148)": "string",
      "source (p. 148)": enum
    },
    "adAvailOffset (p. 110)": integer,
    "nielsenConfiguration (p. 110)": {
      "distributorId (p. 135)": "string",
      "breakoutCode (p. 135)": integer
    },
    "inputs (p. 110)": [
      {
        "audioSelectors (p. 104)": {
          },
        "audioSelectorGroups (p. 105)": {
          },
        "filterEnable (p. 105)": enum,
        "deblockFilter (p. 105)": enum,
        "videoSelector (p. 105)": {
          "colorSpace (p. 153)": enum,
          "hdr10Metadata (p. 153)": {
            "redPrimaryY (p. 93)": integer,
            "greenPrimaryY (p. 93)": integer,
            "whitePointX (p. 94)": integer,
            "maxLuminance (p. 94)": integer,
            "greenPrimaryX (p. 94)": integer,
            "whitePointY (p. 94)": integer,
            "redPrimaryX (p. 94)": integer,
            "bluePrimaryX (p. 94)": integer,
            "maxFrameAverageLightLevel (p. 95)": integer,
            "BluePrimaryY (p. 95)": integer,
            "maxContentLightLevel (p. 95)": integer,
            "minLuminance (p. 95)": integer
          },
          "programNumber (p. 153)": integer,
          "pid (p. 154)": integer,
          "colorSpaceUsage (p. 154)": enum
        },
        "filterStrength (p. 105)": integer,
        "programNumber (p. 105)": integer,
        "timecodeSource (p. 105)": enum,
        "captionSelectors (p. 105)": {
          },
        "denoiseFilter (p. 106)": enum,
        "psiControl (p. 106)": enum,
        "inputClippings (p. 106)": [
          {
            "startTimecode (p. 103)": "string",
            "endTimecode (p. 103)": "string"
          }
        ]
      }
    ]
  }
}
},
"outputGroups (p. 110)": [
    "outputs (p. 139)": [
        "extension (p. 137)": "string",
        "videoDescription (p. 138)": {
            "fixedAf (p. 150)": integer,
            "scalingBehavior (p. 151)": enum,
            "respondToAf (p. 151)": enum,
            "codecSettings (p. 151)": {
                "h265Settings (p. 150)": {
                    "slices (p. 87)": integer,
                    "minIInterval (p. 88)": integer,
                    "parNumerator (p. 88)": integer,
                    "flickerAdaptiveQuantization (p. 88)": enum,
                    "gopSizeUnits (p. 88)": enum,
                    "hrdBufferSize (p. 88)": integer,
                    "qualityTuningLevel (p. 88)": enum,
                    "maxBitrate (p. 88)": integer,
                    "bitrate (p. 89)": integer,
                    "spatialAdaptiveQuantization (p. 89)": enum,
                    "sampleAdaptiveOffsetFilterMode (p. 89)": enum,
                    "temporalIds (p. 89)": enum,
                    "slowPal (p. 89)": enum,
                    "tiles (p. 89)": enum,
                    "codeProfile (p. 89)": enum,
                    "alternateTransferFunctionSei (p. 89)": enum,
                    "unregisteredSeiTimecode (p. 89)": enum,
                    "framerateControl (p. 90)": enum,
                    "telecine (p. 90)": enum,
                    "framerateConversionAlgorithm (p. 90)": enum,
                    "codeLevel (p. 90)": enum,
                    "numberReferenceFrames (p. 90)": integer,
                    "hrdBufferInitialFillPercentage (p. 90)": integer,
                    "framerateNumerator (p. 90)": integer,
                    "framerateDenominator (p. 90)": integer,
                    "gopClosedCadence (p. 91)": integer,
                    "framerateNumerator (p. 91)": integer,
                    "framerateDenominator (p. 91)": integer,
                    "gopSize (p. 91)": number,
                    "gopReference (p. 91)": enum,
                    "sceneChangeDetect (p. 92)": enum,
                    "parDenominator (p. 92)": integer,
                    "parControl (p. 92)": enum,
                    "rateControlMode (p. 92)": enum
                },
                "proresSettings (p. 150)": {
                    "framerateControl (p. 141)": enum,
                    "telecine (p. 142)": enum,
                    "framerateDenominator (p. 142)": integer,
                    "framerateConversionAlgorithm (p. 142)": enum,
                    "interlaceMode (p. 142)": enum,
                    "codeProfile (p. 142)": enum,
                    "parNumerator (p. 142)": integer,
                    "parControl (p. 142)": enum,
                    "parDenominator (p. 142)": integer,
                    "framerateNumerator (p. 143)": integer,
                    "gopSize (p. 143)": number,
                    "gopReference (p. 143)": enum,
                    "sceneChangeDetect (p. 144)": enum,
                    "parDenominator (p. 144)": integer,
                    "parControl (p. 144)": enum,
                    "framerateNumerator (p. 145)": integer,
                    "framerateDenominator (p. 145)": integer
                },
                "mpeg2Settings (p. 150)": {
                    "minIInterval (p. 129)": integer,
                    "parNumerator (p. 130)": integer,
"gopSizeUnits (p. 130)": enum,
"hrdBufferSize (p. 130)": integer,
"qualityTuningLevel (p. 130)": enum,
"maxBitrate (p. 130)": integer,
"bitrate (p. 130)": integer,
"spatialAdaptiveQuantization (p. 131)": enum,
"slowPal (p. 131)": enum,
"codecProfile (p. 131)": enum,
"intraDcPrecision (p. 131)": enum,
"softness (p. 131)": integer,
"framerateControl (p. 131)": enum,
"telecine (p. 131)": enum,
"framerateConversionAlgorithm (p. 131)": enum,
"codecLevel (p. 131)": enum,
"temporalAdaptiveQuantization (p. 132)": enum,
"framerateNumerator (p. 132)": integer,
"numberBFramesBetweenReferenceFrames (p. 132)": integer,
"gopClosedCadence (p. 132)": integer,
"framerateDenominator (p. 132)": integer,
"adaptiveQuantization (p. 132)": enum,
"interlaceMode (p. 133)": enum,
"gopSize (p. 133)": number,
"sceneChangeDetect (p. 133)": enum,
"parDenominator (p. 133)": integer,
"parControl (p. 133)": enum,
"syntax (p. 133)": enum,
"rateControlMode (p. 133)": enum
},
"h264Settings (p. 150)": {
  "slices (p. 79)": integer,
  "minIInterval (p. 79)": integer,
  "parNumerator (p. 79)": integer,
  "flickerAdaptiveQuantization (p. 79)": enum,
  "gopSizeUnits (p. 79)": enum,
  "hrdBufferSize (p. 79)": integer,
  "qualityTuningLevel (p. 80)": enum,
  "maxBitrate (p. 80)": integer,
  "bitrate (p. 80)": integer,
  "spatialAdaptiveQuantization (p. 80)": enum,
  "slowPal (p. 80)": enum,
  "codecProfile (p. 80)": enum,
  "unregisteredSeiTimecode (p. 80)": enum,
  "softness (p. 80)": integer,
  "framerateControl (p. 81)": enum,
  "telecine (p. 81)": enum,
  "framerateConversionAlgorithm (p. 81)": enum,
  "codecLevel (p. 81)": enum,
  "numberReferenceFrames (p. 81)": integer,
  "temporalAdaptiveQuantization (p. 81)": enum,
  "repeatPps (p. 81)": enum,
  "hrdBufferInitialFillPercentage (p. 81)": integer,
  "framerateNumerator (p. 82)": integer,
  "numberBFramesBetweenReferenceFrames (p. 82)": integer,
  "gopClosedCadence (p. 82)": integer,
  "fieldEncoding (p. 82)": enum,
  "entropyEncoding (p. 82)": enum,
  "framerateDenominator (p. 82)": integer,
  "adaptiveQuantization (p. 83)": enum,
  "interlaceMode (p. 83)": enum,
  "gopSize (p. 83)": number,
  "gopBReference (p. 83)": enum,
  "sceneChangeDetect (p. 83)": enum,
  "parDenominator (p. 83)": integer,
  "parControl (p. 83)": enum,
  "syntax (p. 83)": enum,
"rateControlMode (p. 83)" : enum
},
"frameCaptureSettings (p. 150)" : {
  "framerateDenominator (p. 75)" : integer,
  "maxCaptures (p. 75)" : integer,
  "framerateNumerator (p. 75)" : integer,
  "quality (p. 76)" : integer
},
"afdSignaling (p. 151)" : enum,
"colorMetadata (p. 151)" : enum,
"timecodeInsertion (p. 151)" : enum,
"width (p. 151)" : integer,
"sharpness (p. 151)" : integer,
"antiAlias (p. 152)" : enum,
"videoPreprocessors (p. 152)" : {
  "timecodeBurnin (p. 152)" : {
    "prefix (p. 147)" : "string",
    "fontSize (p. 147)" : integer,
    "position (p. 147)" : enum
  },
  "noiseReducer (p. 153)" : {
    "filter (p. 136)" : enum,
    "filterSettings (p. 136)" : {
      "strength (p. 136)" : integer
    },
    "spatialFilterSettings (p. 136)" : {
      "strength (p. 137)" : integer,
      "postFilterSharpenStrength (p. 137)" : integer,
      "speed (p. 137)" : integer
    }
  },
  "colorCorrector (p. 153)" : {
    "saturation (p. 56)" : integer,
    "brightness (p. 57)" : integer,
    "hdr10Metadata (p. 57)" : {
      "redPrimaryY (p. 93)" : integer,
      "greenPrimaryY (p. 93)" : integer,
      "whitePointX (p. 94)" : integer,
      "maxLuminance (p. 94)" : integer,
      "greenPrimaryX (p. 94)" : integer,
      "whitePointY (p. 94)" : integer,
      "redPrimaryX (p. 94)" : integer,
      "bluePrimaryX (p. 94)" : integer,
      "maxFrameAverageLightLevel (p. 95)" : integer,
      "bluePrimaryY (p. 95)" : integer,
      "maxContentLightLevel (p. 95)" : integer,
      "minLuminance (p. 95)" : integer
    },
    "contrast (p. 57)" : integer,
    "hue (p. 57)" : integer,
    "colorSpaceConversion (p. 57)" : enum
  },
  "imageInserter (p. 153)" : {
    "insertableImages (p. 103)" : [
      {
        "duration (p. 106)" : integer,
        "fadeOut (p. 106)" : integer,
        "imageY (p. 107)" : integer,
        "fadeIn (p. 107)" : integer,
        "imageX (p. 107)" : integer,
        "width (p. 107)" : integer,
        "startTime (p. 107)" : "string",
        "opacity (p. 108)" : integer,
        "layer (p. 108)" : integer,
        "height (p. 108)" : integer,
        "position (p. 108)" : integer,
        "timecodeBurnin (p. 108)" : enum
      }
    ]
  }
}
}
"imageInserterInput (p. 108)" : "string"
}
],
"deinterlacer (p. 153)" : {
"mode (p. 62)" : enum,
"control (p. 62)" : enum,
"algorithm (p. 62)" : enum
}
],
"position (p. 152)" : {
"width (p. 143)" : integer,
"x (p. 143)" : integer,
"y (p. 143)" : integer,
"height (p. 144)" : integer
},
"dropFrameTimecode (p. 152)" : enum,
"crop (p. 152)" : {
"width (p. 143)" : integer,
"x (p. 143)" : integer,
"y (p. 143)" : integer,
"height (p. 144)" : integer
},
"height (p. 152)" : integer
],
"audioDescriptions (p. 138)" : [
{
"languageCodeControl (p. 45)" : enum,
"audioTypeControl (p. 45)" : enum,
"remixSettings (p. 45)" : {
"channelsOut (p. 144)" : integer,
"channelMapping (p. 144)" : {
"outputChannels (p. 56)" : [
{ "inputChannels (p. 139)" : [ integer ]
]
},
"channelsIn (p. 144)" : integer
},
"audioType (p. 45)" : integer,
"audioSourceName (p. 45)" : "string",
"codecSettings (p. 46)" : {
"codec (p. 44)" : enum,
"wavSettings (p. 44)" : {
"channels (p. 154)" : integer,
"bitDepth (p. 154)" : integer,
"sampleRate (p. 154)" : integer
},
"ac3Settings (p. 44)" : {
"dynamicRangeCompressionProfile (p. 41)" : enum,
"dailnorm (p. 41)" : integer,
"codingMode (p. 42)" : enum,
"metadataControl (p. 42)" : enum,
"lifeFilter (p. 42)" : enum,
"bitrate (p. 42)" : integer,
"bitstreamMode (p. 42)" : enum,
"sampleRate (p. 42)" : integer
},
"aacSettings (p. 44)" : {
"vbrQuality (p. 39)" : enum,
"codecProfile (p. 39)" : enum,
"codingMode (p. 39)" : enum,
"specification (p. 39)" : enum,
"bitrate (p. 40)": integer,
"rawFormat (p. 40)": enum,
"rateControlMode (p. 40)": enum,
"sampleRate (p. 40)": integer,
"audioDescriptionBroadcasterMix (p. 40)": enum
},
"aiffSettings (p. 44)": {
  "channels (p. 43)": integer,
  "bitDepth (p. 43)": integer,
  "sampleRate (p. 43)": integer
},
"eac3Settings (p. 44)": {
  "dialnorm (p. 70)": integer,
  "passthroughControl (p. 70)": enum,
  "metadataControl (p. 70)": enum,
  "bitrate (p. 70)": integer,
  "dynamicRangeCompressionRf (p. 70)": enum,
  "sampleRate (p. 70)": integer,
  "ltRtSurroundMixLevel (p. 71)": number,
  "surroundExMode (p. 71)": enum,
  "dynamicRangeCompressionLine (p. 71)": enum,
  "lfeControl (p. 71)": enum,
  "codingMode (p. 71)": enum,
  "surroundMode (p. 71)": enum,
  "attenuationControl (p. 71)": enum,
  "lfeFilter (p. 71)": enum,
  "phaseControl (p. 71)": enum,
  "ltRtCenterMixLevel (p. 72)": number,
  "dcFilter (p. 72)": enum,
  "stereoDownmix (p. 72)": enum,
  "bitstreamMode (p. 72)": enum,
  "loRoSurroundMixLevel (p. 72)": number,
  "loRoCenterMixLevel (p. 72)": number
},
"mp2Settings (p. 44)": {
  "channels (p. 126)": integer,
  "bitrate (p. 126)": integer,
  "sampleRate (p. 126)": integer
}
"languageCode (p. 46)": enum,
"streamName (p. 46)": "string",
"audioNormalizationSettings (p. 46)": {
  "targetLkfs (p. 47)": number,
  "algorithmControl (p. 47)": enum,
  "peakCalculation (p. 47)": enum,
  "loudnessLogging (p. 47)": enum,
  "correctionGateLevel (p. 47)": integer,
  "algorithm (p. 48)": enum
}
"containerSettings (p. 138)": {
  "container (p. 58)": enum,
  "mp4Settings (p. 58)": {
    "mp4MajorBrand (p. 127)": "string",
    "moovPlacement (p. 127)": enum,
    "cs1gAtom (p. 127)": enum,
    "freeSpaceBox (p. 127)": enum
  },
  "m3u8Settings (p. 58)": {
    "pmtPid (p. 122)": integer,
    "pcrPid (p. 122)": integer,
    "audioPids (p. 123)": [ integer
]}
}
"audioFramesPerPes (p. 123)": integer,
"scte35Pid (p. 123)": integer,
"transportStreamId (p. 123)": integer,
"videoPid (p. 123)": integer,
"pcrControl (p. 123)": enum,
"privateMetadataPid (p. 124)": integer,
"pmtInterval (p. 124)": integer,
"patInterval (p. 124)": integer,
"programNumber (p. 124)": integer,
"timedMetadataPid (p. 124)": integer,
"timedMetadata (p. 124)": enum,
"scte35Source (p. 124)": enum
},
"m2tsSettings (p. 58)": {
  "dvbTeletextPid (p. 117)": integer,
  "bitrate (p. 117)": integer,
  "segmentationTime (p. 118)": number,
  "audioPids (p. 118)": [integer
  ],
  "rateMode (p. 118)": enum,
  "ebpAudioInterval (p. 118)": enum,
  "fragmentTime (p. 118)": number,
  "audioFramesPerPes (p. 118)": integer,
  "maxPcrInterval (p. 118)": integer,
  "scte35Pid (p. 119)": integer,
  "privateMetadataPid (p. 119)": integer,
  "pmtInterval (p. 119)": integer,
  "segmentationStyle (p. 119)": enum,
  "audioBufferModel (p. 119)": enum,
  "programNumber (p. 119)": integer,
  "dvbNitSettings (p. 120)": {
    "networkName (p. 63)": "string",
    "networkId (p. 63)": integer,
    "nitInterval (p. 63)": integer
  },
  "scte35Source (p. 120)": enum,
  "pmtPid (p. 120)": integer,
  "bufferModel (p. 120)": enum,
  "ebpPlacement (p. 120)": enum,
  "dvbSdtSettings (p. 120)": {
    "sdtInterval (p. 63)": integer,
    "serviceName (p. 63)": "string",
    "serviceProviderName (p. 63)": "string",
    "outputSdt (p. 64)": enum
  },
  "nullPacketBitrate (p. 120)": number,
  "pcrPid (p. 120)": integer,
  "minEbpInterval (p. 121)": integer,
  "transportStreamId (p. 121)": integer,
  "pcrControl (p. 121)": enum,
  "videoPid (p. 121)": integer,
  "esRateInPes (p. 121)": enum,
  "segmentationMarkers (p. 121)": enum,
  "dvbTdtSettings (p. 121)": {
    "tdtInterval (p. 68)": integer
  },
  "patInterval (p. 122)": integer,
  "dvbSubPids (p. 122)": [integer
  ]
},
"movSettings (p. 58)": {
  "reference (p. 125)": enum,
  "paddingControl (p. 125)": enum,
  "mpeg2FourCCControl (p. 126)": enum,
"cslgAtom (p. 126)”: enum,
"clapAtom (p. 126)”: enum
},
"f4vSettings (p. 59)”: {
  "moovPlacement (p. 74)”: enum
}
},
"preset (p. 138)”: "string",
"outputSettings (p. 138)”: {
  "hlsSettings (p. 140)”: {
    "iFrameOnlyManifest (p. 102)”: enum,
    "segmentModifier (p. 102)”: "string",
    "audioTrackType (p. 102)”: enum,
    "audioRenditionSets (p. 102)”: "string",
    "audioGroupId (p. 102)”: "string"
  }
},
"captionDescriptions (p. 138)”: [
  {
    "captionSelectorName (p. 54)”: "string",
    "languageDescription (p. 54)”: "string",
    "languageCode (p. 54)”: enum,
    "destinationSettings (p. 54)”: {
      "burninDestinationSettings (p. 54)”: {
        "xPosition (p. 50)”: integer,
        "backgroundColor (p. 50)”: enum,
        "teletextSpacing (p. 50)”: enum,
        "yPosition (p. 50)”: integer,
        "backgroundOpacity (p. 51)”: integer,
        "fontOpacity (p. 51)”: integer,
        "shadowOpacity (p. 51)”: integer,
        "fontResolution (p. 51)”: integer,
        "shadowYOffset (p. 51)”: integer,
        "outlineSize (p. 51)”: integer,
        "foregroundColor (p. 52)”: enum,
        "shadowColor (p. 52)”: enum,
        "fontColor (p. 52)”: enum
      },
      "teletextDestinationSettings (p. 54)”: {
        "pageNumber (p. 146)”: "string"
      },
      "ttmlDestinationSettings (p. 54)": {
        "stylePassthrough (p. 149)”: enum
      },
      "destinationType (p. 54)”: enum,
      "dvbSubDestinationSettings (p. 55)”: {
        "xPosition (p. 64)”: integer,
        "backgroundColor (p. 64)”: enum,
        "teletextSpacing (p. 64)”: enum,
        "yPosition (p. 64)”: integer,
        "backgroundOpacity (p. 65)”: integer,
        "fontOpacity (p. 65)”: integer,
        "shadowOpacity (p. 65)": integer,
        "fontResolution (p. 65)": integer,
        "shadowYOffset (p. 65)": integer,
        "outlineSize (p. 65)": integer,
        "foregroundColor (p. 66)": enum,
        "shadowColor (p. 66)": integer,
        "fontColor (p. 66)": enum
      }
    }
  }
},
"16"
"sccDestinationSettings (p. 55)": {
  "framerate (p. 145)": enum
}
],
"nameModifier (p. 138)": "string"
}
],
"outputGroupSettings (p. 139)": {
  "dashIsoGroupSettings (p. 139)": {
    "fragmentLength (p. 60)": integer,
    "baseUrl (p. 60)": "string",
    "minBufferTime (p. 60)": integer,
    "encryption (p. 60)": {
      "spekeKeyProvider (p. 60)": {
        "resourceId (p. 145)": "string",
        "systemIds (p. 145)": [
          "string"
        ],
        "url (p. 146)": "string"
      }
    },
    "destination (p. 61)": "string",
    "segmentLength (p. 61)": integer,
    "segmentControl (p. 61)": enum,
    "hbbtvCompliance (p. 61)": enum
  },
  "fileGroupSettings (p. 139)": {
    "destination (p. 74)": "string"
  },
  "msSmoothGroupSettings (p. 140)": {
    "fragmentLength (p. 134)": integer,
    "encryption (p. 135)": {
      "spekeKeyProvider (p. 134)": {
        "resourceId (p. 145)": "string",
        "systemIds (p. 145)": [
          "string"
        ],
        "url (p. 146)": "string"
      }
    },
    "audioDeduplication (p. 135)": enum,
    "manifestEncoding (p. 135)": enum,
    "destination (p. 135)": "string"
  },
  "type (p. 140)": enum,
  "hlsGroupSettings (p. 140)": {
    "segmentsPerSubdirectory (p. 98)": integer,
    "streamInfResolution (p. 98)": enum,
    "timestampDeltaMilliseconds (p. 98)": integer,
    "outputSelection (p. 98)": enum,
    "captionLanguageMappings (p. 98)": [
      {
        "languageDescription (p. 96)": "string",
        "captionChannel (p. 96)": integer,
        "languageCode (p. 96)": enum
      }
    ],
    "clientCache (p. 98)": enum,
    "codecSpecification (p. 98)": enum,
    "destination (p. 99)": "string",
    "segmentControl (p. 99)": enum,
    "timedMetadataId3Frame (p. 99)": enum,
    "timedMetadataId3Period (p. 99)": integer,
    "captionLanguageSetting (p. 99)": enum,
"minSegmentLength (p. 99)": integer,
"directoryStructure (p. 99)": enum,
"programDateTime (p. 99)": enum,
"baseUrl (p. 100)": "string",
"encryption (p. 100)": {
  "initializationVectorInManifest (p. 97)": enum,
  "constantInitializationVector (p. 97)": "string",
  "staticKeyProvider (p. 97)": {
    "keyFormatVersions (p. 146)": "string",
    "keyFormat (p. 146)": "string",
    "staticKeyValue (p. 146)": "string",
    "url (p. 146)": "string"
  },
  "type (p. 97)": enum,
  "spekeKeyProvider (p. 97)": {
    "resourceId (p. 145)": "string",
    "systemIds (p. 145)": [
      "string"
    ],
    "url (p. 146)": "string"
  },
  "encryptionMethod (p. 97)": enum
},
"adMarkers (p. 100)": [
  enum
],
"programDateTimePeriod (p. 100)": integer,
"manifestCompression (p. 100)": enum,
"segmentLength (p. 100)": integer,
"manifestDurationFormat (p. 100)": enum
},
"name (p. 139)": "string",
"customName (p. 139)": "string"
},
"timedMetadataInsertion (p. 111)": {
  "id3Insertions (p. 149)": [
    {
      "id3 (p. 103)": "string",
      "timecode (p. 103)": "string"
    }
  ],
  "availBlanking (p. 111)": {
    "availBlankingImage (p. 50)": "string"
  },
  "name (p. 59)": "string",
  "description (p. 59)": "string",
  "category (p. 59)": "string",
  "queue (p. 59)": "string"
}

Response Bodies

Example ListJobTemplatesResponse

{
  "nextToken (p. 115)": "string",
  "jobTemplates (p. 115)": [
    {
      "settings (p. 108)": {
        "timecodeConfig (p. 110)": {
          "timecode (p. 110)": "string",
          "id3 (p. 103)": "string",
          "url (p. 146)": "string"
        },
        "encryption (p. 100)": {
          "initializationVectorInManifest (p. 97)": enum,
          "constantInitializationVector (p. 97)": "string",
          "staticKeyProvider (p. 97)": {
            "keyFormatVersions (p. 146)": "string",
            "keyFormat (p. 146)": "string",
            "staticKeyValue (p. 146)": "string",
            "url (p. 146)": "string"
          },
          "type (p. 97)": enum,
          "spekeKeyProvider (p. 97)": {
            "resourceId (p. 145)": "string",
            "systemIds (p. 145)": [
              "string"
            ],
            "url (p. 146)": "string"
          },
          "encryptionMethod (p. 97)": enum
        },
        "adMarkers (p. 100)": [
          enum
        ],
        "programDateTimePeriod (p. 100)": integer,
        "manifestCompression (p. 100)": enum,
        "segmentLength (p. 100)": integer,
        "manifestDurationFormat (p. 100)": enum
      }
    }
  ]
}
"timestampOffset (p. 148)": "string",
"anchor (p. 148)": "string",
"start (p. 148)": "string",
"source (p. 148)": "string",
"adAvailOffset (p. 110)": integer,
"nielsenConfiguration (p. 110)": {
  "distributorId (p. 135)": "string",
  "breakoutCode (p. 135)": integer
},
"inputs (p. 110)": [
  {
    "audioSelectors (p. 104)": {
    },
    "audioSelectorGroups (p. 105)": {
    },
    "filterEnable (p. 105)": enum,
    "deblockFilter (p. 105)": enum,
    "videoSelector (p. 105)": {
      "colorSpace (p. 153)": enum,
      "hdr10Metadata (p. 153)": {
        "redPrimaryY (p. 93)": integer,
        "greenPrimaryY (p. 93)": integer,
        "whitePointX (p. 94)": integer,
        "maxLuminance (p. 94)": integer,
        "greenPrimaryX (p. 94)": integer,
        "whitePointY (p. 94)": integer,
        "redPrimaryX (p. 94)": integer,
        "bluePrimaryX (p. 94)": integer,
        "maxFrameAverageLightLevel (p. 95)": integer,
        "bluePrimaryY (p. 95)": integer,
        "maxContentLightLevel (p. 95)": integer,
        "minLuminance (p. 95)": integer
      },
      "programNumber (p. 153)": integer,
      "pid (p. 154)": integer,
      "colorSpaceUsage (p. 154)": enum
    },
    "filterStrength (p. 105)": integer,
    "programNumber (p. 105)": integer,
    "timecodeSource (p. 105)": enum,
    "captionSelectors (p. 105)": {
    },
    "denoiseFilter (p. 106)": enum,
    "psiControl (p. 106)": enum,
    "inputClippings (p. 106)": [
      {
        "startTimecode (p. 103)": "string",
        "endTimecode (p. 103)": "string"
      }
    ]
  }
},
"outputGroups (p. 110)": [
  {
    "outputs (p. 139)": [
      {
        "extension (p. 137)": "string",
        "videoDescription (p. 138)": {
          "fixedAfd (p. 150)": integer,
          "scalingBehavior (p. 151)": enum,
          "respondToAfd (p. 151)": enum,
          "codecSettings (p. 151)": {
            "h265Settings (p. 150)": {
              "slices (p. 87)": integer,
              "minIInterval (p. 88)": integer,
              "profile (p. 149)": "string",
              "level (p. 149)": "string",
              "sliceSegmentation (p. 149)": enum,
              "bitrate (p. 86)": integer
            },
            "h264Settings (p. 150)": {
              "profile (p. 86)": "string",
              "level (p. 86)": "string",
              "sliceSegmentation (p. 86)": enum,
              "bitrate (p. 85)": integer
            }
          },
          "videoSelector (p. 154)": {
            "colorSpace (p. 154)": enum,
            "filterEnable (p. 154)": enum,
            "filterStrength (p. 154)": integer
          }
        }
      }
    ]
  }
]
"parNumerator (p. 88)": integer,
"flickerAdaptiveQuantization (p. 88)": enum,
"gopSizeUnits (p. 88)": enum,
"hrdBufferSize (p. 88)": integer,
"qualityTuningLevel (p. 88)": enum,
"maxBitrate (p. 88)": integer,
"bitrate (p. 89)": integer,
"spatialAdaptiveQuantization (p. 89)": enum,
"sampleAdaptiveOffsetFilterMode (p. 89)": enum,
"temporalIds (p. 89)": enum,
"slowPal (p. 89)": enum,
"tiles (p. 89)": enum,
"codecProfile (p. 89)": enum,
"alternateTransferFunctionSei (p. 89)": enum,
"unregisteredSeiTimecode (p. 89)": enum,
"framerateControl (p. 90)": enum,
"telecine (p. 90)": enum,
"framerateConversionAlgorithm (p. 90)": enum,
"codecLevel (p. 90)": enum,
"numberReferenceFrames (p. 90)": integer,
"temporalAdaptiveQuantization (p. 90)": enum,
"hrdBufferInitialFillPercentage (p. 90)": integer,
"framerateNumerator (p. 90)": integer,
"numberBFramesBetweenReferenceFrames (p. 91)": integer,
"gopClosedCadence (p. 91)": integer,
"framerateDenominator (p. 91)": integer,
"adaptiveQuantization (p. 91)": enum,
"framerateMode (p. 91)": enum,
"gopSize (p. 91)": number,
"gopReference (p. 91)": enum,
"sceneChangeDetect (p. 92)": enum,
"parDenominator (p. 92)": integer,
"parControl (p. 92)": enum,
"rateControlMode (p. 92)": enum
},
"codec (p. 150)": enum,
"proresSettings (p. 150)": {
  "slowPal (p. 141)": enum,
  "framerateControl (p. 141)": enum,
  "telecine (p. 142)": enum,
  "framerateConversionAlgorithm (p. 142)": enum,
  "telecine (p. 142)": enum,
  "framerateConversionAlgorithm (p. 142)": enum,
  "gopSize (p. 142)": number,
  "intraDcPrecision (p. 131)": enum,
  "softness (p. 131)": integer,
  "framerateControl (p. 131)": enum,
  "telecine (p. 131)": enum,
  "framerateConversionAlgorithm (p. 131)": enum,
  "codecProfile (p. 131)": enum,
  "codeLevel (p. 131)": enum,
"temporalAdaptiveQuantization (p. 132)": enum,
"hrdBufferInitialFillPercentage (p. 132)": integer,
"framerateNumerator (p. 132)": integer,
"numberBFramesBetweenReferenceFrames (p. 132)": integer,
"gopClosedCadence (p. 132)": integer,
"framerateDenominator (p. 132)": integer,
"adaptiveQuantization (p. 132)": enum,
"interlaceMode (p. 133)": enum,
"gopSize (p. 133)": number,
"sceneChangeDetect (p. 133)": enum,
"parDenominator (p. 133)": integer,
"parControl (p. 133)": enum,
"syntax (p. 133)": enum,
"rateControlMode (p. 133)": enum
},
"h264Settings (p. 150)": {
  "slices (p. 79)": integer,
  "minIInterval (p. 79)": integer,
  "parNumerator (p. 79)": integer,
  "flickerAdaptiveQuantization (p. 79)": enum,
  "gopSizeUnits (p. 79)": enum,
  "hrdBufferSize (p. 79)": integer,
  "qualityTuningLevel (p. 80)": enum,
  "maxBitrate (p. 80)": integer,
  "bitrate (p. 80)": integer,
  "spatialAdaptiveQuantization (p. 80)": enum,
  "slowPal (p. 80)": enum,
  "codecProfile (p. 80)": enum,
  "unregisteredSeiTimecode (p. 80)": enum,
  "softness (p. 80)": integer,
  "framerateControl (p. 81)": enum,
  "telecine (p. 81)": enum,
  "framerateConversionAlgorithm (p. 81)": enum,
  "codecLevel (p. 81)": enum,
  "numberReferenceFrames (p. 81)": integer,
  "temporalAdaptiveQuantization (p. 81)": enum,
  "repeatPps (p. 81)": enum,
  "hrdBufferInitialFillPercentage (p. 81)": integer,
  "framerateNumerator (p. 82)": integer,
  "numberBFramesBetweenReferenceFrames (p. 82)": integer,
  "gopClosedCadence (p. 82)": integer,
  "fieldEncoding (p. 82)": enum,
  "entropyEncoding (p. 82)": enum,
  "framerateDenominator (p. 82)": integer,
  "adaptiveQuantization (p. 83)": enum,
  "interlaceMode (p. 83)": enum,
  "gopSize (p. 83)": number,
  "gopBReference (p. 83)": enum,
  "sceneChangeDetect (p. 83)": enum,
  "parDenominator (p. 83)": integer,
  "parControl (p. 83)": enum,
  "syntax (p. 83)": enum,
  "rateControlMode (p. 83)": enum
},
"frameCaptureSettings (p. 150)": {
  "framerateDenominator (p. 75)": integer,
  "maxCaptures (p. 75)": integer,
  "framerateNumerator (p. 75)": integer,
  "quality (p. 76)": integer
},
"antiAlias (p. 152)": enum,
"videoPreprocessors (p. 152)": {
  "timecodeBurnin (p. 152)": {
    "prefix (p. 147)": "string",
    "fontSize (p. 147)": integer,
    "position (p. 147)": enum
  },
  "noiseReducer (p. 153)": {
    "filter (p. 136)": enum,
    "filterSettings (p. 136)": {
      "strength (p. 136)": integer
    },
    "spatialFilterSettings (p. 136)": {
      "strength (p. 137)": integer,
      "postFilterSharpenStrength (p. 137)": integer,
      "speed (p. 137)": integer
    }
  },
  "colorCorrector (p. 153)": {
    "saturation (p. 56)": integer,
    "brightness (p. 57)": integer,
    "hdr10Metadata (p. 57)": {
      "redPrimaryY (p. 93)": integer,
      "greenPrimaryY (p. 93)": integer,
      "whitePointX (p. 94)": integer,
      "maxLuminance (p. 94)": integer,
      "greenPrimaryX (p. 94)": integer,
      "whitePointY (p. 94)": integer,
      "redPrimaryX (p. 94)": integer,
      "bluePrimaryX (p. 94)": integer,
      "maxFrameAverageLightLevel (p. 95)": integer,
      "bluePrimaryY (p. 95)": integer,
      "maxContentLightLevel (p. 95)": integer,
      "minLuminance (p. 95)": integer
    },
    "contrast (p. 57)": integer,
    "hue (p. 57)": integer,
    "colorSpaceConversion (p. 57)": enum
  },
  "imageInserter (p. 153)": {
    "insertableImages (p. 103)": [
      {
        "duration (p. 106)": integer,
        "fadeOut (p. 106)": integer,
        "imageY (p. 107)": integer,
        "fadeIn (p. 107)": integer,
        "imageX (p. 107)": integer,
        "width (p. 107)": integer,
        "startTime (p. 107)": "string",
        "opacity (p. 108)": integer,
        "layer (p. 108)": integer,
        "height (p. 108)": integer,
        "imageInserterInput (p. 108)": "string"
      }
    ]
  },
  "deinterlacer (p. 153)": {
    "mode (p. 62)": enum,
    "control (p. 62)": enum,
    "algorithm (p. 62)": enum
  }
},
"position (p. 152)": {
  "width (p. 143)": integer,
  "x (p. 143)": integer,
  "y (p. 143)": integer,
"height (p. 144)": integer,
"dropFrameTimecode (p. 152)": enum,
"crop (p. 152)": {
  "width (p. 143)": integer,
  "x (p. 143)": integer,
  "y (p. 143)": integer,
  "height (p. 144)": integer
},
"height (p. 152)": integer
},
"audioDescriptions (p. 138)": [
  {
    "languageCodeControl (p. 45)": enum,
    "audioTypeControl (p. 45)": enum,
    "remixSettings (p. 45)": {
      "channelsOut (p. 144)": integer,
      "channelMapping (p. 144)": {
        "outputChannels (p. 56)": [
          "inputChannels (p. 139)": [integer]
        ]
      }
    },
    "channelsIn (p. 144)": integer,
    "audioType (p. 45)": integer,
    "audioSourceName (p. 45)": "string",
    "codecSettings (p. 46)": {
      "codec (p. 44)": enum,
      "wavSettings (p. 44)": {
        "channels (p. 154)": integer,
        "bitDepth (p. 154)": integer,
        "sampleRate (p. 154)": integer
      },
      "ac3Settings (p. 44)": {
        "dynamicRangeCompressionProfile (p. 41)": enum,
        "dialnorm (p. 41)": integer,
        "codingMode (p. 42)": enum,
        "metadataControl (p. 42)": enum,
        "lfeFilter (p. 42)": enum,
        "bitrate (p. 42)": integer,
        "bitstreamMode (p. 42)": enum,
        "sampleRate (p. 42)": integer
      },
      "aacSettings (p. 44)": {
        "vbrQuality (p. 39)": enum,
        "codecProfile (p. 39)": enum,
        "codingMode (p. 39)": enum,
        "specification (p. 39)": enum,
        "bitrate (p. 40)": integer,
        "rawFormat (p. 40)": enum,
        "rateControlMode (p. 40)": enum,
        "sampleRate (p. 40)": integer,
        "audioDescriptionBroadcasterMix (p. 40)": enum
      },
      "aiffSettings (p. 44)": {
        "channels (p. 43)": integer,
        "bitDepth (p. 43)": integer,
        "sampleRate (p. 43)": integer
      },
      "eac3Settings (p. 44)": {
        "dialnorm (p. 70)": integer,
        "passthroughControl (p. 70)": enum,
"metadataControl (p. 70)": enum,
"bitrate (p. 70)": integer,
"dynamicRangeCompressionRf (p. 70)": enum,
"sampleRate (p. 70)": integer,
"ltRtSurroundMixLevel (p. 71)": number,
"surroundExMode (p. 71)": enum,
"dynamicRangeCompressionLine (p. 71)": enum,
"lfeControl (p. 71)": enum,
"codingMode (p. 71)": enum,
"surroundMode (p. 71)": enum,
"attenuationControl (p. 71)": enum,
"lfeFilter (p. 71)": enum,
"phaseControl (p. 71)": enum,
"ltRtCenterMixLevel (p. 72)": number,
"dcFilter (p. 72)": enum,
"stereoDownmix (p. 72)": enum,
"bitstreamMode (p. 72)": enum,
"loRoSurroundMixLevel (p. 72)": number,
"loRoCenterMixLevel (p. 72)": number,
"mp2Settings (p. 44)": {
  "channels (p. 126)": integer,
  "bitrate (p. 126)": integer,
  "sampleRate (p. 126)": integer
},
"languageCode (p. 46)": enum,
"streamName (p. 46)": "string",
"audioNormalizationSettings (p. 46)": {
  "targetLkfs (p. 47)": number,
  "algorithmControl (p. 47)": enum,
  "peakCalculation (p. 47)": enum,
  "loudnessLogging (p. 47)": enum,
  "correctionGateLevel (p. 47)": integer,
  "algorithm (p. 48)": enum
},
"containerSettings (p. 138)": {
  "container (p. 58)": enum,
  "mp4Settings (p. 58)": {
    "mp4MajorBrand (p. 127)": "string",
    "moovPlacement (p. 127)": enum,
    "cslgAtom (p. 127)": enum,
    "freeSpaceBox (p. 127)": enum
  },
  "m3u8Settings (p. 58)": {
    "pmtPid (p. 122)": integer,
    "pcrPid (p. 122)": integer,
    "audioPids (p. 123)": [
      integer
    ],
    "audioFramesPerPes (p. 123)": integer,
    "scte35Pid (p. 123)": integer,
    "transportStreamId (p. 123)": integer,
    "videoPid (p. 123)": integer,
    "pcrControl (p. 123)": enum,
    "privateMetadataPid (p. 124)": integer,
    "pmtInterval (p. 124)": integer,
    "patInterval (p. 124)": integer,
    "programNumber (p. 124)": integer,
    "timedMetadataPid (p. 124)": integer,
    "timedMetadata (p. 124)": enum,
    "scte35Source (p. 124)": enum
  },
  "m2tsSettings (p. 58)": {
    ...
"dvbTeletextPid (p. 117)": integer,
"bitrate (p. 117)": integer,
"segmentationTime (p. 118)": number,
"audioPids (p. 118)": [
  integer
],
"rateMode (p. 118)": enum,
"ebpAudioInterval (p. 118)": enum,
"fragmentTime (p. 118)": number,
"audioFramesPerPes (p. 118)": integer,
"maxPcrInterval (p. 118)": integer,
"scte35Pid (p. 119)": integer,
"privateMetadataPid (p. 119)": integer,
"pmtInterval (p. 119)": integer,
"segmentationStyle (p. 119)": enum,
"audioBufferModel (p. 119)": enum,
"programNumber (p. 119)": integer,
"dvbNitSettings (p. 120)": {
  "networkName (p. 63)": "string",
  "networkId (p. 63)": integer,
  "nitInterval (p. 63)": integer
},
"scte35Source (p. 120)": enum,
"pmtPid (p. 120)": integer,
"bufferModel (p. 120)": enum,
"ebpPlacement (p. 120)": enum,
"dvbSdtSettings (p. 120)": {
  "sdtInterval (p. 63)": integer,
  "serviceName (p. 63)": "string",
  "serviceProviderName (p. 63)": "string",
  "outputSdt (p. 64)": "string"
},
"nullPacketBitrate (p. 120)": number,
"pcrPid (p. 120)": integer,
"minEbpInterval (p. 121)": integer,
"transportStreamId (p. 121)": integer,
"pcrControl (p. 121)": enum,
"videoPid (p. 121)": integer,
"esRateInPes (p. 121)": enum,
"segmentationMarkers (p. 121)": enum,
"dvbTdtSettings (p. 121)": {
  "tdtInterval (p. 68)": integer
},
"patInterval (p. 122)": integer,
"dvbSubPids (p. 122)": [
  integer
]
},
"movSettings (p. 58)": {
  "reference (p. 125)": enum,
  "paddingControl (p. 125)": enum,
  "mpeg2FourCCControl (p. 126)": enum,
  "cslgAtom (p. 126)": enum,
  "clapAtom (p. 126)": enum
},
"f4vSettings (p. 59)": {
  "moovPlacement (p. 74)": enum
},
"preset (p. 138)": "string",
"outputSettings (p. 138)": {
  "hlsSettings (p. 140)": {
    "iFrameOnlyManifest (p. 102)": enum,
    "segmentModifier (p. 102)": "string",
    "audioTrackType (p. 102)": "string",
    "audioRenditionSets (p. 102)": "string",
}
"audioGroupId (p. 102)": "string"
],
"captionDescriptions (p. 138)": [
{
"captionSelectorName (p. 54)": "string",
"languageDescription (p. 54)": "string",
"languageCode (p. 54)": enum,
"destinationSettings (p. 54)": {
"burninDestinationSettings (p. 54)": {
"xPosition (p. 50)": integer,
"backgroundColor (p. 50)": enum,
"teletextSpacing (p. 50)": enum,
"yPosition (p. 50)": integer,
"backgroundOpacity (p. 51)": integer,
"fontOpacity (p. 51)": integer,
"shadowOpacity (p. 51)": integer,
"fontResolution (p. 51)": integer,
"shadowOffset (p. 51)": integer,
"outlineSize (p. 51)": integer,
"outlineColor (p. 52)": enum,
"fontSize (p. 52)": integer,
"shadowXOffset (p. 52)": integer,
"alignment (p. 52)": enum,
"shadowColor (p. 52)": enum,
"fontColor (p. 52)": enum
},
"teletextDestinationSettings (p. 54)": {
"pageNumber (p. 146)": "string"
},
"ttmlDestinationSettings (p. 54)": {
"stylePassthrough (p. 149)": enum
},
"destinationType (p. 54)": enum,
"dvbSubDestinationSettings (p. 55)": {
"xPosition (p. 64)": integer,
"backgroundColor (p. 64)": enum,
"teletextSpacing (p. 64)": enum,
"yPosition (p. 64)": integer,
"backgroundOpacity (p. 64)": integer,
"fontOpacity (p. 65)": integer,
"shadowOpacity (p. 65)": integer,
"fontResolution (p. 65)": integer,
"shadowOffset (p. 65)": integer,
"outlineSize (p. 65)": integer,
"outlineColor (p. 66)": enum,
"fontSize (p. 66)": integer,
"shadowXOffset (p. 66)": integer,
"alignment (p. 66)": enum,
"shadowColor (p. 66)": enum,
"fontColor (p. 66)": enum
},
"sccDestinationSettings (p. 55)": {
"framerate (p. 145)": enum
}
}
],
"nameModifier (p. 138)": "string"
],
"outputGroupSettings (p. 139)": {
"dashIsoGroupSettings (p. 139)": {
"fragmentLength (p. 60)": integer,
"baseUrl (p. 60)": "string",
"minBufferTime (p. 60)": integer,
"encryption (p. 60)": {  
  "spekeKeyProvider (p. 60)": {  
    "resourceId (p. 145)": "string",
    "systemIds (p. 145)": [
      "string"
    ],
    "url (p. 146)": "string"
  }
},  
"destination (p. 61)": "string",
"segmentLength (p. 61)": integer,
"segmentControl (p. 61)": enum,
"hbbtvCompliance (p. 61)": enum
},
"fileGroupSettings (p. 139)": {  
  "destination (p. 74)": "string"
},
"msSmoothGroupSettings (p. 140)": {  
  "fragmentLength (p. 134)": integer,
  "encryption (p. 135)": {  
    "spekeKeyProvider (p. 134)": {  
      "resourceId (p. 145)": "string",
      "systemIds (p. 145)": [
        "string"
      ],
      "url (p. 146)": "string"
    }
  },
  "audioDeduplication (p. 135)": enum,
  "manifestEncoding (p. 135)": enum,
  "destination (p. 135)": "string"
},
"type (p. 140)": enum,
"hlsGroupSettings (p. 140)": {  
  "segmentsPerSubdirectory (p. 98)": integer,
  "streamInfResolution (p. 98)": enum,
  "timestampDeltaMilliseconds (p. 98)": integer,
  "outputSelection (p. 98)": enum,
  "captionLanguageMappings (p. 98)": [
    {  
      "languageDescription (p. 96)": "string",
      "captionChannel (p. 96)": integer,
      "languageCode (p. 96)": enum
    }
  ],
  "clientCache (p. 98)": enum,
  "codecSpecification (p. 98)": enum,
  "destination (p. 99)": "string",
  "segmentControl (p. 99)": enum,
  "timedMetadataId3Frame (p. 99)": enum,
  "timedMetadataId3Period (p. 99)": integer,
  "captionLanguageSetting (p. 99)": enum,
  "minSegmentLength (p. 99)": integer,
  "directoryStructure (p. 99)": enum,
  "programDateTime (p. 99)": enum,
  "baseUrl (p. 100)": "string",
  "encryption (p. 100)": {  
    "initializationVectorInManifest (p. 97)": enum,
    "constantInitializationVector (p. 97)": "string",
    "staticKeyProvider (p. 97)": {  
      "keyFormatVersions (p. 146)": "string",
      "keyFormat (p. 146)": "string",
      "staticKeyValue (p. 146)": "string",
      "url (p. 146)": "string"
    }
  },
  "type (p. 97)": enum,
"spekeKeyProvider (p. 97)": {
  "resourceId (p. 145)": "string",
  "systemIds (p. 145)": [
    "string"
  ],
  "url (p. 146)": "string"
},
"encryptionMethod (p. 97)": enum
},
"adMarkers (p. 100)": [
  enum
],
"programDateTimePeriod (p. 100)": integer,
"manifestCompression (p. 100)": enum,
"segmentLength (p. 100)": integer,
"manifestDurationFormat (p. 100)": enum
},
"name (p. 139)": "string",
"customName (p. 139)": "string"
},
"timedMetadataInsertion (p. 111)": {
  "id3Insertions (p. 149)": [
    {
      "id3 (p. 103)": "string",
      "timecode (p. 103)": "string"
    }
  ],
  "availBlanking (p. 111)": {
    "availBlankingImage (p. 50)": "string"
  },
  "lastUpdated (p. 108)": "string",
  "createdAt (p. 109)": "string",
  "name (p. 109)": "string",
  "description (p. 109)": "string",
  "category (p. 109)": "string",
  "type (p. 109)": enum,
  "arn (p. 109)": "string",
  "queue (p. 109)": "string"
}
}

Example CreateJobTemplateResponse

{
  "jobTemplate (p. 60)": {
    "settings (p. 108)": {
      "timecodeConfig (p. 110)": {
        "timestampOffset (p. 148)": "string",
        "anchor (p. 148)": "string",
        "start (p. 148)": "string",
        "source (p. 148)": enum
      },
      "adAvailOffset (p. 110)": integer,
      "nielsenConfiguration (p. 110)": {
        "distributorId (p. 135)": "string",
        "breakoutCode (p. 135)": integer
      },
      "inputs (p. 110)": [
      ]
    }
  }
}
"audioSelectors (p. 104)": {
},
"audioSelectorGroups (p. 105)": {
},
"filterEnable (p. 105)": enum,
"deblockFilter (p. 105)": enum,
"videoSelector (p. 105)": {
  "colorSpace (p. 153)": enum,
  "hdr10Metadata (p. 153)": {
    "redPrimaryY (p. 93)": integer,
    "greenPrimaryY (p. 93)": integer,
    "whitePointX (p. 94)": integer,
    "maxLuminance (p. 94)": integer,
    "greenPrimaryX (p. 94)": integer,
    "whitePointY (p. 94)": integer,
    "redPrimaryX (p. 94)": integer,
    "bluePrimaryX (p. 94)": integer,
    "maxFrameAverageLightLevel (p. 95)": integer,
    "bluePrimaryY (p. 95)": integer,
    "maxContentLightLevel (p. 95)": integer,
    "minLuminance (p. 95)": integer
  },
  "programNumber (p. 153)": integer,
  "pid (p. 154)": integer,
  "colorSpaceUsage (p. 154)": enum
},
"filterStrength (p. 105)": integer,
"programNumber (p. 105)": integer,
"timecodeSource (p. 105)": enum,
"captionSelectors (p. 105)": {
},
"denoiseFilter (p. 106)": enum,
"psiControl (p. 106)": enum,
"inputClippings (p. 106)": [
  {
    "startTimecode (p. 103)": "string",
    "endTimecode (p. 103)": "string"
  }
],
"outputGroups (p. 110)": [
  {
    "outputs (p. 139)": [
      {
        "extension (p. 137)": "string",
        "videoDescription (p. 138)": {
          "fixedAfd (p. 150)": integer,
          "scalingBehavior (p. 151)": enum,
          "respondToAfd (p. 151)": enum,
          "codecSettings (p. 151)": {
            "h265Settings (p. 150)": {
              "slices (p. 87)": integer,
              "minInterval (p. 88)": integer,
              "parNumerator (p. 88)": integer,
              "flickerAdaptiveQuantization (p. 88)": enum,
              "gopSizeUnits (p. 88)": enum,
              "hrdBufferSize (p. 88)": integer,
              "qualityTuningLevel (p. 88)": enum,
              "maxBitrate (p. 88)": integer,
              "bitrate (p. 89)": integer,
              "spatialAdaptiveQuantization (p. 89)": enum,
              "sampleAdaptiveOffsetFilterMode (p. 89)": enum,
              "temporalIds (p. 89)": enum,
              "slowPal (p. 89)": enum,
              "tiles (p. 89)": enum,
            }
          }
        }
      }
    ]
  }
]
"codecProfile (p. 89)" : enum,
"alternateTransferFunctionSei (p. 89)" : enum,
"unregisteredSeiTimecode (p. 89)" : enum,
"framerateControl (p. 90)" : enum,
"telecine (p. 90)" : enum,
"framerateConversionAlgorithm (p. 90)" : enum,
"codecLevel (p. 90)" : enum,
"numberReferenceFrames (p. 90)" : integer,
"temporalAdaptiveQuantization (p. 90)" : enum,
"hrdBufferInitialFillPercentage (p. 90)" : integer,
"framerateNumerator (p. 90)" : integer,
"numberBFramesBetweenReferenceFrames (p. 91)" : integer,
"gopClosedCadence (p. 91)" : integer,
"framerateDenominator (p. 91)" : integer,
"adaptiveQuantization (p. 91)" : enum,
"interlaceMode (p. 91)" : enum,
"gopSize (p. 91)" : number,
"gopBReference (p. 91)" : enum,
"sceneChangeDetect (p. 92)" : enum,
"parDenominator (p. 92)" : integer,
"parControl (p. 92)" : enum,
"rateControlMode (p. 92)" : enum
},
"codec (p. 150)" : enum,
"proresSettings (p. 150)" : {
  "slowPal (p. 141)" : enum,
  "framerateControl (p. 141)" : enum,
  "telecine (p. 142)" : enum,
  "framerateDenominator (p. 142)" : integer,
  "framerateConversionAlgorithm (p. 142)" : enum,
  "interlaceMode (p. 142)" : enum,
  "gopSize (p. 142)" : number,
  "gopBReference (p. 142)" : enum,
  "sceneChangeDetect (p. 142)" : enum,
  "parNumerator (p. 142)" : integer,
  "parControl (p. 142)" : enum,
  "parDenominator (p. 142)" : integer,
  "framerateNumerator (p. 142)" : integer
},
"mpeg2Settings (p. 150)" : {
  "minIInterval (p. 129)" : integer,
  "parNumerator (p. 130)" : integer,
  "gopSizeUnits (p. 130)" : enum,
  "hrdBufferSize (p. 130)" : integer,
  "qualityTuningLevel (p. 130)" : enum,
  "maxBitrate (p. 130)" : integer,
  "bitrate (p. 130)" : integer,
  "spatialAdaptiveQuantization (p. 131)" : enum,
  "slowPal (p. 131)" : enum,
  "codecProfile (p. 131)" : enum,
  "intraDcPrecision (p. 131)" : enum,
  "softness (p. 131)" : integer,
  "framerateControl (p. 131)" : enum,
  "telecine (p. 131)" : enum,
  "framerateConversionAlgorithm (p. 131)" : enum,
  "codecLevel (p. 131)" : enum,
  "temporalAdaptiveQuantization (p. 132)" : enum,
  "hrdBufferInitialFillPercentage (p. 132)" : integer,
  "framerateNumerator (p. 132)" : integer,
  "numberBFramesBetweenReferenceFrames (p. 132)" : integer,
  "gopClosedCadence (p. 132)" : integer,
  "framerateDenominator (p. 132)" : integer,
  "adaptiveQuantization (p. 132)" : enum,
  "interlaceMode (p. 133)" : enum,
  "gopSize (p. 133)" : number,
  "sceneChangeDetect (p. 133)" : enum,
  "parDenominator (p. 133)" : integer,
  "parControl (p. 133)" : enum,
"syntax (p. 133)" : enum,
"rateControlMode (p. 133)" : enum
},
"h264Settings (p. 150)" : {
  "slices (p. 79)" : integer,
  "minIInterval (p. 79)" : integer,
  "parNumerator (p. 79)" : integer,
  "flickerAdaptiveQuantization (p. 79)" : enum,
  "gopSizeUnits (p. 79)" : enum,
  "hrdBufferSize (p. 79)" : integer,
  "qualityTuningLevel (p. 80)" : enum,
  "maxBitrate (p. 80)" : integer,
  "bitrate (p. 80)" : integer,
  "spatialAdaptiveQuantization (p. 80)" : enum,
  "slowPal (p. 80)" : enum,
  "codecProfile (p. 80)" : enum,
  "unregisteredSeiTimecode (p. 80)" : enum,
  "softness (p. 80)" : integer,
  "framerateControl (p. 81)" : enum,
  "telecine (p. 81)" : enum,
  "framerateConversionAlgorithm (p. 81)" : enum,
  "codecLevel (p. 81)" : enum,
  "numberReferenceFrames (p. 81)" : integer,
  "temporalAdaptiveQuantization (p. 81)" : enum,
  "repeatPps (p. 81)" : enum,
  "hrdBufferInitialFillPercentage (p. 81)" : integer,
  "framerateNumerato (p. 82)" : integer,
  "numberBFramesBetweenReferenceFrames (p. 82)" : integer,
  "gopClosedCadence (p. 82)" : integer,
  "fieldEncoding (p. 82)" : enum,
  "entropyEncoding (p. 82)" : enum,
  "framerateDenominator (p. 82)" : integer,
  "adaptiveQuantization (p. 83)" : enum,
  "interlaceMode (p. 83)" : enum,
  "gopSize (p. 83)" : number,
  "gopBReference (p. 83)" : enum,
  "sceneChangeDetect (p. 83)" : enum,
  "parDenominator (p. 83)" : integer,
  "parControl (p. 83)" : enum,
  "syntax (p. 83)" : enum,
  "rateControlMode (p. 83)" : enum
},
"frameCaptureSettings (p. 150)" : {
  "framerateDenominator (p. 75)" : integer,
  "maxCaptures (p. 75)" : integer,
  "framerateNumerato (p. 75)" : integer,
  "quality (p. 76)" : integer
}
"afdSignaling (p. 151)" : enum,
"colorMetadata (p. 151)" : enum,
"timecodeInsertion (p. 151)" : enum,
"width (p. 151)" : integer,
"sharpcness (p. 151)" : integer,
"antiAlias (p. 152)" : enum,
"videoPreprocessors (p. 152)" : {
  "timecodeBurnin (p. 152)" : {
    "prefix (p. 147)" : "string",
    "fontSize (p. 147)" : integer,
    "position (p. 147)" : enum
  },
  "noiseReduction (p. 153)" : {
    "filter (p. 136)" : enum,
    "filterSettings (p. 136)" : {
      "strength (p. 136)" : integer
    }
  }
}
"spatialFilterSettings (p. 136)": {
   "strength (p. 137)": integer,
   "postFilterSharpenStrength (p. 137)": integer,
   "speed (p. 137)": integer
 },
"colorCorrector (p. 153)": {
   "saturation (p. 56)": integer,
   "brightness (p. 57)": integer,
   "hdr10Metadata (p. 57)": {
      "redPrimaryY (p. 93)": integer,
      "greenPrimaryY (p. 93)": integer,
      "whitePointX (p. 94)": integer,
      "maxLuminance (p. 94)": integer,
      "greenPrimaryX (p. 94)": integer,
      "whitePointY (p. 94)": integer,
      "redPrimaryX (p. 94)": integer,
      "bluePrimaryX (p. 94)": integer,
      "maxFrameAverageLightLevel (p. 95)": integer,
      "bluePrimaryY (p. 95)": integer,
      "maxContentLightLevel (p. 95)": integer,
      "minLuminance (p. 95)": integer
   },
   "contrast (p. 57)": integer,
   "hue (p. 57)": integer,
   "colorSpaceConversion (p. 57)": enum
 },
"imageInserter (p. 153)": {
   "insertableImages (p. 103)": [
      {
         "duration (p. 106)": integer,
         "fadeOut (p. 106)": integer,
         "imageY (p. 107)": integer,
         "fadeIn (p. 107)": integer,
         "imageX (p. 107)": integer,
         "width (p. 107)": integer,
         "startTime (p. 107)": "string",
         "opacity (p. 108)": integer,
         "layer (p. 108)": integer,
         "height (p. 108)": integer,
         "imageInserterInput (p. 108)": "string"
      }
   ]
 },
"deinterlacer (p. 153)": {
   "mode (p. 62)": enum,
   "control (p. 62)": enum,
   "algorithm (p. 62)": enum
 }
"position (p. 152)": {
   "width (p. 143)": integer,
   "x (p. 143)": integer,
   "y (p. 143)": integer,
   "height (p. 144)": integer
 },
"dropFrameTimecode (p. 152)": enum,
"crop (p. 152)": {
   "width (p. 143)": integer,
   "x (p. 143)": integer,
   "y (p. 143)": integer,
   "height (p. 144)": integer
 },
"height (p. 152)": integer
"audioDescriptions (p. 138)": [
{
  "languageCodeControl (p. 45)": enum,
  "audioTypeControl (p. 45)": enum,
  "remixSettings (p. 45)": {
    "channelsOut (p. 144)": integer,
    "channelMapping (p. 144)": {
      "outputChannels (p. 56)": [  
        "inputChannels (p. 139)": [  
          integer
        ]
      ]
    },
    "channelsIn (p. 144)": integer
  },
  "audioType (p. 45)": integer,
  "audioSourceName (p. 45)": "string",
  "codecSettings (p. 46)": {
    "codec (p. 44)": enum,
    "wavSettings (p. 44)": {
      "channels (p. 154)": integer,
      "bitDepth (p. 154)": integer,
      "sampleRate (p. 154)": integer
    },
    "ac3Settings (p. 44)": {
      "dynamicRangeCompressionProfile (p. 41)": enum,
      "dialnorm (p. 41)": integer,
      "codingMode (p. 42)": enum,
      "metadataControl (p. 42)": enum,
      "lfeFilter (p. 42)": enum,
      "bitrate (p. 42)": integer,
      "bitstreamMode (p. 42)": enum,
      "sampleRate (p. 42)": integer
    },
    "aacSettings (p. 44)": {
      "vbrQuality (p. 39)": enum,
      "codecProfile (p. 39)": enum,
      "codingMode (p. 39)": enum,
      "specification (p. 39)": enum,
      "bitrate (p. 40)": integer,
      "rawFormat (p. 40)": enum,
      "rateControlMode (p. 40)": enum,
      "sampleRate (p. 40)": integer,
      "audioDescriptionBroadcasterMix (p. 40)": enum
    },
    "aiffSettings (p. 44)": {
      "channels (p. 43)": integer,
      "bitDepth (p. 43)": integer,
      "sampleRate (p. 43)": integer
    },
    "eac3Settings (p. 44)": {
      "dialnorm (p. 70)": integer,
      "passthroughControl (p. 70)": enum,
      "metadataControl (p. 70)": enum,
      "bitrate (p. 70)": integer,
      "dynamicRangeCompressionRf (p. 70)": enum,
      "sampleRate (p. 70)": integer,
      "l1r5SurroundMixLevel (p. 71)": number,
      "surroundExMode (p. 71)": enum,
      "dynamicRangeCompressionLine (p. 71)": enum,
      "lfeControl (p. 71)": enum,
      "codingMode (p. 71)": enum,
      "surroundMode (p. 71)": enum,
      "attenuationControl (p. 71)": enum,
      "lfeFilter (p. 71)": enum
    }
  }
}
"phaseControl (p. 71)": enum,
"ltRtCenterMixLevel (p. 72)": number,
"dcFilter (p. 72)": enum,
"stereoDownmix (p. 72)": enum,
"bitstreamMode (p. 72)": enum,
"loRoSurroundMixLevel (p. 72)": number,
"loRoCenterMixLevel (p. 72)": number,
"mp2Settings (p. 44)": {
  "channels (p. 126)": integer,
  "bitrate (p. 126)": integer,
  "sampleRate (p. 126)": integer
},
"languageCode (p. 46)": enum,
"StreamName (p. 46)": "string",
"audioNormalizationSettings (p. 46)": {
  "targetLkfs (p. 47)": number,
  "algorithmControl (p. 47)": enum,
  "peakCalculation (p. 47)": enum,
  "loudnessLogging (p. 47)": enum,
  "correctionGateLevel (p. 47)": integer,
  "algorithm (p. 48)": enum
},
"containerSettings (p. 138)": {
  "container (p. 58)": enum,
  "mp4Settings (p. 58)": {
    "mp4MajorBrand (p. 127)": "string",
    "moovPlacement (p. 127)": enum,
    "cslgAtom (p. 127)": enum,
    "freeSpaceBox (p. 127)": enum
  },
  "m3u8Settings (p. 58)": {
    "pmtPid (p. 122)": integer,
    "pcrPid (p. 122)": integer,
    "audioPids (p. 123)": [
      integer
    ],
    "audioFramesPerPes (p. 123)": integer,
    "scte35Pid (p. 123)": integer,
    "transportStreamId (p. 123)": integer,
    "videoPid (p. 123)": integer,
    "pcrControl (p. 123)": enum,
    "privateMetadataPid (p. 124)": integer,
    "pmtInterval (p. 124)": integer,
    "patInterval (p. 124)": integer,
    "programNumber (p. 124)": integer,
    "timedMetadataPid (p. 124)": integer,
    "timedMetadata (p. 124)": enum,
    "scte35Source (p. 124)": enum
  },
  "m2tsSettings (p. 58)": {
    "dvbTeletextPid (p. 117)": integer,
    "bitrate (p. 117)": integer,
    "segmentationTime (p. 118)": number,
    "audioPids (p. 118)": [
      integer
    ],
    "rateMode (p. 118)": enum,
    "ebpAudioInterval (p. 118)": enum,
    "fragmentTime (p. 118)": number,
    "audioFramesPerPes (p. 118)": integer,
    "maxPcrInterval (p. 118)": integer,
    "scte35Pid (p. 119)": integer,
"privateMetadataPid (p. 119)": integer,
"pmtInterval (p. 119)": integer,
"segmentationStyle (p. 119)": enum,
"audioBufferModel (p. 119)": enum,
"programNumber (p. 119)": integer,
"dvbNitSettings (p. 120)": {
  "networkName (p. 63)": "string",
  "networkId (p. 63)": integer,
  "nitInterval (p. 63)": integer
},
"scte35Source (p. 120)": enum,
"pmtPid (p. 120)": integer,
"bufferModel (p. 120)": enum,
"ebpPlacement (p. 120)": enum,
"dvbSdtSettings (p. 120)": {
  "sdtInterval (p. 63)": integer,
  "serviceName (p. 63)": "string",
  "serviceProviderName (p. 63)": "string",
  "outputSdt (p. 64)": enum
},
"nullPacketBitrate (p. 120)": number,
"pcrPid (p. 120)": integer,
"minEbpInterval (p. 121)": integer,
"transportStreamId (p. 121)": integer,
"pcrControl (p. 121)": enum,
"videoPid (p. 121)": integer,
"esRateInPes (p. 121)": enum,
"segmentationMarkers (p. 121)": enum,
"dvbTdtSettings (p. 121)": {
  "tdtInterval (p. 68)": integer
},
"patInterval (p. 122)": integer,
"dvbSubPids (p. 122)": [integer
],
"mobSettings (p. 58)": {
  "reference (p. 125)": enum,
  "paddingControl (p. 125)": enum,
  "mpeg2FourCCControl (p. 126)": enum,
  "celgAtom (p. 126)": enum,
  "clapAtom (p. 126)": enum
},
"f4vSettings (p. 59)": {
  "moovPlacement (p. 74)": enum
},
"preset (p. 138)": "string",
"outputSettings (p. 138)": {
  "hisSettings (p. 140)": {
    "iFrameOnlyManifest (p. 102)": enum,
    "segmentModifier (p. 102)": "string",
    "audioTrackType (p. 102)": enum,
    "audioRenditionSets (p. 102)": "string",
    "audioGroupId (p. 102)": "string"
  }
},
"captionDescriptions (p. 138)": [
  {
    "captionSelectorName (p. 54)": "string",
    "languageDescription (p. 54)": "string",
    "languageCode (p. 54)": enum,
    "destinationSettings (p. 54)": {
      "burninDestinationSettings (p. 54)": {
        "xPosition (p. 50)": integer,
        "backgroundColor (p. 50)": enum,
        "textColor (p. 50)": enum
      }
    }
  }
]}
"teletextSpacing (p. 50)": enum,
"yPosition (p. 50)": integer,
"backgroundColor (p. 51)": enum,
"fontOpacity (p. 51)": integer,
"shadowOpacity (p. 51)": integer,
"fontResolution (p. 51)": integer,
"shadowXOffset (p. 51)": integer,
"outlineSize (p. 51)": integer,
"outlineColor (p. 52)": enum,
"fontSize (p. 52)": integer,
"shadowYOffset (p. 52)": integer,
"alignment (p. 52)": enum,
"shadowColor (p. 52)": enum,
"fontColor (p. 52)": enum
},
"teletextDestinationSettings (p. 54)": {
"pageNumber (p. 146)": "string"
},
"ttmlDestinationSettings (p. 54)": {
"stylePassthrough (p. 149)": enum
},
"destinationType (p. 54)": enum,
"dvbSubDestinationSettings (p. 55)": {
"xPosition (p. 64)": integer,
"backgroundColor (p. 64)": enum,
"teletextSpacing (p. 64)": enum,
"yPosition (p. 64)": integer,
"backgroundColor (p. 64)": integer,
"fontOpacity (p. 64)": integer,
"shadowOpacity (p. 64)": integer,
"fontResolution (p. 64)": integer,
"shadowXOffset (p. 64)": integer,
"alignment (p. 64)": enum,
"shadowColor (p. 64)": enum,
"fontColor (p. 64)": enum
},
"sccDestinationSettings (p. 55)": {
"framerate (p. 145)": enum
}
},
"nameModifier (p. 138)": "string"
],
"outputGroupSettings (p. 139)": {
"dashIsoGroupSettings (p. 139)": {
"fragmentLength (p. 60)": integer,
"baseUrl (p. 60)": "string",
"minBufferTime (p. 60)": integer,
"encryption (p. 60)": {
"spekeKeyProvider (p. 60)": {
"resourceId (p. 145)": "string",
"systemIds (p. 145)": [
"string"
],
"url (p. 146)": "string"
}
},
"destination (p. 61)": "string",
"segmentLength (p. 61)": integer,
"segmentControl (p. 61)": enum,
"hbbtvCompliance (p. 61)" : enum

"fileGroupSettings (p. 139)" : {
  "destination (p. 74)" : "string"
},

"msSmoothGroupSettings (p. 140)" : {
  "fragmentLength (p. 134)" : integer,
  "encryption (p. 135)" : {
    "spekeKeyProvider (p. 134)" : {
      "resourceId (p. 145)" : "string",
      "systemIds (p. 145)" : [
        "string"
      ],
      "url (p. 146)" : "string"
    }
  },
  "audioDeduplication (p. 135)" : enum,
  "manifestEncoding (p. 135)" : enum,
  "destination (p. 135)" : "string"
},

"type (p. 140)" : enum,

"hlsGroupSettings (p. 140)" : {
  "segmentsPerSubdirectory (p. 98)" : integer,
  "streamInfResolution (p. 98)" : enum,
  "timestampDeltaMilliseconds (p. 98)" : integer,
  "outputSelection (p. 98)" : enum,
  "captionLanguageMappings (p. 98)" : [
    {
      "languageDescription (p. 96)" : "string",
      "captionChannel (p. 96)" : integer,
      "languageCode (p. 96)" : enum
    }
  ],
  "clientCache (p. 98)" : enum,
  "codecSpecification (p. 98)" : enum,
  "destination (p. 99)" : "string",
  "segmentControl (p. 99)" : enum,
  "timedMetadataId3Frame (p. 99)" : enum,
  "timedMetadataId3Period (p. 99)" : integer,
  "captionLanguageSetting (p. 99)" : enum,
  "minSegmentLength (p. 99)" : integer,
  "directoryStructure (p. 99)" : enum,
  "programDateTime (p. 99)" : integer,
  "baseUrl (p. 100)" : "string",
  "encryption (p. 100)" : {
    "initializationVectorInManifest (p. 97)" : enum,
    "constantInitializationVector (p. 97)" : "string",
    "spekeKeyProvider (p. 97)" : {
      "keyFormatVersions (p. 146)" : "string",
      "keyFormat (p. 146)" : "string",
      "staticKeyValue (p. 146)" : "string",
      "url (p. 146)" : "string"
    },
    "type (p. 97)" : enum,
    "spekeKeyProvider (p. 97)" : {
      "resourceId (p. 145)" : "string",
      "systemIds (p. 145)" : [
        "string"
      ],
      "url (p. 146)" : "string"
    },
    "encryptionMethod (p. 97)" : enum
  },
  "adMarkers (p. 100)" : enum
}

37


**Example ExceptionBody**

```json
{
  "message (p. 74)": "string"
}
```

---

**Properties**

**AacAudioDescriptionBroadcasterMix (Enum)**

Choose BROADCASTER_MIXED_AD when the input contains pre-mixed main audio + audio description (AD) as a stereo pair. The value for AudioType will be set to 3, which signals to downstream systems that this stream contains "broadcaster mixed AD". Note that the input received by the encoder must contain pre-mixed audio; the encoder does not perform the mixing. When you choose BROADCASTER_MIXED_AD, the encoder ignores any values you provide in AudioType and FollowInputAudioType. Choose NORMAL when the input does not contain pre-mixed audio + audio description (AD). In this case, the encoder will use any values you provide for AudioType and FollowInputAudioType.

- BROADCASTER_MIXED_AD
- NORMAL

**AacCodecProfile (Enum)**

AAC Profile.

- LC
HEV1
HEV2

**AacCodingMode (Enum)**

Mono (Audio Description), Mono, Stereo, or 5.1 channel layout. Valid values depend on rate control mode and profile. "1.0 - Audio Description (Receiver Mix)" setting receives a stereo description plus control track and emits a mono AAC encode of the description track, with control data emitted in the PES header as per ETSI TS 101 154 Annex E.

- AD_RECEIVER_MIX
- CODING_MODE_1_0
- CODING_MODE_1_1
- CODING_MODE_2_0
- CODING_MODE_5_1

**AacRateControlMode (Enum)**

Rate Control Mode.

- CBR
- VBR

**AacRawFormat (Enum)**

Enables LATM/LOAS AAC output. Note that if you use LATM/LOAS AAC in an output, you must choose "No container" for the output container.

- LATM_LOAS
- NONE

**AacSettings**

**vbrQuality**

Type: AacVbrQuality (p. 40)
Required: False

codecProfile

Type: AacCodecProfile (p. 38)
Required: False

codingMode

Type: AacCodingMode (p. 39)
Required: True

**specification**

Type: AacSpecification (p. 40)
Required: False
bitrate

Average bitrate in bits/second. Valid values depend on rate control mode and profile.

Type: integer
Required: False
Minimum: 6000
Maximum: 1024000

rawFormat

Type: AacRawFormat (p. 39)
Required: False

rateControlMode

Type: AacRateControlMode (p. 39)
Required: False

sampleRate

Sample rate in Hz. Valid values depend on rate control mode and profile.

Type: integer
Required: True
Minimum: 8000
Maximum: 96000

audioDescriptionBroadcasterMix

Type: AacAudioDescriptionBroadcasterMix (p. 38)
Required: False

AacSpecification (Enum)

Use MPEG-2 AAC instead of MPEG-4 AAC audio for raw or MPEG-2 Transport Stream containers.

MPEG2
MPEG4

AacVbrQuality (Enum)

VBR quality level. Only used if the rate control mode (AacRateControlMode) is VBR.

LOW
MEDIUM_LOW
MEDIUM_HIGH
HIGH

Ac3BitstreamMode (Enum)

Specifies the "Bitstream Mode" (bsmod) for the emitted AC-3 stream. See ATSC A/52-2012 for background on these values.
**Ac3CodingMode (Enum)**

Dolby Digital coding mode. Determines number of channels.

- CODING_MODE_1_0
- CODING_MODE_1_1
- CODING_MODE_2_0
- CODING_MODE_3_2_LFE

**Ac3DynamicRangeCompressionProfile (Enum)**

If set to FILM_STANDARD, adds dynamic range compression signaling to the output bitstream as defined in the Dolby Digital specification.

- FILM_STANDARD
- NONE

**Ac3LfeFilter (Enum)**

Applies a 120Hz lowpass filter to the LFE channel prior to encoding. Only valid with 3_2_LFE coding mode.

- ENABLED
- DISABLED

**Ac3MetadataControl (Enum)**

When set to FOLLOW_INPUT, encoder metadata will be sourced from the DD, DD+, or DolbyE decoder that supplied this audio data. If audio was not supplied from one of these streams, then the static metadata settings will be used.

- FOLLOW_INPUT
- USE_CONFIGURED

**Ac3Settings**

**dynamicRangeCompressionProfile**

*Type:* Ac3DynamicRangeCompressionProfile (p. 41)

*Required:* False

**dialnorm**

Sets the dialnorm for the output. If blank and input audio is Dolby Digital, dialnorm will be passed through.
Type: integer  
Required: False  
Minimum: 1  
Maximum: 31

codingMode
Type: Ac3CodingMode (p. 41)  
Required: False

metadataControl
Type: Ac3MetadataControl (p. 41)  
Required: False

lfeFilter
Type: Ac3LfeFilter (p. 41)  
Required: False

bitrate
Average bitrate in bits/second. Valid bitrates depend on the coding mode.
Type: integer  
Required: False  
Minimum: 64000  
Maximum: 640000

bitstreamMode
Type: Ac3BitstreamMode (p. 40)  
Required: False

sampleRate
Sample rate in hz. Sample rate is always 48000.
Type: integer  
Required: False  
Minimum: 48000  
Maximum: 48000

AfdSignaling (Enum)
This setting only applies to H.264 and MPEG2 outputs. Use Insert AFD signaling (AfdSignaling) to whether there are AFD values in the output video data and what those values are.

* Choose None to remove all AFD values from this output.

* Choose Fixed to ignore input AFD values and instead encode the value specified in the job.
* Choose Auto to calculate output AFD values based on the input AFD scaler data.

NONE
AUTO
FIXED

**AiffSettings**

**channels**

Set Channels to specify the number of channels in this output audio track. Choosing Mono in the console will give you 1 output channel; choosing Stereo will give you 2. In the API, valid values are 1 and 2.

  - **Type:** integer
  - **Required:** False
  - **Minimum:** 1
  - **Maximum:** 2

**bitDepth**

Specify Bit depth (BitDepth), in bits per sample, to choose the encoding quality for this audio track.

  - **Type:** integer
  - **Required:** False
  - **Minimum:** 16
  - **Maximum:** 24

**sampleRate**

Sample rate in hz.

  - **Type:** integer
  - **Required:** False
  - **Minimum:** 8000
  - **Maximum:** 192000

**AncillarySourceSettings**

**sourceAncillaryChannelNumber**

Specifies the 608 channel number in the ancillary data track from which to extract captions. Unused for passthrough.

  - **Type:** integer
  - **Required:** False
  - **Minimum:** 1
  - **Maximum:** 4

**AntiAlias (Enum)**

Enable Anti-alias (AntiAlias) to enhance sharp edges in video output when your input resolution is much larger than your output resolution. Default is enabled.
**Properties**

### AudioCodec (Enum)

Type of Audio codec.

- AAC
- MP2
- WAV
- AIFF
- AC3
- EAC3
- PASSTHROUGH

### AudioCodecSettings

**codec**

- **Type**: AudioCodec (p. 44)
- **Required**: True

**wavSettings**

- **Type**: WavSettings (p. 154)
- **Required**: False

**ac3Settings**

- **Type**: Ac3Settings (p. 41)
- **Required**: False

**aacSettings**

- **Type**: AacSettings (p. 39)
- **Required**: False

**aiffSettings**

- **Type**: AiffSettings (p. 43)
- **Required**: False

**eac3Settings**

- **Type**: Eac3Settings (p. 70)
- **Required**: False

**mp2Settings**

- **Type**: Mp2Settings (p. 126)
**Properties**

**Required**: False

**AudioDefaultSelection (Enum)**

When an "Audio Description" specifies an AudioSelector or AudioSelectorGroup for which no matching source is found in the input, then the audio selector marked as DEFAULT will be used. If none are marked as default, silence will be inserted for the duration of the input.

- DEFAULT
- NOT_DEFAULT

**AudioDescription**

**languageCodeControl**

- **Type**: AudioLanguageCodeControl (p. 46)
- **Required**: False

**audioTypeControl**

- **Type**: AudioTypeControl (p. 49)
- **Required**: False

**remixSettings**

Advanced audio remixing settings.

- **Type**: RemixSettings (p. 144)
- **Required**: False

**audioType**

Applies only if Follow Input Audio Type is unchecked (false). A number between 0 and 255. The following are defined in ISO-IEC 13818-1: 0 = Undefined, 1 = Clean Effects, 2 = Hearing Impaired, 3 = Visually Impaired Commentary, 4-255 = Reserved.

- **Type**: integer
- **Required**: False
  - **Minimum**: 0
  - **Maximum**: 255

**audioSourceName**

Specifies which audio data to use from each input. In the simplest case, specify an "Audio Selector" by name based on its order within each input. For example if you specify "Audio Selector 3", then the third audio selector will be used from each input. If an input does not have an "Audio Selector 3", then the audio selector marked as "default" in that input will be used. If there is no audio selector marked as "default", silence will be inserted for the duration of that input. Alternatively, an "Audio Selector Group" name may be specified, with similar default/silence behavior. If no audio_source_name is specified, then "Audio Selector 1" will be chosen automatically.

- **Type**: string
- **Required**: False
codecSettings

Type: AudioCodecSettings (p. 44)
Required: True

languageCode

Indicates the language of the audio output track. The ISO 639 language specified in the 'Language Code' drop down will be used when 'Follow Input Language Code' is not selected or when 'Follow Input Language Code' is selected but there is no ISO 639 language code specified by the input.

Type: LanguageCode (p. 111)
Required: False

streamName

Used for Microsoft Smooth Streaming and Apple HLS outputs. Indicates the name displayed by the player (eg. English, or Director Commentary). Alphanumeric characters, spaces, and underscore are legal.

Type: string
Required: False
Pattern: ^\w\s*$

audioNormalizationSettings

Advanced audio normalization settings.

Type: AudioNormalizationSettings (p. 47)
Required: False

AudioLanguageCodeControl (Enum)

Choosing FOLLOW_INPUT will cause the ISO 639 language code of the output to follow the ISO 639 language code of the input. The language specified for languageCode' will be used when USE_CONFIGURED is selected or when FOLLOW_INPUT is selected but there is no ISO 639 language code specified by the input.

FOLLOW_INPUT
USE_CONFIGURED

AudioNormalizationAlgorithm (Enum)

Audio normalization algorithm to use. 1770-1 conforms to the CALM Act specification, 1770-2 conforms to the EBU R-128 specification.

ITU_BS_1770_1
ITU_BS_1770_2

AudioNormalizationAlgorithmControl (Enum)

When enabled the output audio is corrected using the chosen algorithm. If disabled, the audio will be measured but not adjusted.

CORRECT_AUDIO
MEASURE_ONLY

**AudioNormalizationLoudnessLogging (Enum)**

If set to LOG, log each output's audio track loudness to a CSV file.

- LOG
- DONT_LOG

**AudioNormalizationPeakCalculation (Enum)**

If set to TRUE_PEAK, calculate and log the TruePeak for each output's audio track loudness.

- TRUE_PEAK
- NONE

**AudioNormalizationSettings**

**targetLkfs**

Target LKFS(loudness) to adjust volume to. If no value is entered, a default value will be used according to the chosen algorithm. The CALM Act (1770-1) recommends a target of -24 LKFS. The EBU R-128 specification (1770-2) recommends a target of -23 LKFS.

- **Type**: number
- **Required**: False
- **Format**: float
- **Minimum**: -59.0
- **Maximum**: 0.0

**algorithmControl**

- **Type**: AudioNormalizationAlgorithmControl (p. 46)
- **Required**: False

**peakCalculation**

- **Type**: AudioNormalizationPeakCalculation (p. 47)
- **Required**: False

**loudnessLogging**

- **Type**: AudioNormalizationLoudnessLogging (p. 47)
- **Required**: False

**correctionGateLevel**

Content measuring above this level will be corrected to the target level. Content measuring below this level will not be corrected. Gating only applies when not using real_time_correction.

- **Type**: integer
- **Required**: False
- **Minimum**: -70
Maximum: 0

algorithm
Type: AudioNormalizationAlgorithm (p. 46)
Required: False

AudioSelector

remixSettings
Advanced audio remixing settings.

Type: RemixSettings (p. 144)
Required: False

programSelection
Applies only when input streams contain Dolby E. Enter the program ID (according to the metadata in the audio) of the Dolby E program to extract from the specified track. One program extracted per audio selector. To select multiple programs, create multiple selectors with the same Track and different Program numbers. "All channels" means to ignore the program IDs and include all the channels in this selector; useful if metadata is known to be incorrect.

Type: integer
Required: False
Minimum: 0
Maximum: 8

offset
Specifies a time delta in milliseconds to offset the audio from the input video.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

defaultSelection
Type: AudioDefaultSelection (p. 45)
Required: False

selectorType
Type: AudioSelectorType (p. 49)
Required: False

languageCode
Selects a specific language code from within an audio source.

Type: LanguageCode (p. 111)
Required: False
**pids**

Selects a specific PID from within an audio source (e.g. 257 selects PID 0x101).

**Type:** Array of type integer  
**Required:** False

**externalAudioFileInput**

Specifies audio data from an external file source. Auto populated when Infer External Filename is checked.

**Type:** string  
**Required:** False

**tracks**

Identify the channel to include in this selector by entering the 1-based track index. To combine several tracks, enter a comma-separated list, e.g. "1,2,3" for tracks 1-3.

**Type:** Array of type integer  
**Required:** False
in the output. Otherwise the value in Audio Type is included in the output. Note that this field and audioType are both ignored if audioDescriptionBroadcasterMix is set to BROADCASTER_MIXED_AD.

FOLLOW_INPUT
USE_CONFIGURED

**AvailBlanking**

**availBlankingImage**

Blanking image to be used. Leave empty for solid black. Only bmp and png images are supported.

*Type:* string  
*Required:* False  
*Pattern:* `^(s3://)(.*)(bmp|BMP|png|PNG)$`

**BurninDestinationSettings**

**xPosition**

Specifies the horizontal position of the caption relative to the left side of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the left of the output. If no explicit x_position is provided, the horizontal caption position will be determined by the alignment parameter. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

*Type:* integer  
*Required:* False  
*Minimum:* 0  
*Maximum:* 2147483647

**backgroundColor**

*Type:* BurninSubtitleBackgroundColor (p. 53)  
*Required:* False

**teletextSpacing**

*Type:* BurninSubtitleTeletextSpacing (p. 53)  
*Required:* False

**yPosition**

Specifies the vertical position of the caption relative to the top of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the top of the output. If no explicit y_position is provided, the caption will be positioned towards the bottom of the output. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

*Type:* integer  
*Required:* False  
*Minimum:* 0  
*Maximum:* 2147483647
**backgroundOpacity**

Specifies the opacity of the background rectangle. 255 is opaque; 0 is transparent. Leaving this parameter blank is equivalent to setting it to 0 (transparent). All burn-in and DVB-Sub font settings must match.

**Properties**

Type: integer  
Required: False  
Minimum: 0  
Maximum: 255

**fontOpacity**

Specifies the opacity of the burned-in captions. 255 is opaque; 0 is transparent. All burn-in and DVB-Sub font settings must match.

**Properties**

Type: integer  
Required: True  
Minimum: 0  
Maximum: 255

**shadowOpacity**

Specifies the opacity of the shadow. 255 is opaque; 0 is transparent. Leaving this parameter blank is equivalent to setting it to 0 (transparent). All burn-in and DVB-Sub font settings must match.

**Properties**

Type: integer  
Required: False  
Minimum: 0  
Maximum: 255

**fontResolution**

Font resolution in DPI (dots per inch); default is 96 dpi. All burn-in and DVB-Sub font settings must match.

**Properties**

Type: integer  
Required: False  
Minimum: 96  
Maximum: 600

**shadowYOffset**

Specifies the vertical offset of the shadow relative to the captions in pixels. A value of -2 would result in a shadow offset 2 pixels above the text. All burn-in and DVB-Sub font settings must match.

**Properties**

Type: integer  
Required: False  
Minimum: -2147483648  
Maximum: 2147483647

**outlineSize**

Specifies font outline size in pixels. This option is not valid for source captions that are either 608/ embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.
Properties

Type: integer  
Required: True  
Minimum: 0  
Maximum: 10

**outlineColor**

Type: `BurninSubtitleOutlineColor (p. 53)`  
Required: True

**fontSize**

A positive integer indicates the exact font size in points. Set to 0 for automatic font size selection. All burn-in and DVB-Sub font settings must match.

Type: integer  
Required: False  
Minimum: 0  
Maximum: 96

**shadowXOffset**

Specifies the horizontal offset of the shadow relative to the captions in pixels. A value of -2 would result in a shadow offset 2 pixels to the left. All burn-in and DVB-Sub font settings must match.

Type: integer  
Required: False  
Minimum: -2147483648  
Maximum: 2147483647

**alignment**

Type: `BurninSubtitleAlignment (p. 52)`  
Required: True

**shadowColor**

Type: `BurninSubtitleShadowColor (p. 53)`  
Required: False

**fontColor**

Type: `BurninSubtitleFontColor (p. 53)`  
Required: False

**BurninSubtitleAlignment (Enum)**

If no explicit x_position or y_position is provided, setting alignment to centered will place the captions at the bottom center of the output. Similarly, setting a left alignment will align captions to the bottom left of the output. If x and y positions are given in conjunction with the alignment parameter, the font will be justified (either left or centered) relative to those coordinates. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.
BurninSubtitleBackgroundColor (Enum)

Specifies the color of the rectangle behind the captions. All burn-in and DVB-Sub font settings must match.

- NONE
- BLACK
- WHITE

BurninSubtitleFontColor (Enum)

Specifies the color of the burned-in captions. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

- WHITE
- BLACK
- YELLOW
- RED
- GREEN
- BLUE

BurninSubtitleOutlineColor (Enum)

Specifies font outline color. This option is not valid for source captions that are either 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

- BLACK
- WHITE
- YELLOW
- RED
- GREEN
- BLUE

BurninSubtitleShadowColor (Enum)

Specifies the color of the shadow cast by the captions. All burn-in and DVB-Sub font settings must match.

- NONE
- BLACK
- WHITE

BurninSubtitleTeletextSpacing (Enum)

Controls whether a fixed grid size or proportional font spacing will be used to generate the output subtitles bitmap. Only applicable for Teletext inputs and DVB-Sub/Burn-in outputs.

- FIXED_GRID
- PROPORTIONAL
CaptionDescription

captionSelectorName

Specifies which "Caption Selector" to use from each input when generating captions. The name should be of the format "Caption Selector <N>", which denotes that the Nth Caption Selector will be used from each input.

Type: string
Required: True

languageDescription

Human readable information to indicate captions available for players (eg, English, or Spanish). Alphanumeric characters, spaces, and underscore are legal.

Type: string
Required: False
Pattern: ^[\w ]*$

languageCode

Indicates the language of the caption output track.

Type: LanguageCode (p. 111)
Required: False

destinationSettings

Type: CaptionDestinationSettings (p. 54)
Required: False

CaptionDestinationSettings

burninDestinationSettings

Type: BurninDestinationSettings (p. 50)
Required: False

teletextDestinationSettings

Type: TeletextDestinationSettings (p. 146)
Required: False

ttmlDestinationSettings

Type: TtmlDestinationSettings (p. 149)
Required: False

destinationType

Type: CaptionDestinationType (p. 55)
Required: True

dvbSubDestinationSettings

Type: DvbSubDestinationSettings (p. 64)
Required: False

sccDestinationSettings

Type: SccDestinationSettings (p. 145)
Required: False

CaptionDestinationType (Enum)

BURN_IN
DVB_SUB
EMBEDDED
SCC
SRT
TELETEXT
TTML
WEBVTT

CaptionSelector

sourceSettings

Type: CaptionSourceSettings (p. 55)
Required: True

languageCode

The specific language to extract from source. If input is SCTE-27, complete this field and/or PID to select the caption language to extract. If input is DVB-Sub and output is Burn-in or SMPTE-TT, complete this field and/or PID to select the caption language to extract. If input is DVB-Sub that is being passed through, omit this field (and PID field); there is no way to extract a specific language with pass-through captions.

Type: LanguageCode (p. 111)
Required: False

CaptionSourceSettings

fileSourceSettings

Type: FileSourceSettings (p. 74)
Required: False

ancillarySourceSettings

Type: AncillarySourceSettings (p. 43)
Required: False
embeddedSourceSettings
   Type: EmbeddedSourceSettings (p. 73)
   Required: False

sourceType
   Type: CaptionSourceType (p. 56)
   Required: True

dvbSubSourceSettings
   Type: DvbSubSourceSettings (p. 66)
   Required: False

teletextSourceSettings
   Type: TeletextSourceSettings (p. 147)
   Required: False

CaptionSourceType (Enum)

Use Source (SourceType) to identify the format of your input captions. The service cannot auto-detect caption format.

ANCILLARY
DVB_SUB
EMBEDDED
SCC
TTML
STL
SRT
TELETEXT
NULL_SOURCE

ChannelMapping

outputChannels
   Type: Array of type OutputChannelMapping (p. 139)
   Required: True

ColorCorrector

saturation

Saturation level.

Type: integer
Required: False
Minimum: 1
Maximum: 100
brightness

Brightness level.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 100

hdr10Metadata

- **Type**: Hdr10Metadata (p. 93)
- **Required**: False

contrast

Contrast level.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 100

hue

Hue in degrees.

- **Type**: integer
- **Required**: False
- **Minimum**: -180
- **Maximum**: 180

colorSpaceConversion

- **Type**: ColorSpaceConversion (p. 58)
- **Required**: False

ColorMetadata (Enum)

Enable insert color metadata (ColorMetadata) to include color metadata in this output. This setting is enabled by default.

- IGNORE
- INSERT

ColorSpace (Enum)

Specifies the colorspace of an input. This setting works in tandem with "Color Corrector":"\color_corrector > color_space_conversion to determine if any conversion will be performed.

- FOLLOW
- REC_601
- REC_709
- HDR10
HLG_2020

**ColorSpaceConversion (Enum)**

Determines if colorspace conversion will be performed. If set to _None_, no conversion will be performed. If _Force 601_ or _Force 709_ are selected, conversion will be performed for inputs with differing colorspaces. An input's colorspace can be specified explicitly in the "Video Selector".#inputs-video_selector if necessary.

NONE
FORCE_601
FORCE_709
FORCE_HDR10
FORCE_HLG_2020

**ColorSpaceUsage (Enum)**

There are two sources for color metadata, the input file and the job configuration. This enum controls which takes precedence. FORCE: System will use color metadata supplied by user, if any. If the user does not supply color metadata the system will use data from the source. FALLBACK: System will use color metadata from the source. If source has no color metadata, the system will use user-supplied color metadata values if available.

FORCE
FALLBACK

**ContainerSettings**

container

_Type: ContainerType (p. 59)  
Required: True

**mp4Settings**

_Type: Mp4Settings (p. 127)  
Required: False

**m3u8Settings**

_Type: M3u8Settings (p. 122)  
Required: False

**m2tsSettings**

_Type: M2tsSettings (p. 117)  
Required: False

**movSettings**

_Type: MovSettings (p. 125)  
Required: False
f4vSettings

Type: F4vSettings (p. 74)
Required: False

ContainerType (Enum)

Container for this output. Some containers require a container settings object. If not specified, the default object will be created.

F4V
ISMV
M2TS
M3U8
MOV
MP4
MPD
MXF
RAW

CreateJobTemplateRequest

settings

Type: JobTemplateSettings (p. 110)
Required: True

name

The name of the job template you are creating.

Type: string
Required: True

description

Optional. A description of the job template you are creating.

Type: string
Required: False

category

Optional. A category for the job template you are creating.

Type: string
Required: False

queue

Optional. The queue that jobs created from this template are assigned to. If you don't specify this, jobs will go to the default queue.

Type: string
Required: False

CreateJobTemplateResponse

jobTemplate

  Type: JobTemplate (p. 108)
  Required: False

DashIsoEncryptionSettings

spekeKeyProvider

  Type: SpekeKeyProvider (p. 145)
  Required: True

DashIsoGroupSettings

fragmentLength

Length of fragments to generate (in seconds). Fragment length must be compatible with GOP size and Framerate. Note that fragments will end on the next keyframe after this number of seconds, so actual fragment length may be longer. When Emit Single File is checked, the fragmentation is internal to a single output file and it does not cause the creation of many output files as in other output types.

  Type: integer
  Required: True
  Minimum: 1
  Maximum: 2147483647

baseUrl

A partial URI prefix that will be put in the manifest (.mpd) file at the top level BaseURL element. Can be used if streams are delivered from a different URL than the manifest file.

  Type: string
  Required: False

minBufferTime

Minimum time of initially buffered media that is needed to ensure smooth playout.

  Type: integer
  Required: False
  Minimum: 0
  Maximum: 2147483647

encryption

DRM settings.

  Type: DashIsoEncryptionSettings (p. 60)
  Required: False
**destination**

Use Destination (Destination) to specify the S3 output location and the output filename base. Destination accepts format identifiers. If you do not specify the base filename in the URI, the service will use the filename of the input file. If your job has multiple inputs, the service uses the filename of the first input file.

- **Type:** string
- **Required:** True
- **Pattern:** ^s3:/\/

**segmentLength**

Length of mpd segments to create (in seconds). Note that segments will end on the next keyframe after this number of seconds, so actual segment length may be longer. When Emit Single File is checked, the segmentation is internal to a single output file and it does not cause the creation of many output files as in other output types.

- **Type:** integer
- **Required:** True
- **Minimum:** 1
- **Maximum:** 2147483647

**segmentControl**

- **Type:** DashIsoSegmentControl (p. 61)
- **Required:** False

**hbbtvCompliance**

- **Type:** DashIsoHbbtvCompliance (p. 61)
- **Required:** False

**DashIsoHbbtvCompliance (Enum)**

Supports HbbTV specification as indicated

- HBBTV_1_5
- NONE

**DashIsoSegmentControl (Enum)**

When set to SINGLE_FILE, a single output file is generated, which is internally segmented using the Fragment Length and Segment Length. When set to SEGMENTED_FILES, separate segment files will be created.

- SINGLE_FILE
- SEGMENTED_FILES

**DeinterlaceAlgorithm (Enum)**

Only applies when you set Deinterlacer (DeinterlaceMode) to Deinterlace (DEINTERLACE) or Adaptive (ADAPTIVE). Motion adaptive interpolate (INTERPOLATE) produces sharper pictures, while blend (BLEND)
produces smoother motion. Use (INTERPOLATE_TICKER) OR (BLEND_TICKER) if your source file includes a ticker, such as a scrolling headline at the bottom of the frame.

INTERPOLATE
INTERPOLATE_TICKER
BLEND
BLEND_TICKER

Deinterlacer

mode

Type: DeinterlacerMode (p. 62)
Required: False

type

control

Type: DeinterlacerControl (p. 62)
Required: False

algorithm

Type: DeinterlaceAlgorithm (p. 61)
Required: False

DeinterlacerControl (Enum)

- When set to NORMAL (default), the deinterlacer does not convert frames that are tagged in metadata as progressive. It will only convert those that are tagged as some other type. - When set to FORCE_ALL_FRAMES, the deinterlacer converts every frame to progressive - even those that are already tagged as progressive. Turn Force mode on only if there is a good chance that the metadata has tagged frames as progressive when they are not progressive. Do not turn on otherwise; processing frames that are already progressive into progressive will probably result in lower quality video.

FORCE_ALL_FRAMES
NORMAL

DeinterlacerMode (Enum)

Use Deinterlacer (DeinterlaceMode) to choose how the service will do deinterlacing. Default is Deinterlace. - Deinterlace converts interlaced to progressive. - Inverse telecine converts Hard Telecine 29.97i to progressive 23.976p. - Adaptive auto-detects and converts to progressive.

DEINTERLACE
INVERSE_TELECINE
ADAPTIVE

DropFrameTimecode (Enum)

Applies only to 29.97 fps outputs. When this feature is enabled, the service will use drop-frame timecode on outputs. If it is not possible to use drop-frame timecode, the system will fall back to non-drop-frame. This setting is enabled by default when Timecode insertion (TimecodeInsertion) is enabled.

DISABLED
ENABLED

DvbNitSettings

networkName

The network name text placed in the network_nameDescriptor inside the Network Information Table. Maximum length is 256 characters.

Type: string
    Required: True

networkId

The numeric value placed in the Network Information Table (NIT).

    Type: integer
    Required: True
    Minimum: 0
    Maximum: 65535

nitInterval

The number of milliseconds between instances of this table in the output transport stream.

    Type: integer
    Required: True
    Minimum: 25
    Maximum: 10000

DvbSdtSettings

sdtInterval

The number of milliseconds between instances of this table in the output transport stream.

    Type: integer
    Required: False
    Minimum: 25
    Maximum: 2000

serviceName

The service name placed in the serviceDescriptor in the Service Description Table. Maximum length is 256 characters.

    Type: string
    Required: False

serviceProviderName

The service provider name placed in the serviceDescriptor in the Service Description Table. Maximum length is 256 characters.
Properties

Type: string
Required: False

outputSdt

Type: OutputSdt (p. 140)
Required: False

DvbSubDestinationSettings

xPosition

Specifies the horizontal position of the caption relative to the left side of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the left of the output. If no explicit x_position is provided, the horizontal caption position will be determined by the alignment parameter. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

backgroundColor

Type: DvbSubtitleBackgroundColor (p. 67)
Required: False

teletextSpacing

Type: DvbSubtitleTeletextSpacing (p. 68)
Required: False

yPosition

Specifies the vertical position of the caption relative to the top of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the top of the output. If no explicit y_position is provided, the caption will be positioned towards the bottom of the output. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

backgroundOpacity

Specifies the opacity of the background rectangle. 255 is opaque; 0 is transparent. Leaving this parameter blank is equivalent to setting it to 0 (transparent). All burn-in and DVB-Sub font settings must match.

Type: integer
Properties

**fontOpacity**

Specifies the opacity of the burned-in captions. 255 is opaque; 0 is transparent. All burn-in and DVB-Sub font settings must match.

- **Type**: integer
- **Required**: True
- **Minimum**: 0
- **Maximum**: 255

**shadowOpacity**

 Specifies the opacity of the shadow. 255 is opaque; 0 is transparent. Leaving this parameter blank is equivalent to setting it to 0 (transparent). All burn-in and DVB-Sub font settings must match.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 255

**fontResolution**

Font resolution in DPI (dots per inch); default is 96 dpi. All burn-in and DVB-Sub font settings must match.

- **Type**: integer
- **Required**: False
- **Minimum**: 96
- **Maximum**: 600

**shadowYOffset**

Specifies the vertical offset of the shadow relative to the captions in pixels. A value of -2 would result in a shadow offset 2 pixels above the text. All burn-in and DVB-Sub font settings must match.

- **Type**: integer
- **Required**: False
- **Minimum**: -2147483648
- **Maximum**: 2147483647

**outlineSize**

Specifies font outline size in pixels. This option is not valid for source captions that are either 608/ embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

- **Type**: integer
- **Required**: True
- **Minimum**: 0
Maximum: 10

**outlineColor**

Type: DvbSubtitleOutlineColor (p. 67)
Required: True

**fontSize**

A positive integer indicates the exact font size in points. Set to 0 for automatic font size selection. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: 0
Maximum: 96

**shadowXOffset**

Specifies the horizontal offset of the shadow relative to the captions in pixels. A value of -2 would result in a shadow offset 2 pixels to the left. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

**alignment**

Type: DvbSubtitleAlignment (p. 67)
Required: True

**shadowColor**

Type: DvbSubtitleShadowColor (p. 67)
Required: False

**fontColor**

Type: DvbSubtitleFontColor (p. 67)
Required: False

**DvbSubSourceSettings**

**pid**

When using DVB-Sub with Burn-In or SMPTE-TT, use this PID for the source content. Unused for DVB-Sub passthrough. All DVB-Sub content is passed through, regardless of selectors.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

DvbSubtitleAlignment (Enum)

If no explicit x_position or y_position is provided, setting alignment to centered will place the captions at the bottom center of the output. Similarly, setting a left alignment will align captions to the bottom left of the output. If x and y positions are given in conjunction with the alignment parameter, the font will be justified (either left or centered) relative to those coordinates. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

  CENTERED
  LEFT

DvbSubtitleBackgroundColor (Enum)

Specifies the color of the rectangle behind the captions. All burn-in and DVB-Sub font settings must match.

  NONE
  BLACK
  WHITE

DvbSubtitleFontColor (Enum)

Specifies the color of the burned-in captions. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

  WHITE
  BLACK
  YELLOW
  RED
  GREEN
  BLUE

DvbSubtitleOutlineColor (Enum)

Specifies font outline color. This option is not valid for source captions that are either 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

  BLACK
  WHITE
  YELLOW
  RED
  GREEN
  BLUE

DvbSubtitleShadowColor (Enum)

Specifies the color of the shadow cast by the captions. All burn-in and DVB-Sub font settings must match.

  NONE
**DvbSubtitleTeletextSpacing (Enum)**

Controls whether a fixed grid size or proportional font spacing will be used to generate the output subtitles bitmap. Only applicable for Teletext inputs and DVB-Sub/Burn-in outputs.

- FIXED_GRID
- PROPORTIONAL

**DvbTdtSettings**

**tdtInterval**

The number of milliseconds between instances of this table in the output transport stream.

- **Type:** integer
- **Required:** True
- **Minimum:** 1000
- **Maximum:** 30000

**Eac3AttenuationControl (Enum)**

If set to ATTENUATE_3_DB, applies a 3 dB attenuation to the surround channels. Only used for 3/2 coding mode.

- ATTENUATE_3_DB
- NONE

**Eac3BitstreamMode (Enum)**

Specifies the "Bitstream Mode" (bsmod) for the emitted E-AC-3 stream. See ATSC A/52-2012 (Annex E) for background on these values.

- COMPLETE_MAIN
- COMMENTARY
- EMERGENCY
- HEARING_IMPAIRED
- VISUALLY_IMPAIRED

**Eac3CodingMode (Enum)**

Dolby Digital Plus coding mode. Determines number of channels.

- CODING_MODE_1_0
- CODING_MODE_2_0
- CODING_MODE_3_2

**Eac3DcFilter (Enum)**

Activates a DC highpass filter for all input channels.

- ENABLED
DISABLED

**Eac3DynamicRangeCompressionLine (Enum)**
Enables Dynamic Range Compression that restricts the absolute peak level for a signal.

- NONE
- FILM_STANDARD
- FILM_LIGHT
- MUSIC_STANDARD
- MUSIC_LIGHT
- SPEECH

**Eac3DynamicRangeCompressionRf (Enum)**
Enables Heavy Dynamic Range Compression, ensures that the instantaneous signal peaks do not exceed specified levels.

- NONE
- FILM_STANDARD
- FILM_LIGHT
- MUSIC_STANDARD
- MUSIC_LIGHT
- SPEECH

**Eac3LfeControl (Enum)**
When encoding 3/2 audio, controls whether the LFE channel is enabled

- LFE
- NO_LFE

**Eac3LfeFilter (Enum)**
Applies a 120Hz lowpass filter to the LFE channel prior to encoding. Only valid with 3_2_LFE coding mode.

- ENABLED
- DISABLED

**Eac3MetadataControl (Enum)**
When set to FOLLOW_INPUT, encoder metadata will be sourced from the DD, DD+, or DolbyE decoder that supplied this audio data. If audio was not supplied from one of these streams, then the static metadata settings will be used.

- FOLLOW_INPUT
- USE_CONFIGURED

**Eac3PassthroughControl (Enum)**
When set to WHEN_POSSIBLE, input DD+ audio will be passed through if it is present on the input. This detection is dynamic over the life of the transcode. Inputs that alternate between DD+ and non-DD+ content will have a consistent DD+ output as the system alternates between passthrough and encoding.
WHEN_POSSIBLE
NO_PASSTHROUGH

Eac3PhaseControl (Enum)
Controls the amount of phase-shift applied to the surround channels. Only used for 3/2 coding mode.

SHIFT_90_DEGREES
NO_SHIFT

Eac3Settings

dialnorm
Sets the dialnorm for the output. If blank and input audio is Dolby Digital Plus, dialnorm will be passed through.

Type: integer
Required: False
Minimum: 1
Maximum: 31

passthroughControl

Type: Eac3PassthroughControl (p. 69)
Required: False

metadataControl

Type: Eac3MetadataControl (p. 69)
Required: False

bitrate
Average bitrate in bits/second. Valid bitrates depend on the coding mode.

Type: integer
Required: False
Minimum: 64000
Maximum: 640000

dynamicRangeCompressionRf

Type: Eac3DynamicRangeCompressionRf (p. 69)
Required: False

sampleRate
Sample rate in hz. Sample rate is always 48000.

Type: integer
Required: False
Minimum: 48000
Maximum: 48000

**LtRtSurroundMixLevel**

Left total/Right total surround mix level. Only used for 3/2 coding mode. Valid values: -1.5 -3.0 -4.5 -6.0 -60

- **Type:** number
- **Required:** False
- **Format:** float
- **Minimum:** -60.0
- **Maximum:** -1.5

**surroundExMode**

- **Type:** Eac3SurroundExMode (p. 73)
- **Required:** False

**dynamicRangeCompressionLine**

- **Type:** Eac3DynamicRangeCompressionLine (p. 69)
- **Required:** False

**lfeControl**

- **Type:** Eac3LfeControl (p. 69)
- **Required:** False

**codingMode**

- **Type:** Eac3CodingMode (p. 68)
- **Required:** False

**surroundMode**

- **Type:** Eac3SurroundMode (p. 73)
- **Required:** False

**attenuationControl**

- **Type:** Eac3AttenuationControl (p. 68)
- **Required:** False

**lfeFilter**

- **Type:** Eac3LfeFilter (p. 69)
- **Required:** False

**phaseControl**

- **Type:** Eac3PhaseControl (p. 70)
Properties

**Required**: False

**ltRtCenterMixLevel**

Left total/Right total center mix level. Only used for 3/2 coding mode. Valid values: 3.0, 1.5, 0.0, -1.5 -3.0 -4.5 -6.0 -60

*Type*: number
*Required*: False
*Format*: float
*Minimum*: -60.0
*Maximum*: 3.0

**dcFilter**

*Type*: Eac3DcFilter (p. 68)
*Required*: False

**stereoDownmix**

*Type*: Eac3StereoDownmix (p. 72)
*Required*: False

**bitstreamMode**

*Type*: Eac3BitstreamMode (p. 68)
*Required*: False

**loRoSurroundMixLevel**

Left only/Right only surround mix level. Only used for 3/2 coding mode. Valid values: -1.5 -3.0 -4.5 -6.0 -60

*Type*: number
*Required*: False
*Format*: float
*Minimum*: -60.0
*Maximum*: -1.5

**loRoCenterMixLevel**

Left only/Right only center mix level. Only used for 3/2 coding mode. Valid values: 3.0, 1.5, 0.0, -1.5 -3.0 -4.5 -6.0 -60

*Type*: number
*Required*: False
*Format*: float
*Minimum*: -60.0
*Maximum*: 3.0

**Eac3StereoDownmix (Enum)**

Stereo downmix preference. Only used for 3/2 coding mode.
Eac3SurroundExMode (Enum)

When encoding 3/2 audio, sets whether an extra center back surround channel is matrix encoded into the left and right surround channels.

- NOT_INDICATED
- ENABLED
- DISABLED

Eac3SurroundMode (Enum)

When encoding 2/0 audio, sets whether Dolby Surround is matrix encoded into the two channels.

- NOT_INDICATED
- ENABLED
- DISABLED

EmbeddedConvert608To708 (Enum)

When set to UPCONVERT, 608 data is both passed through via the "608 compatibility bytes" fields of the 708 wrapper as well as translated into 708. 708 data present in the source content will be discarded.

- UPCONVERT
- DISABLED

EmbeddedSourceSettings

source608ChannelNumber

Specifies the 608/708 channel number within the video track from which to extract captions. Unused for passthrough.

- Type: integer
- Required: False
- Minimum: 1
- Maximum: 4

convert608To708

Type: EmbeddedConvert608To708 (p. 73)
- Required: False

source608TrackNumber

Specifies the video track index used for extracting captions. The system only supports one input video track, so this should always be set to ‘1’.

- Type: integer
Required: False
Minimum: 1
Maximum: 1

**ExceptionBody**

**message**

Type: string
Required: False

**F4vMoovPlacement (Enum)**

If set to PROGRESSIVE_DOWNLOAD, the MOOV atom is relocated to the beginning of the archive as required for progressive downloading. Otherwise it is placed normally at the end.

- PROGRESSIVE_DOWNLOAD
- NORMAL

**F4vSettings**

**moovPlacement**

Type: F4vMoovPlacement (p. 74)
Required: False

**FileGroupSettings**

**destination**

Use Destination (Destination) to specify the S3 output location and the output filename base. Destination accepts format identifiers. If you do not specify the base filename in the URI, the service will use the filename of the input file. If your job has multiple inputs, the service uses the filename of the first input file.

Type: string
Required: True
Pattern: ^s3:\/\/

**FileSourceConvert608To708 (Enum)**

If set to UPCONVERT, 608 caption data is both passed through via the "608 compatibility bytes" fields of the 708 wrapper as well as translated into 708. 708 data present in the source content will be discarded.

- UPCONVERT
- DISABLED

**FileSourceSettings**

**timeDelta**

Specifies a time delta in seconds to offset the captions from the source file.
**Type**: integer  
**Required**: False  
**Minimum**: -2147483648  
**Maximum**: 2147483647

**convert608To708**

**Type**: FileSourceConvert608To708 (p. 74)  
**Required**: False

**sourceFile**

External caption file used for loading captions. Accepted file extensions are 'scc', 'ttml', 'dfxp', 'stl', 'srt', and 'smi'. Auto-populated when Infer External Filename is checked.

**Type**: string  
**Required**: True  
**Pattern**: ^(s3://)(.*?)\.(scc|SCC|ttml|TTML|dfxp|DFXP|stl|STL|srt|SRT|smi|SMI)$

**FrameCaptureSettings**

**framerateDenominator**

Frame capture will encode the first frame of the output stream, then one frame every framerateDenominator/framerateNumerator seconds. For example, settings of framerateNumerator = 1 and framerateDenominator = 3 (a rate of 1/3 frame per second) will capture the first frame, then 1 frame every 3s. Files will be named as filename.n.jpg where n is the 0-based sequence number of each Capture.

**Type**: integer  
**Required**: False  
**Minimum**: 1  
**Maximum**: 2147483647

**maxCaptures**

Maximum number of captures (encoded jpg output files).

**Type**: integer  
**Required**: False  
**Minimum**: 1  
**Maximum**: 10000000

**framerateNumerator**

Frame capture will encode the first frame of the output stream, then one frame every framerateDenominator/framerateNumerator seconds. For example, settings of framerateNumerator = 1 and framerateDenominator = 3 (a rate of 1/3 frame per second) will capture the first frame, then 1 frame every 3s. Files will be named as filename.NNNNNNN.jpg where N is the 0-based frame sequence number zero padded to 7 decimal places.

**Type**: integer  
**Required**: False
Minimum: 1
Maximum: 2147483647

**quality**

JPEG Quality - a higher value equals higher quality.

- **Type:** integer
- **Required:** False
- **Minimum:** 1
- **Maximum:** 100

**H264AdaptiveQuantization (Enum)**

Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

- OFF
- LOW
- MEDIUM
- HIGH
- HIGHER
- MAX

**H264CodecLevel (Enum)**

H.264 Level.

- AUTO
- LEVEL_1
- LEVEL_1_1
- LEVEL_1_2
- LEVEL_1_3
- LEVEL_2
- LEVEL_2_1
- LEVEL_2_2
- LEVEL_3
- LEVEL_3_1
- LEVEL_3_2
- LEVEL_4
- LEVEL_4_1
- LEVEL_4_2
- LEVEL_5
- LEVEL_5_1
- LEVEL_5_2

**H264CodecProfile (Enum)**

H.264 Profile. High 4:2:2 and 10-bit profiles are only available with the AVC-I License.

- BASELINE
- HIGH
- HIGH_10BIT
- HIGH_422
HIGH_422_10BIT
MAIN

**H264EntropyEncoding (Enum)**

Entropy encoding mode. Use CABAC (must be in Main or High profile) or CAVLC.

- CABAC
- CAVLC

**H264FieldEncoding (Enum)**

Choosing FORCE_FIELD disables PAFF encoding for interlaced outputs.

- PAFF
- FORCE_FIELD

**H264FlickerAdaptiveQuantization (Enum)**

Adjust quantization within each frame to reduce flicker or 'pop' on I-frames.

- DISABLED
- ENABLED

**H264FramerateControl (Enum)**

Using the API, set FramerateControl to INITIALIZE_FROM_SOURCE if you want the service to use the framerate from the input. Using the console, do this by choosing INITIALIZE_FROM_SOURCE for Framerate.

- INITIALIZE_FROM_SOURCE
- SPECIFIED

**H264FramerateConversionAlgorithm (Enum)**

When set to INTERPOLATE, produces smoother motion during framerate conversion.

- DUPLICATE_DROP
- INTERPOLATE

**H264GopBReference (Enum)**

If enable, use reference B frames for GOP structures that have B frames > 1.

- DISABLED
- ENABLED

**H264GopSizeUnits (Enum)**

Indicates if the GOP Size in H264 is specified in frames or seconds. If seconds the system will convert the GOP Size into a frame count at run time.

- FRAMES
- SECONDS
H264InterlaceMode (Enum)

Use Interlace mode (InterlaceMode) to choose the scan line type for the output.

* Top Field First (TOP_FIELD) and Bottom Field First (BOTTOM_FIELD) produce interlaced output with the entire output having the same field polarity (top or bottom first).

* Follow, Default Top (FOLLOW_TOP_FIELD) and Follow, Default Bottom (FOLLOW_BOTTOM_FIELD) use the same field polarity as the source. Therefore, behavior depends on the input scan type. If the source is interlaced, the output will be interlaced with the same polarity as the source (it will follow the source). The output could therefore be a mix of "top field first" and "bottom field first". If the source is progressive, the output will be interlaced with "top field first" or "bottom field first" polarity, depending on which of the Follow options you chose.

```plaintext
PROGRESSIVE
TOP_FIELD
BOTTOM_FIELD
FOLLOW_TOP_FIELD
FOLLOW_BOTTOM_FIELD
```

H264ParControl (Enum)

Using the API, enable ParFollowSource if you want the service to use the pixel aspect ratio from the input. Using the console, do this by choosing Follow source for Pixel aspect ratio.

```
INITIALIZE_FROM_SOURCE
SPECIFIED
```

H264QualityTuningLevel (Enum)

Use Quality tuning level (H264QualityTuningLevel) to specify whether to use fast single-pass, high-quality singlepass, or high-quality multipass video encoding.

```
SINGLE_PASS
SINGLE_PASS_HQ
MULTI_PASS_HQ
```

H264RateControlMode (Enum)

Rate control mode. CQ uses constant quantizer (qp), ABR (average bitrate) does not write HRD parameters.

```
VBR
CBR
```

H264RepeatPps (Enum)

Places a PPS header on each encoded picture, even if repeated.

```
DISABLED
ENABLED
```

H264SceneChangeDetect (Enum)

Scene change detection (inserts I-frames on scene changes).
DISABLED
ENABLED

H264Settings

slices
Number of slices per picture. Must be less than or equal to the number of macroblock rows for progressive pictures, and less than or equal to half the number of macroblock rows for interlaced pictures.

Type: integer
Required: False
Minimum: 1
Maximum: 32

minIInterval
Enforces separation between repeated (cadence) I-frames and I-frames inserted by Scene Change Detection. If a scene change I-frame is within I-interval frames of a cadence I-frame, the GOP is shrunk and/or stretched to the scene change I-frame. GOP stretch requires enabling lookahead as well as setting I-interval. The normal cadence resumes for the next GOP. This setting is only used when Scene Change Detect is enabled. Note: Maximum GOP stretch = GOP size + Min-I-interval - 1

Type: integer
Required: False
Minimum: 0
Maximum: 30

parNumerator
Pixel Aspect Ratio numerator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

flickerAdaptiveQuantization

Type: H264FlickerAdaptiveQuantization (p. 77)
Required: False

gopSizeUnits

Type: H264GopSizeUnits (p. 77)
Required: False

hrdBufferSize
Size of buffer (HRD buffer model). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.

Type: integer
**qualityTuningLevel**

Type: \texttt{H264QualityTuningLevel (p. 78)}

**maxBitrate**

Maximum bitrate in bits/second (for VBR mode only). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.

**bitrate**

Average bitrate in bits/second. Required for VBR, CBR, and ABR. Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.

**spatialAdaptiveQuantization**

Type: \texttt{H264SpatialAdaptiveQuantization (p. 84)}

**slowPal**

Type: \texttt{H264SlowPal (p. 84)}

**codecProfile**

Type: \texttt{H264CodecProfile (p. 76)}

**unregisteredSeiTimecode**

Type: \texttt{H264UnregisteredSeiTcTimecode (p. 84)}

**softness**

Softness. Selects quantizer matrix, larger values reduce high-frequency content in the encoded image.
Type: integer
Required: False
Minimum: 0
Maximum: 128

framerateControl
Type: H264FramerateControl (p. 77)
Required: False

telecine
Type: H264Telecine (p. 84)
Required: False

framerateConversionAlgorithm
Type: H264FramerateConversionAlgorithm (p. 77)
Required: False

codecLevel
Type: H264CodecLevel (p. 76)
Required: False

numberReferenceFrames
Number of reference frames to use. The encoder may use more than requested if using B-frames and/or
interlaced encoding.
Type: integer
Required: False
Minimum: 1
Maximum: 6

temporalAdaptiveQuantization
Type: H264TemporalAdaptiveQuantization (p. 84)
Required: False

repeatPps
Type: H264RepeatPps (p. 78)
Required: False

hrdBufferInitialFillPercentage
Percentage of the buffer that should initially be filled (HRD buffer model).
Type: integer
Required: False
Minimum: 0
Maximum: 100

framerateNumerator

Framerate numerator - framerate is a fraction, e.g. 24000 / 1001 = 23.976 fps.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

numberBFramesBetweenReferenceFrames

Number of B-frames between reference frames.

Type: integer
Required: False
Minimum: 0
Maximum: 7

gopClosedCadence

Frequency of closed GOPs. In streaming applications, it is recommended that this be set to 1 so a decoder joining mid-stream will receive an IDR frame as quickly as possible. Setting this value to 0 will break output segmenting.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

fieldEncoding

Type: H264FieldEncoding (p. 77)
Required: False

entropyEncoding

Type: H264EntropyEncoding (p. 77)
Required: False

framerateDenominator

When you use the API for transcode jobs that use framerate conversion, specify the framerate as a fraction. For example, 24000 / 1001 = 23.976 fps. Use FramerateDenominator to specify the denominator of this fraction. In this example, use 1001 for the value of FramerateDenominator. When you use the console for transcode jobs that use framerate conversion, provide the value as a decimal number for Framerate. In this example, specify 23.976.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647
adaptiveQuantization
Type: H264AdaptiveQuantization (p. 76)
Required: False

interlaceMode
Type: H264InterlaceMode (p. 78)
Required: False

gopSize
GOP Length (keyframe interval) in frames or seconds. Must be greater than zero.
Type: number
Required: False
Format: float
Minimum: 0.0

parDenominator
Pixel Aspect Ratio denominator.
Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

parControl
Type: H264ParControl (p. 78)
Required: False

rateControlMode
Type: H264RateControlMode (p. 78)
Required: False

H264SlowPal (Enum)

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

DISABLED
ENABLED

H264SpatialAdaptiveQuantization (Enum)

Adjust quantization within each frame based on spatial variation of content complexity.

DISABLED
ENABLED

H264Syntax (Enum)

Produces a bitstream compliant with SMPTE RP-2027.

DEFAULT
RP2027

H264Telecine (Enum)

This field applies only if the Streams > Advanced > Framerate (framerate) field is set to 29.970. This field works with the Streams > Advanced > Preprocessors > Deinterlacer field (deinterlace_mode) and the Streams > Advanced > Interlaced Mode field (interlace_mode) to identify the scan type for the output: Progressive, Interlaced, Hard Telecine or Soft Telecine. - Hard: produces 29.97i output from 23.976 input. - Soft: produces 23.976; the player converts this output to 29.97i.

NONE
SOFT
HARD

H264TemporalAdaptiveQuantization (Enum)

Adjust quantization within each frame based on temporal variation of content complexity.

DISABLED
ENABLED

H264UnregisteredSeiTImecode (Enum)

Inserts timecode for each frame as 4 bytes of an unregistered SEI message.

DISABLED
ENABLED

H265AdaptiveQuantization (Enum)

Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.
H265AlternateTransferFunctionSei (Enum)

Enables Alternate Transfer Function SEI message for outputs using Hybrid Log Gamma (HLG) Electro-Optical Transfer Function (EOTF).

DISABLED
ENABLED

H265CodecLevel (Enum)

H.265 Level.

AUTO
LEVEL_1
LEVEL_2
LEVEL_2_1
LEVEL_3
LEVEL_3_1
LEVEL_4
LEVEL_4_1
LEVEL_5
LEVEL_5_1
LEVEL_5_2
LEVEL_6
LEVEL_6_1
LEVEL_6_2

H265CodecProfile (Enum)

Represents the Profile and Tier, per the HEVC (H.265) specification. Selections are grouped as [Profile] / [Tier], so “Main/High” represents Main Profile with High Tier. 4:2:2 profiles are only available with the HEVC 4:2:2 License.

MAIN_MAIN
MAIN_HIGH
MAIN10_MAIN
MAIN10_HIGH
MAIN_422_8BIT_MAIN
MAIN_422_8BIT_HIGH
MAIN_422_10BIT_MAIN
MAIN_422_10BIT_HIGH

H265FlickerAdaptiveQuantization (Enum)

Adjust quantization within each frame to reduce flicker or 'pop' on I-frames.
DISABLED
ENABLED

H265FramerateControl (Enum)

Using the API, set FramerateControl to INITIALIZE_FROM_SOURCE if you want the service to use the framerate from the input. Using the console, do this by choosing INITIALIZE_FROM_SOURCE for Framerate.

INITIALIZE_FROM_SOURCE
SPECIFIED

H265FramerateConversionAlgorithm (Enum)

When set to INTERPOLATE, produces smoother motion during framerate conversion.

DUPLICATE_DROP
INTERPOLATE

H265GopBReference (Enum)

If enable, use reference B frames for GOP structures that have B frames > 1.

DISABLED
ENABLED

H265GopSizeUnits (Enum)

Indicates if the GOP Size in H265 is specified in frames or seconds. If seconds the system will convert the GOP Size into a frame count at run time.

FRAMES
SECONDS

H265InterlaceMode (Enum)

Use Interlace mode (InterlaceMode) to choose the scan line type for the output.

* Top Field First (TOP_FIELD) and Bottom Field First (BOTTOM_FIELD) produce interlaced output with the entire output having the same field polarity (top or bottom first).

* Follow, Default Top (FOLLOW_TOP_FIELD) and Follow, Default Bottom (FOLLOW_BOTTOM_FIELD) use the same field polarity as the source. Therefore, behavior depends on the input scan type. If the source is interlaced, the output will be interlaced with the same polarity as the source (it will follow the source). The output could therefore be a mix of “top field first” and “bottom field first”. If the source is progressive, the output will be interlaced with “top field first” or “bottom field first” polarity, depending on which of the Follow options you chose.

PROGRESSIVE
TOP_FIELD
BOTTOM_FIELD
FOLLOW_TOP_FIELD
FOLLOW_BOTTOM_FIELD
**H265ParControl (Enum)**

Using the API, enable ParFollowSource if you want the service to use the pixel aspect ratio from the input. Using the console, do this by choosing Follow source for Pixel aspect ratio.

- INITIALIZE_FROM_SOURCE
- SPECIFIED

**H265QualityTuningLevel (Enum)**

Use Quality tuning level (H265QualityTuningLevel) to specify whether to use fast single-pass, high-quality singlepass, or high-quality multipass video encoding.

- SINGLE_PASS
- SINGLE_PASS_HQ
- MULTI_PASS_HQ

**H265RateControlMode (Enum)**

Rate control mode. CQ uses constant quantizer (qp), ABR (average bitrate) does not write HRD parameters.

- VBR
- CBR

**H265SampleAdaptiveOffsetFilterMode (Enum)**

Specify Sample Adaptive Offset (SAO) filter strength. Adaptive mode dynamically selects best strength based on content.

- DEFAULT
- ADAPTIVE
- OFF

**H265SceneChangeDetect (Enum)**

Scene change detection (inserts I-frames on scene changes).

- DISABLED
- ENABLED

**H265Settings**

**slices**

Number of slices per picture. Must be less than or equal to the number of macroblock rows for progressive pictures, and less than or equal to half the number of macroblock rows for interlaced pictures.

- **Type:** integer
- **Required:** False
- **Minimum:** 1
- **Maximum:** 32
**minIInterval**

Enforces separation between repeated (cadence) I-frames and I-frames inserted by Scene Change Detection. If a scene change I-frame is within I-interval frames of a cadence I-frame, the GOP is shrunk and/or stretched to the scene change I-frame. GOP stretch requires enabling lookahead as well as setting I-interval. The normal cadence resumes for the next GOP. This setting is only used when Scene Change Detect is enabled. Note: Maximum GOP stretch = GOP size + Min-I-interval - 1

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 30

**parNumerator**

Pixel Aspect Ratio numerator.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 2147483647

**flickerAdaptiveQuantization**

- **Type**: H265FlickerAdaptiveQuantization (p. 85)
- **Required**: False

**gopSizeUnits**

- **Type**: H265GopSizeUnits (p. 86)
- **Required**: False

**hrdBufferSize**

Size of buffer (HRD buffer model). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.

- **Type**: integer
- **Required**: False
- **Minimum**: -2147483648
- **Maximum**: 2147483647

**qualityTuningLevel**

- **Type**: H265QualityTuningLevel (p. 87)
- **Required**: False

**maxBitrate**

Maximum bitrate in bits/second (for VBR mode only). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.

- **Type**: integer
- **Required**: False
Properties

Minimum: -2147483648
Maximum: 2147483647

**bitrate**

Average bitrate in bits/second. Required for VBR, CBR, and ABR. Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.

Type: integer
Required: False
Minimum: 1000
Maximum: 2147483647

**spatialAdaptiveQuantization**

Type: H265SpatialAdaptiveQuantization (p. 92)
Required: False

**sampleAdaptiveOffsetFilterMode**

Type: H265SampleAdaptiveOffsetFilterMode (p. 87)
Required: False

**temporalIds**

Type: H265TemporalIds (p. 93)
Required: False

**slowPal**

Type: H265SlowPal (p. 92)
Required: False

**tiles**

Type: H265Tiles (p. 93)
Required: False

**codecProfile**

Type: H265CodecProfile (p. 85)
Required: False

**alternateTransferFunctionSei**

Type: H265AlternateTransferFunctionSei (p. 85)
Required: False

**unregisteredSeiTimecode**

Type: H265UnregisteredSeiTimecode (p. 93)
Required: False

framerateControl

Type: H265FramerateControl (p. 86)  
Required: False

telecine

Type: H265Telecine (p. 92)  
Required: False

framerateConversionAlgorithm

Type: H265FramerateConversionAlgorithm (p. 86)  
Required: False

codecLevel

Type: H265CodecLevel (p. 85)  
Required: False

numberReferenceFrames

Number of reference frames to use. The encoder may use more than requested if using B-frames and/or interlaced encoding.

Type: integer  
Required: False  
Minimum: 1  
Maximum: 6

temporalAdaptiveQuantization

Type: H265TemporalAdaptiveQuantization (p. 93)  
Required: False

hrdBufferInitialFillPercentage

Percentage of the buffer that should initially be filled (HRD buffer model).

Type: integer  
Required: False  
Minimum: 0  
Maximum: 100

framerateNumerator

Framerate numerator - framerate is a fraction, e.g. 24000 / 1001 = 23.976 fps.

Type: integer  
Required: False  
Minimum: 1
Maximum: 2147483647

**numberBFramesBetweenReferenceFrames**

Number of B-frames between reference frames.

Type: integer  
Required: False  
Minimum: 0  
Maximum: 7

**gopClosedCadence**

Frequency of closed GOPs. In streaming applications, it is recommended that this be set to 1 so a decoder joining mid-stream will receive an IDR frame as quickly as possible. Setting this value to 0 will break output segmenting.

Type: integer  
Required: False  
Minimum: 0  
Maximum: 2147483647

**framerateDenominator**

Framerate denominator.

Type: integer  
Required: False  
Minimum: 1  
Maximum: 2147483647

**adaptiveQuantization**

Type: H265AdaptiveQuantization (p. 84)  
Required: False

**interlaceMode**

Type: H265InterlaceMode (p. 86)  
Required: False

**gopSize**

GOP Length (keyframe interval) in frames or seconds. Must be greater than zero.

Type: number  
Required: False  
Format: float  
Minimum: 0.0

**gopBReference**

Type: H265GopBReference (p. 86)
**Required**: False

**sceneChangeDetect**

**Type**: H265SceneChangeDetect (p. 87)

**Required**: False

**parDenominator**

Pixel Aspect Ratio denominator.

**Type**: integer

**Required**: False

**Minimum**: 1

**Maximum**: 2147483647

**parControl**

**Type**: H265ParControl (p. 87)

**Required**: False

**rateControlMode**

**Type**: H265RateControlMode (p. 87)

**Required**: False

**H265SlowPal (Enum)**

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

- **DISABLED**
- **ENABLED**

**H265SpatialAdaptiveQuantization (Enum)**

Adjust quantization within each frame based on spatial variation of content complexity.

- **DISABLED**
- **ENABLED**

**H265Telecine (Enum)**

This field applies only if the Streams > Advanced > Framerate (framerate) field is set to 29.970. This field works with the Streams > Advanced > Preprocessors > Deinterlacer field (deinterlace_mode) and the Streams > Advanced > Interlaced Mode field (interlace_mode) to identify the scan type for the output: Progressive, Interlaced, Hard Telecine or Soft Telecine. - Hard: produces 29.97i output from 23.976 input. - Soft: produces 23.976; the player converts this output to 29.97i.

- **NONE**
- **SOFT**
- **HARD**
H265TemporalAdaptiveQuantization (Enum)
Adjust quantization within each frame based on temporal variation of content complexity.

DISABLED
ENABLED

H265TemporalIds (Enum)
Enables temporal layer identifiers in the encoded bitstream. Up to 3 layers are supported depending on GOP structure: I- and P-frames form one layer, reference B-frames can form a second layer and non-reference b-frames can form a third layer. Decoders can optionally decode only the lower temporal layers to generate a lower frame rate output. For example, given a bitstream with temporal IDs and with b-frames = 1 (i.e. IbPbPb display order), a decoder could decode all the frames for full frame rate output or only the I and P frames (lowest temporal layer) for a half frame rate output.

DISABLED
ENABLED

H265Tiles (Enum)
Enable use of tiles, allowing horizontal as well as vertical subdivision of the encoded pictures.

DISABLED
ENABLED

H265UnregisteredSeiTimecode (Enum)
Inserts timecode for each frame as 4 bytes of an unregistered SEI message.

DISABLED
ENABLED

Hdr10Metadata
redPrimaryY
HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

Type: integer
Required: False
Minimum: 0
Maximum: 50000

greenPrimaryY
HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

Type: integer
Required: False
Minimum: 0
Maximum: 50000
whitePointX

HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

Type: integer
Required: False
Minimum: 0
Maximum: 50000

maxLuminance

Nominal maximum mastering display luminance in units of 0.0001 candelas per square meter.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

greenPrimaryX

HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

Type: integer
Required: False
Minimum: 0
Maximum: 50000

whitePointY

HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

Type: integer
Required: False
Minimum: 0
Maximum: 50000

redPrimaryX

HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

Type: integer
Required: False
Minimum: 0
Maximum: 50000

bluePrimaryX

HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

Type: integer
maxFrameAverageLightLevel

Maximum average light level of any frame in the coded video sequence, in units of candelas per square meter.

Type: integer
Required: True
Minimum: 0
Maximum: 65535

bluePrimaryY

HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

Type: integer
Required: False
Minimum: 0
Maximum: 50000

maxContentLightLevel

Maximum light level among all samples in the coded video sequence, in units of candelas per square meter.

Type: integer
Required: True
Minimum: 0
Maximum: 65535

minLuminance

Nominal minimum mastering display luminance in units of of 0.0001 candelas per square meter

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

HlsAdMarkers (Enum)

- ELEMENTAL
- ELEMENTAL_SCTE35

HlsAudioTrackType (Enum)

Four types of audio-only tracks are supported: Audio-Only Variant Stream The client can play back this audio-only stream instead of video in low-bandwidth scenarios. Represented as an EXT-X-STREAM-INF in the HLS manifest. Alternate Audio, Auto Select, Default Alternate rendition that the client should
try to play back by default. Represented as an EXT-X-MEDIA in the HLS manifest with DEFAULT=YES, AUTOSELECT=YES Alternate Audio, Auto Select, Not Default Alternate rendition that the client may try to play back by default. Represented as an EXT-X-MEDIA in the HLS manifest with DEFAULT=NO, AUTOSELECT=YES Alternate Audio, not Auto Select Alternate rendition that the client will not try to play back by default. Represented as an EXT-X-MEDIA in the HLS manifest with DEFAULT=NO, AUTOSELECT=NO

ALTERNATE_AUDIO_AUTO_SELECT_DEFAULT
ALTERNATE_AUDIO_AUTO_SELECT
ALTERNATE_AUDIO_NOT_AUTO_SELECT
AUDIO_ONLY_VARIANT_STREAM

HlsCaptionLanguageMapping

languageDescription

Caption language description.

Type: string
Required: False

captionChannel

Caption channel.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

languageCode

Type: LanguageCode (p. 111)
Required: False

HlsCaptionLanguageSetting (Enum)

Applies only to 608 Embedded output captions. Insert: Include CLOSED-CAPTIONS lines in the manifest. Specify at least one language in the CC1 Language Code field. One CLOSED-CAPTION line is added for each Language Code you specify. Make sure to specify the languages in the order in which they appear in the original source (if the source is embedded format) or the order of the caption selectors (if the source is other than embedded). Otherwise, languages in the manifest will not match up properly with the output captions. None: Include CLOSED-CAPTIONS=NULL line in the manifest. Omit: Omit any CLOSED-CAPTIONS line from the manifest.

INSERT
OMIT
NONE

HlsClientCache (Enum)

When set to ENABLED, sets #EXT-X-ALLOW-CACHE:no tag, which prevents client from saving media segments for later replay.
DISABLED
ENABLED

**HlsCodecSpecification (Enum)**

Specification to use (RFC-6381 or the default RFC-4281) during m3u8 playlist generation.

RFC_6381
RFC_4281

**HlsDirectoryStructure (Enum)**

Indicates whether segments should be placed in subdirectories.

SINGLE_DIRECTORY
SUBDIRECTORY_PER_STREAM

**HlsEncryptionSettings**

**initializationVectorInManifest**

Type: HlsInitializationVectorInManifest (p. 101)
Required: False

**constantInitializationVector**

This is a 128-bit, 16-byte hex value represented by a 32-character text string. If this parameter is not set then the Initialization Vector will follow the segment number by default.

Type: string
Required: False
Pattern: ^[0-9a-zA-F]{32}$

**staticKeyProvider**

Type: StaticKeyProvider (p. 146)
Required: False

**type**

Type: HlsKeyProviderType (p. 101)
Required: True

**spekeKeyProvider**

Type: SpekeKeyProvider (p. 145)
Required: False

**encryptionMethod**

Type: HlsEncryptionType (p. 98)
Required: False
HlsEncryptionType (Enum)

Encrypts the segments with the given encryption scheme. Leave blank to disable. Selecting 'Disabled' in the web interface also disables encryption.

- AES128
- SAMPLE_AES

HlsGroupSettings

segmentsPerSubdirectory

Number of segments to write to a subdirectory before starting a new one. directoryStructure must be SINGLE_DIRECTORY for this setting to have an effect.

- Type: integer
- Required: False
- Minimum: 1
- Maximum: 2147483647

streamInfResolution

- Type: HlsStreamInfResolution (p. 102)
- Required: False

timestampDeltaMilliseconds

Provides an extra millisecond delta offset to fine tune the timestamps.

- Type: integer
- Required: False
- Minimum: -2147483648
- Maximum: 2147483647

outputSelection

- Type: HlsOutputSelection (p. 101)
- Required: False

captionLanguageMappings

- Type: Array of type HlsCaptionLanguageMapping (p. 96)
- Required: False

clientCache

- Type: HlsClientCache (p. 96)
- Required: False

codecSpecification

- Type: HlsCodecSpecification (p. 97)
- Required: False
destination

Use Destination (Destination) to specify the S3 output location and the output filename base. Destination accepts format identifiers. If you do not specify the base filename in the URI, the service will use the filename of the input file. If your job has multiple inputs, the service uses the filename of the first input file.

Type: string
Required: True
Pattern: ^s3:\/\/\/

segmentControl

Type: HlsSegmentControl (p. 101)
Required: False

timedMetadataId3Frame

Type: HlsTimedMetadataId3Frame (p. 102)
Required: False

timedMetadataId3Period

Timed Metadata interval in seconds.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

captionLanguageSetting

Type: HlsCaptionLanguageSetting (p. 96)
Required: False

minSegmentLength

When set, Minimum Segment Size is enforced by looking ahead and back within the specified range for a nearby avail and extending the segment size if needed.

Type: integer
Required: True
Minimum: 0
Maximum: 2147483647

directoryStructure

Type: HlsDirectoryStructure (p. 97)
Required: False

programDateTime

Type: HlsProgramDateTime (p. 101)
Required: False
basePath
A partial URI prefix that will be prepended to each output in the media.m3u8 file. Can be used if base manifest is delivered from a different URL than the main.m3u8 file.

- **Type:** string
- **Required:** False

encryption
DRM settings.

- **Type:** HlsEncryptionSettings (p. 97)
- **Required:** False

adMarkers
Choose one or more ad marker types to pass SCTE35 signals through to this group of Apple HLS outputs.

- **Type:** Array of type HlsAdMarkers (p. 95)
- **Required:** False

programDateTimePeriod
Period of insertion of EXT-X-PROGRAM-DATE-TIME entry, in seconds.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 3600

manifestCompression

- **Type:** HlsManifestCompression (p. 101)
- **Required:** False

segmentLength
Length of MPEG-2 Transport Stream segments to create (in seconds). Note that segments will end on the next keyframe after this number of seconds, so actual segment length may be longer.

- **Type:** integer
- **Required:** True
- **Minimum:** 1
- **Maximum:** 2147483647

manifestDurationFormat

- **Type:** HlsManifestDurationFormat (p. 101)
- **Required:** False

HlsIFrameOnlyManifest (Enum)
When set to INCLUDE, writes I-Frame Only Manifest in addition to the HLS manifest
HlsInitializationVectorInManifest (Enum)

The Initialization Vector is a 128-bit number used in conjunction with the key for encrypting blocks. If set to INCLUDE, Initialization Vector is listed in the manifest. Otherwise Initialization Vector is not in the manifest.

INCLUDE
EXCLUDE

HlsKeyProviderType (Enum)

Indicates which type of key provider is used for encryption.

SPEKE
STATIC_KEY

HlsManifestCompression (Enum)

When set to GZIP, compresses HLS playlist.

GZIP
NONE

HlsManifestDurationFormat (Enum)

Indicates whether the output manifest should use floating point values for segment duration.

FLOATING_POINT
INTEGER

HlsOutputSelection (Enum)

Indicates whether the .m3u8 manifest file should be generated for this HLS output group.

MANIFESTS_AND_SEGMENTS
SEGMENTS_ONLY

HlsProgramDateTime (Enum)

Includes or excludes EXT-X-PROGRAM-DATE-TIME tag in .m3u8 manifest files. The value is calculated as follows: either the program date and time are initialized using the input timecode source, or the time is initialized using the input timecode source and the date is initialized using the timestamp_offset.

INCLUDE
EXCLUDE

HlsSegmentControl (Enum)

When set to SINGLE_FILE, emits program as a single media resource (.ts) file, uses #EXT-X-BYTERANGE tags to index segment for playback.
SINGLE_FILE
SEGMENTED_FILES

HlsSettings

iFrameOnlyManifest

Type: HlsIFrameOnlyManifest (p. 100)
Required: False

segmentModifier

String concatenated to end of segment filenames. Accepts "Format Identifiers":#format_identifier_parameters.

Type: string
Required: False

audioTrackType

Type: HlsAudioTrackType (p. 95)
Required: False

audioRenditionSets

List all the audio groups that are used with the video output stream. Input all the audio GROUP-IDs that are associated to the video, separate by ','.

Type: string
Required: False

audioGroupId

Specifies the group to which the audio Rendition belongs.

Type: string
Required: False

HlsStreamInfResolution (Enum)

Include or exclude RESOLUTION attribute for video in EXT-X-STREAM-INF tag of variant manifest.

INCLUDE
EXCLUDE

HlsTimedMetadataId3Frame (Enum)

Indicates ID3 frame that has the timecode.

NONE
PRIV
TDRL
**Id3Insertion**

**id3**
Use ID3 tag (id3) to provide a tag value in base64-encode format.

- **Type:** string
- **Required:** True
- **Pattern:** ^[A-Za-z0-9+/]+={0,2}$

**timecode**
Provide a Timecode (TimeCode) in HH:MM:SS:FF or HH:MM:SS;FF format.

- **Type:** string
- **Required:** True
- **Format:** timecode
- **Pattern:** ^([01][0-9]|2[0-4]):[0-5][0-9]:[0-5][0-9]:[0-9]{2}$

**ImageInserter**

**insertableImages**
Image to insert. Must be 32 bit windows BMP, PNG, or TGA file. Must not be larger than the output frames.

- **Type:** Array of type InsertableImage (p. 106)
- **Required:** True

**InputClipping**

**startTimecode**
Set Start timecode (StartTimecode) to the beginning of the portion of the input you are clipping. The frame corresponding to the Start timecode value is included in the clip. Start timecode or End timecode may be left blank, but not both. When choosing this value, take into account your setting for Input timecode source. For example, if you have embedded timecodes that start at 01:00:00:00 and you want your clip to begin five minutes into the video, use 01:00:05:00.

- **Type:** string
- **Required:** False
- **Format:** timecode
- **Pattern:** ^([01][0-9]|2[0-4]):[0-5][0-9]:[0-5][0-9]:[0-9]{2}$

**endTimecode**
Set End timecode (EndTimecode) to the end of the portion of the input you are clipping. The frame corresponding to the End timecode value is included in the clip. Start timecode or End timecode may be left blank, but not both. When choosing this value, take into account your setting for Input timecode source. For example, if you have embedded timecodes that start at 01:00:00:00 and you want your clip to begin five minutes into the video, use 01:00:05:00.

- **Type:** string
- **Required:** False
- **Format:** timecode
- **Pattern:** ^([01][0-9]|2[0-4]):[0-5][0-9]:[0-5][0-9]:[0-9]{2}$
**Pattern:** ^([01][0-9]|2[0-4]):[0-5][0-9]:[0-5][0-9][;][0-9]{2}$

**InputDeblockFilter (Enum)**

Enable Deblock (InputDeblockFilter) to produce smoother motion in the output. Default is disabled. Only manually controllable for MPEG2 and uncompressed video inputs.

- ENABLED
- DISABLED

**InputDenoiseFilter (Enum)**

Enable Denoise (InputDenoiseFilter) to filter noise from the input. Default is disabled. Only applicable to MPEG2, H.264, H.265, and uncompressed video inputs.

- ENABLED
- DISABLED

**InputFilterEnable (Enum)**

Use Filter enable (InputFilterEnable) to specify how the transcoding service applies the denoise and deblock filters. You must also enable the filters separately, with Denoise (InputDenoiseFilter) and Debloc (InputDeblockFilter).

- Auto - The transcoding service determines whether to apply filtering, depending on input type and quality.
- Disable - The input is not filtered. This is true even if you use the API to enable them in (InputDeblockFilter) and (InputDeblockFilter).
- Force - The input is filtered regardless of input type.

- AUTO
- DISABLE
- FORCE

**InputPsiControl (Enum)**

Set PSI control (InputPsiControl) for transport stream inputs to specify which data the demux process to scans.

- Ignore PSI - Scan all PIDs for audio and video.
- Use PSI - Scan only PSI data.

- IGNORE_PSI
- USE_PSI

**InputTemplate**

**audioSelectors**

Use Audio selectors (AudioSelectors) to specify a track or set of tracks from the input that you will use in your outputs. You can use multiple Audio selectors per input.

**Type**: object

**Required**: False
**audioSelectorGroups**

Specifies set of audio selectors within an input to combine. An input may have multiple audio selector groups. See "Audio Selector Group":#inputs-audio_selector_group for more information.

- **Type**: object
- **Required**: False

**filterEnable**

- **Type**: InputFilterEnable (p. 104)
- **Required**: False

**deblockFilter**

- **Type**: InputDeblockFilter (p. 104)
- **Required**: False

**videoSelector**

- **Type**: VideoSelector (p. 153)
- **Required**: False

**filterStrength**

Use Filter strength (FilterStrength) to adjust the magnitude the input filter settings (Deblock and Denoise). The range is -5 to 5. Default is 0.

- **Type**: integer
- **Required**: False
- **Minimum**: -5
- **Maximum**: 5

**programNumber**

Use Program (programNumber) to select a specific program from within a multi-program transport stream. Note that Quad 4K is not currently supported. Default is the first program within the transport stream. If the program you specify doesn’t exist, the transcoding service will use this default.

- **Type**: integer
- **Required**: False
- **Minimum**: -2147483648
- **Maximum**: 2147483647

**timecodeSource**

- **Type**: InputTimecodeSource (p. 106)
- **Required**: False

**captionSelectors**

Use Captions selectors (CaptionSelectors) to specify the captions data from the input that you will use in your outputs. You can use multiple captions selectors per input.
Type: object  
Required: False

denoiseFilter
Type: InputDenoiseFilter (p. 104)  
Required: False

psiControl
Type: InputPsiControl (p. 104)  
Required: False

inputClippings
(InputClippings) contains sets of start and end times that together specify a portion of the input to be used in the outputs. If you provide only a start time, the clip will be the entire input from that point to the end. If you provide only an end time, it will be the entire input up to that point. When you specify more than one input clip, the transcoding service creates the job outputs by stringing the clips together in the order you specify them.

Type: Array of type InputClipping (p. 103)  
Required: False

InputTimecodeSource (Enum)

Use Timecode source (InputTimecodeSource) to specify how timecode information from your input is adjusted and encoded in all outputs for the job. Default is embedded. Set to Embedded (EMBEDDED) to use the timecode that is in the input video. If no embedded timecode is in the source, will set the timecode for the first frame to 00:00:00:00. Set to Start at 0 (ZEROBASED) to set the timecode of the initial frame to 00:00:00:00. Set to Specified start (SPECIFIEDSTART) to provide the initial timecode yourself the setting (Start).

EMBEDDED  
ZEROBASED  
SPECIFIEDSTART

InsertableImage

duration
Use Duration (Duration) to set the time, in milliseconds, for the image to remain on the output video.

Type: integer  
Required: False  
Minimum: -2147483648  
Maximum: 2147483647

fadeOut
Use Fade out (FadeOut) to set the length, in milliseconds, of the inserted image fade out. If you don’t specify a value for Fade out, the image will disappear abruptly at the end of the inserted image duration.
Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

**imageY**

Use Top (ImageY) to set the distance, in pixels, between the inserted image and the top edge of the video frame. Required for BMP, PNG and TGA input.

Type: integer
Required: True
Minimum: -2147483648
Maximum: 2147483647

**fadeIn**

Use Fade in (FadeIn) to set the length, in milliseconds, of the inserted image fade in. If you don't specify a value for Fade in, the image will appear abruptly at the Start time.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

**imageX**

Use Left (ImageX) to set the distance, in pixels, between the inserted image and the left edge of the frame. Required for BMP, PNG and TGA input.

Type: integer
Required: True
Minimum: -2147483648
Maximum: 2147483647

**width**

Specify the Width (Width) of the inserted image. Use a value that is less than or equal to the video resolution width. Leave this setting blank to use the native width of the image.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

**startTime**

Use Start time (StartTime) to specify the video timecode when the image is inserted in the output. This must be in timecode format (HH:MM:SS:FF)

Type: string
Required: False
opacity

Use Opacity (Opacity) to specify how much of the underlying video shows through the inserted image. 0 is transparent and 100 is fully opaque. Default is 50.

  Type: integer  
  Required: True  
  Minimum: 0  
  Maximum: 100

layer

Use Layer (Layer) to specify how overlapping inserted images appear. Images with higher values of layer appear on top of images with lower values of layer.

  Type: integer  
  Required: True  
  Minimum: 0  
  Maximum: 7

height

Specify the Height (Height) of the inserted image. Use a value that is less than or equal to the video resolution height. Leave this setting blank to use the native height of the image.

  Type: integer  
  Required: False  
  Minimum: -2147483648  
  Maximum: 2147483647

imageInserterInput

Use Image location (imageInserterInput) to specify the Amazon S3 location of the image to be inserted into the output. Use a 32 bit BMP, PNG, or TGA file that fits inside the video frame.

  Type: string  
  Required: True  
  Pattern: ^s3:\/(.*)\.(bmp|BMP|png|PNG|tga|TGA)$

JobTemplate

settings

  Type: JobTemplateSettings (p. 110)  
  Required: True

lastUpdated

The timestamp in epoch seconds when the Job template was last updated.

  Type: string  
  Required: False  
  Format: date-time
createdAt

The timestamp in epoch seconds for Job template creation.

- **Type**: string
- **Required**: False
- **Format**: date-time

name

A name you create for each job template. Each name must be unique within your account.

- **Type**: string
- **Required**: True

description

An optional description you create for each job template.

- **Type**: string
- **Required**: False

category

An optional category you create to organize your job templates.

- **Type**: string
- **Required**: False

type

- **Type**: Type (p. 149)
- **Required**: False

arn

An identifier for this resource that is unique within all of AWS.

- **Type**: string
- **Required**: False

queue

Optional. The queue that jobs created from this template are assigned to. If you don't specify this, jobs will go to the default queue.

- **Type**: string
- **Required**: False

JobTemplateListBy (Enum)

Optional. When you request a list of job templates, you can choose to list them alphabetically by NAME or chronologically by CREATION_DATE. If you don't specify, the service will list them by name.

- **NAME**
CREATION_DATE
SYSTEM

JobTemplateSettings

timecodeConfig
Contains settings used to acquire and adjust timecode information from inputs.

  Type: TimecodeConfig (p. 148)
  Required: False

adAvailOffset
When specified, this offset (in milliseconds) is added to the input Ad Avail PTS time.

  Type: integer
  Required: False
  Minimum: -1000
  Maximum: 1000

nielsenConfiguration
Nielsen configuration settings

  Type: NielsenConfiguration (p. 135)
  Required: False

inputs
Use Inputs (inputs) to define the source file used in the transcode job. There can only be one input in a job template. Using the API, you can include multiple inputs when referencing a job template.

  Type: Array of type InputTemplate (p. 104)
  Required: False

outputGroups
(OutputGroups) contains one group of settings for each set of outputs that share a common package type. All unpackaged files (MPEG-4, MPEG-2 TS, Quicktime, MXF, and no container) are grouped in a single output group as well. Required in (OutputGroups) is a group of settings that apply to the whole group. This required object depends on the value you set for (Type) under (OutputGroups)->(OutputGroupSettings). Type, settings object pairs are as follows:

* FILE_GROUP_SETTINGS, FileGroupSettings
* HLS_GROUP_SETTINGS, HlsGroupSettings
* DASH_ISO_GROUP_SETTINGS, DashIsoGroupSettings
* MS_SMOOTH_GROUP_SETTINGS, MsSmoothGroupSettings

  Type: Array of type OutputGroup (p. 139)
  Required: True
timedMetadataInsertion

Type: TimedMetadataInsertion (p. 149)
Required: False

availBlanking

Settings for ad avail blanking. Video can be blanked or overlaid with an image, and audio muted during SCTE-35 triggered ad avails.

Type: AvailBlanking (p. 50)
Required: False

LanguageCode (Enum)

Selects a specific language code from within an audio source.

ENG
SPA
FRA
DEU
GER
ZHO
ARA
HIN
JPN
RUS
POR
ITA
URD
vie
KOR
PAN
ABK
AAR
AFR
AKA
SQI
AMH
ARG
HYE
ASM
AVA
AVE
AYM
AZE
BAM
BAK
EUS
BEL
BEN
BIH
BIS
BOS
BRE
BUL
MYA
CAT
KHM
CHA
CHE
NYA
CHU
CHV
COR
COS
CRE
HRV
CES
DAN
DIV
NLD
DZ0
ENM
EPO
EST
EWE
FAO
FIJ
FIN
FRM
FUL
GLA
GLG
LUG
KAT
ELL
GRN
GUJ
HAT
HAU
HEB
HER
HMO
HUN
ISL
IDO
IBO
IND
INA
ILE
IKU
IPK
GLE
JAV
KAL
KAN
KAU
KAS
KAZ
KIK
KIN
KIR
KOM
KON
KUA
KUR
LAO
LAT
LAV
LIM
LIN
LIT
LUB
LTZ
MKD
MLG
MSA
MAL
MLT
GLV
MRI
MAR
MAH
MON
NAU
NAV
NDE
NBL
NDO
NEP
SME
NOR
NOB
NNO
OCI
OJI
ORI
ORM
OSS
PLI
FAS
POL
PUS
QUE
QAA
RON
ROH
RUN
SMO
SAG
SAN
SRD
Properties

SRB
SNA
III
SND
SIN
SLK
SLV
SOM
SOT
SUN
SWA
SSW
SWE
TGL
TAH
TGK
TAM
TAT
TEL
THA
BOD
TIR
TON
TSO
TSN
TUR
TUK
TWI
UIG
UKR
UZB
VEN
VOL
WLN
CYM
FRY
WOL
XHO
YID
YOR
ZHA
ZUL
ORJ
QPC
TNG

**ListJobTemplatesRequest**

**nextToken**

Use this string, provided with the response to a previous request, to request the next batch of job templates.

*Type*: string
Required: False

maxResults

Optional. Number of job templates, up to twenty, that will be returned at one time.

  Type: integer
  Required: False
  Format: int32

category

Optionally, specify a job template category to limit responses to only job templates from that category.

  Type: string
  Required: False

listBy

  Type: JobTemplateListBy (p. 109)
  Required: False

order

  Type: Order (p. 137)
  Required: False

ListJobTemplatesResponse

nextToken

Use this string to request the next batch of job templates.

  Type: string
  Required: False

jobTemplates

  Type: Array of type JobTemplate (p. 108)
  Required: False

M2tsAudioBufferModel (Enum)

Selects between the DVB and ATSC buffer models for Dolby Digital audio.

  DVB
  ATSC

M2tsBufferModel (Enum)

Controls what buffer model to use for accurate interleaving. If set to MULTIPLEX, use multiplex buffer model. If set to NONE, this can lead to lower latency, but low-memory devices may not be able to play back the stream without interruptions.
MULTIPLEX
NONE

**M2tsEbpAudioInterval (Enum)**

When set to VIDEO_AND_FIXED_INTERVALS, audio EBP markers will be added to partitions 3 and 4. The interval between these additional markers will be fixed, and will be slightly shorter than the video EBP marker interval. When set to VIDEO_INTERVAL, these additional markers will not be inserted. Only applicable when EBP segmentation markers are is selected (segmentationMarkers is EBP or EBP_LEGACY).

- VIDEO_AND_FIXED_INTERVALS
- VIDEO_INTERVAL

**M2tsEbpPlacement (Enum)**

Selects which PIDs to place EBP markers on. They can either be placed only on the video PID, or on both the video PID and all audio PIDs. Only applicable when EBP segmentation markers are is selected (segmentationMarkers is EBP or EBP_LEGACY).

- VIDEO_AND_AUDIO_PIDS
- VIDEO_PID

**M2tsEsRateInPes (Enum)**

Controls whether to include the ES Rate field in the PES header.

- INCLUDE
- EXCLUDE

**M2tsPcrControl (Enum)**

When set to PCR_EVERY_PES_PACKET, a Program Clock Reference value is inserted for every Packetized Elementary Stream (PES) header. This is effective only when the PCR PID is the same as the video or audio elementary stream.

- PCR_EVERY_PES_PACKET
- CONFIGURED_PCR_PERIOD

**M2tsRateMode (Enum)**

When set to CBR, inserts null packets into transport stream to fill specified bitrate. When set to VBR, the bitrate setting acts as the maximum bitrate, but the output will not be padded up to that bitrate.

- VBR
- CBR

**M2tsScte35Source (Enum)**

Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output. This is only available for certain containers.

- PASSTHROUGH
NONE

M2tsSegmentationMarkers (Enum)

Inserts segmentation markers at each segmentation_time period. rai_segstart sets the Random Access
Indicator bit in the adaptation field. rai_adapt sets the RAI bit and adds the current timecode in the
private data bytes. psi_segstart inserts PAT and PMT tables at the start of segments. ebp adds Encoder
Boundary Point information to the adaptation field as per OpenCable specification OC-SP-EBP-I01-130118. ebp_legacy adds Encoder Boundary Point information to the adaptation field using a legacy
proprietary format.

NONE
RAI_SEGSTART
RAI_ADAPT
PSI_SEGSTART
EBP
EBP_LEGACY

M2tsSegmentationStyle (Enum)

The segmentation style parameter controls how segmentation markers are inserted into the transport
stream. With avails, it is possible that segments may be truncated, which can influence where future
segmentation markers are inserted. When a segmentation style of "reset_cadence" is selected and
a segment is truncated due to an avail, we will reset the segmentation cadence. This means the
subsequent segment will have a duration of of $segmentation_time seconds. When a segmentation
style of "maintain_cadence" is selected and a segment is truncated due to an avail, we will not reset the
segmentation cadence. This means the subsequent segment will likely be truncated as well. However, all
segments after that will have a duration of $segmentation_time seconds. Note that EBP lookahead is a
slight exception to this rule.

MAINTAIN_CADENCE
RESET_CADENCE

M2tsSettings

dvbTeletextPid

Packet Identifier (PID) for input source DVB Teletext data to this output. Can be entered as a decimal or
hexadecimal value.

Type: integer
Required: False
Minimum: 32
Maximum: 8182

bitrate

The output bitrate of the transport stream in bits per second. Setting to 0 lets the muxer automatically
determine the appropriate bitrate. Other common values are 3750000, 7500000, and 15000000.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

**segmentationTime**

The length in seconds of each segment. Required unless markers is set to _none_.

- **Type**: number
- **Required**: False
- **Format**: float
- **Minimum**: 0.0

**audioPids**

Packet Identifier (PID) of the elementary audio stream(s) in the transport stream. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values.

- **Type**: Array of type integer
- **Required**: False

**rateMode**

- **Type**: M2tsRateMode (p. 116)
- **Required**: False

**ebpAudioInterval**

- **Type**: M2tsEbpAudioInterval (p. 116)
- **Required**: False

**fragmentTime**

The length in seconds of each fragment. Only used with EBP markers.

- **Type**: number
- **Required**: False
- **Format**: float
- **Minimum**: 0.0

**audioFramesPerPes**

The number of audio frames to insert for each PES packet.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 2147483647

**maxPcrInterval**

Maximum time in milliseconds between Program Clock References (PCRs) inserted into the transport stream.
Type: integer
Required: False
Minimum: 0
Maximum: 500

scte35Pid
Packet Identifier (PID) of the SCTE-35 stream in the transport stream. Can be entered as a decimal or hexadecimal value.

Type: integer
Required: False
Minimum: 32
Maximum: 8182

privateMetadataPid
Packet Identifier (PID) of the private metadata stream in the transport stream. Can be entered as a decimal or hexadecimal value.

Type: integer
Required: False
Minimum: 32
Maximum: 8182

pmtInterval
The number of milliseconds between instances of this table in the output transport stream.

Type: integer
Required: False
Minimum: 0
Maximum: 1000

segmentationStyle
Type: M2tsSegmentationStyle (p. 117)
Required: False

audioBufferModel
Type: M2tsAudioBufferModel (p. 115)
Required: False

programNumber
The value of the program number field in the Program Map Table.

Type: integer
Required: False
Minimum: 0
Maximum: 65535
**dvbNitSettings**

Type: DvbNitSettings (p. 63)
Required: False

**scte35Source**

Type: M2tsScte35Source (p. 116)
Required: False

**pmtPid**

Packet Identifier (PID) for the Program Map Table (PMT) in the transport stream. Can be entered as a decimal or hexadecimal value.

Type: integer
Required: False
Minimum: 32
Maximum: 8182

**bufferModel**

Type: M2tsBufferModel (p. 115)
Required: False

**ebpPlacement**

Type: M2tsEbpPlacement (p. 116)
Required: False

**dvbSdtSettings**

Type: DvbSdtSettings (p. 63)
Required: False

**nullPacketBitrate**

Value in bits per second of extra null packets to insert into the transport stream. This can be used if a downstream encryption system requires periodic null packets.

Type: number
Required: False
Format: float
Minimum: 0.0

**pcrPid**

Packet Identifier (PID) of the Program Clock Reference (PCR) in the transport stream. When no value is given, the encoder will assign the same value as the Video PID. Can be entered as a decimal or hexadecimal value.

Type: integer
Required: False
Minimum: 32
**Properties**

**Minimum**: 8182

**minEbpInterval**

When set, enforces that Encoder Boundary Points do not come within the specified time interval of each other by looking ahead at input video. If another EBP is going to come in within the specified time interval, the current EBP is not emitted, and the segment is "stretched" to the next marker. The lookahead value does not add latency to the system. The Live Event must be configured elsewhere to create sufficient latency to make the lookahead accurate.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 10000

**transportStreamId**

The value of the transport stream ID field in the Program Map Table.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 65535

**pcrControl**

- **Type**: M2tsPcrControl (p. 116)
- **Required**: False

**videoPid**

Packet Identifier (PID) of the elementary video stream in the transport stream. Can be entered as a decimal or hexadecimal value.

- **Type**: integer
- **Required**: False
- **Minimum**: 32
- **Maximum**: 8182

**esRateInPes**

- **Type**: M2tsEsRateInPes (p. 116)
- **Required**: False

**segmentationMarkers**

- **Type**: M2tsSegmentationMarkers (p. 117)
- **Required**: False

**dvbTdtSettings**

- **Type**: DvbTdtSettings (p. 68)
- **Required**: False
**Properties**

**patInterval**

The number of milliseconds between instances of this table in the output transport stream.

*Type*: integer  
*Required*: False  
*Minimum*: 0  
*Maximum*: 1000

**dvbSubPids**

Packet Identifier (PID) for input source DVB Subtitle data to this output. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values.

*Type*: Array of type integer  
*Required*: False

**M3u8PcrControl (Enum)**

When set to PCR_EVERY_PES_PACKET a Program Clock Reference value is inserted for every Packetized Elementary Stream (PES) header. This parameter is effective only when the PCR PID is the same as the video or audio elementary stream.

- PCR_EVERY_PES_PACKET
- CONFIGURED_PCR_PERIOD

**M3u8Scte35Source (Enum)**

Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output. This is only available for certain containers.

- PASSTHROUGH
- NONE

**M3u8Settings**

**pmtPid**

Packet Identifier (PID) for the Program Map Table (PMT) in the transport stream. Can be entered as a decimal or hexadecimal value.

*Type*: integer  
*Required*: False  
*Minimum*: 32  
*Maximum*: 8182

**pcrPid**

Packet Identifier (PID) of the Program Clock Reference (PCR) in the transport stream. When no value is given, the encoder will assign the same value as the Video PID. Can be entered as a decimal or hexadecimal value.

*Type*: integer  
*Required*: False
Minimum: 32
Maximum: 8182

**audioPids**

Packet Identifier (PID) of the elementary audio stream(s) in the transport stream. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values.

Type: Array of type integer
Required: False

**audioFramesPerPes**

The number of audio frames to insert for each PES packet.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

**scte35Pid**

Packet Identifier (PID) of the SCTE-35 stream in the transport stream. Can be entered as a decimal or hexadecimal value.

Type: integer
Required: False
Minimum: 32
Maximum: 8182

**transportStreamId**

The value of the transport stream ID field in the Program Map Table.

Type: integer
Required: False
Minimum: 0
Maximum: 65535

**videoPid**

Packet Identifier (PID) of the elementary video stream in the transport stream. Can be entered as a decimal or hexadecimal value.

Type: integer
Required: False
Minimum: 32
Maximum: 8182

**pcrControl**

Type: M3u8PcrControl (p. 122)
Required: False

123
**privateMetadataPid**

Packet Identifier (PID) of the private metadata stream in the transport stream. Can be entered as a decimal or hexadecimal value.

*Type:* integer  
*Required:* False  
*Minimum:* 32  
*Maximum:* 8182

**pmtInterval**

The number of milliseconds between instances of this table in the output transport stream.

*Type:* integer  
*Required:* False  
*Minimum:* 0  
*Maximum:* 1000

**patInterval**

The number of milliseconds between instances of this table in the output transport stream.

*Type:* integer  
*Required:* False  
*Minimum:* 0  
*Maximum:* 1000

**programNumber**

The value of the program number field in the Program Map Table.

*Type:* integer  
*Required:* False  
*Minimum:* 0  
*Maximum:* 65535

**timedMetadataPid**

Packet Identifier (PID) of the timed metadata stream in the transport stream. Can be entered as a decimal or hexadecimal value.

*Type:* integer  
*Required:* False  
*Minimum:* 32  
*Maximum:* 8182

**timedMetadata**

*Type:* TimedMetadata (p. 149)  
*Required:* False

**scte35Source**

*Type:* M3u8Scte35Source (p. 122)
**Required:** False

**MovClapAtom (Enum)**

When enabled, include 'clap' atom if appropriate for the video output settings.

- **INCLUDE**
- **EXCLUDE**

**MovCslgAtom (Enum)**

When enabled, file composition times will start at zero, composition times in the 'ctts' (composition time to sample) box for B-frames will be negative, and a 'cslg' (composition shift least greatest) box will be included per 14496-1 amendment 1. This improves compatibility with Apple players and tools.

- **INCLUDE**
- **EXCLUDE**

**MovMpeg2FourCCControl (Enum)**

When set to XDCAM, writes MPEG2 video streams into the QuickTime file using XDCAM fourcc codes. This increases compatibility with Apple editors and players, but may decrease compatibility with other players. Only applicable when the video codec is MPEG2.

- **XDCAM**
- **MPEG**

**MovPaddingControl (Enum)**

If set to OMNEON, inserts Omneon-compatible padding

- **OMNEON**
- **NONE**

**MovReference (Enum)**

A value of 'external' creates separate media files and the wrapper file (.mov) contains references to these media files. A value of 'self_contained' creates only a wrapper (.mov) file and this file contains all of the media.

- **SELF_CONTAINED**
- **EXTERNAL**

**MovSettings**

**reference**

- **Type:** MovReference (p. 125)
- **Required:** False

**paddingControl**

- **Type:** MovPaddingControl (p. 125)
Properties

Required: False

mpeg2FourCCControl

Type: MovMpeg2FourCCControl (p. 125)
Required: False

cslgAtom

Type: MovCslgAtom (p. 125)
Required: False

clapAtom

Type: MovClapAtom (p. 125)
Required: False

Mp2Settings

channels

Set Channels to specify the number of channels in this output audio track. Choosing Mono in the console will give you 1 output channel; choosing Stereo will give you 2. In the API, valid values are 1 and 2.

Type: integer
Required: False
Minimum: 1
Maximum: 2

bitrate

Average bitrate in bits/second.

Type: integer
Required: False
Minimum: 32000
Maximum: 384000

sampleRate

Sample rate in hz.

Type: integer
Required: False
Minimum: 32000
Maximum: 48000

Mp4CslgAtom (Enum)

When enabled, file composition times will start at zero, composition times in the 'ctts' (composition time to sample) box for B-frames will be negative, and a 'cslg' (composition shift least greatest) box will be included per 14496-1 amendment 1. This improves compatibility with Apple players and tools.
**Epilogue**

**Mp4FreeSpaceBox (Enum)**

Inserts a free-space box immediately after the moov box.

**Mp4MoovPlacement (Enum)**

If set to PROGRESSIVE_DOWNLOAD, the MOOV atom is relocated to the beginning of the archive as required for progressive downloading. Otherwise it is placed normally at the end.

**Mp4Settings**

**mp4MajorBrand**

Overrides the "Major Brand" field in the output file. Usually not necessary to specify.

**Type**: string  
**Required**: False

**moovPlacement**

**Type**: Mp4MoovPlacement (p. 127)  
**Required**: False

**cslgAtom**

**Type**: Mp4CslgAtom (p. 126)  
**Required**: False

**freeSpaceBox**

**Type**: Mp4FreeSpaceBox (p. 127)  
**Required**: False

**Mpeg2AdaptiveQuantization (Enum)**

Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

**OFF**  
**LOW**  
**MEDIUM**  
**HIGH**

**Mpeg2CodecLevel (Enum)**

Use Level (Mpeg2CodecLevel) to set the MPEG-2 level for the video output.
Mpeg2CodecProfile (Enum)

Use Profile (Mpeg2CodecProfile) to set the MPEG-2 profile for the video output.

- MAIN
- PROFILE_422

Mpeg2FramerateControl (Enum)

Using the API, set FramerateControl to INITIALIZE_FROM_SOURCE if you want the service to use the framerate from the input. Using the console, do this by choosing INITIALIZE_FROM_SOURCE for Framerate.

- INITIALIZE_FROM_SOURCE
- SPECIFIED

Mpeg2FramerateConversionAlgorithm (Enum)

When set to INTERPOLATE, produces smoother motion during framerate conversion.

- DUPLICATE_DROP
- INTERPOLATE

Mpeg2GopSizeUnits (Enum)

Indicates if the GOP Size in MPEG2 is specified in frames or seconds. If seconds the system will convert the GOP Size into a frame count at run time.

- FRAMES
- SECONDS

Mpeg2InterlaceMode (Enum)

Use Interlace mode (InterlaceMode) to choose the scan line type for the output.

* Top Field First (TOP_FIELD) and Bottom Field First (BOTTOM_FIELD) produce interlaced output with the entire output having the same field polarity (top or bottom first).

* Follow, Default Top (FOLLOW_TOP_FIELD) and Follow, Default Bottom (FOLLOW_BOTTOM_FIELD) use the same field polarity as the source. Therefore, behavior depends on the input scan type. If the source is interlaced, the output will be interlaced with the same polarity as the source (it will follow the source). The output could therefore be a mix of "top field first" and "bottom field first". If the source is progressive, the output will be interlaced with "top field first" or "bottom field first" polarity, depending on which of the Follow options you chose.

- PROGRESSIVE
- TOP_FIELD
- BOTTOM_FIELD
Mpeg2IntraDcPrecision (Enum)

Use Intra DC precision (Mpeg2IntraDcPrecision) to set quantization precision for intra-block DC coefficients. If you choose the value auto, the service will automatically select the precision based on the per-frame compression ratio.

AUTO
INTRA_DC_PRECISION_8
INTRA_DC_PRECISION_9
INTRA_DC_PRECISION_10
INTRA_DC_PRECISION_11

Mpeg2ParControl (Enum)

Using the API, enable ParFollowSource if you want the service to use the pixel aspect ratio from the input. Using the console, do this by choosing Follow source for Pixel aspect ratio.

INITIALIZE_FROM_SOURCE
SPECIFIED

Mpeg2QualityTuningLevel (Enum)

Use Quality tuning level (Mpeg2QualityTuningLevel) to specify whether to use single-pass or multipass video encoding.

SINGLE_PASS
MULTI_PASS

Mpeg2RateControlMode (Enum)

Use Rate control mode (Mpeg2RateControlMode) to specify whether the bitrate is variable (vbr) or constant (cbr).

VBR
CBR

Mpeg2SceneChangeDetect (Enum)

Scene change detection (inserts I-frames on scene changes).

DISABLED
ENABLED

Mpeg2Settings

minIInterval

Enforces separation between repeated (cadence) I-frames and I-frames inserted by Scene Change Detection. If a scene change I-frame is within I-interval frames of a cadence I-frame, the GOP is shrunk and/or stretched to the scene change I-frame. GOP stretch requires enabling lookahead as well as
setting I-interval. The normal cadence resumes for the next GOP. This setting is only used when Scene Change Detect is enabled. Note: Maximum GOP stretch = GOP size + Min-I-interval - 1

**Type:** integer
**Required:** False
**Minimum:** 0
**Maximum:** 30

**parNumerator**
Pixel Aspect Ratio numerator.

**Type:** integer
**Required:** False
**Minimum:** 1
**Maximum:** 2147483647

**gopSizeUnits**

**Type:** Mpeg2GopSizeUnits (p. 128)
**Required:** False

**hrdBufferSzie**
Size of buffer (HRD buffer model). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.

**Type:** integer
**Required:** False
**Minimum:** -2147483648
**Maximum:** 2147483647

**qualityTuningLevel**

**Type:** Mpeg2QualityTuningLevel (p. 129)
**Required:** False

**maxBitrate**
Maximum bitrate in bits/second (for VBR mode only). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.

**Type:** integer
**Required:** False
**Minimum:** -2147483648
**Maximum:** 2147483647

**bitrate**
Average bitrate in bits/second. Required for VBR, CBR, and ABR. Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.

**Type:** integer
Required: False
Minimum: 1000
Maximum: 2147483647

spatialAdaptiveQuantization

Type: Mpeg2SpatialAdaptiveQuantization (p. 134)
Required: False

slowPal

Type: Mpeg2SlowPal (p. 133)
Required: False

codecProfile

Type: Mpeg2CodecProfile (p. 128)
Required: False

intraDcPrecision

Type: Mpeg2IntraDcPrecision (p. 129)
Required: False

softness

Softness. Selects quantizer matrix, larger values reduce high-frequency content in the encoded image.

Type: integer
Required: False
Minimum: 0
Maximum: 128

framerateControl

Type: Mpeg2FramerateControl (p. 128)
Required: False

telecine

Type: Mpeg2Telecine (p. 134)
Required: False

framerateConversionAlgorithm

Type: Mpeg2FramerateConversionAlgorithm (p. 128)
Required: False

codecLevel

Type: Mpeg2CodecLevel (p. 127)
Required: False
temporalAdaptiveQuantization

Type: Mpeg2TemporalAdaptiveQuantization (p. 134)
Required: False

hrdBufferInitialFillPercentage

Percentage of the buffer that should initially be filled (HRD buffer model).

Type: integer
Required: False
Minimum: 0
Maximum: 100

framerateNumerator

Framerate numerator - framerate is a fraction, e.g. 24000 / 1001 = 23.976 fps.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

numberBFramesBetweenReferenceFrames

Number of B-frames between reference frames.

Type: integer
Required: False
Minimum: 0
Maximum: 7

gopClosedCadence

Frequency of closed GOPs. In streaming applications, it is recommended that this be set to 1 so a decoder joining mid-stream will receive an IDR frame as quickly as possible. Setting this value to 0 will break output segmenting.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

framerateDenominator

Framerate denominator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

adaptiveQuantization

Type: Mpeg2AdaptiveQuantization (p. 127)
Properties

**interlaceMode**

- **Type**: Mpeg2InterlaceMode (p. 128)
- **Required**: False

**gopSize**

GOP Length (keyframe interval) in frames or seconds. Must be greater than zero.

- **Type**: number
- **Required**: False
- **Format**: float
- **Minimum**: 0.0

**sceneChangeDetect**

- **Type**: Mpeg2SceneChangeDetect (p. 129)
- **Required**: False

**parDenominator**

Pixel Aspect Ratio denominator.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 2147483647

**parControl**

- **Type**: Mpeg2ParControl (p. 129)
- **Required**: False

**syntax**

- **Type**: Mpeg2Syntax (p. 134)
- **Required**: False

**rateControlMode**

- **Type**: Mpeg2RateControlMode (p. 129)
- **Required**: False

**Mpeg2SlowPal (Enum)**

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

- **DISABLED**
- **ENABLED**
**Mpeg2SpatialAdaptiveQuantization (Enum)**

Adjust quantization within each frame based on spatial variation of content complexity.

- DISABLED
- ENABLED

**Mpeg2Syntax (Enum)**

Produces a Type D-10 compatible bitstream (SMPTE 356M-2001).

- DEFAULT
- D_10

**Mpeg2Telecine (Enum)**

Only use Telecine (Mpeg2Telecine) when you set Framerate (Framerate) to 29.970. Set Telecine (Mpeg2Telecine) to Hard (hard) to produce a 29.97i output from a 23.976 input. Set it to Soft (soft) to produce 23.976 output and leave conversion to the player.

- NONE
- SOFT
- HARD

**Mpeg2TemporalAdaptiveQuantization (Enum)**

Adjust quantization within each frame based on temporal variation of content complexity.

- DISABLED
- ENABLED

**MsSmoothAudioDeduplication (Enum)**

COMBINE_DUPLICATE_STREAMS combines identical audio encoding settings across a Microsoft Smooth output group into a single audio stream.

- COMBINE_DUPLICATE_STREAMS
- NONE

**MsSmoothEncryptionSettings**

**spekeKeyProvider**

- Type: SpekeKeyProvider (p. 145)
- Required: True

**MsSmoothGroupSettings**

**fragmentLength**

Use Fragment length (FragmentLength) to specify the mp4 fragment sizes in seconds. Fragment length must be compatible with GOP size and framerate.

- Type: integer
Properties

encryption

Type: MsSmoothEncryptionSettings (p. 134)
Required: False

audioDeduplication

Type: MsSmoothAudioDeduplication (p. 134)
Required: False

manifestEncoding

Type: MsSmoothManifestEncoding (p. 135)
Required: False

destination

Use Destination (Destination) to specify the S3 output location and the output filename base. Destination accepts format identifiers. If you do not specify the base filename in the URI, the service will use the filename of the input file. If your job has multiple inputs, the service uses the filename of the first input file.

Type: string
Required: True
Pattern: ^s3:\/\/

MsSmoothManifestEncoding (Enum)

Use Manifest encoding (MsSmoothManifestEncoding) to specify the encoding format for the server and client manifest. Valid options are utf8 and utf16.

UTF8
UTF16

NielsenConfiguration

distributorId

Use Distributor ID (DistributorID) to specify the distributor ID that is assigned to your organization by Neilsen.

Type: string
Required: False

breakoutCode

Use Nielsen Configuration (NielsenConfiguration) to set the Nielsen measurement system breakout code. Supported values are 0, 3, 7, and 9.
Properties

**Type:** integer
**Required:** False
**Minimum:** 0
**Maximum:** 9

**NoiseReducer**

**filter**

**Type:** NoiseReducerFilter (p. 136)
**Required:** True

**filterSettings**

**Type:** NoiseReducerFilterSettings (p. 136)
**Required:** False

**spatialFilterSettings**

**Type:** NoiseReducerSpatialFilterSettings (p. 137)
**Required:** False

**NoiseReducerFilter (Enum)**

Use Noise reducer filter (NoiseReducerFilter) to select one of the following spatial image filtering functions. To use this setting, you must also enable Noise reducer (NoiseReducer).

* Bilateral is an edge preserving noise reduction filter
* Mean (softest), Gaussian, Lanczos, and Sharpen (sharpest) are convolution filters
* Conserve is a min/max noise reduction filter
* Spatial is frequency-domain filter based on JND principles.

BILATERAL
MEAN
GAUSSIAN
LANCZOS
SHARPEN
CONSERVE
SPATIAL

**NoiseReducerFilterSettings**

**strength**

Relative strength of noise reducing filter. Higher values produce stronger filtering.

**Type:** integer
**Required:** False
**Minimum:** 0
**Maximum:** 3
NoiseReducerSpatialFilterSettings

strength
Relative strength of noise reducing filter. Higher values produce stronger filtering.

Type: integer
Required: False
Minimum: 0
Maximum: 16

postFilterSharpenStrength
Specify strength of post noise reduction sharpening filter, with 0 disabling the filter and 3 enabling it at maximum strength.

Type: integer
Required: False
Minimum: 0
Maximum: 3

speed
The speed of the filter, from -2 (lower speed) to 3 (higher speed), with 0 being the nominal value.

Type: integer
Required: False
Minimum: -2
Maximum: 3

Order (Enum)
When you request lists of resources, you can optionally specify whether they are sorted in ASCENDING or DESCENDING order. Default varies by resource.

ASCENDING
DESCENDING

Output

extension
Use Extension (Extension) to specify the file extension for outputs in File output groups. If you do not specify a value, the service will use default extensions by container type as follows:

* MPEG-2 transport stream, m2ts
* Quicktime, mov
* MXF container, mxf
* MPEG-4 container, mp4
* No Container, the service will use codec extensions (e.g. AAC, H265, H265, AC3)
videoDescription

(VideoDescription) contains a group of video encoding settings. The specific video settings depend on the video codec you choose when you specify a value for Video codec (codec). Include one instance of (VideoDescription) per output.

Type: VideoDescription (p. 150)
Required: False

audioDescriptions

(AudioDescriptions) contains groups of audio encoding settings organized by audio codec. Include one instance of (AudioDescriptions) per output. (AudioDescriptions) can contain multiple groups of encoding settings.

Type: Array of type AudioDescription (p. 45)
Required: False

captionDescriptions

(CaptionDescriptions) contains groups of captions settings. For each output that has captions, include one instance of (CaptionDescriptions). (CaptionDescriptions) can contain multiple groups of captions settings.

Type: Array of type CaptionDescription (p. 54)
Required: False

nameModifier

Use Name modifier (NameModifier) to have the service add a string to the end of each output filename. You specify the base filename as part of your destination URI. When you create multiple outputs in the same output group, Name modifier is required. Name modifier also accepts format identifiers. For DASH
ISO outputs, if you use the format identifiers $Number$ or $Time$ in one output, you must use them in the same way in all outputs of the output group.

- **Type**: string
- **Required**: False

**OutputChannelMapping**

**inputChannels**

- **Type**: Array of type integer
- **Required**: True

**OutputGroup**

**outputs**

This object holds groups of encoding settings, one group of settings per output.

- **Type**: Array of type **Output** (p. 137)
- **Required**: True

**outputGroupSettings**

- **Type**: **OutputGroupSettings** (p. 139)
- **Required**: True

**name**

- **Type**: string
- **Required**: False

**customName**

Use Custom Group Name (CustomName) to specify a name for the output group. This value is displayed on the console and can make your job settings JSON more human-readable. It does not affect your outputs. Use up to twelve characters that are either letters, numbers, spaces, or underscores.

- **Type**: string
- **Required**: False

**OutputGroupSettings**

**dashIsoGroupSettings**

- **Type**: **DashIsoGroupSettings** (p. 60)
- **Required**: False

**fileGroupSettings**

- **Type**: **FileGroupSettings** (p. 74)
Required: False

**msSmoothGroupSettings**

Type: MsSmoothGroupSettings (p. 134)

Required: False

**type**

Type: OutputGroupType (p. 140)

Required: True

**hlsGroupSettings**

Type: HlsGroupSettings (p. 98)

Required: False

**OutputGroupType (Enum)**

HLS_GROUP_SETTINGS
DASH_ISO_GROUP_SETTINGS
FILE_GROUP_SETTINGS
MS_SMOOTH_GROUP_SETTINGS

**OutputSdt (Enum)**

Selects method of inserting SDT information into output stream. "Follow input SDT" copies SDT information from input stream to output stream. "Follow input SDT if present" copies SDT information from input stream to output stream if SDT information is present in the input, otherwise it will fall back on the user-defined values. Enter "SDT Manually" means user will enter the SDT information. "No SDT" means output stream will not contain SDT information.

SDT_FOLLOW
SDT_FOLLOW_IF_PRESENT
SDT_MANUAL
SDT_NONE

**OutputSettings**

**hlsSettings**

Type: HlsSettings (p. 102)

Required: False

**ProresCodecProfile (Enum)**

Use Profile (ProResCodecProfile) to specify the type of Apple ProRes codec to use for this output.

APPLE_PRORES_422
APPLE_PRORES_422_HQ
APPLE_PRORES_422_LT
APPLE_PRORES_422_PROXY
ProresFramerateControl (Enum)

Using the API, set FramerateControl to INITIALIZE_FROM_SOURCE if you want the service to use the framerate from the input. Using the console, do this by choosing INITIALIZE_FROM_SOURCE for Framerate.

    INITIALIZE_FROM_SOURCE
    SPECIFIED

ProresFramerateConversionAlgorithm (Enum)

When set to INTERPOLATE, produces smoother motion during framerate conversion.

    DUPLICATE_DROP
    INTERPOLATE

ProresInterlaceMode (Enum)

Use Interlace mode (InterlaceMode) to choose the scan line type for the output.

* Top Field First (TOP_FIELD) and Bottom Field First (BOTTOM_FIELD) produce interlaced output with the entire output having the same field polarity (top or bottom first).

* Follow, Default Top (FOLLOW_TOP_FIELD) and Follow, Default Bottom (FOLLOW_BOTTOM_FIELD) use the same field polarity as the source. Therefore, behavior depends on the input scan type. If the source is interlaced, the output will be interlaced with the same polarity as the source (it will follow the source). The output could therefore be a mix of "top field first" and "bottom field first". If the source is progressive, the output will be interlaced with "top field first" or "bottom field first" polarity, depending on which of the Follow options you chose.

    PROGRESSIVE
    TOP_FIELD
    BOTTOM_FIELD
    FOLLOW_TOP_FIELD
    FOLLOW_BOTTOM_FIELD

ProresParControl (Enum)

Use (ProresParControl) to specify how the service determines the pixel aspect ratio. Set to Follow source (INITIALIZE_FROM_SOURCE) to use the pixel aspect ratio from the input. To specify a different pixel aspect ratio: Using the console, choose it from the dropdown menu. Using the API, set ProresParControl to (SPECIFIED) and provide for (ParNumerator) and (ParDenominator).

    INITIALIZE_FROM_SOURCE
    SPECIFIED

ProresSettings

slowPal

    Type: ProresSlowPal (p. 143)
    Required: False

framerateControl

    Type: ProresFramerateControl (p. 141)
telecine

Type: ProresTelecine
Required: False

framerateDenominator

Framerate denominator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

framerateConversionAlgorithm

Type: ProresFramerateConversionAlgorithm
Required: False

interlaceMode

Type: ProresInterlaceMode
Required: False

codecProfile

Type: ProresCodecProfile
Required: False

parNumerator

Pixel Aspect Ratio numerator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

parControl

Type: ProresParControl
Required: False

parDenominator

Pixel Aspect Ratio denominator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

framerateNumerator

When you use the API for transcoding jobs that use framerate conversion, specify the framerate as a fraction. For example, 24000 / 1001 = 23.976 fps. Use FramerateNumerator to specify the numerator of this fraction. In this example, use 24000 for the value of FramerateNumerator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

ProresSlowPal (Enum)

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

DISABLED
ENABLED

ProresTelecine (Enum)

Only use Telecine (ProresTelecine) when you set Framerate (Framerate) to 29.970. Set Telecine (ProresTelecine) to Hard (hard) to produce a 29.97i output from a 23.976 input. Set it to Soft (soft) to produce 23.976 output and leave conversion to the player.

NONE
HARD

Rectangle

width

Width of rectangle in pixels.

Type: integer
Required: True
Minimum: -2147483648
Maximum: 2147483647

x

The distance, in pixels, between the rectangle and the left edge of the video frame.

Type: integer
Required: True
Minimum: -2147483648
Maximum: 2147483647

y

The distance, in pixels, between the rectangle and the top edge of the video frame.
Type: integer  
Required: True  
Minimum: -2147483648  
Maximum: 2147483647

height

Height of rectangle in pixels.

Type: integer  
Required: True  
Minimum: -2147483648  
Maximum: 2147483647

RemixSettings

channelsOut

Specify the number of channels in this output after remixing. Valid values: 1, 2, 4, 6, 8

Type: integer  
Required: True  
Minimum: 1  
Maximum: 8

channelMapping

Type: ChannelMapping (p. 56)  
Required: True

channelsIn

Specify the number of audio channels from your input that you want to use in your output. With remixing, you might combine or split the data in these channels, so the number of channels in your final output might be different.

Type: integer  
Required: True  
Minimum: 1  
Maximum: 16

RespondToAfd (Enum)

Use Respond to AFD (RespondToAfd) to specify how the service changes the video itself in response to AFD values in the input.

* Choose Respond to clip the input video frame according to the AFD value, input display aspect ratio, and output display aspect ratio.

* Choose Passthrough to include the input AFD values. Do not choose this when AfdSignaling is set to (NONE). A preferred implementation of this workflow is to set RespondToAfd to (NONE) and set AfdSignaling to (AUTO).
* Choose None to remove all input AFD values from this output.

NONE
RESPOND
PASSTHROUGH

**ScalingBehavior (Enum)**

Applies only if your input aspect ratio is different from your output aspect ratio. Enable Stretch to output (StretchToOutput) to have the service stretch your video image to fit. Leave this setting disabled to allow the service to letterbox your video instead. This setting overrides any positioning value you specify elsewhere in the job.

DEFAULT
STRETCH_TO_OUTPUT

**SccDestinationFramerate (Enum)**

Set Framerate (SccDestinationFramerate) to make sure that the captions and the video are synchronized in the output. Specify a framerate that matches the framerate of the associated video. If the video framerate is 29.97, choose 29.97 dropframe (FRAMERATE_29_97_DROPFRAME) only if the video has video_insertion=true and drop_frame_timecode=true; otherwise, choose 29.97 non-dropframe (FRAMERATE_29_97_NON_DROPFRAME).

FRAMERATE_23_97
FRAMERATE_24
FRAMERATE_29_97_DROPFRAME
FRAMERATE_29_97_NON_DROPFRAME

**SccDestinationSettings**

framerate

**Type:** SccDestinationFramerate (p. 145)

**Required:** False

**SpekeKeyProvider**

**resourceId**

The SPEKE-compliant server uses Resource ID (ResourceId) to identify content.

**Type:** string

**Required:** True

**Pattern:** ^[\w-]+$**

**systemIds**

Relates to SPEKE implementation. DRM system identifiers. DASH output groups support a max of two system ids. Other group types support one system id.

**Type:** Array of type string

**Required:** True
url

Use URL (Url) to specify the SPEKE-compliant server that will provide keys for content.

Type: string
Required: True
Format: uri
Pattern: ^https:/\/

StaticKeyProvider

keyFormatVersions

Relates to DRM implementation. Either a single positive integer version value or a slash delimited list of version values (1/2/3).

Type: string
Required: False
Pattern: ^\d+(\/\d+)*$

keyFormat

Relates to DRM implementation. Sets the value of the KEYFORMAT attribute. Must be 'identity' or a reverse DNS string. May be omitted to indicate an implicit value of 'identity'.

Type: string
Required: False
Pattern: ^\{identity|[A-Za-z]{2,6}(\.[A-Za-z0-9-]{1,63})\{1,63\}\}$

staticKeyValue

Relates to DRM implementation. Use a 32-character hexadecimal string to specify Key Value (StaticKeyValue).

Type: string
Required: True
Pattern: ^[A-Za-z0-9]{32}$

url

Relates to DRM implementation. The location of the license server used for protecting content.

Type: string
Required: True
Format: uri

TeletextDestinationSettings

pageNumber

Set pageNumber to the Teletext page number for the destination captions for this output. This value must be a three-digit hexadecimal string; strings ending in -FF are invalid. If you are passing through the entire set of Teletext data, do not use this field.
Properties

**TeletextSourceSettings**

**pageNumber**

Use Page Number (PageNumber) to specify the three-digit hexadecimal page number that will be used for Teletext captions. Do not use this setting if you are passing through teletext from the input source to output.

- **Type:** string
- **Required:** False
- **Pattern:** `^[1-8][0-9a-fA-F][0-9a-eA-E]$`

**TimecodeBurnin**

**prefix**

Use Prefix (Prefix) to place ASCII characters before any burned-in timecode. For example, a prefix of "EZ-" will result in the timecode "EZ-00:00:00:00". Provide either the characters themselves or the ASCII code equivalents. The supported range of characters is 0x20 through 0x7e. This includes letters, numbers, and all special characters represented on a standard English keyboard.

- **Type:** string
- **Required:** False
- **Pattern:** `^[ -~]+$`

**fontSize**

Use Font Size (FontSize) to set the font size of any burned-in timecode. Valid values are 10, 16, 32, 48.

- **Type:** integer
- **Required:** False
- **Minimum:** 10
- **Maximum:** 48

**position**

- **Type:** `TimecodeBurninPosition (p. 147)`
- **Required:** False

**TimecodeBurninPosition (Enum)**

Use Position (Position) under Timecode burn-in (TimecodeBurnIn) to specify the location the burned-in timecode on output video.

- `TOP_CENTER`
- `TOP_LEFT`
- `TOP_RIGHT`
- `MIDDLE_LEFT`
**TimecodeConfig**

**timestampOffset**

Only applies to outputs that support program-date-time stamp. Use TimestampOffset to overwrite the timecode date without affecting the time and frame number. To use this, you must also enable Insert program-date-time (InsertProgramDateTime) in the output settings.

- **Type**: string
- **Required**: False
- **Pattern**: `^([0-9]{4})-(0[1-9]|1[0-2])-(0[1-9]|12)(0-9)|3(01)$`

**anchor**

If you use an editing platform that relies on an anchor timecode, use Anchor Timecode (Anchor) to specify a timecode that will match the input video frame to the output video frame. Use 24-hour format with frame number, (HH:MM:SS:FF) or (HH:MM:SS;FF). This setting ignores framerate conversion. System behavior for Anchor Timecode varies depending on your setting for Timecode source (TimecodeSource).

* If Timecode source (TimecodeSource) is set to Specified Start (specifiedstart), the first input frame is the specified value in Start Timecode (Start). Anchor Timecode (Anchor) and Start Timecode (Start) are used to calculate output timecode.

* If Timecode source (TimecodeSource) is set to Start at 0 (zerobased) the first frame is 00:00:00:00.

* If Timecode source (TimecodeSource) is set to Embedded (embedded), the first frame is the timecode value on the first input frame of the input.

- **Type**: string
- **Required**: False
- **Format**: timecode
- **Pattern**: `^([01][0-9]|2[0-4]):[0-5][0-9]:[0-5][0-9]:[0-9]{2}$`

**start**

Only use when you set Timecode Source (TimecodeSource) to Specified Start (SPECIFIEDSTART). Use Start timecode (Start) to specify the timecode for the initial frame. Use 24-hour format with frame number, (HH:MM:SS:FF) or (HH:MM:SS;FF).

- **Type**: string
- **Required**: False
- **Format**: timecode
- **Pattern**: `^([01][0-9]|2[0-4]):[0-5][0-9]:[0-5][0-9]:[0-9]{2}$`

**source**

- **Type**: TimecodeSource (p. 149)
- **Required**: False
**TimecodeSource (Enum)**

Use Timecode source (TimecodeSource) to set how timecodes are handled within this input. To make sure that your video, audio, captions, and markers are synchronized and that time-based features, such as image inserter, work correctly, choose the Timecode source option that matches your assets. All timecodes are in a 24-hour format with frame number (HH:MM:SS:FF).

* Embedded (EMBEDDED) - Use the timecode that is in the input video. If no embedded timecode is in the source, the service will use Start at 0 (ZEROBASED) instead.

* Start at 0 (ZEROBASED) - Set the timecode of the initial frame to 00:00:00:00.

* Specified Start (SPECIFIEDSTART) - Set the timecode of the initial frame to a value other than zero. You use Start timecode (Start) to provide this value.

  EMBEDDED
  ZEROBASED
  SPECIFIEDSTART

**TimedMetadata (Enum)**

If PASSTHROUGH, inserts ID3 timed metadata from the timed_metadata REST command into this output. Only available for certain containers.

  PASSTHROUGH
  NONE

**TimedMetadataInsertion**

**id3Insertions**

Id3Insertions contains the array of Id3Insertion instances.

- **Type:** Array of type [Id3Insertion](#)
- **Required:** True

**TtmlDestinationSettings**

**stylePassthrough**

- **Type:** [TtmlStylePassthrough](#)
- **Required:** False

**TtmlStylePassthrough (Enum)**

Pass through style and position information from a TTML-like input source (TTML, SMPTE-TT, CFF-TT) to the CFF-TT output or TTML output.

  ENABLED
  DISABLED

**Type (Enum)**

  SYSTEM
  CUSTOM
VideoCodec (Enum)

Type of video codec

- FRAME_CAPTURE
- H_264
- H_265
- MPEG2
- PRORES

VideoCodecSettings

h265Settings

- Type: H265Settings (p. 87)
- Required: False

codec

- Type: VideoCodec (p. 150)
- Required: True

proresSettings

- Type: ProresSettings (p. 141)
- Required: False

mpeg2Settings

- Type: Mpeg2Settings (p. 129)
- Required: False

h264Settings

- Type: H264Settings (p. 79)
- Required: False

frameCaptureSettings

- Type: FrameCaptureSettings (p. 75)
- Required: False

VideoDescription

fixedAfd

Applies only if you set AFD Signaling(AfdSignaling) to Fixed (FIXED). Use Fixed (FixedAfd) to specify a four-bit AFD value which the service will write on all frames of this video output.

- Type: integer
- Required: False
- Minimum: 0
Maximum: 15

scalingBehavior

Type: ScalingBehavior (p. 145)
Required: False

respondToAfd

Type: RespondToAfd (p. 144)
Required: False

codecSettings

Type: VideoCodecSettings (p. 150)
Required: True

afdSignaling

Type: AfdSignaling (p. 42)
Required: False

colorMetadata

Type: ColorMetadata (p. 57)
Required: False

timecodeInsertion

Type: VideoTimecodeInsertion (p. 154)
Required: False

width

Use Width (Width) to define the video resolution width, in pixels, for this output. If you don't provide a value here, the service will use the input width.

Type: integer
Required: False
Minimum: 32
Maximum: 4096

sharpness

Use Sharpness (Sharpness) setting to specify the strength of anti-aliasing. This setting changes the width of the anti-alias filter kernel used for scaling. Sharpness only applies if your output resolution is different from your input resolution, and if you set Anti-alias (AntiAlias) to ENABLED. 0 is the softest setting, 100 the sharpest, and 50 recommended for most content.

Type: integer
Required: False
Minimum: 0
Maximum: 100

**antiAlias**

_Type:_ AntiAlias (p. 43)  
_Required:_ False

**videoPreprocessors**

Find additional transcoding features under Preprocessors (VideoPreprocessors). Enable the features at each output individually. These features are disabled by default.

_Type:_ VideoPreprocessor (p. 152)  
_Required:_ False

**position**

Use Position (Position) to point to a rectangle object to define your position. This setting overrides any other aspect ratio.

_Type:_ Rectangle (p. 143)  
_Required:_ False

**dropFrameTimecode**

_Type:_ DropFrameTimecode (p. 62)  
_Required:_ False

**crop**

Applies only if your input aspect ratio is different from your output aspect ratio. Use Input cropping rectangle (Crop) to specify the video area the service will include in the output. This will crop the input source, causing video pixels to be removed on encode. Do not use this setting if you have enabled Stretch to output (stretchToOutput) in your output settings.

_Type:_ Rectangle (p. 143)  
_Required:_ False

**height**

Use the Height (Height) setting to define the video resolution height for this output. Specify in pixels. If you don’t provide a value here, the service will use the input height.

_Type:_ integer  
_Required:_ False  
_Minimum:_ 32  
_Maximum:_ 2160

**VideoPreprocessor**

**timecodeBurnin**

Timecode burn-in (TimecodeBurnIn)—Burns the output timecode and specified prefix into the output.
Properties

- **Type**: TimecodeBurnin (p. 147)
  - **Required**: False

- **noiseReducer**
  Enable the Noise reducer (NoiseReducer) feature to remove noise from your video output if necessary. Enable or disable this feature for each output individually. This setting is disabled by default.
  - **Type**: NoiseReducer (p. 136)
  - **Required**: False

- **colorCorrector**
  Enable the Color corrector (ColorCorrector) feature if necessary. Enable or disable this feature for each output individually. This setting is disabled by default.
  - **Type**: ColorCorrector (p. 56)
  - **Required**: False

- **imageInserter**
  Enable the Image inserter (ImageInserter) feature to include a graphic overlay on your video. Enable or disable this feature for each output individually. This setting is disabled by default.
  - **Type**: ImageInserter (p. 103)
  - **Required**: False

- **deinterlacer**
  Use Deinterlacer (Deinterlacer) to produce smoother motion and a clearer picture.
  - **Type**: Deinterlacer (p. 62)
  - **Required**: False

**VideoSelector**

- **colorSpace**
  - **Type**: ColorSpace (p. 57)
  - **Required**: False

- **hdr10Metadata**
  - **Type**: Hdr10Metadata (p. 93)
  - **Required**: False

- **programNumber**
  Selects a specific program from within a multi-program transport stream. Note that Quad 4K is not currently supported.
  - **Type**: integer
  - **Required**: False
Minimum: -2147483648  
Maximum: 2147483647

**pid**

Use PID (Pid) to select specific video data from an input file. Specify this value as an integer; the system automatically converts it to the hexadecimal value. For example, 257 selects PID 0x101. A PID, or packet identifier, is an identifier for a set of data in an MPEG-2 transport stream container.

**Type:** integer  
**Required:** False  
**Minimum:** 1  
**Maximum:** 2147483647

**colorSpaceUsage**

**Type:** ColorSpaceUsage (p. 58)  
**Required:** False

**VideoTimecodeInsertion (Enum)**

Enable Timecode insertion to include timecode information in this output. Do this in the API by setting (VideoTimecodeInsertion) to (PIC_TIMING_SEI). To get timecodes to appear correctly in your output, also set up the timecode configuration for your job in the input settings. Only enable Timecode insertion when the input framerate is identical to output framerate. Disable this setting to remove the timecode from the output. Default is disabled.

**DISABLED**  
**PIC_TIMING_SEI**

**WavSettings**

**channels**

Set Channels to specify the number of channels in this output audio track. With WAV, valid values 1, 2, 4, and 8. In the console, these values are Mono, Stereo, 4-Channel, and 8-Channel, respectively.

**Type:** integer  
**Required:** False  
**Minimum:** 1  
**Maximum:** 8

**bitDepth**

Specify Bit depth (BitDepth), in bits per sample, to choose the encoding quality for this audio track.

**Type:** integer  
**Required:** False  
**Minimum:** 16  
**Maximum:** 24

**sampleRate**

Sample rate in Hz.
A Specific Job Template

URI

/jobTemplates/name

HTTP Methods

GET

Operation ID: GetJobTemplate

Retrieve the JSON for a specific job template.

Path Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>True</td>
<td></td>
</tr>
</tbody>
</table>

Responses

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>GetJobTemplateResponse (p. 169)</td>
<td>200: OkResponse</td>
</tr>
<tr>
<td>400</td>
<td>ExceptionBody (p. 188)</td>
<td>400: BadRequestException The conditional request failed. The service can't process your request because of a problem in the request. Please check your request form and syntax.</td>
</tr>
<tr>
<td>500</td>
<td>ExceptionBody (p. 188)</td>
<td>500: InternalServiceException The service encountered an unexpected condition and cannot fulfill your request.</td>
</tr>
<tr>
<td>403</td>
<td>ExceptionBody (p. 188)</td>
<td>403: AccessDeniedException You don't have permissions for this action with the credentials you sent. Please check your authorization credentials. You should be sending credentials</td>
</tr>
</tbody>
</table>
### Status Code | Response Model | Description
--- | --- | ---
404 | ExceptionBody (p. 188) | 404: ResourceNotFoundException

The resource you requested does not exist.

429 | ExceptionBody (p. 188) | 429: LimitExceededException

Too many requests have been sent in too short of a time. The service limits the rate at which it will accept requests. For example, you may be hitting your account limits for preset creation or job submission.

409 | ExceptionBody (p. 188) | 409: ResourceInUseException

The service could not complete your request because there is a conflict with the current state of the resource. For example, you may be trying to delete a Queue that has jobs processing.

---

**PUT**

Operation ID: UpdateJobTemplate

Modify one of your existing job templates.

### Path Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>True</td>
<td></td>
</tr>
</tbody>
</table>

### Responses

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>UpdateJobTemplateResponse</td>
<td>200: OkResponse</td>
</tr>
</tbody>
</table>
| 400 | ExceptionBody (p. 188) | 400: BadRequestException

The conditional request failed. The service can't process your request because of a problem in the request. Please check your request form and syntax.
### DELETE

Operation ID: DeleteJobTemplate

Permanently delete a job template you have created.

#### Path Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>True</td>
<td></td>
</tr>
</tbody>
</table>
### Responses

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
</table>
| 400         | ExceptionBody (p. 188) | 400: BadRequestException  
The conditional request failed. The service can’t process your request because of a problem in the request. Please check your request form and syntax. |
| 202         | DeleteJobTemplateResponse | 202: AcceptedResponse  
Your request has been accepted. Processing has not yet begun. |
| 500         | ExceptionBody (p. 188) | 500: InternalServiceException  
The service encountered an unexpected condition and cannot fulfill your request. |
| 403         | ExceptionBody (p. 188) | 403: AccessDeniedException  
You don't have permissions for this action with the credentials you sent. Please check your authorization credentials. You should be sending credentials using the [AWS Signature Version 4 signing process](https://docs.aws.amazon.com/general/latest/gr/aws-ml-signature-version-4.html). |
| 404         | ExceptionBody (p. 188) | 404: ResourceNotFoundException  
The resource you requested does not exist. |
| 429         | ExceptionBody (p. 188) | 429: LimitExceededException  
Too many requests have been sent in too short of a time. The service limits the rate at which it will accept requests. For example, you may be hitting your account limits for preset creation or job submission. |
| 409         | ExceptionBody (p. 188) | 409: ResourceInUseException  
The service could not complete your request because there is a conflict with the current state of the resource. For example, you may be trying to delete a Queue that has jobs processing. |
Schemas

Request Bodies

Example GET

```
{
  "name (p. 225)": "string"
}
```

Example PUT

```
{
  "settings (p. 298)": {
    "timecodeConfig (p. 259)": {
      "timestampOffset (p. 296)": "string",
      "anchor (p. 296)": "string",
      "start (p. 296)": "string",
      "source (p. 297)": enum
    },
    "adAvailOffset (p. 259)": integer,
    "nielsenConfiguration (p. 259)": {
      "distributorId (p. 284)": "string",
      "breakoutCode (p. 284)": integer
    },
    "inputs (p. 259)": [
      {
        "audioSelectors (p. 254)": {
        },
        "audioSelectorGroups (p. 254)": {
        },
        "filterEnable (p. 254)": enum,
        "deblockFilter (p. 254)": enum,
        "videoSelector (p. 254)": {
          "colorSpace (p. 302)": enum,
          "hdr10Metadata (p. 302)": {
            "redPrimaryY (p. 243)": integer,
            "greenPrimaryY (p. 243)": integer,
            "whitePointX (p. 243)": integer,
            "maxLuminance (p. 243)": integer,
            "greenPrimaryX (p. 243)": integer,
            "whitePointY (p. 244)": integer,
            "redPrimaryX (p. 244)": integer,
            "bluePrimaryX (p. 244)": integer,
            "maxFrameAverageLightLevel (p. 244)": integer,
            "bluePrimaryY (p. 244)": integer,
            "maxContentLightLevel (p. 244)": integer,
            "minLuminance (p. 245)": integer
          },
          "programNumber (p. 302)": integer,
          "pid (p. 303)": integer,
          "colorSpaceUsage (p. 303)": enum
        },
        "filterStrength (p. 255)": integer,
        "programNumber (p. 255)": integer,
        "timecodeSource (p. 255)": enum,
        "captionSelectors (p. 255)": {
        },
        "denoiseFilter (p. 255)": enum,
        "psiControl (p. 255)": enum,
        "inputClippings (p. 255)": [
        ]
      }
    ]
  }
}
```
"startTimecode (p. 253)": "string",
"endTimecode (p. 253)": "string"
]}
]
"outputGroups (p. 260)": [
"outputs (p. 287)": [
"extension (p. 286)": "string",
"videoDescription (p. 286)": {
"fixedAfd (p. 299)": integer,
"scalingBehavior (p. 300)": enum,
"respondToFad (p. 300)": enum,
"codecSettings (p. 300)": {
"h265Settings (p. 299)": {
"slices (p. 237)": integer,
"minInterval (p. 237)": integer,
"parNumerator (p. 237)": integer,
"flickerAdaptiveQuantization (p. 237)": enum,
"gopSizeUnits (p. 237)": enum,
"hrdBufferSize (p. 238)": integer,
"qualityTuningLevel (p. 238)": enum,
"maxBitrate (p. 238)": integer,
"bitrate (p. 238)": integer,
"spatialAdaptiveQuantization (p. 238)": enum,
"sampleAdaptiveOffsetFilterMode (p. 238)": enum,
"temporalIds (p. 238)": enum,
"slowPal (p. 239)": enum,
"tiles (p. 239)": enum,
"codecProfile (p. 239)": enum,
"alternateTransferFunctionSei (p. 239)": enum,
"unregisteredSeiTimecode (p. 239)": enum,
"framerateControl (p. 239)": enum,
"telecine (p. 239)": enum,
"framerateConversionAlgorithm (p. 239)": enum,
"codecLevel (p. 239)": enum,
"numberReferenceFrames (p. 239)": integer,
"temporalAdaptiveQuantization (p. 240)": enum,
"hrdBufferInitialFillPercentage (p. 240)": integer,
"framerateNumerator (p. 240)": integer,
"framerateDenominator (p. 240)": integer,
"framerateConversionAlgorithm (p. 240)": enum,
"interlaceMode (p. 241)": enum,
"gopSize (p. 241)": number,
"gopBReference (p. 241)": enum,
"sceneChangeDetect (p. 241)": enum,
"parDenominator (p. 241)": integer,
"parControl (p. 241)": enum,
"rateControlMode (p. 241)": enum
},
"proresSettings (p. 299)": {
"slowPal (p. 290)": enum,
"framerateControl (p. 290)": enum,
"telecine (p. 290)": enum,
"framerateNumerator (p. 290)": integer,
"framerateDenominator (p. 290)": integer,
"framerateConversionAlgorithm (p. 290)": enum,
"adaptiveQuantization (p. 241)": enum,
"interlaceMode (p. 241)": enum,
"gopSize (p. 241)": number,
"gopBReference (p. 241)": enum,
"sceneChangeDetect (p. 241)": enum,
"parDenominator (p. 241)": integer,
"parControl (p. 241)": enum,
"parBReference (p. 241)": integer,
"parControl (p. 291)": enum,
"parReferenceFramesBetweenReferenceFrames (p. 291)": integer
},
"codec (p. 299)": enum,
"proresSettings (p. 299)": {
"slowPal (p. 290)": enum,
"framerateControl (p. 290)": enum,
"telecine (p. 290)": enum,
"framerateNumerator (p. 290)": integer,
"framerateDenominator (p. 290)": integer,
"framerateConversionAlgorithm (p. 290)": enum,
"adaptiveQuantization (p. 241)": enum,
"interlaceMode (p. 241)": enum,
"gopSize (p. 241)": number,
"gopBReference (p. 241)": enum,
"sceneChangeDetect (p. 241)": enum,
"parDenominator (p. 241)": integer,
"parControl (p. 241)": enum,
"rateControlMode (p. 241)": enum
}
"framerateNumerator (p. 291)" : integer,
"mpeg2Settings (p. 299)" : {
  "minIInterval (p. 278)" : integer,
  "parNumerator (p. 278)" : integer,
  "gopSizeUnits (p. 278)" : enum,
  "hrdBufferSize (p. 278)" : integer,
  "qualityTuningLevel (p. 279)" : enum,
  "maxBitrate (p. 279)" : integer,
  "bitrate (p. 279)" : integer,
  "spatialAdaptiveQuantization (p. 279)" : enum,
  "slowPal (p. 279)" : enum,
  "codecProfile (p. 279)" : enum,
  "intraDcPrecision (p. 279)" : enum,
  "softness (p. 279)" : integer,
  "framerateControl (p. 280)" : enum,
  "telecine (p. 280)" : enum,
  "framerateConversionAlgorithm (p. 280)" : enum,
  "codeLevel (p. 280)" : enum,
  "temporalAdaptiveQuantization (p. 280)" : enum,
  "hrdBufferInitialFillPercentage (p. 280)" : integer,
  "framerateNumerator (p. 280)" : integer,
  "numberBFramesBetweenReferenceFrames (p. 280)" : integer,
  "gopClosedCadence (p. 281)" : integer,
  "framerateDenominator (p. 281)" : integer,
  "adaptiveQuantization (p. 281)" : enum,
  "interlaceMode (p. 281)" : enum,
  "gopSize (p. 281)" : number,
  "sceneChangeDetect (p. 281)" : enum,
  "parDenominator (p. 281)" : integer,
  "parControl (p. 282)" : enum,
  "syntax (p. 282)" : enum,
  "rateControlMode (p. 282)" : enum
},
"h264Settings (p. 299)" : {
  "slices (p. 228)" : integer,
  "minIInterval (p. 228)" : integer,
  "parNumerator (p. 228)" : integer,
  "flickerAdaptiveQuantization (p. 229)" : enum,
  "gopSizeUnits (p. 229)" : enum,
  "hrdBufferSize (p. 229)" : integer,
  "qualityTuningLevel (p. 229)" : enum,
  "maxBitrate (p. 229)" : integer,
  "bitrate (p. 229)" : integer,
  "spatialAdaptiveQuantization (p. 229)" : enum,
  "slowPal (p. 230)" : enum,
  "codecProfile (p. 230)" : enum,
  "unregisteredSeiTimecode (p. 230)" : enum,
  "softness (p. 230)" : integer,
  "framerateControl (p. 230)" : enum,
  "telecine (p. 230)" : enum,
  "framerateConversionAlgorithm (p. 230)" : enum,
  "codeLevel (p. 230)" : enum,
  "numberReferenceFrames (p. 230)" : integer,
  "temporalAdaptiveQuantization (p. 231)" : enum,
  "repeatPps (p. 231)" : enum,
  "hrdBufferInitialFillPercentage (p. 231)" : integer,
  "framerateNumerator (p. 231)" : integer,
  "numberBFramesBetweenReferenceFrames (p. 231)" : integer,
  "gopClosedCadence (p. 231)" : integer,
  "fieldEncoding (p. 231)" : enum,
  "entropyEncoding (p. 232)" : enum,
  "framerateDenominator (p. 232)" : integer,
  "adaptiveQuantization (p. 232)" : enum,
  "interlaceMode (p. 232)" : enum,
  "gopSize (p. 232)" : number,
"gopBReference (p. 232)": enum,
"sceneChangeDetect (p. 232)": enum,
"parDenominator (p. 232)": integer,
"parControl (p. 233)": enum,
"syntax (p. 233)": enum,
"rateControlMode (p. 233)": enum
},
"frameCaptureSettings (p. 299)": {
"framerateDenominator (p. 224)": integer,
"maxCaptures (p. 224)": integer,
"framerateNumerator (p. 224)": integer,
"quality (p. 225)": integer
},
"afdSignaling (p. 300)": enum,
"colorMetadata (p. 300)": enum,
"timecodeInsertion (p. 300)": enum,
"width (p. 300)": integer,
"sharpness (p. 300)": integer,
"antiAlias (p. 301)": enum,
"videoPreprocessors (p. 301)": {
"timecodeBurnin (p. 301)": {
  "prefix (p. 295)": "string",
  "fontSize (p. 295)": integer,
  "position (p. 295)": enum
},
"noiseReducer (p. 302)": {
  "filter (p. 284)": enum,
  "filterSettings (p. 284)": {
    "strength (p. 285)": integer
  },
  "spatialFilterSettings (p. 284)": {
    "strength (p. 285)": integer,
    "postFilterSharpenStrength (p. 285)": integer,
    "speed (p. 285)": integer
  }
},
"colorCorrector (p. 302)": {
  "saturation (p. 206)": integer,
  "brightness (p. 207)": integer,
  "hdr10Metadata (p. 207)": {
    "redPrimaryY (p. 243)": integer,
    "greenPrimaryY (p. 243)": integer,
    "whitePointX (p. 243)": integer,
    "maxLuminance (p. 243)": integer,
    "greenPrimaryX (p. 243)": integer,
    "whitePointY (p. 244)": integer,
    "redPrimaryX (p. 244)": integer,
    "bluePrimaryX (p. 244)": integer,
    "maxFrameAverageLightLevel (p. 244)": integer,
    "bluePrimaryY (p. 244)": integer,
    "maxContentLightLevel (p. 244)": integer,
    "minLuminance (p. 245)": integer
  },
  "contrast (p. 207)": integer,
  "hue (p. 207)": integer,
  "colorSpaceConversion (p. 207)": enum
},
"imageInserter (p. 302)": {
  "insertableImages (p. 252)": [
  {"duration (p. 256)": integer,
   "fadeOut (p. 256)": integer,
   "imageY (p. 256)": integer,
   "fadeIn (p. 256)": integer,
   "imageX (p. 257)": integer,
"width (p. 257)" : integer,
"startTime (p. 257)" : "string",
"opacity (p. 257)" : integer,
"layer (p. 257)" : integer,
"height (p. 257)" : integer,
"imageInserterInput (p. 258)" : "string"
]
"
deinterlacer (p. 302)" : {
  "mode (p. 211)" : enum,
  "control (p. 211)" : enum,
  "algorithm (p. 211)" : enum
}
"
"position (p. 301)" : {
  "width (p. 291)" : integer,
  "x (p. 292)" : integer,
  "y (p. 292)" : integer,
  "height (p. 292)" : integer
},
"dropFrameTimecode (p. 301)" : enum,
"crop (p. 301)" : {
  "width (p. 291)" : integer,
  "x (p. 292)" : integer,
  "y (p. 292)" : integer,
  "height (p. 292)" : integer
},
"height (p. 301)" : integer
},
"audioDescriptions (p. 286)" : [
  {
    "languageCodeControl (p. 195)" : enum,
    "audioTypeControl (p. 195)" : enum,
    "remixSettings (p. 195)" : {
      "channelsOut (p. 292)" : integer,
      "channelMapping (p. 292)" : {
        "outputChannels (p. 206)" : [
          {
            "inputChannels (p. 287)" : [integer
          ]
        ]
      }
    },
    "audioType (p. 195)" : integer,
    "audioSourceName (p. 195)" : "string",
    "codecSettings (p. 196)" : {
      "codec (p. 194)" : enum,
      "wavSettings (p. 194)" : {
        "channels (p. 303)" : integer,
        "bitDepth (p. 303)" : integer,
        "sampleRate (p. 303)" : integer
      },
      "ac3Settings (p. 194)" : {
        "dynamicRangeCompressionProfile (p. 192)" : enum,
        "dialnorm (p. 192)" : integer,
        "codingMode (p. 192)" : enum,
        "metadataControl (p. 192)" : enum,
        "lfeFilter (p. 192)" : enum,
        "bitrate (p. 192)" : integer,
        "bitstreamMode (p. 192)" : enum,
        "sampleRate (p. 192)" : integer
      }
    }
  }
]
"aacSettings (p. 194)": {
  "vbrQuality (p. 189)": enum,
  "codecProfile (p. 189)": enum,
  "codingMode (p. 190)": enum,
  "specification (p. 190)": enum,
  "bitrate (p. 190)": integer,
  "rawFormat (p. 190)": enum,
  "rateControlMode (p. 190)": enum,
  "sampleRate (p. 190)": integer,
  "audioDescriptionBroadcasterMix (p. 190)": enum
},
"aiffSettings (p. 194)": {
  "channels (p. 193)": integer,
  "bitDepth (p. 193)": integer,
  "sampleRate (p. 193)": integer
},
"eac3Settings (p. 194)": {
  "dialnorm (p. 219)": integer,
  "passthroughControl (p. 219)": enum,
  "metadataControl (p. 219)": enum,
  "bitrate (p. 219)": integer,
  "dynamicRangeCompressionRf (p. 220)": enum,
  "sampleRate (p. 220)": integer,
  "ltRtSurroundMixLevel (p. 220)": number,
  "surroundEXMode (p. 220)": enum,
  "dynamicRangeCompressionLine (p. 220)": enum,
  "lfeControl (p. 220)": enum,
  "codingMode (p. 220)": enum,
  "surroundMode (p. 220)": enum,
  "attenuationControl (p. 221)": enum,
  "lfeFilter (p. 221)": enum,
  "phaseControl (p. 221)": enum,
  "ltRtCenterMixLevel (p. 221)": number,
  "dcFilter (p. 221)": enum,
  "stereoDownmix (p. 221)": enum,
  "bitstreamMode (p. 221)": enum,
  "loRoSurroundMixLevel (p. 221)": number,
  "loRoCenterMixLevel (p. 222)": number
},
"mp2Settings (p. 195)": {
  "channels (p. 274)": integer,
  "bitrate (p. 275)": integer,
  "sampleRate (p. 275)": integer
},
"languageCode (p. 196)": enum,
"streamName (p. 196)": "string",
"audioNormalizationSettings (p. 196)": {
  "targetLkfs (p. 197)": number,
  "algorithmControl (p. 197)": enum,
  "peakCalculation (p. 197)": enum,
  "loudnessLogging (p. 197)": enum,
  "correctionGateLevel (p. 197)": integer,
  "algorithm (p. 198)": enum
},
"containerSettings (p. 286)": {
  "container (p. 208)": enum,
  "mp4Settings (p. 208)": {
    "mp4MajorBrand (p. 275)": "string",
    "moovPlacement (p. 276)": enum,
    "cs1gAtom (p. 276)": enum,
    "freeSpaceBox (p. 276)": enum
  },
  "m3u8Settings (p. 208)": {
    "mp4MajorBrand (p. 275)": "string",
    "moovPlacement (p. 276)": enum,
    "cs1gAtom (p. 276)": enum,
    "freeSpaceBox (p. 276)": enum
  }
}
"pmtPid (p. 271)" : integer,
"pcrPid (p. 271)" : integer,
"audioPids (p. 271)" : [  
  integer  
],
"audioFramesPerPES (p. 271)" : integer,
"scte35Pid (p. 271)" : integer,
"transportStreamId (p. 272)" : integer,
"videoPid (p. 272)" : integer,
"pcrControl (p. 272)" : enum,
"privateMetadataPid (p. 272)" : integer,
"pmtInterval (p. 272)" : integer,
"patInterval (p. 272)" : integer,
"programNumber (p. 273)" : integer,
"timedMetadataPid (p. 273)" : integer,
"timedMetadata (p. 273)" : enum,
"scte35Source (p. 273)" : enum,
"m2tsSettings (p. 208)" : {  
  "dvbTeletextPid (p. 266)" : integer,
  "bitrate (p. 266)" : integer,
  "segmentationTime (p. 266)" : number,
  "audioPids (p. 266)" : [  
    integer  
  ],
  "rateMode (p. 266)" : enum,
  "ebpAudioInterval (p. 266)" : enum,
  "fragmentTime (p. 267)" : number,
  "audioFramesPerPES (p. 267)" : integer,
  "maxPcrInterval (p. 267)" : integer,
  "scte35Pid (p. 267)" : integer,
  "privateMetadataPid (p. 267)" : integer,
  "pmtInterval (p. 267)" : integer,
  "segmentationStyle (p. 268)" : enum,
  "audioBufferModel (p. 268)" : enum,
  "programNumber (p. 268)" : integer,
  "dvbNitSettings (p. 268)" : {  
    "networkName (p. 212)" : "string",
    "networkId (p. 212)" : integer,
    "nitInterval (p. 212)" : integer
  },
  "scte35Source (p. 268)" : enum,
  "pmtPid (p. 268)" : integer,
  "bufferModel (p. 268)" : enum,
  "ebpPlacement (p. 268)" : enum,
  "dvbSdtSettings (p. 269)" : {  
    "sdtInterval (p. 213)" : integer,
    "serviceName (p. 213)" : "string",
    "serviceProviderName (p. 213)" : "string",
    "outputSdt (p. 213)" : "string",
    "nullPacketBitrate (p. 269)" : number,
    "pcrPid (p. 269)" : integer,
    "minEbpInterval (p. 269)" : integer,
    "transportStreamId (p. 269)" : integer,
    "pcrControl (p. 269)" : enum,
    "videoPid (p. 270)" : integer,
    "esRateInPES (p. 270)" : integer,
    "segmentationMarkers (p. 270)" : enum,
    "dvbTdtSettings (p. 270)" : {  
      "tdtInterval (p. 217)" : integer
    },
    "patInterval (p. 270)" : integer,
    "dvbSubPids (p. 270)" : [  
      integer
    ]
  },
  "programNumber (p. 273)" : integer,
"movSettings (p. 209)": {
    "reference (p. 274)": enum,
    "paddingControl (p. 274)": enum,
    "mpeg2FourCCControl (p. 274)": enum,
    "cs1gAtom (p. 274)": enum,
    "clapAtom (p. 274)": enum
},
"f4vSettings (p. 209)": {
    "moovPlacement (p. 223)": enum
},
"preset (p. 286)": "string",
"outputSettings (p. 286)": {
    "hlsSettings (p. 288)": {
        "iFrameOnlyManifest (p. 251)": enum,
        "segmentModifier (p. 251)": "string",
        "audioTrackType (p. 251)": enum,
        "audioRenditionSets (p. 252)": "string",
        "audioGroupId (p. 252)": "string"
    },
    "captionDescriptions (p. 287)": [
        {
            "captionSelectorName (p. 204)": "string",
            "languageDescription (p. 204)": "string",
            "languageCode (p. 204)": enum,
            "destinationSettings (p. 204)": {
                "burninDestinationSettings (p. 204)": {
                    "xPosition (p. 200)": integer,
                    "backgroundColor (p. 200)": enum,
                    "teletextSpacing (p. 200)": enum,
                    "yPosition (p. 200)": integer,
                    "backgroundOpacity (p. 201)": integer,
                    "fontOpacity (p. 201)": integer,
                    "shadowOpacity (p. 201)": integer,
                    "fontResolution (p. 201)": integer,
                    "shadowXOffset (p. 201)": integer,
                    "outlineSize (p. 201)": integer,
                    "outlineColor (p. 202)": enum,
                    "fontSize (p. 202)": integer,
                    "shadowXOffset (p. 202)": integer,
                    "alignment (p. 202)": enum,
                    "shadowColor (p. 202)": enum,
                    "fontColor (p. 202)": enum
                },
                "teletextDestinationSettings (p. 204)": {
                    "pageNumber (p. 295)": "string"
                },
                "ttmlDestinationSettings (p. 204)": {
                    "stylePassthrough (p. 297)": enum
                },
                "destinationType (p. 204)": enum,
                "dvbSubDestinationSettings (p. 205)": {
                    "xPosition (p. 213)": integer,
                    "backgroundColor (p. 213)": enum,
                    "teletextSpacing (p. 213)": enum,
                    "yPosition (p. 213)": integer,
                    "backgroundOpacity (p. 214)": integer,
                    "fontOpacity (p. 214)": integer,
                    "shadowOpacity (p. 214)": integer,
                    "fontResolution (p. 214)": integer,
                    "shadowXOffset (p. 215)": integer,
                    "outlineSize (p. 215)": integer,
                    "outlineColor (p. 215)": enum,
                    "fontSize (p. 215)": integer,
                    "shadowColor (p. 215)": enum,
                    "fontColor (p. 215)": enum
                }
            }
        }
    ]
}
"shadowXOffset (p. 215)": integer,
"alignment (p. 215)": enum,
"shadowColor (p. 216)": enum,
"fontColor (p. 216)": enum
},
"sccDestinationSettings (p. 205)": {
"framerate (p. 293)": enum
}
]
,"nameModifier (p. 287)": "string"
}
,"outputGroupSettings (p. 287)": {
"dashIsoGroupSettings (p. 288)": {
"fragmentLength (p. 209)": integer,
"baseUrl (p. 209)": "string",
"minBufferTime (p. 210)": integer,
"encryption (p. 210)": {
"spekeKeyProvider (p. 209)": {
"resourceId (p. 293)": "string",
"systemIds (p. 294)": ["string"
],
"url (p. 294)": "string"
}
},
"destination (p. 210)": "string",
"segmentLength (p. 210)": integer,
"segmentControl (p. 210)": enum,
"hbbtvCompliance (p. 210)": enum
},
"fileGroupSettings (p. 288)": {
"destination (p. 223)": "string"
},
"msSmoothGroupSettings (p. 288)": {
"fragmentLength (p. 283)": integer,
"encryption (p. 283)": {
"spekeKeyProvider (p. 283)": {
"resourceId (p. 293)": "string",
"systemIds (p. 294)": ["string"
],
"url (p. 294)": "string"
}
},
"audioDeduplication (p. 283)": enum,
"manifestEncoding (p. 283)": enum,
"destination (p. 283)": "string"
},
"type (p. 288)": enum,
"hlsGroupSettings (p. 288)": {
"segmentsPerSubdirectory (p. 247)": integer,
"streamInfResolution (p. 247)": enum,
"timestampDeltaMilliseconds (p. 247)": integer,
"outputSelection (p. 248)": enum,
"captionLanguageMappings (p. 248)": ["languageDescription (p. 245)": "string",
"captionChannel (p. 245)": integer,
"languageCode (p. 246)": enum
],
"clientCache (p. 248)": enum,
"codecSpecification (p. 248)": enum,
Example DELETE

```json
{
    "name": "string"
}
```
Response Bodies

Example GetJobTemplateResponse

```
{
  "jobTemplate (p. 225)": {
    "settings (p. 258)": {
      "timecodeConfig (p. 259)": {
        "timestampOffset (p. 296)": "string",
        "anchor (p. 296)": "string",
        "start (p. 296)": "string",
        "source (p. 297)": enum
      },
      "adAvailOffset (p. 259)": integer,
      "nielsenConfiguration (p. 259)": {
        "distributorId (p. 284)": "string",
        "breakoutCode (p. 284)": integer
      },
      "inputs (p. 259)": [
        "audioSelectors (p. 254)": {
        },
        "audioSelectorGroups (p. 254)": {
        },
        "filterEnable (p. 254)": enum,
        "deblockFilter (p. 254)": enum,
        "videoSelector (p. 254)": {
          "colorSpace (p. 302)": enum,
          "hdr10Metadata (p. 302)": {
            "redPrimaryY (p. 243)": integer,
            "greenPrimaryY (p. 243)": integer,
            "whitePointX (p. 243)": integer,
            "maxLuminance (p. 243)": integer,
            "greenPrimaryX (p. 243)": integer,
            "whitePointY (p. 244)": integer,
            "redPrimaryX (p. 244)": integer,
            "bluePrimaryX (p. 244)": integer,
            "maxFrameAverageLightLevel (p. 244)": integer,
            "bluePrimaryY (p. 244)": integer,
            "maxContentLightLevel (p. 244)": integer,
            "minLuminance (p. 245)": integer
          },
          "programNumber (p. 302)": integer,
          "pid (p. 303)": integer,
          "colorSpaceUsage (p. 303)": enum
        },
        "filterStrength (p. 255)": integer,
        "programNumber (p. 255)": integer,
        "timecodeSource (p. 255)": enum,
        "captionSelectors (p. 255)": {
        },
        "denoiseFilter (p. 255)": enum,
        "psiControl (p. 255)": enum,
        "inputClippings (p. 255)": [
          {
            "startTimecode (p. 253)": "string",
            "endTimecode (p. 253)": "string"
          }
        ],
      },
      "outputGroups (p. 260)": [
        {
          "outputs (p. 287)": [
          ]
        }
      ]
    }
  }
}
```
Schemas

{
  "extension (p. 286)": "string",
  "videoDescription (p. 286)": {
    "fixedAfd (p. 299)": integer,
    "scalingBehavior (p. 300)": enum,
    "respondToAfd (p. 300)": enum,
    "codecSettings (p. 300)": {
      "h265Settings (p. 299)": {
        "slices (p. 237)": integer,
        "minInterval (p. 237)": integer,
        "parNumerator (p. 237)": integer,
        "flickerAdaptiveQuantization (p. 237)": enum,
        "gopSizeUnits (p. 237)": enum,
        "hrdBufferSize (p. 238)": integer,
        "qualityTuningLevel (p. 238)": enum,
        "maxBitrate (p. 238)": integer,
        "bitrate (p. 238)": integer,
        "respondToAfd (p. 238)": enum,
        "spatialAdaptiveQuantization (p. 238)": enum,
        "sampleAdaptiveOffsetFilterMode (p. 238)": enum,
        "temporalIds (p. 238)": enum,
        "slowPal (p. 239)": enum,
        "tiles (p. 239)": enum,
        "codecProfile (p. 239)": enum,
        "alternateTransferFunctionSei (p. 239)": enum,
        "unregisteredSeiTimecode (p. 239)": enum,
        "framerateControl (p. 239)": enum,
        "telecine (p. 239)": enum,
        "framerateConversionAlgorithm (p. 239)": enum,
        "codecLevel (p. 239)": enum,
        "numberReferenceFrames (p. 239)": integer,
        "temporalAdaptiveQuantization (p. 240)": enum,
        "hrdBufferInitialFillPercentage (p. 240)": integer,
        "framerateNumerator (p. 240)": integer,
        "framerateDenominator (p. 240)": integer,
        "framerateConversionAlgorithm (p. 240)": enum,
        "gopClosedCadence (p. 240)": integer,
        "framerateDenominator (p. 240)": integer,
        "adaptiveQuantization (p. 241)": enum,
        "interlaceMode (p. 241)": enum,
        "gopSize (p. 241)": number,
        "gopReference (p. 241)": enum,
        "sceneChangeDetect (p. 241)": enum,
        "parDenominator (p. 241)": integer,
        "parControl (p. 241)": enum,
        "rateControlMode (p. 241)": enum
      },
      "proresSettings (p. 299)": {
        "slowPal (p. 290)": enum,
        "framerateControl (p. 290)": enum,
        "telecine (p. 290)": enum,
        "framerateDenominator (p. 290)": integer,
        "framerateConversionAlgorithm (p. 290)": enum,
        "interlaceMode (p. 241)": enum,
        "gopSize (p. 241)": number,
        "gopReference (p. 241)": enum,
        "sceneChangeDetect (p. 241)": enum,
        "parDenominator (p. 241)": integer,
        "parControl (p. 241)": enum,
        "rateControlMode (p. 241)": enum
      },
      "mpeg2Settings (p. 299)": {
        "minInterval (p. 278)": integer,
        "parNumerator (p. 278)": integer,
        "gopSizeUnits (p. 278)": enum,
        "hrdBufferSize (p. 278)": integer,
        "qualityTuningLevel (p. 279)": enum,
        "maxBitrate (p. 279)": integer,
      }
    }
  }
}
"bitrate (p. 279)": integer,
"spatialAdaptiveQuantization (p. 279)": enum,
"slowPal (p. 279)": enum,
"codecProfile (p. 279)": enum,
"intraDcPrecision (p. 279)": enum,
"softness (p. 279)": integer,
"framerateControl (p. 280)": enum,
"telecine (p. 280)": enum,
"framerateConversionAlgorithm (p. 280)": enum,
"codecLevel (p. 280)": enum,
"temporalAdaptiveQuantization (p. 280)": enum,
"hrdBuffersInitialFillPercentage (p. 280)": integer,
"framerateNumerator (p. 280)": integer,
"framerateDenominator (p. 281)": integer,
"adaptiveQuantization (p. 281)": enum,
"interlaceMode (p. 281)": enum,
"gopSize (p. 281)": number,
"sceneChangeDetect (p. 281)": enum,
"parDenominator (p. 281)": integer,
"parControl (p. 282)": enum,
"syntax (p. 282)": enum,
"rateControlMode (p. 282)": enum,

"h264Settings (p. 299)": {
  "slices (p. 228)": integer,
  "minIInterval (p. 228)": integer,
  "parNumerator (p. 228)": integer,
  "flickerAdaptiveQuantization (p. 229)": enum,
  "gopSizeUnits (p. 229)": enum,
  "hrdBuffersSize (p. 229)": integer,
  "qualityTuningLevel (p. 229)": enum,
  "maxBitrate (p. 229)": integer,
  "bitrate (p. 229)": integer,
  "spatialAdaptiveQuantization (p. 229)": enum,
  "slowPal (p. 230)": enum,
  "codecProfile (p. 230)": enum,
  "unregisteredSeiTimecode (p. 230)": enum,
  "softness (p. 230)": integer,
  "framerateControl (p. 230)": enum,
  "telecine (p. 230)": enum,
  "framerateConversionAlgorithm (p. 230)": enum,
  "codecLevel (p. 230)": enum,
  "numberReferenceFrames (p. 230)": integer,
  "temporalAdaptiveQuantization (p. 231)": enum,
  "repeatPps (p. 231)": integer,
  "hrdBuffersInitialFillPercentage (p. 231)": integer,
  "framerateNumerator (p. 231)": integer,
  "framerateDenominator (p. 231)": integer,
  "fieldEncoding (p. 231)": enum,
  "entropyEncoding (p. 232)": enum,
  "framerateDenominator (p. 232)": integer,
  "adaptiveQuantization (p. 232)": enum,
  "interlaceMode (p. 232)": enum,
  "gopSize (p. 232)": number,
  "gopReference (p. 232)": enum,
  "sceneChangeDetect (p. 232)": enum,
  "parDenominator (p. 232)": integer,
  "parControl (p. 233)": enum,
  "syntax (p. 233)": enum,
  "rateControlMode (p. 233)": enum
},

"frameCaptureSettings (p. 299)": {
  "framerateDenominator (p. 224)": integer,
"maxCaptures (p. 224)" : integer,
"framerateNumerator (p. 225)" : integer,
"quality (p. 225)" : integer
}
},
"afdsSignaling (p. 300)" : enum,
"colorMetadata (p. 300)" : enum,
"timecodeInsertion (p. 300)" : enum,
"width (p. 300)" : integer,
"sharpness (p. 300)" : integer,
"antiAlias (p. 301)" : enum,
"videoPreprocessors (p. 301)" : {
  "timecodeBurnin (p. 301)" : {
    "prefix (p. 295)" : "string",
    "fontSize (p. 295)" : integer,
    "position (p. 295)" : enum
  },
  "noiseReducer (p. 302)" : {
    "filter (p. 284)" : enum,
    "filterSettings (p. 284)" : {
      "strength (p. 285)" : integer
    },
    "spatialFilterSettings (p. 284)" : {
      "strength (p. 285)" : integer,
      "postFilterSharpenStrength (p. 285)" : integer,
      "speed (p. 285)" : integer
    }
  },
  "colorCorrector (p. 302)" : {
    "saturation (p. 206)" : integer,
    "brightness (p. 207)" : integer,
    "hdr10Metadata (p. 207)" : {
      "redPrimaryY (p. 243)" : integer,
      "greenPrimaryY (p. 243)" : integer,
      "whitePointX (p. 243)" : integer,
      "maxLuminance (p. 243)" : integer,
      "greenPrimaryX (p. 243)" : integer,
      "whitePointY (p. 244)" : integer,
      "redPrimaryX (p. 244)" : integer,
      "bluePrimaryX (p. 244)" : integer,
      "maxFrameAverageLightLevel (p. 244)" : integer,
      "bluePrimaryY (p. 244)" : integer,
      "maxContentLightLevel (p. 244)" : integer,
      "minLuminance (p. 245)" : integer
    },
    "contrast (p. 207)" : integer,
    "hue (p. 207)" : integer,
    "colorSpaceConversion (p. 207)" : enum
  },
  "imageInserter (p. 302)" : {
    "insertableImages (p. 252)" : [
      {
        "duration (p. 256)" : integer,
        "fadeOut (p. 256)" : integer,
        "imageY (p. 256)" : integer,
        "fadeIn (p. 256)" : integer,
        "imageX (p. 257)" : integer,
        "width (p. 257)" : integer,
        "startTime (p. 257)" : "string",
        "opacity (p. 257)" : integer,
        "layer (p. 257)" : integer,
        "height (p. 257)" : integer,
        "imageInserterInput (p. 258)" : "string"
      }
    ]
  }
},
"deinterlacer (p. 302)": {
  "mode (p. 211)": enum,
  "control (p. 211)": enum,
  "algorithm (p. 211)": enum
}
},
"position (p. 301)": {
  "width (p. 291)": integer,
  "x (p. 292)": integer,
  "y (p. 292)": integer,
  "height (p. 292)": integer
}
},
"dropFrameTimecode (p. 301)": enum,
"crop (p. 301)": {
  "width (p. 291)": integer,
  "x (p. 292)": integer,
  "y (p. 292)": integer,
  "height (p. 292)": integer
}
},
"height (p. 301)": integer
},
"audioDescriptions (p. 286)": [
  {
    "languageCodeControl (p. 195)": enum,
    "audioTypeControl (p. 195)": enum,
    "remixSettings (p. 195)": {
      "channelsOut (p. 292)": integer,
      "channelMapping (p. 292)": {
        "outputChannels (p. 206)": [
          {
            "inputChannels (p. 287)": [integer
          ]
        ]
      }
    }
  }
},
"channelsIn (p. 292)": integer
},
"audioType (p. 195)": integer,
"audioSourceName (p. 195)": "string",
"codecSettings (p. 196)": {
  "codec (p. 194)": enum,
  "wavSettings (p. 194)": {
    "channels (p. 303)": integer,
    "bitDepth (p. 303)": integer,
    "sampleRate (p. 303)": integer
  }
},
"ac3Settings (p. 194)": {
  "dynamicRangeCompressionProfile (p. 192)": enum,
  "dialnorm (p. 192)": integer,
  "codingMode (p. 192)": enum,
  "metadataControl (p. 192)": enum,
  "lfeFilter (p. 192)": enum,
  "bitrate (p. 192)": integer,
  "bitstreamMode (p. 192)": enum,
  "sampleRate (p. 192)": integer
}
},
"aacSettings (p. 194)": {
  "vbrQuality (p. 189)": enum,
  "codecProfile (p. 189)": enum,
  "codingMode (p. 190)": enum,
  "specification (p. 190)": enum,
  "bitrate (p. 190)": integer,
  "rawFormat (p. 190)": enum,
  "rateControlMode (p. 190)": enum,
  "sampleRate (p. 190)": integer,
  "channelCount (p. 190)": integer
}
"audioDescriptionBroadcasterMix (p. 190)": enum
},
"aiffSettings (p. 194)": {
  "channels (p. 193)": integer,
  "bitDepth (p. 193)": integer,
  "sampleRate (p. 193)": integer
},
"eac3Settings (p. 194)": {
  "dialnorm (p. 219)": integer,
  "passthroughControl (p. 219)": enum,
  "metadataControl (p. 219)": enum,
  "bitrate (p. 219)": integer,
  "dynamicRangeCompressionRf (p. 220)": enum,
  "sampleRate (p. 220)": integer,
  "ltRtSurroundMixLevel (p. 220)": number,
  "surroundExMode (p. 220)": enum,
  "dynamicRangeCompressionLine (p. 220)": enum,
  "lifeControl (p. 220)": enum,
  "codingMode (p. 220)": enum,
  "surroundMode (p. 220)": enum,
  "attenuationControl (p. 221)": enum,
  "lifeFilter (p. 221)": enum,
  "phaseControl (p. 221)": enum,
  "ltRtCenterMixLevel (p. 221)": number,
  "dcFilter (p. 221)": enum,
  "stereoDownmix (p. 221)": enum,
  "bitstreamMode (p. 221)": enum,
  "loRoSurroundMixLevel (p. 221)": number,
  "loRoCenterMixLevel (p. 222)": number
},
"mp2Settings (p. 195)": {
  "channels (p. 274)": integer,
  "bitrate (p. 275)": integer,
  "sampleRate (p. 275)": integer
},
"languageCode (p. 196)": enum,
"streamName (p. 196)": "string",
"audioNormalizationSettings (p. 196)": {
  "targetLkfs (p. 197)": number,
  "algorithmControl (p. 197)": enum,
  "peakCalculation (p. 197)": enum,
  "loudnessLogging (p. 197)": enum,
  "correctionGateLevel (p. 197)": integer,
  "algorithm (p. 198)": enum
},
"containerSettings (p. 286)": {
  "container (p. 208)": enum,
  "mp4Settings (p. 208)": {
    "mp4MajorBrand (p. 275)": "string",
    "moovPlacement (p. 276)": enum,
    "cs1gAtom (p. 276)": enum,
    "freeSpaceBox (p. 276)": enum
  },
  "m3u8Settings (p. 208)": {
    "pmtPid (p. 271)": integer,
    "pcrPid (p. 271)": integer,
    "audioPids (p. 271)": [
      integer
    ],
    "audioFramesPerPes (p. 271)": integer,
    "scte35Pid (p. 271)": integer,
    "transportStreamId (p. 272)": integer,
    "videoPid (p. 272)": integer,
"pcrControl (p. 272)": enum,
"privateMetadataPid (p. 272)": integer,
"pmtInterval (p. 272)": integer,
"patInterval (p. 272)": integer,
"programNumber (p. 273)": integer,
"timedMetadataPid (p. 273)": integer,
"timedMetadata (p. 273)": enum,
"scte35Source (p. 273)": enum
},
"m2tsSettings (p. 208)": {
  "dvbTeletextPid (p. 266)": integer,
  "bitrate (p. 266)": integer,
  "segmentationTime (p. 266)": number,
  "audioPids (p. 266)": [ integer
  ],
  "rateMode (p. 266)": enum,
  "ebpAudioInterval (p. 266)": enum,
  "fragmentTime (p. 267)": number,
  "audioFramesPerPes (p. 267)": integer,
  "maxPcrInterval (p. 267)": integer,
  "scte35Pid (p. 267)": integer,
  "privateMetadataPid (p. 267)": integer,
  "pmtInterval (p. 267)": integer,
  "segmentationStyle (p. 268)": enum,
  "audioBufferModel (p. 268)": enum,
  "ebpAudioInterval (p. 268)": enum,
  "audioPids (p. 268)": [ integer
  ],
  "networkName (p. 212)": "string",
  "networkId (p. 212)": integer,
  "nitInterval (p. 212)": integer
},
"scte35Source (p. 268)": enum,
"pmtPld (p. 268)": integer,
"bufferModel (p. 268)": enum,
"ebpPlacement (p. 268)": enum,
"dvbSdtSettings (p. 269)": {
  "sdtInterval (p. 213)": integer,
  "serviceName (p. 213)": "string",
  "serviceProviderName (p. 213)": "string",
  "outputSdt (p. 213)": "string",
  "nullPacketBitrate (p. 269)": number,
  "pcrPid (p. 269)": integer,
  "minEbPInterval (p. 269)": integer,
  "transportStreamId (p. 269)": integer,
  "pcrControl (p. 269)": enum,
  "videoPid (p. 270)": integer,
  "esRateInPes (p. 270)": number,
  "segmentationMarkers (p. 270)": enum,
  "dvbTdtSettings (p. 270)": {
    "tdtInterval (p. 217)": integer
  },
  "patInterval (p. 270)": integer,
  "dvbSubPld (p. 270)": [ integer
  ]
},
"movSettings (p. 209)": {
  "reference (p. 274)": enum,
  "paddingControl (p. 274)": enum,
  "mpeg2FourCCControl (p. 274)": enum,
  "cslgAtom (p. 274)": enum,
  "clapAtom (p. 274)": enum
},
"f4vSettings (p. 209)": {
"moovPlacement (p. 223)" : enum
}
,"preset (p. 286)" : "string",
"outputSettings (p. 286)" : {
    "hlsSettings (p. 288)" : {
        "iFrameOnlyManifest (p. 251)" : enum,
        "segmentModifier (p. 251)" : "string",
        "audioTrackType (p. 251)" : enum,
        "audioRenditionSets (p. 252)" : "string",
        "audioGroupId (p. 252)" : "string"
    }
},
"captionDescriptions (p. 287)" : [
    {
        "captionSelectorName (p. 204)" : "string",
        "languageDescription (p. 204)" : "string",
        "languageCode (p. 204)" : enum,
        "destinationSettings (p. 204)" : {
            "burninDestinationSettings (p. 204)" : {
                "xPosition (p. 200)" : integer,
                "backgroundColor (p. 200)" : enum,
                "teletextSpacing (p. 200)" : enum,
                "yPosition (p. 200)" : integer,
                "backgroundColor (p. 201)" : integer,
                "shadowOpacity (p. 201)" : integer,
                "fontOpacity (p. 201)" : integer,
                "shadowOpacity (p. 201)" : integer,
                "fontResolution (p. 201)" : integer,
                "shadowXOffset (p. 201)" : integer,
                "outlineSize (p. 201)" : integer,
                "outlineColor (p. 202)" : enum,
                "fontSize (p. 202)" : integer,
                "shadowXOffset (p. 202)" : integer,
                "alignment (p. 202)" : enum,
                "shadowColor (p. 202)" : enum,
                "fontColor (p. 202)" : enum
            },
            "teletextDestinationSettings (p. 204)" : {
                "pageNumber (p. 295)" : "string"
            },
            "ttmlDestinationSettings (p. 204)" : {
                "stylePassthrough (p. 297)" : enum
            }
        },
        "destinationType (p. 204)" : enum,
        "dvbSubDestinationSettings (p. 205)" : {
            "xPosition (p. 213)" : integer,
            "backgroundColor (p. 213)" : enum,
            "teletextSpacing (p. 213)" : enum,
            "yPosition (p. 213)" : integer,
            "backgroundColor (p. 214)" : integer,
            "shadowOpacity (p. 214)" : integer,
            "fontOpacity (p. 214)" : integer,
            "shadowOpacity (p. 214)" : integer,
            "fontResolution (p. 214)" : integer,
            "shadowXOffset (p. 214)" : integer,
            "outlineSize (p. 215)" : integer,
            "outlineColor (p. 215)" : enum,
            "fontSize (p. 215)" : integer,
            "shadowXOffset (p. 215)" : integer,
            "alignment (p. 215)" : enum,
            "shadowColor (p. 216)" : enum,
            "fontColor (p. 216)" : enum
        },
        "sccDestinationSettings (p. 205)" : {
            "framerate (p. 293)" : enum
        }
    }
}
"nameModifier (p. 287)" : "string",
"dashIsoGroupSettings (p. 288)" : {
"fragmentLength (p. 209)" : integer,
"baseUrl (p. 209)" : "string",
"minBufferTime (p. 210)" : integer,
"encryption (p. 210)" : {
"spekeKeyProvider (p. 209)" : {
"resourceId (p. 293)" : "string",
"systemIds (p. 294)" : [ "string"
],
"url (p. 294)" : "string"
}
},
"destination (p. 210)" : "string",
"segmentLength (p. 210)" : integer,
"segmentControl (p. 210)" : enum,
"hbbtvCompliance (p. 210)" : enum
},
"fileGroupSettings (p. 288)" : {
"destination (p. 223)" : "string"
},
"msSmoothGroupSettings (p. 288)" : {
"fragmentLength (p. 283)" : integer,
"encryption (p. 283)" : {
"spekeKeyProvider (p. 283)" : {
"resourceId (p. 293)" : "string",
"systemIds (p. 294)" : [ "string"
],
"url (p. 294)" : "string"
}
},
"audioDeduplication (p. 283)" : enum,
"manifestEncoding (p. 283)" : enum,
"destination (p. 283)" : "string"
},
"type (p. 288)" : enum,
"hlsGroupSettings (p. 288)" : {
"segmentsPerSubdirectory (p. 247)" : integer,
"streamInfResolution (p. 247)" : enum,
"timestampDeltaMilliseconds (p. 247)" : integer,
"outputSelection (p. 248)" : enum,
"captionLanguageMappings (p. 248)" : [ {
"languageDescription (p. 245)" : "string",
"captionChannel (p. 245)" : integer,
"languageCode (p. 246)" : enum
} ]
},
"clientCache (p. 248)" : enum,
"codecSpecification (p. 248)" : enum,
"destination (p. 248)" : "string",
"segmentControl (p. 248)" : enum,
"timedMetadataId3Frame (p. 248)" : enum,
"timedMetadataId3Period (p. 248)" : integer,
"captionLanguageSetting (p. 249)" : enum,
"minSegmentLength (p. 249)" : integer,
"directoryStructure (p. 249)" : enum,
"programDateTime (p. 249)" : enum,
"baseUrl (p. 249)" : "string",
"segmentLength (p. 210)" : integer,
"segmentControl (p. 210)" : enum,
"hbbtvCompliance (p. 210)" : enum
},
"fileGroupSettings (p. 288)" : {
"destination (p. 223)" : "string"
},
"msSmoothGroupSettings (p. 288)" : {
"fragmentLength (p. 283)" : integer,
"encryption (p. 283)" : {
"spekeKeyProvider (p. 283)" : {
"resourceId (p. 293)" : "string",
"systemIds (p. 294)" : [ "string"
],
"url (p. 294)" : "string"
}
},
"audioDeduplication (p. 283)" : enum,
"manifestEncoding (p. 283)" : enum,
"destination (p. 283)" : "string"
},
"type (p. 288)" : enum,
"hlsGroupSettings (p. 288)" : {
"segmentsPerSubdirectory (p. 247)" : integer,
"streamInfResolution (p. 247)" : enum,
"timestampDeltaMilliseconds (p. 247)" : integer,
"outputSelection (p. 248)" : enum,
"captionLanguageMappings (p. 248)" : [ {
"languageDescription (p. 245)" : "string",
"captionChannel (p. 245)" : integer,
"languageCode (p. 246)" : enum
} ]
},
"clientCache (p. 248)" : enum,
"codecSpecification (p. 248)" : enum,
"destination (p. 248)" : "string",
"segmentControl (p. 248)" : enum,
"timedMetadataId3Frame (p. 248)" : enum,
"timedMetadataId3Period (p. 248)" : integer,
"captionLanguageSetting (p. 249)" : enum,
"minSegmentLength (p. 249)" : integer,
"directoryStructure (p. 249)" : enum,
"programDateTime (p. 249)" : enum,
"baseUrl (p. 249)" : "string",
"segmentLength (p. 210)" : integer,
"segmentControl (p. 210)" : enum,
"hbbtvCompliance (p. 210)" : enum
},
"fileGroupSettings (p. 288)" : {
"destination (p. 223)" : "string"
},
"msSmoothGroupSettings (p. 288)" : {
"fragmentLength (p. 283)" : integer,
"encryption (p. 283)" : {
"spekeKeyProvider (p. 283)" : {
"resourceId (p. 293)" : "string",
"systemIds (p. 294)" : [ "string"
],
"url (p. 294)" : "string"
}
},
"audioDeduplication (p. 283)" : enum,
"manifestEncoding (p. 283)" : enum,
"destination (p. 283)" : "string"
},
"type (p. 288)" : enum,
"encryption (p. 249)": {
  "initializationVectorInManifest (p. 246)": enum,
  "constantInitializationVector (p. 246)": "string",
  "staticKeyProvider (p. 247)": {
    "keyFormatVersions (p. 294)": "string",
    "keyFormat (p. 294)": "string",
    "staticKeyValue (p. 294)": "string",
    "url (p. 294)": "string"
  },
  "type (p. 247)": enum,
  "spekeKeyProvider (p. 247)": {
    "resourceId (p. 293)": "string",
    "systemIds (p. 294)": [
      "string"
    ],
    "url (p. 294)": "string"
  },
  "encryptionMethod (p. 247)": enum
},
"adMarkers (p. 249)": [
  enum
],
"programDateTimePeriod (p. 249)": integer,
"manifestCompression (p. 250)": enum,
"segmentLength (p. 250)": integer,
"manifestDurationFormat (p. 250)": enum
"name (p. 287)": "string",
"customName (p. 287)": "string"
},
"timedMetadataInsertion (p. 260)": {
  "id3Insertions (p. 297)": [
    {
      "id3 (p. 252)": "string",
      "timecode (p. 252)": "string"
    }
  ],
  "availBlanking (p. 260)": {
    "availBlankingImage (p. 200)": "string"
  },
  "lastUpdated (p. 258)": "string",
  "createdAt (p. 258)": "string",
  "name (p. 258)": "string",
  "description (p. 258)": "string",
  "category (p. 258)": "string",
  "type (p. 259)": enum,
  "arn (p. 259)": "string",
  "queue (p. 259)": "string"
}
}

**Example UpdateJobTemplateResponse**

```json
{
  "jobTemplate (p. 298)": {
    "settings (p. 258)": {
      "timecodeConfig (p. 259)": {
        "timestampOffset (p. 296)": "string",
        "anchor (p. 296)": "string",
        "start (p. 296)": "string"
      }
    }
  }
}
```
"source (p. 297)": enum,
"adAvailOffset (p. 259)": integer,
"nielsenConfiguration (p. 259)": {
  "distributorId (p. 284)": "string",
  "breakoutCode (p. 284)": integer
},
"inputs (p. 259)": [
  {
    "audioSelectors (p. 254)": {
    },
    "audioSelectorGroups (p. 254)": {
    },
    "filterEnable (p. 254)": enum,
    "deblockFilter (p. 254)": enum,
    "videoSelector (p. 254)": {
      "colorSpace (p. 302)": enum,
      "hdr10Metadata (p. 302)": {
        "redPrimaryY (p. 243)": integer,
        "greenPrimaryY (p. 243)": integer,
        "whitePointX (p. 243)": integer,
        "maxLuminance (p. 243)": integer,
        "greenPrimaryX (p. 243)": integer,
        "whitePointY (p. 244)": integer,
        "redPrimaryX (p. 244)": integer,
        "bluePrimaryX (p. 244)": integer,
        "maxFrameAverageLightLevel (p. 244)": integer,
        "bluePrimaryY (p. 244)": integer,
        "maxContentLightLevel (p. 244)": integer,
        "minLuminance (p. 245)": integer
      },
      "programNumber (p. 302)": integer,
      "pid (p. 303)": integer,
      "colorSpaceUsage (p. 303)": enum
    },
    "filterStrength (p. 255)": integer,
    "programNumber (p. 255)": integer,
    "timecodeSource (p. 255)": enum,
    "captionSelectors (p. 255)": {
    },
    "denoiseFilter (p. 255)": enum,
    "psiControl (p. 255)": enum,
    "inputClippings (p. 255)": [
      {
        "startTimecode (p. 253)": "string",
        "endTimecode (p. 253)": "string"
      }
    ]
  }
},
"outputGroups (p. 260)": [
  {
    "outputs (p. 287)": [
    {
      "extension (p. 286)": "string",
      "videoDescription (p. 286)": {
        "fixedAfd (p. 299)": integer,
        "scalingBehavior (p. 300)": enum,
        "respondToAfd (p. 300)": enum,
        "codecSettings (p. 300)": {
          "h265Settings (p. 299)": {
            "slices (p. 237)": integer,
            "minInterval (p. 237)": integer,
            "parNumerator (p. 237)": integer,
            "flickerAdaptiveQuantization (p. 237)": enum,
            "gopSizeUnits (p. 237)": enum,
            "parDenominator (p. 237)": integer,
            "mode (p. 237)": enum,
            "iFrameInterval (p. 237)": integer,
            "parDenominator (p. 237)": integer,
            "gopSizeUnits (p. 237)": enum,
            "gopSize (p. 237)": integer
          }
        }
      }
    }
  }
]
"hrdBufferSize (p. 238)": integer,
"qualityTuningLevel (p. 238)": enum,
"maxBitrate (p. 238)": integer,
"bitrate (p. 238)": integer,
"spatialAdaptiveQuantization (p. 238)": enum,
"sampleAdaptiveOffsetFilterMode (p. 238)": enum,
"temporalIds (p. 238)": enum,
"slowPal (p. 239)": enum,
"tiles (p. 239)": enum,
"codecProfile (p. 239)": enum,
"alternateTransferFunctionSei (p. 239)": enum,
"unregisteredSeiTimecode (p. 239)": enum,
"framerateControl (p. 239)": enum,
"telecine (p. 239)": enum,
"framerateConversionAlgorithm (p. 239)": enum,
"codecLevel (p. 239)": enum,
"numberReferenceFrames (p. 239)": integer,
"temporalAdaptiveQuantization (p. 240)": enum,
"hrdBufferInitialFillPercentage (p. 240)": integer,
"framerateClosedCadence (p. 240)": integer,
"framerateConversionAlgorithm (p. 240)": integer,
"parNumerator (p. 240)": integer,
"parDenominator (p. 241)": integer,
"framerateControl (p. 241)": enum,
"telecine (p. 241)": enum,
"framerateConversionAlgorithm (p. 241)": enum,
"interlaceMode (p. 241)": enum,
"gopSize (p. 241)": number,
"gopReference (p. 241)": enum,
"sceneChangeDetect (p. 241)": enum,
"parDenominator (p. 241)": integer,
"parControl (p. 241)": enum,
"rateControlMode (p. 241)": enum,
 codec (p. 299)": enum,
"proresSettings (p. 299)": {
  "slowPal (p. 290)": enum,
  "framerateControl (p. 290)": enum,
  "telecine (p. 290)": enum,
  "framerateConversionAlgorithm (p. 290)": enum,
  "numberReferenceFrames (p. 290)": integer,
  "codecProfile (p. 290)": enum,
  "parNumerator (p. 290)": integer,
  "parControl (p. 291)": enum,
  "parDenominator (p. 291)": integer,
  "framerateNumerator (p. 291)": integer,
},
"mpeg2Settings (p. 299)": {
  "minInterval (p. 278)": integer,
  "parNumerator (p. 278)": integer,
  "gopSizeUnits (p. 278)": enum,
  "hrdBufferSize (p. 278)": integer,
  "qualityTuningLevel (p. 279)": enum,
  "maxBitrate (p. 279)": integer,
  "bitrate (p. 279)": integer,
  "spatialAdaptiveQuantization (p. 279)": enum,
  "slowPal (p. 279)": enum,
  "codecProfile (p. 279)": enum,
  "intraDcPrecision (p. 279)": enum,
  "softness (p. 279)": integer,
  "framerateControl (p. 280)": enum,
  "telecine (p. 280)": enum,
  "framerateConversionAlgorithm (p. 280)": enum,
  "codecLevel (p. 280)": enum,
  "temporalAdaptiveQuantization (p. 280)": enum,
  "hrdBufferInitialFillPercentage (p. 280)": integer,
  "framerateNumerator (p. 281)": integer,
"
"numberBFramesBetweenReferenceFrames (p. 280)": integer,
"gopClosedCadence (p. 281)": integer,
"framerateDenominator (p. 281)": integer,
"adaptiveQuantization (p. 281)": enum,
"interlaceMode (p. 281)": enum,
"gopSize (p. 281)": number,
"sceneChangeDetect (p. 281)": enum,
"parDenominator (p. 281)": integer,
"parControl (p. 282)": enum,
"syntax (p. 282)": enum,
"rateControlMode (p. 282)": enum
},
"h264Settings (p. 299)": {
  "slices (p. 228)": integer,
  "minIInterval (p. 228)": integer,
  "parNumerator (p. 228)": integer,
  "flickerAdaptiveQuantization (p. 229)": enum,
  "gopSizeUnits (p. 229)": enum,
  "hrdBufferSize (p. 229)": integer,
  "qualityTuningLevel (p. 229)": enum,
  "maxBitrate (p. 229)": integer,
  "bitrate (p. 229)": integer,
  "spatialAdaptiveQuantization (p. 229)": enum,
  "slowPal (p. 230)": enum,
  "codecProfile (p. 230)": enum,
  "unregisteredSeiTimecode (p. 230)": enum,
  "softness (p. 230)": integer,
  "framerateControl (p. 230)": enum,
  "telecine (p. 230)": enum,
  "framerateConversionAlgorithm (p. 230)": enum,
  "codecLevel (p. 230)": enum,
  "numberReferenceFrames (p. 230)": integer,
  "temporalAdaptiveQuantization (p. 231)": enum,
  "repeatPps (p. 231)": enum,
  "hrdBufferInitialFillPercentage (p. 231)": integer,
  "framerateNumerator (p. 231)": integer,
  "numberBFramesBetweenReferenceFrames (p. 231)": integer,
  "gopClosedCadence (p. 231)": integer,
  "fieldEncoding (p. 231)": enum,
  "entropyEncoding (p. 232)": enum,
  "framerateDenominator (p. 232)": integer,
  "adaptiveQuantization (p. 232)": enum,
  "interlaceMode (p. 232)": enum,
  "gopSize (p. 232)": number,
  "gopBReference (p. 232)": enum,
  "parDenominator (p. 232)": integer,
  "parControl (p. 233)": enum,
  "syntax (p. 233)": enum,
  "rateControlMode (p. 233)": enum
},
"frameCaptureSettings (p. 299)": {
  "framerateDenominator (p. 224)": integer,
  "maxCaptures (p. 224)": integer,
  "framerateNumerator (p. 225)": integer,
  "quality (p. 225)": integer
},
"afdSignaling (p. 300)": enum,
"colorMetadata (p. 300)": enum,
"timecodeInsertion (p. 300)": enum,
"width (p. 300)": integer,
"sharpness (p. 300)": integer,
"antiAlias (p. 301)": enum,
"videoPreprocessors (p. 301)": {
  "timecodeBurnin (p. 301)": {
"prefix (p. 295)": "string",
"fontSize (p. 295)": integer,
"position (p. 295)": enum
},
"noiseReducer (p. 302)": {
  "filter (p. 284)": enum,
  "filterSettings (p. 284)": {
    "strength (p. 285)": integer
  },
  "spatialFilterSettings (p. 284)": {
    "strength (p. 285)": integer,
    "postFilterSharpenStrength (p. 285)": integer,
    "speed (p. 285)": integer
  }
},
"colorCorrector (p. 302)": {
  "saturation (p. 206)": integer,
  "brightness (p. 207)": integer,
  "hdr10Metadata (p. 207)": {
    "redPrimaryY (p. 243)": integer,
    "greenPrimaryY (p. 243)": integer,
    "whitePointX (p. 243)": integer,
    "maxLuminance (p. 243)": integer,
    "greenPrimaryX (p. 243)": integer,
    "whitePointY (p. 244)": integer,
    "redPrimaryX (p. 244)": integer,
    "bluePrimaryX (p. 244)": integer,
    "maxFrameAverageLightLevel (p. 244)": integer,
    "bluePrimaryY (p. 244)": integer,
    "maxContentLightLevel (p. 244)": integer,
    "minLuminance (p. 245)": integer
  },
  "contrast (p. 207)": integer,
  "hue (p. 207)": integer,
  "colorSpaceConversion (p. 207)": enum
},
"imageInserter (p. 302)": {
  "insertableImages (p. 252)": [
    {
      "duration (p. 256)": integer,
      "fadeOut (p. 256)": integer,
      "imageY (p. 256)": integer,
      "fadeIn (p. 256)": integer,
      "imageX (p. 257)": integer,
      "width (p. 257)": integer,
      "startTime (p. 257)": "string",
      "opacity (p. 257)": integer,
      "layer (p. 257)": integer,
      "height (p. 257)": integer,
      "imageInserterInput (p. 258)": "string"
    }
  ]
},
"deinterlacer (p. 302)": {
  "mode (p. 211)": enum,
  "control (p. 211)": enum,
  "algorithm (p. 211)": enum
},
"position (p. 301)": {
  "width (p. 291)": integer,
  "x (p. 292)": integer,
  "y (p. 292)": integer,
  "height (p. 292)": integer
},
"dropFrameTimecode (p. 301)": enum,
"crop (p. 301)": {
    "width (p. 291)": integer,
    "x (p. 292)": integer,
    "y (p. 292)": integer,
    "height (p. 292)": integer
},
"height (p. 301)": integer
},
"audioDescriptions (p. 286)": [
    {
        "languageCodeControl (p. 195)": enum,
        "audioTypeControl (p. 195)": enum,
        "remixSettings (p. 195)": {
            "channelsOut (p. 292)": integer,
            "channelMapping (p. 292)": {
                "outputChannels (p. 206)": [
                    {
                        "inputChannels (p. 287)": [integer
                    ]
                ]
            }
        },
        "channelsIn (p. 292)": integer
    },
    "audioType (p. 195)": integer,
    "audioSourceName (p. 195)": "string",
    "codecSettings (p. 196)": {
        "codec (p. 194)": enum,
        "wavSettings (p. 194)": {
            "channels (p. 303)": integer,
            "bitDepth (p. 303)": integer,
            "sampleRate (p. 303)": integer
        },
        "ac3Settings (p. 194)": {
            "dynamicRangeCompressionProfile (p. 192)": enum,
            "dialnorm (p. 192)": integer,
            "codingMode (p. 192)": enum,
            "metadataControl (p. 192)": enum,
            "lfeFilter (p. 192)": enum,
            "bitrate (p. 192)": integer,
            "bitstreamMode (p. 192)": enum,
            "sampleRate (p. 192)": integer
        },
        "aacSettings (p. 194)": {
            "vbrQuality (p. 189)": enum,
            "codecProfile (p. 189)": enum,
            "codingMode (p. 190)": enum,
            "specification (p. 190)": enum,
            "bitrate (p. 190)": integer,
            "rawFormat (p. 190)": enum,
            "rateControlMode (p. 190)": enum,
            "sampleRate (p. 190)": integer,
            "audioDescriptionBroadcasterMix (p. 190)": enum
        },
        "aiffSettings (p. 194)": {
            "channels (p. 193)": integer,
            "bitDepth (p. 193)": integer,
            "sampleRate (p. 193)": integer
        },
        "eac3Settings (p. 194)": {
            "dialnorm (p. 219)": integer,
            "passthroughControl (p. 219)": enum,
            "metadataControl (p. 219)": enum,
            "bitrate (p. 219)": integer,
            "dynamicRangeCompressionRf (p. 220)": enum,
            "dynamicRangeCompressionProfile (p. 220)": enum,
"sampleRate (p. 220)": integer,
"lRtSurroundMixLevel (p. 220)": number,
"surroundExMode (p. 220)": enum,
"dynamicRangeCompressionLine (p. 220)": enum,
"lfeControl (p. 220)": enum,
"codingMode (p. 220)": enum,
"surroundMode (p. 220)": enum,
"attenuationControl (p. 221)": enum,
"lfeFilter (p. 221)": enum,
"phaseControl (p. 221)": enum,
"lRtCenterMixLevel (p. 221)": number,
"dcFilter (p. 221)": enum,
"stereoDownmix (p. 221)": enum,
"bitstreamMode (p. 221)": enum,
"loRoSurroundMixLevel (p. 221)": number,
"loRoCenterMixLevel (p. 222)": number
},
"mp2Settings (p. 195)": {
  "channels (p. 274)": integer,
  "bitrate (p. 275)": integer,
  "sampleRate (p. 275)": integer
},
"languageCode (p. 196)": enum,
"StreamName (p. 196)": "string",
"audioNormalizationSettings (p. 196)": {
  "targetLkfs (p. 197)": number,
  "algorithmControl (p. 197)": enum,
  "peakCalculation (p. 197)": enum,
  "loudnessLogging (p. 197)": enum,
  "correctionGateLevel (p. 197)": integer,
  "algorithm (p. 198)": enum
},
"containerSettings (p. 286)": {
  "container (p. 208)": enum,
  "mp4Settings (p. 208)": {
    "mp4MajorBrand (p. 275)": "string",
    "moovPlacement (p. 276)": enum,
    "csilogAtom (p. 276)": enum,
    "freeSpaceBox (p. 276)": enum
  },
  "m3u8Settings (p. 208)": {
    "pmtPid (p. 271)": integer,
    "pcrPid (p. 271)": integer,
    "audioPids (p. 271)": [integer,
    "audioFramesPerPes (p. 271)": integer,
    "scte35Pid (p. 271)": integer,
    "transportStreamId (p. 272)": integer,
    "videoPid (p. 272)": integer,
    "pcrControl (p. 272)": enum,
    "privateMetadataPid (p. 272)": integer,
    "pmtInterval (p. 272)": integer,
    "patInterval (p. 272)": integer,
    "programNumber (p. 273)": integer,
    "timedMetadataPid (p. 273)": integer,
    "timedMetadata (p. 273)": enum,
    "scte35Source (p. 273)": enum
  },
  "m2tsSettings (p. 208)": {
    "dvbTeletextPid (p. 266)": integer,
    "bitrate (p. 266)": integer,
    "segmentationTime (p. 266)": number,
"audioPids (p. 266)": [ integer ],
"rateMode (p. 266)": enum,
"ebpAudioInterval (p. 266)": enum,
"fragmentTime (p. 267)": number,
"audioFramesPerPes (p. 267)": integer,
"maxPcrInterval (p. 267)": integer,
"scte35Pid (p. 267)": integer,
"privateMetadataPid (p. 267)": integer,
"pmtInterval (p. 267)": integer,
"segmentationStyle (p. 268)": enum,
"audioBufferModel (p. 268)": enum,
"programNumber (p. 268)": integer,
"dvbNitSettings (p. 268)": {
  "networkName (p. 212)": "string",
  "networkId (p. 212)": integer,
  "nitInterval (p. 212)": integer
},
"scte35Source (p. 268)": enum,
"pmtPid (p. 268)": integer,
"bufferModel (p. 268)": enum,
"ebpPlacement (p. 268)": enum,
"dvbSdtSettings (p. 269)": {
  "sdtInterval (p. 213)": integer,
  "serviceName (p. 213)": "string",
  "serviceProviderName (p. 213)": "string",
  "outputSdt (p. 213)": enum
},
"nullPacketBitrate (p. 269)": number,
"pcrPid (p. 269)": integer,
"minEbpInterval (p. 269)": integer,
"transportStreamId (p. 269)": integer,
"pcrControl (p. 269)": enum,
"videoPid (p. 270)": integer,
"esRateInPes (p. 270)": integer,
"segmentationMarkers (p. 270)": enum,
"dvbTdtSettings (p. 270)": {
  "tdtInterval (p. 217)": integer
},
"patInterval (p. 270)": integer,
"dvbSubPids (p. 270)": [ integer ]
},
"movSettings (p. 209)": {
  "reference (p. 274)": enum,
  "paddingControl (p. 274)": enum,
  "mpeg2FourCCControl (p. 274)": enum,
  "csigAtom (p. 274)": enum,
  "clapAtom (p. 274)": enum
},
"f4vSettings (p. 209)": {
  "moovPlacement (p. 223)": enum
},
"preset (p. 286)": "string",
"outputSettings (p. 286)": {
  "hlsSettings (p. 288)": {
    "iFrameOnlyManifest (p. 251)": enum,
    "segmentModifier (p. 251)": "string",
    "audioTrackType (p. 251)": enum,
    "audioRenditionSets (p. 252)": "string",
    "audioGroupId (p. 252)": "string"
  }
}
"captionDescriptions (p. 287)": [
    
    "captionSelectorName (p. 204)": "string",
    "languageDescription (p. 204)": "string",
    "languageCode (p. 204)": enum,
    "destinationSettings (p. 204)": {
        "burninDestinationSettings (p. 204)": {
            "xPosition (p. 200)": integer,
            "backgroundColor (p. 200)": enum,
            "teletextSpacing (p. 200)": enum,
            "yPosition (p. 200)": integer,
            "backgroundOpacity (p. 201)": integer,
            "fontOpacity (p. 201)": integer,
            "fontResolution (p. 201)": integer,
            "shadowOpacity (p. 201)": integer,
            "fontName (p. 201)": integer,
            "fontColor (p. 202)": enum,
            "fontSize (p. 202)": integer,
            "shadowXOffset (p. 202)": integer,
            "alignment (p. 202)": enum,
            "shadowColor (p. 202)": enum,
            "fontColor (p. 202)": enum
        },
        "teletextDestinationSettings (p. 204)": {
            "pageNumber (p. 295)": "string"
        },
        "ttmlDestinationSettings (p. 204)": {
            "stylePassthrough (p. 297)": enum
        },
        "destinationType (p. 204)": enum,
        "dvbSubDestinationSettings (p. 205)": {
            "xPosition (p. 213)": integer,
            "backgroundColor (p. 213)": enum,
            "teletextSpacing (p. 214)": enum,
            "yPosition (p. 214)": integer,
            "backgroundOpacity (p. 214)": integer,
            "fontOpacity (p. 214)": integer,
            "fontResolution (p. 214)": integer,
            "shadowOpacity (p. 214)": integer,
            "fontName (p. 214)": integer,
            "fontColor (p. 216)": enum,
            "fontSize (p. 215)": integer,
            "shadowXOffset (p. 215)": integer,
            "alignment (p. 215)": enum,
            "shadowColor (p. 215)": enum,
            "fontColor (p. 215)": enum
        },
        "sccDestinationSettings (p. 205)": {
            "framerate (p. 293)": enum
        }
    },
    "nameModifier (p. 287)": "string"
],

"outputGroupSettings (p. 287)": {
    "dashIsoGroupSettings (p. 288)": {
        "fragmentLength (p. 209)": integer,
        "baseUrl (p. 209)": "string",
        "minBufferTime (p. 210)": integer,
        "encryption (p. 210)": {
            "spekeKeyProvider (p. 209)": {
                "resourceId (p. 293)": "string",
            }
        },
    }
}
"systemIds (p. 294)": [ "string" ],
"url (p. 294)": "string"
},
"destination (p. 210)": "string",
"segmentLength (p. 210)": integer,
"segmentControl (p. 210)": enum,
"hbbtvCompliance (p. 210)": enum
},
"fileGroupSettings (p. 288)": {
"destination (p. 223)": "string"
},
"msSmoothGroupSettings (p. 288)": {
"fragmentLength (p. 283)": integer,
"encryption (p. 283)": {
"spekeKeyProvider (p. 283)": {
"resourceId (p. 293)": "string",
"systemIds (p. 294)": [ "string"
],
"url (p. 294)": "string"
}
},
"audioDeduplication (p. 283)": enum,
"manifestEncoding (p. 283)": enum,
"destination (p. 283)": "string"
},
"type (p. 288)": enum,
"hlsGroupSettings (p. 288)": {
"segmentsPerSubdirectory (p. 247)": integer,
"streamInfResolution (p. 247)": enum,
"timestampDeltaMilliseconds (p. 247)": integer,
"outputSelection (p. 248)": enum,
"captionLanguageMappings (p. 248)": [ {
"languageDescription (p. 245)": "string",
"captionChannel (p. 245)": integer,
"languageCode (p. 246)": enum
}
],
"clientCache (p. 248)": enum,
"codecSpecification (p. 248)": enum,
"destination (p. 248)": "string",
"segmentControl (p. 248)": enum,
"timedMetadataId3Frame (p. 248)": enum,
"timedMetaDataId3Period (p. 248)": integer,
"captionLanguageSetting (p. 249)": enum,
"minSegmentLength (p. 249)": integer,
"directoryStructure (p. 249)": enum,
"programDateTime (p. 249)": integer,
"baseUrl (p. 249)": "string",
"encryption (p. 249)": {
"initializationVectorInManifest (p. 246)": enum,
"constantInitializationVector (p. 246)": "string",
"spekeKeyProvider (p. 247)": {
"resourceId (p. 293)": "string",
"systemIds (p. 294)": [ "string"
],
"url (p. 294)": "string"
}
},
"type (p. 247)": enum,
"spekeKeyProvider (p. 247)": {
"resourceId (p. 293)": "string",
"systemIds (p. 294)": [ "string"
],
"url (p. 294)": "string"
}
```
"string"
},
  "url (p. 294)": "string"
},
  "encryptionMethod (p. 247)": enum
},
  "adMarkers (p. 249)": [
    enum
  ],
  "programDateTimePeriod (p. 249)": integer,
  "manifestCompression (p. 250)": enum,
  "segmentLength (p. 250)": integer,
  "manifestDurationFormat (p. 250)": enum
}

"name (p. 287)": "string",
"customName (p. 287)": "string"
}],
  "timedMetadataInsertion (p. 260)": {
    "id3Insertions (p. 297)": [
      {
        "id3 (p. 252)": "string",
        "timecode (p. 252)": "string"
      }
    ],
    "availBlanking (p. 260)": {
      "availBlankingImage (p. 200)": "string"
    }
  }
},
  "lastUpdated (p. 258)": "string",
  "createdAt (p. 258)": "string",
  "name (p. 258)": "string",
  "description (p. 258)": "string",
  "category (p. 258)": "string",
  "type (p. 259)": enum,
  "arn (p. 259)": "string",
  "queue (p. 259)": "string"
}
```

**Example DeleteJobTemplateResponse**

```json
{}
```

**Example ExceptionBody**

```json
{
  "message (p. 223)": "string"
}
```

**Properties**

**AacAudioDescriptionBroadcasterMix (Enum)**

Choose BROADCASTER_MIXED_AD when the input contains pre-mixed main audio + audio description (AD) as a stereo pair. The value for AudioType will be set to 3, which signals to downstream systems that this stream contains “broadcaster mixed AD”. Note that the input received by the encoder must contain pre-mixed audio; the encoder does not perform the mixing. When you
choose BROADCASTER_MIXED_AD, the encoder ignores any values you provide in AudioType and FollowInputAudioType. Choose NORMAL when the input does not contain pre-mixed audio + audio description (AD). In this case, the encoder will use any values you provide for AudioType and FollowInputAudioType.

BROADCASTER_MIXED_AD
NORMAL

**AacCodecProfile (Enum)**

AAC Profile.

LC
HEV1
HEV2

**AacCodingMode (Enum)**

Mono (Audio Description), Mono, Stereo, or 5.1 channel layout. Valid values depend on rate control mode and profile. "1.0 - Audio Description (Receiver Mix)" setting receives a stereo description plus control track and emits a mono AAC encode of the description track, with control data emitted in the PES header as per ETSI TS 101 154 Annex E.

AD_RECEIVER_MIX
CODING_MODE_1_0
CODING_MODE_1_1
CODING_MODE_2_0
CODING_MODE_5_1

**AacRateControlMode (Enum)**

Rate Control Mode.

CBR
VBR

**AacRawFormat (Enum)**

Enables LATM/LOAS AAC output. Note that if you use LATM/LOAS AAC in an output, you must choose "No container" for the output container.

LATM_LOAS
NONE

**AacSettings**

vbrQuality

Type: AacVbrQuality (p. 191)
Required: False

codecProfile

Type: AacCodecProfile (p. 189)
Properties

**Required**: False

**codingMode**

Type: AacCodingMode (p. 189)
Required: True

**specification**

Type: AacSpecification (p. 190)
Required: False

**bitrate**

Average bitrate in bits/second. Valid values depend on rate control mode and profile.

Type: integer
Required: False
Minimum: 6000
Maximum: 1024000

**rawFormat**

Type: AacRawFormat (p. 189)
Required: False

**rateControlMode**

Type: AacRateControlMode (p. 189)
Required: False

**sampleRate**

Sample rate in Hz. Valid values depend on rate control mode and profile.

Type: integer
Required: True
Minimum: 8000
Maximum: 96000

**audioDescriptionBroadcasterMix**

Type: AacAudioDescriptionBroadcasterMix (p. 188)
Required: False

**AacSpecification (Enum)**

Use MPEG-2 AAC instead of MPEG-4 AAC audio for raw or MPEG-2 Transport Stream containers.

MPEG2
MPEG4
AacVbrQuality (Enum)

VBR quality level. Only used if the rate control mode (AacRateControlMode) is VBR.

LOW
MEDIUM_LOW
MEDIUM_HIGH
HIGH

Ac3BitstreamMode (Enum)

Specifies the "Bitstream Mode" (bsmod) for the emitted AC-3 stream. See ATSC A/52-2012 for background on these values.

COMPLETE_MAIN
COMMENTARY
DIALOGUE
EMERGENCY
HEARING_IMPAIRED
MUSIC_AND_EFFECTS
VISUALLY_IMPAIRED
VOICE_OVER

Ac3CodingMode (Enum)

Dolby Digital coding mode. Determines number of channels.

CODING_MODE_1_0
CODING_MODE_1_1
CODING_MODE_2_0
CODING_MODE_3_2_LFE

Ac3DynamicRangeCompressionProfile (Enum)

If set to FILM_STANDARD, adds dynamic range compression signaling to the output bitstream as defined in the Dolby Digital specification.

FILM_STANDARD
NONE

Ac3LfeFilter (Enum)

Applies a 120Hz lowpass filter to the LFE channel prior to encoding. Only valid with 3_2_LFE coding mode.

ENABLED
DISABLED

Ac3MetadataControl (Enum)

When set to FOLLOW_INPUT, encoder metadata will be sourced from the DD, DD+, or DolbyE decoder that supplied this audio data. If audio was not supplied from one of these streams, then the static metadata settings will be used.

FOLLOW_INPUT
USE_CONFIGURED
**Ac3Settings**

**dynamicRangeCompressionProfile**

- **Type:** Ac3DynamicRangeCompressionProfile (p. 191)
- **Required:** False

**dialnorm**
Sets the dialnorm for the output. If blank and input audio is Dolby Digital, dialnorm will be passed through.

- **Type:** integer
- **Required:** False
- **Minimum:** 1
- **Maximum:** 31

**codingMode**

- **Type:** Ac3CodingMode (p. 191)
- **Required:** False

**metadataControl**

- **Type:** Ac3MetadataControl (p. 191)
- **Required:** False

**lfeFilter**

- **Type:** Ac3LfeFilter (p. 191)
- **Required:** False

**bitrate**
Average bitrate in bits/second. Valid bitrates depend on the coding mode.

- **Type:** integer
- **Required:** False
- **Minimum:** 64000
- **Maximum:** 640000

**bitstreamMode**

- **Type:** Ac3BitstreamMode (p. 191)
- **Required:** False

**sampleRate**
Sample rate in hz. Sample rate is always 48000.

- **Type:** integer
- **Required:** False
- **Minimum:** 48000
- **Maximum:** 48000
Properties

**AfdSignaling (Enum)**

This setting only applies to H.264 and MPEG2 outputs. Use Insert AFD signaling (AfdSignaling) to whether there are AFD values in the output video data and what those values are.

* Choose None to remove all AFD values from this output.
* Choose Fixed to ignore input AFD values and instead encode the value specified in the job.
* Choose Auto to calculate output AFD values based on the input AFD scaler data.

    NONE
    AUTO
    FIXED

**AiffSettings**

**channels**

Set Channels to specify the number of channels in this output audio track. Choosing Mono in the console will give you 1 output channel; choosing Stereo will give you 2. In the API, valid values are 1 and 2.

* Type: integer
* Required: False
* Minimum: 1
* Maximum: 2

**bitDepth**

Specify Bit depth (BitDepth), in bits per sample, to choose the encoding quality for this audio track.

* Type: integer
* Required: False
* Minimum: 16
* Maximum: 24

**sampleRate**

Sample rate in hz.

* Type: integer
* Required: False
* Minimum: 8000
* Maximum: 192000

**AncillarySourceSettings**

**sourceAncillaryChannelNumber**

Specifies the 608 channel number in the ancillary data track from which to extract captions. Unused for passthrough.

* Type: integer
* Required: False
* Minimum: 1
* Maximum: 4
Properties

**AntiAlias (Enum)**

Enable Anti-alias (AntiAlias) to enhance sharp edges in video output when your input resolution is much larger than your output resolution. Default is enabled.

- DISABLED
- ENABLED

**AudioCodec (Enum)**

Type of Audio codec.

- AAC
- MP2
- WAV
- AIFF
- AC3
- EAC3
- PASSTHROUGH

**AudioCodecSettings**

- **codec**
  - *Type:* AudioCodec (p. 194)
  - *Required:* True

- **wavSettings**
  - *Type:* WavSettings (p. 303)
  - *Required:* False

- **ac3Settings**
  - *Type:* Ac3Settings (p. 192)
  - *Required:* False

- **aacSettings**
  - *Type:* AacSettings (p. 189)
  - *Required:* False

- **aiffSettings**
  - *Type:* AiffSettings (p. 193)
  - *Required:* False

- **eac3Settings**
  - *Type:* Eac3Settings (p. 219)
  - *Required:* False
mp2Settings

Type: Mp2Settings (p. 274)
Required: False

AudioDefaultSelection (Enum)

When an "Audio Description" specifies an AudioSelector or AudioSelectorGroup for which no matching source is found in the input, then the audio selector marked as DEFAULT will be used. If none are marked as default, silence will be inserted for the duration of the input.

- DEFAULT
- NOT_DEFAULT

AudioDescription

languageCodeControl

Type: AudioLanguageCodeControl (p. 196)
Required: False

audioTypeControl

Type: AudioTypeControl (p. 199)
Required: False

remixSettings

Advanced audio remixing settings.

Type: RemixSettings (p. 292)
Required: False

audioType

Applies only if Follow Input Audio Type is unchecked (false). A number between 0 and 255. The following are defined in ISO-IEC 13818-1: 0 = Undefined, 1 = Clean Effects, 2 = Hearing Impaired, 3 = Visually Impaired Commentary, 4-255 = Reserved.

Type: integer
Required: False
Minimum: 0
Maximum: 255

audioSourceName

Specifies which audio data to use from each input. In the simplest case, specify an "Audio Selector" by name based on its order within each input. For example if you specify "Audio Selector 3", then the third audio selector will be used from each input. If an input does not have an "Audio Selector 3", then the audio selector marked as "default" in that input will be used. If there is no audio selector marked as "default", silence will be inserted for the duration of that input. Alternatively, an "Audio Selector Group" name may be specified, with similar default/silence behavior. If no audio_source_name is specified, then "Audio Selector 1" will be chosen automatically.
**Properties**

**Type**: string  
**Required**: False

**codecSettings**

**Type**: AudioCodecSettings (p. 194)  
**Required**: True

**languageCode**

Indicates the language of the audio output track. The ISO 639 language specified in the 'Language Code' drop down will be used when 'Follow Input Language Code' is not selected or when 'Follow Input Language Code' is selected but there is no ISO 639 language code specified by the input.

**Type**: LanguageCode (p. 260)  
**Required**: False

**streamName**

Used for Microsoft Smooth Streaming and Apple HLS outputs. Indicates the name displayed by the player (eg. English, or Director Commentary). Alphanumeric characters, spaces, and underscore are legal.

**Type**: string  
**Required**: False  
**Pattern**: ^\[\w\s\]*$

**audioNormalizationSettings**

Advanced audio normalization settings.

**Type**: AudioNormalizationSettings (p. 197)  
**Required**: False

**AudioLanguageCodeControl (Enum)**

Choosing FOLLOW_INPUT will cause the ISO 639 language code of the output to follow the ISO 639 language code of the input. The language specified for languageCode' will be used when USE_CONFIGURED is selected or when FOLLOW_INPUT is selected but there is no ISO 639 language code specified by the input.

- FOLLOW_INPUT
- USE_CONFIGURED

**AudioNormalizationAlgorithm (Enum)**

Audio normalization algorithm to use. 1770-1 conforms to the CALM Act specification, 1770-2 conforms to the EBU R-128 specification.

- ITU_BS_1770_1
- ITU_BS_1770_2

**AudioNormalizationAlgorithmControl (Enum)**

When enabled the output audio is corrected using the chosen algorithm. If disabled, the audio will be measured but not adjusted.
CORRECT_AUDIO
MEASURE_ONLY

AudioNormalizationLoudnessLogging (Enum)
If set to LOG, log each output's audio track loudness to a CSV file.

LOG
DONT_LOG

AudioNormalizationPeakCalculation (Enum)
If set to TRUE_PEAK, calculate and log the TruePeak for each output's audio track loudness.

TRUE_PEAK
NONE

AudioNormalizationSettings

targetLkfs
Target LKFS(loudness) to adjust volume to. If no value is entered, a default value will be used according to the chosen algorithm. The CALM Act (1770-1) recommends a target of -24 LKFS. The EBU R-128 specification (1770-2) recommends a target of -23 LKFS.

Type: number
Required: False
Format: float
Minimum: -59.0
Maximum: 0.0

algorithmControl

Type: AudioNormalizationAlgorithmControl (p. 196)
Required: False

peakCalculation

Type: AudioNormalizationPeakCalculation (p. 197)
Required: False

loudnessLogging

Type: AudioNormalizationLoudnessLogging (p. 197)
Required: False

correctionGateLevel
Content measuring above this level will be corrected to the target level. Content measuring below this level will not be corrected. Gating only applies when not using real_time_correction.

Type: integer
Required: False
Minimum: -70
Maximum: 0

**algorithm**

*Type: AudioNormalizationAlgorithm (p. 196)*  
*Required: False*

**AudioSelector**

**remixSettings**

Advanced audio remixing settings.

*Type: RemixSettings (p. 292)*  
*Required: False*

**programSelection**

Applies only when input streams contain Dolby E. Enter the program ID (according to the metadata in the audio) of the Dolby E program to extract from the specified track. One program extracted per audio selector. To select multiple programs, create multiple selectors with the same Track and different Program numbers. "All channels" means to ignore the program IDs and include all the channels in this selector; useful if metadata is known to be incorrect.

*Type: integer*  
*Required: False*  
*Minimum: 0*  
*Maximum: 8*

**offset**

Specifies a time delta in milliseconds to offset the audio from the input video.

*Type: integer*  
*Required: False*  
*Minimum: -2147483648*  
*Maximum: 2147483647*

**defaultSelection**

*Type: AudioDefaultSelection (p. 195)*  
*Required: False*

**selectorType**

*Type: AudioSelectorType (p. 199)*  
*Required: False*

**languageCode**

Selects a specific language code from within an audio source.

*Type: LanguageCode (p. 260)*  
*Required: False*
**pids**

Selects a specific PID from within an audio source (e.g. 257 selects PID 0x101).

**Type:** Array of type integer  
**Required:** False

**externalAudioFileInput**

Specifies audio data from an external file source. Auto populated when Infer External Filename is checked

**Type:** string  
**Required:** False

** tracks**

Identify the channel to include in this selector by entering the 1-based track index. To combine several tracks, enter a comma-separated list, e.g. "1,2,3" for tracks 1-3.

**Type:** Array of type integer  
**Required:** False

**AudioSelectorGroup**

**audioSelectorNames**

Name of an "Audio Selector" in the same input to include in the group. Audio selector names are standardized, based on their order within the input (e.g. "Audio Selector 1"). The audio_selector_name parameter can be repeated to add any number of audio selectors to the group.

**Type:** Array of type string  
**Required:** True

**AudioSelectorType (Enum)**

Specifies the type of the audio selector.

- **PID**
- **TRACK**
- **LANGUAGE_CODE**

**AudioTypeControl (Enum)**

When set to FOLLOW_INPUT, if the input contains an ISO 639 audio_type, then that value is passed through to the output. If the input contains no ISO 639 audio_type, the value in Audio Type is included...
in the output. Otherwise the value in Audio Type is included in the output. Note that this field and
audioType are both ignored if audioDescriptionBroadcasterMix is set to BROADCASTER_MIXED_AD.

FOLLOW_INPUT
USE_CONFIGURED

AvailBlanking

availBlankingImage

Blanking image to be used. Leave empty for solid black. Only bmp and png images are supported.

Type: string
Required: False
Pattern: ^(s3:\/\/)\.(bmp|BMP|png|PNG)$

BurninDestinationSettings

xPosition

Specifies the horizontal position of the caption relative to the left side of the output in pixels. A value of 10
would result in the captions starting 10 pixels from the left of the output. If no explicit x_position is
provided, the horizontal caption position will be determined by the alignment parameter. This option is
not valid for source captions that are STL, 608/embedded or teletext. These source settings are already
pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

backgroundColor

Type: BurninSubtitleBackgroundColor (p. 203)
Required: False

teletextSpacing

Type: BurninSubtitleTeletextSpacing (p. 203)
Required: False

yPosition

Specifies the vertical position of the caption relative to the top of the output in pixels. A value of 10
would result in the captions starting 10 pixels from the top of the output. If no explicit y_position is
provided, the caption will be positioned towards the bottom of the output. This option is not valid for
source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined
by the caption stream. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647
Properties

**backgroundOpacity**

Specifies the opacity of the background rectangle. 255 is opaque; 0 is transparent. Leaving this parameter blank is equivalent to setting it to 0 (transparent). All burn-in and DVB-Sub font settings must match.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 255

**fontOpacity**

Specifies the opacity of the burned-in captions. 255 is opaque; 0 is transparent. All burn-in and DVB-Sub font settings must match.

- **Type:** integer
- **Required:** True
- **Minimum:** 0
- **Maximum:** 255

**shadowOpacity**

Specifies the opacity of the shadow. 255 is opaque; 0 is transparent. Leaving this parameter blank is equivalent to setting it to 0 (transparent). All burn-in and DVB-Sub font settings must match.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 255

**fontResolution**

Font resolution in DPI (dots per inch); default is 96 dpi. All burn-in and DVB-Sub font settings must match.

- **Type:** integer
- **Required:** False
- **Minimum:** 96
- **Maximum:** 600

**shadowYOffset**

Specifies the vertical offset of the shadow relative to the captions in pixels. A value of -2 would result in a shadow offset 2 pixels above the text. All burn-in and DVB-Sub font settings must match.

- **Type:** integer
- **Required:** False
- **Minimum:** -2147483648
- **Maximum:** 2147483647

**outlineSize**

Specifies font outline size in pixels. This option is not valid for source captions that are either 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.
Properties

Type: integer
Required: True
Minimum: 0
Maximum: 10

outlineColor

Type: BurninSubtitleOutlineColor (p. 203)
Required: True

fontSize

A positive integer indicates the exact font size in points. Set to 0 for automatic font size selection. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: 0
Maximum: 96

shadowXOffset

Specifies the horizontal offset of the shadow relative to the captions in pixels. A value of -2 would result in a shadow offset 2 pixels to the left. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

alignment

Type: BurninSubtitleAlignment (p. 202)
Required: True

shadowColor

Type: BurninSubtitleShadowColor (p. 203)
Required: False

fontColor

Type: BurninSubtitleFontColor (p. 203)
Required: False

BurninSubtitleAlignment (Enum)

If no explicit x_position or y_position is provided, setting alignment to centered will place the captions at the bottom center of the output. Similarly, setting a left alignment will align captions to the bottom left of the output. If x and y positions are given in conjunction with the alignment parameter, the font will be justified (either left or centered) relative to those coordinates. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.
### CENTERED

### LEFT

**BurninSubtitleBackgroundColor (Enum)**

Specifies the color of the rectangle behind the captions. All burn-in and DVB-Sub font settings must match.

- NONE
- BLACK
- WHITE

**BurninSubtitleFontColor (Enum)**

Specifies the color of the burned-in captions. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

- WHITE
- BLACK
- YELLOW
- RED
- GREEN
- BLUE

**BurninSubtitleOutlineColor (Enum)**

Specifies font outline color. This option is not valid for source captions that are either 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

- BLACK
- WHITE
- YELLOW
- RED
- GREEN
- BLUE

**BurninSubtitleShadowColor (Enum)**

Specifies the color of the shadow cast by the captions. All burn-in and DVB-Sub font settings must match.

- NONE
- BLACK
- WHITE

**BurninSubtitleTeletextSpacing (Enum)**

Controls whether a fixed grid size or proportional font spacing will be used to generate the output subtitles bitmap. Only applicable for Teletext inputs and DVB-Sub/Burn-in outputs.

- FIXED_GRID
- PROPORTIONAL
CaptionDescription

captionSelectorName

Specifies which "Caption Selector":#inputs-captions selector to use from each input when generating captions. The name should be of the format "Caption Selector <N>", which denotes that the Nth Caption Selector will be used from each input.

Type: string
Required: True

languageDescription

Human readable information to indicate captions available for players (eg, English, or Spanish). Alphanumeric characters, spaces, and underscore are legal.

Type: string
Required: False
Pattern: ^\[\w\ ]*$

languageCode

Indicates the language of the caption output track.

Type: LanguageCode (p. 260)
Required: False

destinationSettings

Type: CaptionDestinationSettings (p. 204)
Required: False

CaptionDestinationSettings

burninDestinationSettings

Type: BurninDestinationSettings (p. 200)
Required: False

teletextDestinationSettings

Type: TeletextDestinationSettings (p. 295)
Required: False

ttmlDestinationSettings

Type: TtmlDestinationSettings (p. 297)
Required: False

destinationType

Type: CaptionDestinationType (p. 205)
Required: True

dvbSubDestinationSettings

Type: DvbSubDestinationSettings (p. 213)
Required: False

sccDestinationSettings

Type: SccDestinationSettings (p. 293)
Required: False

CaptionDestinationType (Enum)

BURN_IN
DVB_SUB
EMBEDDED
SCC
SRT
TELETEXT
TTML
WEBVTT

CaptionSelector

sourceSettings

Type: CaptionSourceSettings (p. 205)
Required: True

languageCode

The specific language to extract from source. If input is SCTE-27, complete this field and/or PID to select the caption language to extract. If input is DVB-Sub and output is Burn-in or SMPTE-TT, complete this field and/or PID to select the caption language to extract. If input is DVB-Sub that is being passed through, omit this field (and PID field); there is no way to extract a specific language with pass-through captions.

Type: LanguageCode (p. 260)
Required: False

CaptionSourceSettings

fileSourceSettings

Type: FileSourceSettings (p. 224)
Required: False

ancillarySourceSettings

Type: AncillarySourceSettings (p. 193)
Properties

Required: False

**embeddedSourceSettings**

Type: EmbeddedSourceSettings (p. 222)
Required: False

**sourceType**

Type: CaptionSourceType (p. 206)
Required: True

**dvbSubSourceSettings**

Type: DvbSubSourceSettings (p. 216)
Required: False

**teletextSourceSettings**

Type: TeletextSourceSettings (p. 295)
Required: False

**CaptionSourceType (Enum)**

Use Source (SourceType) to identify the format of your input captions. The service cannot auto-detect caption format.

ANCILLARY
DVB_SUB
EMBEDDED
SCC
TTML
STL
SRT
TELETEXT
NULL_SOURCE

**ChannelMapping**

**outputChannels**

Type: Array of type OutputChannelMapping (p. 287)
Required: True

**ColorCorrector**

**saturation**

Saturation level.
Type: integer
**Required**: False  
**Minimum**: 1  
**Maximum**: 100

**brightness**

Brightness level.

**Type**: integer  
**Required**: False  
**Minimum**: 1  
**Maximum**: 100

**hdr10Metadata**

**Type**: Hdr10Metadata (p. 243)  
**Required**: False

**contrast**

Contrast level.

**Type**: integer  
**Required**: False  
**Minimum**: 1  
**Maximum**: 100

**hue**

Hue in degrees.

**Type**: integer  
**Required**: False  
**Minimum**: -180  
**Maximum**: 180

**colorSpaceConversion**

**Type**: ColorSpaceConversion (p. 208)  
**Required**: False

**ColorMetadata (Enum)**

Enable Insert color metadata (ColorMetadata) to include color metadata in this output. This setting is enabled by default.

- IGNORE
- INSERT

**ColorSpace (Enum)**

Specifies the colorspace of an input. This setting works in tandem with "Color Corrector":#color_corrector > color_space_conversion to determine if any conversion will be performed.
**ColorSpaceConversion (Enum)**

Determines if colorspace conversion will be performed. If set to _None_, no conversion will be performed. If _Force 601_ or _Force 709_ are selected, conversion will be performed for inputs with differing colorspaces. An input's colorspace can be specified explicitly in the "Video Selector" "Video Selector". An input's colorspace can be specified explicitly in the "Video Selector".

NONE
FORCE_601
FORCE_709
FORCE_HDR10
FORCE_HLG_2020

**ColorSpaceUsage (Enum)**

There are two sources for color metadata, the input file and the job configuration. This enum controls which takes precedence. FORCE: System will use color metadata supplied by user, if any. If the user does not supply color metadata the system will use data from the source. FALLBACK: System will use color metadata from the source. If source has no color metadata, the system will use user-supplied color metadata values if available.

FORCE
FALLBACK

**ContainerSettings**

carrier

**Type**: ContainerType (p. 209)
**Required**: True

**mp4Settings**

**Type**: Mp4Settings (p. 275)
**Required**: False

**m3u8Settings**

**Type**: M3u8Settings (p. 271)
**Required**: False

**m2tsSettings**

**Type**: M2tsSettings (p. 266)
**Required**: False
**movSettings**

- **Type:** MovSettings (p. 274)
- **Required:** False

**f4vSettings**

- **Type:** F4vSettings (p. 223)
- **Required:** False

**ContainerType (Enum)**

Container for this output. Some containers require a container settings object. If not specified, the default object will be created.

- F4V
- ISMV
- M2TS
- M3U8
- MOV
- MP4
- MPD
- MXF
- RAW

**DashIsoEncryptionSettings**

**spekeKeyProvider**

- **Type:** SpekeKeyProvider (p. 293)
- **Required:** True

**DashIsoGroupSettings**

**fragmentLength**

Length of fragments to generate (in seconds). Fragment length must be compatible with GOP size and Framerate. Note that fragments will end on the next keyframe after this number of seconds, so actual fragment length may be longer. When Emit Single File is checked, the fragmentation is internal to a single output file and it does not cause the creation of many output files as in other output types.

- **Type:** integer
- **Required:** True
- **Minimum:** 1
- **Maximum:** 2147483647

**baseUrl**

A partial URI prefix that will be put in the manifest (.mpd) file at the top level BaseURL element. Can be used if streams are delivered from a different URL than the manifest file.

- **Type:** string
**Properties**

**Required:** False

**minBufferTime**

Minimum time of initially buffered media that is needed to ensure smooth playout.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 2147483647

**encryption**

DRM settings.

- **Type:** DashIsoEncryptionSettings (p. 209)
- **Required:** False

**destination**

Use Destination (Destination) to specify the S3 output location and the output filename base. Destination accepts format identifiers. If you do not specify the base filename in the URI, the service will use the filename of the input file. If your job has multiple inputs, the service uses the filename of the first input file.

- **Type:** string
- **Required:** True
- **Pattern:** `s3:/

**segmentLength**

Length of mpd segments to create (in seconds). Note that segments will end on the next keyframe after this number of seconds, so actual segment length may be longer. When Emit Single File is checked, the segmentation is internal to a single output file and it does not cause the creation of many output files as in other output types.

- **Type:** integer
- **Required:** True
- **Minimum:** 1
- **Maximum:** 2147483647

**segmentControl**

- **Type:** DashIsoSegmentControl (p. 211)
- **Required:** False

**hbbtvCompliance**

- **Type:** DashIsoHbbtvCompliance (p. 210)
- **Required:** False

**DashIsoHbbtvCompliance (Enum)**

Supports HbbTV specification as indicated
HBBTV_1_5
NONE

DashIsoSegmentControl (Enum)

When set to SINGLE_FILE, a single output file is generated, which is internally segmented using the Fragment Length and Segment Length. When set to SEGMENTED_FILES, separate segment files will be created.

SINGLE_FILE
SEGMENTED_FILES

DeinterlaceAlgorithm (Enum)

Only applies when you set Deinterlace (DeinterlaceMode) to Deinterlace (DEINTERLACE) or Adaptive (ADAPTIVE). Motion adaptive interpolate (INTERPOLATE) produces sharper pictures, while blend (BLEND) produces smoother motion. Use (INTERPOLATE_TICKER) OR (BLEND_TICKER) if your source file includes a ticker, such as a scrolling headline at the bottom of the frame.

INTERPOLATE
INTERPOLATE_TICKER
BLEND
BLEND_TICKER

Deinterlacer

mode

Type: DeinterlaceMode (p. 212)
Required: False

control

Type: DeinterlacerControl (p. 211)
Required: False

algorithm

Type: DeinterlaceAlgorithm (p. 211)
Required: False

DeinterlacerControl (Enum)

- When set to NORMAL (default), the deinterlacer does not convert frames that are tagged in metadata as progressive. It will only convert those that are tagged as some other type. - When set to FORCE_ALL_FRAMES, the deinterlacer converts every frame to progressive - even those that are already tagged as progressive. Turn Force mode on only if there is a good chance that the metadata has tagged frames as progressive when they are not progressive. Do not turn on otherwise; processing frames that are already progressive into progressive will probably result in lower quality video.

FORCE_ALL_FRAMES
NORMAL
DeinterlacerMode (Enum)

Use Deinterlacer (DeinterlaceMode) to choose how the service will do deinterlacing. Default is Deinterlace. - Deinterlace converts interlaced to progressive. - Inverse telecine converts Hard Telecine 29.97i to progressive 23.976p. - Adaptive auto-detects and converts to progressive.

    DEINTERLACE
    INVERSE_TELECINE
    ADAPTIVE

DeleteJobTemplateRequest

name

    Type: string
    Required: True

DeleteJobTemplateResponse

DropFrameTimecode (Enum)

Applies only to 29.97 fps outputs. When this feature is enabled, the service will use drop-frame timecode on outputs. If it is not possible to use drop-frame timecode, the system will fall back to non-drop-frame. This setting is enabled by default when Timecode insertion (TimecodeInsertion) is enabled.

    DISABLED
    ENABLED

DvbNitSettings

networkName

The network name text placed in the network_name_descriptor inside the Network Information Table. Maximum length is 256 characters.

    Type: string
    Required: True

networkId

The numeric value placed in the Network Information Table (NIT).

    Type: integer
    Required: True
    Minimum: 0
    Maximum: 65535

nitInterval

The number of milliseconds between instances of this table in the output transport stream.

    Type: integer
    Required: True
    Minimum: 25
Maximum: 10000

### DvbSdtSettings

#### sdInterval

The number of milliseconds between instances of this table in the output transport stream.

- **Type**: integer
- **Required**: False
- **Minimum**: 25
- **Maximum**: 2000

#### serviceName

The service name placed in the service_descriptor in the Service Description Table. Maximum length is 256 characters.

- **Type**: string
- **Required**: False

#### serviceProviderName

The service provider name placed in the service_descriptor in the Service Description Table. Maximum length is 256 characters.

- **Type**: string
- **Required**: False

#### outputSdt

- **Type**: OutputSdt (p. 288)
- **Required**: False

### DvbSubDestinationSettings

#### xPosition

Specifies the horizontal position of the caption relative to the left side of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the left of the output. If no explicit x_position is provided, the horizontal caption position will be determined by the alignment parameter. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 2147483647

#### backgroundColor

- **Type**: DvbSubtitleBackgroundColor (p. 216)
- **Required**: False
teletextSpacing

Type: DvbSubtitleTeletextSpacing (p. 217)
Required: False

yPosition

Specifies the vertical position of the caption relative to the top of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the top of the output. If no explicit y_position is provided, the caption will be positioned towards the bottom of the output. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

backgroundOpacity

Specifies the opacity of the background rectangle. 255 is opaque; 0 is transparent. Leaving this parameter blank is equivalent to setting it to 0 (transparent). All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: 0
Maximum: 255

fontOpacity

Specifies the opacity of the burned-in captions. 255 is opaque; 0 is transparent. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: True
Minimum: 0
Maximum: 255

shadowOpacity

Specifies the opacity of the shadow. 255 is opaque; 0 is transparent. Leaving this parameter blank is equivalent to setting it to 0 (transparent). All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: 0
Maximum: 255

fontResolution

Font resolution in DPI (dots per inch); default is 96 dpi. All burn-in and DVB-Sub font settings must match.

Type: integer
shadowYOffset

Specifies the vertical offset of the shadow relative to the captions in pixels. A value of -2 would result in a shadow offset 2 pixels above the text. All burn-in and DVB-Sub font settings must match.

- **Type**: integer
- **Required**: False
- **Minimum**: -2147483648
- **Maximum**: 2147483647

outlineSize

Specifies font outline size in pixels. This option is not valid for source captions that are either 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

- **Type**: integer
- **Required**: True
- **Minimum**: 0
- **Maximum**: 10

outlineColor

- **Type**: DvbSubtitleOutlineColor (p. 217)
- **Required**: True

fontSize

A positive integer indicates the exact font size in points. Set to 0 for automatic font size selection. All burn-in and DVB-Sub font settings must match.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 96

shadowXOffset

Specifies the horizontal offset of the shadow relative to the captions in pixels. A value of -2 would result in a shadow offset 2 pixels to the left. All burn-in and DVB-Sub font settings must match.

- **Type**: integer
- **Required**: False
- **Minimum**: -2147483648
- **Maximum**: 2147483647

alignment

- **Type**: DvbSubtitleAlignment (p. 216)
- **Required**: True
shadowColor

Type: DvbSubtitleShadowColor (p. 217)
Required: False

fontColor

Type: DvbSubtitleFontColor (p. 216)
Required: False

DvbSubSourceSettings

pid

When using DVB-Sub with Burn-In or SMPTE-TT, use this PID for the source content. Unused for DVB-Sub passthrough. All DVB-Sub content is passed through, regardless of selectors.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

DvbSubtitleAlignment (Enum)

If no explicit x_position or y_position is provided, setting alignment to centered will place the captions at the bottom center of the output. Similarly, setting a left alignment will align captions to the bottom left of the output. If x and y positions are given in conjunction with the alignment parameter, the font will be justified (either left or centered) relative to those coordinates. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

CENTERED
LEFT

DvbSubtitleBackgroundColor (Enum)

Specifies the color of the rectangle behind the captions. All burn-in and DVB-Sub font settings must match.

NONE
BLACK
WHITE

DvbSubtitleFontColor (Enum)

Specifies the color of the burned-in captions. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

WHITE
BLACK
YELLOW
RED
GREEN
BLUE
DvbSubTitleOutlineColor (Enum)

Specifies font outline color. This option is not valid for source captions that are either 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

- BLACK
- WHITE
- YELLOW
- RED
- GREEN
- BLUE

DvbSubTitleShadowColor (Enum)

Specifies the color of the shadow cast by the captions. All burn-in and DVB-Sub font settings must match.

- NONE
- BLACK
- WHITE

DvbSubTitleTeletextSpacing (Enum)

Controls whether a fixed grid size or proportional font spacing will be used to generate the output subtitles bitmap. Only applicable for Teletext inputs and DVB-Sub/Burn-in outputs.

- FIXED_GRID
- PROPORTIONAL

DvbTdtSettings

tdtInterval

The number of milliseconds between instances of this table in the output transport stream.

- Type: integer
- Required: True
- Minimum: 1000
- Maximum: 30000

Eac3AttenuationControl (Enum)

If set to ATTENUATE_3_DB, applies a 3 dB attenuation to the surround channels. Only used for 3/2 coding mode.

- ATTENUATE_3_DB
- NONE

Eac3BitstreamMode (Enum)

Specifies the "Bitstream Mode" (bsmod) for the emitted E-AC-3 stream. See ATSC A/52-2012 (Annex E) for background on these values.

- COMPLETE_MAIN
Eac3CodingMode (Enum)

Dolby Digital Plus coding mode. Determines number of channels.

- CODING_MODE_1_0
- CODING_MODE_2_0
- CODING_MODE_3_2

Eac3DcFilter (Enum)

Activates a DC highpass filter for all input channels.

- ENABLED
- DISABLED

Eac3DynamicRangeCompressionLine (Enum)

Enables Dynamic Range Compression that restricts the absolute peak level for a signal.

- NONE
- FILM_STANDARD
- FILM_LIGHT
- MUSIC_STANDARD
- MUSIC_LIGHT
- SPEECH

Eac3DynamicRangeCompressionRf (Enum)

Enables Heavy Dynamic Range Compression, ensures that the instantaneous signal peaks do not exceed specified levels.

- NONE
- FILM_STANDARD
- FILM_LIGHT
- MUSIC_STANDARD
- MUSIC_LIGHT
- SPEECH

Eac3LfeControl (Enum)

When encoding 3/2 audio, controls whether the LFE channel is enabled

- LFE
- NO_LFE

Eac3LfeFilter (Enum)

Applies a 120Hz lowpass filter to the LFE channel prior to encoding. Only valid with 3_2_LFE coding mode.
**Eac3MetadataControl (Enum)**

When set to FOLLOW_INPUT, encoder metadata will be sourced from the DD, DD+, or DolbyE decoder that supplied this audio data. If audio was not supplied from one of these streams, then the static metadata settings will be used.

- FOLLOW_INPUT
- USE_CONFIGURED

**Eac3PassthroughControl (Enum)**

When set to WHEN_POSSIBLE, input DD+ audio will be passed through if it is present on the input. This detection is dynamic over the life of the transcode. Inputs that alternate between DD+ and non-DD+ content will have a consistent DD+ output as the system alternates between passthrough and encoding.

- WHEN_POSSIBLE
- NO_PASSTHROUGH

**Eac3PhaseControl (Enum)**

Controls the amount of phase-shift applied to the surround channels. Only used for 3/2 coding mode.

- SHIFT_90_DEGREES
- NO_SHIFT

**Eac3Settings**

- **dialnorm**
  
  Sets the dialnorm for the output. If blank and input audio is Dolby Digital Plus, dialnorm will be passed through.

  - **Type:** integer
  - **Required:** False
  - **Minimum:** 1
  - **Maximum:** 31

- **passthroughControl**
  
  - **Type:** `Eac3PassthroughControl (p. 219)`
  - **Required:** False

- **metadataControl**
  
  - **Type:** `Eac3MetadataControl (p. 219)`
  - **Required:** False

- **bitrate**
  
  Average bitrate in bits/second. Valid bitrates depend on the coding mode.
**Type**: integer  
**Required**: False  
**Minimum**: 64000  
**Maximum**: 640000

**dynamicRangeCompressionRf**

**Type**: `Eac3DynamicRangeCompressionRf` (p. 218)  
**Required**: False

**sampleRate**

Sample rate in hz. Sample rate is always 48000.

**Type**: integer  
**Required**: False  
**Minimum**: 48000  
**Maximum**: 48000

**ltRtSurroundMixLevel**

Left total/Right total surround mix level. Only used for 3/2 coding mode. Valid values: -1.5 -3.0 -4.5 -6.0 -60

**Type**: number  
**Required**: False  
**Format**: float  
**Minimum**: -60.0  
**Maximum**: -1.5

**surroundExMode**

**Type**: `Eac3SurroundExMode` (p. 222)  
**Required**: False

**dynamicRangeCompressionLine**

**Type**: `Eac3DynamicRangeCompressionLine` (p. 218)  
**Required**: False

**lfeControl**

**Type**: `Eac3LfeControl` (p. 218)  
**Required**: False

**codingMode**

**Type**: `Eac3CodingMode` (p. 218)  
**Required**: False

**surroundMode**

**Type**: `Eac3SurroundMode` (p. 222)
Required: False

attenuationControl

Type: Eac3AttenuationControl (p. 217)
Required: False

lfeFilter

Type: Eac3LfeFilter (p. 218)
Required: False

phaseControl

Type: Eac3PhaseControl (p. 219)
Required: False

ltRtCenterMixLevel

Left total/Right total center mix level. Only used for 3/2 coding mode. Valid values: 3.0, 1.5, 0.0, -1.5
-3.0 -4.5 -6.0 -60

Type: number
Required: False
Format: float
Minimum: -60.0
Maximum: 3.0

dcFilter

Type: Eac3DcFilter (p. 218)
Required: False

stereoDownmix

Type: Eac3StereoDownmix (p. 222)
Required: False

bitstreamMode

Type: Eac3BitstreamMode (p. 217)
Required: False

loRoSurroundMixLevel

Left only/Right only surround mix level. Only used for 3/2 coding mode. Valid values: -1.5 -3.0 -4.5 -6.0
-60

Type: number
Required: False
Format: float
Minimum: -60.0
Maximum: -1.5
**loRoCenterMixLevel**

Left only/Right only center mix level. Only used for 3/2 coding mode. Valid values: 3.0, 1.5, 0.0, -1.5, -3.0, -4.5, -6.0, -60

- **Type:** number
- **Required:** False
- **Format:** float
- **Minimum:** -60.0
- **Maximum:** 3.0

**Eac3StereoDownmix (Enum)**

Stereo downmix preference. Only used for 3/2 coding mode.

- NOT_INDICATED
- LO_RO
- LT_RT
- DPL2

**Eac3SurroundExMode (Enum)**

When encoding 3/2 audio, sets whether an extra center back surround channel is matrix encoded into the left and right surround channels.

- NOT_INDICATED
- ENABLED
- DISABLED

**Eac3SurroundMode (Enum)**

When encoding 2/0 audio, sets whether Dolby Surround is matrix encoded into the two channels.

- NOT_INDICATED
- ENABLED
- DISABLED

**EmbeddedConvert608To708 (Enum)**

When set to UPCONVERT, 608 data is both passed through via the "608 compatibility bytes" fields of the 708 wrapper as well as translated into 708. 708 data present in the source content will be discarded.

- UPCONVERT
- DISABLED

**EmbeddedSourceSettings**

**source608ChannelNumber**

Specifies the 608/708 channel number within the video track from which to extract captions. Unused for passthrough.

- **Type:** integer
- **Required:** False
- **Minimum:** 1
- **Maximum:** 4
convert608To708

Type: `EmbeddedConvert608To708 (p. 222)`
Required: False

source608TrackNumber

Specifies the video track index used for extracting captions. The system only supports one input video track, so this should always be set to '1'.

Type: integer
Required: False
Minimum: 1
Maximum: 1

ExceptionBody

message

Type: string
Required: False

F4vMoovPlacement (Enum)

If set to PROGRESSIVE_DOWNLOAD, the MOOV atom is relocated to the beginning of the archive as required for progressive downloading. Otherwise it is placed normally at the end.

PROGRESSIVE_DOWNLOAD
NORMAL

F4vSettings

moovPlacement

Type: `F4vMoovPlacement (p. 223)`
Required: False

FileGroupSettings

destination

Use Destination (Destination) to specify the S3 output location and the output filename base. Destination accepts format identifiers. If you do not specify the base filename in the URI, the service will use the filename of the input file. If your job has multiple inputs, the service uses the filename of the first input file.

Type: string
Required: True
Pattern: ^s3:\/\/

FileSourceConvert608To708 (Enum)

If set to UPCONVERT, 608 caption data is both passed through via the "608 compatibility bytes" fields of the 708 wrapper as well as translated into 708. 708 data present in the source content will be discarded.
UPCONVERT
DISABLED

**FileSourceSettings**

timeDelta

Specifies a time delta in seconds to offset the captions from the source file.

- **Type**: integer
- **Required**: False
- **Minimum**: -2147483648
- **Maximum**: 2147483647

convert608To708

- **Type**: FileSourceConvert608To708 (p. 223)
- **Required**: False

sourceFile

External caption file used for loading captions. Accepted file extensions are 'scc', 'ttml', 'dfxp', 'stl', 'srt', and 'smi'. Auto-populated when Infer External Filename is checked.

- **Type**: string
- **Required**: True
- **Pattern**: ^s3://(.*?)(scc|SCC|ttml|TTML|dfxp|DFXP|stl|STL|srt|SRT|smi|SMI)$

**FrameCaptureSettings**

framerateDenominator

Frame capture will encode the first frame of the output stream, then one frame every framerateDenominator/framerateNumerator seconds. For example, settings of framerateNumerator = 1 and framerateDenominator = 3 (a rate of 1/3 frame per second) will capture the first frame, then 1 frame every 3s. Files will be named as filename.n.jpg where n is the 0-based sequence number of each Capture.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 2147483647

maxCaptures

Maximum number of captures (encoded jpg output files).

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 10000000
framerateNumerator

Frame capture will encode the first frame of the output stream, then one frame every framerateNumerator/framerateDenominator seconds. For example, settings of framerateNumerator = 1 and framerateDenominator = 3 (a rate of 1/3 frame per second) will capture the first frame, then 1 frame every 3s. Files will be named as filename.NNNNNNN.jpg where N is the 0-based frame sequence number zero padded to 7 decimal places.

Type: integer  
Required: False  
Minimum: 1  
Maximum: 2147483647

quality

JPEG Quality - a higher value equals higher quality.

Type: integer  
Required: False  
Minimum: 1  
Maximum: 100

GetJobTemplateRequest

name

Type: string  
Required: True

GetJobTemplateResponse

jobTemplate

Type: JobTemplate (p. 258)  
Required: False

H264AdaptiveQuantization (Enum)

Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

OFF
LOW
MEDIUM
HIGH
HIGHER
MAX

H264CodecLevel (Enum)

H.264 Level.

AUTO
LEVEL_1
LEVEL_1_1
H264CodecProfile (Enum)

H.264 Profile. High 4:2:2 and 10-bit profiles are only available with the AVC-I License.

- BASELINE
- HIGH
- HIGH_10BIT
- HIGH_422
- HIGH_422_10BIT
- MAIN

H264EntropyEncoding (Enum)

Entropy encoding mode. Use CABAC (must be in Main or High profile) or CAVLC.

- CABAC
- CAVLC

H264FieldEncoding (Enum)

Choosing FORCE_FIELD disables PAFF encoding for interlaced outputs.

- PAFF
- FORCE_FIELD

H264FlickerAdaptiveQuantization (Enum)

Adjust quantization within each frame to reduce flicker or 'pop' on I-frames.

- DISABLED
- ENABLED

H264FramerateControl (Enum)

Using the API, set FramerateControl to INITIALIZE_FROM_SOURCE if you want the service to use the framerate from the input. Using the console, do this by choosing INITIALIZE_FROM_SOURCE for Framerate.

- INITIALIZE_FROM_SOURCE
- SPECIFIED
H264FramerateConversionAlgorithm (Enum)

When set to INTERPOLATE, produces smoother motion during framerate conversion.

- DUPLICATE_DROP
- INTERPOLATE

H264GopBReference (Enum)

If enabled, use reference B frames for GOP structures that have B frames > 1.

- DISABLED
- ENABLED

H264GopSizeUnits (Enum)

Indicates if the GOP Size in H264 is specified in frames or seconds. If seconds the system will convert the GOP Size into a frame count at run time.

- FRAMES
- SECONDS

H264InterlaceMode (Enum)

Use Interlace mode (InterlaceMode) to choose the scan line type for the output.

* Top Field First (TOP_FIELD) and Bottom Field First (BOTTOM_FIELD) produce interlaced output with the entire output having the same field polarity (top or bottom first).

* Follow, Default Top (FOLLOW_TOP_FIELD) and Follow, Default Bottom (FOLLOW_BOTTOM_FIELD) use the same field polarity as the source. Therefore, behavior depends on the input scan type. If the source is interlaced, the output will be interlaced with the same polarity as the source (it will follow the source). The output could therefore be a mix of "top field first" and "bottom field first". If the source is progressive, the output will be interlaced with "top field first" or "bottom field first" polarity, depending on which of the Follow options you chose.

- PROGRESSIVE
- TOP_FIELD
- BOTTOM_FIELD
- FOLLOW_TOP_FIELD
- FOLLOW BOTTOM_FIELD

H264ParControl (Enum)

Using the API, enable ParFollowSource if you want the service to use the pixel aspect ratio from the input. Using the console, do this by choosing Follow source for Pixel aspect ratio.

- INITIALIZE_FROM_SOURCE
- SPECIFIED

H264QualityTuningLevel (Enum)

Use Quality tuning level (H264QualityTuningLevel) to specify whether to use fast single-pass, high-quality singlepass, or high-quality multipass video encoding.

- SINGLE_PASS
- SINGLE_PASS_HQ
MULTI_PASS_HQ

**H264RateControlMode (Enum)**

Rate control mode. CQ uses constant quantizer (qp), ABR (average bitrate) does not write HRD parameters.

- VBR
- CBR

**H264RepeatPps (Enum)**

Places a PPS header on each encoded picture, even if repeated.

- DISABLED
- ENABLED

**H264SceneChangeDetect (Enum)**

Scene change detection (inserts I-frames on scene changes).

- DISABLED
- ENABLED

**H264Settings**

**slices**

Number of slices per picture. Must be less than or equal to the number of macroblock rows for progressive pictures, and less than or equal to half the number of macroblock rows for interlaced pictures.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 32

**minIInterval**

Enforces separation between repeated (cadence) I-frames and I-frames inserted by Scene Change Detection. If a scene change I-frame is within I-interval frames of a cadence I-frame, the GOP is shrunk and/or stretched to the scene change I-frame. GOP stretch requires enabling lookahead as well as setting I-interval. The normal cadence resumes for the next GOP. This setting is only used when Scene Change Detect is enabled. Note: Maximum GOP stretch = GOP size + Min-I-interval - 1

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 30

**parNumerator**

Pixel Aspect Ratio numerator.

- **Type**: integer
flickerAdaptiveQuantization
  Type: H264FlickerAdaptiveQuantization (p. 226)
  Required: False

gopSizeUnits
  Type: H264GopSizeUnits (p. 227)
  Required: False

hrdBufferSize
Size of buffer (HRD buffer model). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.
  Type: integer
  Required: False
  Minimum: -2147483648
  Maximum: 2147483647

qualityTuningLevel
  Type: H264QualityTuningLevel (p. 227)
  Required: False

maxBitrate
Maximum bitrate in bits/second (for VBR mode only). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.
  Type: integer
  Required: False
  Minimum: -2147483648
  Maximum: 2147483647

bitrate
Average bitrate in bits/second. Required for VBR, CBR, and ABR. Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.
  Type: integer
  Required: False
  Minimum: 1000
  Maximum: 2147483647

spatialAdaptiveQuantization
  Type: H264SpatialAdaptiveQuantization (p. 233)
  Required: False
slowPal
  Type: H264SlowPal (p. 233)
  Required: False

codecProfile
  Type: H264CodecProfile (p. 226)
  Required: False

unregisteredSeiTimecode
  Type: H264UnregisteredSeiTimecode (p. 234)
  Required: False

softness
Softness. Selects quantizer matrix, larger values reduce high-frequency content in the encoded image.
  Type: integer
  Required: False
  Minimum: 0
  Maximum: 128

framerateControl
  Type: H264FramerateControl (p. 226)
  Required: False

telecine
  Type: H264Telecine (p. 233)
  Required: False

framerateConversionAlgorithm
  Type: H264FramerateConversionAlgorithm (p. 227)
  Required: False

codecLevel
  Type: H264CodecLevel (p. 225)
  Required: False

numberReferenceFrames
Number of reference frames to use. The encoder may use more than requested if using B-frames and/or interlaced encoding.
  Type: integer
  Required: False
  Minimum: 1
Maximum: 6

temporalAdaptiveQuantization

Type: H264TemporalAdaptiveQuantization (p. 234)
Required: False

repeatPps

Type: H264RepeatPps (p. 228)
Required: False

hrdBufferInitialFillPercentage

Percentage of the buffer that should initially be filled (HRD buffer model).

Type: integer
Required: False
Minimum: 0
Maximum: 100

framerateNumerator

Framerate numerator - framerate is a fraction, e.g. 24000 / 1001 = 23.976 fps.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

numberBFramesBetweenReferenceFrames

Number of B-frames between reference frames.

Type: integer
Required: False
Minimum: 0
Maximum: 7

gopClosedCadence

Frequency of closed GOPs. In streaming applications, it is recommended that this be set to 1 so a decoder joining mid-stream will receive an IDR frame as quickly as possible. Setting this value to 0 will break output segmenting.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

fieldEncoding

Type: H264FieldEncoding (p. 226)
Required: False

entropyEncoding

Type: H264EntropyEncoding (p. 226)
Required: False

framerateDenominator

When you use the API for transcode jobs that use framerate conversion, specify the framerate as a fraction. For example, 24000 / 1001 = 23.976 fps. Use FramerateDenominator to specify the denominator of this fraction. In this example, use 1001 for the value of FramerateDenominator. When you use the console for transcode jobs that use framerate conversion, provide the value as a decimal number for Framerate. In this example, specify 23.976.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

adaptiveQuantization

Type: H264AdaptiveQuantization (p. 225)
Required: False

interlaceMode

Type: H264InterlaceMode (p. 227)
Required: False

gopSize

GOP Length (keyframe interval) in frames or seconds. Must be greater than zero.

Type: number
Required: False
Format: float
Minimum: 0.0

gopBReference

Type: H264GopBReference (p. 227)
Required: False

sceneChangeDetect

Type: H264SceneChangeDetect (p. 228)
Required: False

parDenominator

Pixel Aspect Ratio denominator.
**Type**: integer  
**Required**: False  
**Minimum**: 1  
**Maximum**: 2147483647

**parControl**  
**Type**: H264ParControl (p. 227)  
**Required**: False

**syntax**  
**Type**: H264Syntax (p. 233)  
**Required**: False

**rateControlMode**  
**Type**: H264RateControlMode (p. 228)  
**Required**: False

**H264SlowPal (Enum)**  
Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.  
- **DISABLED**  
- **ENABLED**

**H264SpatialAdaptiveQuantization (Enum)**  
Adjust quantization within each frame based on spatial variation of content complexity.  
- **DISABLED**  
- **ENABLED**

**H264Syntax (Enum)**  
Produces a bitstream compliant with SMPTE RP-2027.  
- **DEFAULT**  
- **RP2027**

**H264Telecine (Enum)**  
This field applies only if the Streams > Advanced > Framerate (framerate) field is set to 29.970. This field works with the Streams > Advanced > Preprocessors > Deinterlacer field (deinterlace_mode) and the Streams > Advanced > Interlaced Mode field (interlace_mode) to identify the scan type for the output: Progressive, Interlaced, Hard Telecine or Soft Telecine. - Hard: produces 29.97i output from 23.976 input. - Soft: produces 23.976; the player converts this output to 29.97i.  
- **NONE**  
- **SOFT**
HARD

**H264TemporalAdaptiveQuantization (Enum)**
Adjust quantization within each frame based on temporal variation of content complexity.
- DISABLED
- ENABLED

**H264UnregisteredSeiTimecode (Enum)**
Inserts timecode for each frame as 4 bytes of an unregistered SEI message.
- DISABLED
- ENABLED

**H265AdaptiveQuantization (Enum)**
Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.
- OFF
- LOW
- MEDIUM
- HIGH
- HIGHER
- MAX

**H265AlternateTransferFunctionSei (Enum)**
Enables Alternate Transfer Function SEI message for outputs using Hybrid Log Gamma (HLG) Electro-Optical Transfer Function (EOTF).
- DISABLED
- ENABLED

**H265CodecLevel (Enum)**
H.265 Level.
- AUTO
- LEVEL_1
- LEVEL_2
- LEVEL_2_1
- LEVEL_3
- LEVEL_3_1
- LEVEL_4
- LEVEL_4_1
- LEVEL_5
- LEVEL_5_1
- LEVEL_5_2
- LEVEL_6
- LEVEL_6_1
LEVEL_6_2

H265CodecProfile (Enum)

Represents the Profile and Tier, per the HEVC (H.265) specification. Selections are grouped as [Profile] / [Tier], so "Main/High" represents Main Profile with High Tier. 4:2:2 profiles are only available with the HEVC 4:2:2 License.

- MAIN_MAIN
- MAIN_HIGH
- MAIN10_MAIN
- MAIN10_HIGH
- MAIN_422_8BIT_MAIN
- MAIN_422_8BIT_HIGH
- MAIN_422_10BIT_MAIN
- MAIN_422_10BIT_HIGH

H265FlickerAdaptiveQuantization (Enum)

Adjust quantization within each frame to reduce flicker or 'pop' on I-frames.

- DISABLED
- ENABLED

H265FramerateControl (Enum)

Using the API, set FramerateControl to INITIALIZE_FROM_SOURCE if you want the service to use the framerate from the input. Using the console, do this by choosing INITIALIZE_FROM_SOURCE for Framerate.

- INITIALIZE_FROM_SOURCE
- SPECIFIED

H265FramerateConversionAlgorithm (Enum)

When set to INTERPOLATE, produces smoother motion during framerate conversion.

- DUPLICATE_DROP
- INTERPOLATE

H265GopBReference (Enum)

If enable, use reference B frames for GOP structures that have B frames > 1.

- DISABLED
- ENABLED

H265GopSizeUnits (Enum)

Indicates if the GOP Size in H265 is specified in frames or seconds. If seconds the system will convert the GOP Size into a frame count at run time.

- FRAMES
SECONDS

H265InterlaceMode (Enum)

Use Interlace mode (InterlaceMode) to choose the scan line type for the output.

* Top Field First (TOP_FIELD) and Bottom Field First (BOTTOM_FIELD) produce interlaced output with the entire output having the same field polarity (top or bottom first).

* Follow, Default Top (FOLLOW_TOP_FIELD) and Follow, Default Bottom (FOLLOW_BOTTOM_FIELD) use the same field polarity as the source. Therefore, behavior depends on the input scan type. If the source is interlaced, the output will be interlaced with the same polarity as the source (it will follow the source). The output could therefore be a mix of "top field first" and "bottom field first". If the source is progressive, the output will be interlaced with "top field first" or "bottom field first" polarity, depending on which of the Follow options you chose.

  PROGRESSIVE
  TOP_FIELD
  BOTTOM_FIELD
  FOLLOW_TOP_FIELD
  FOLLOW_BOTTOM_FIELD

H265ParControl (Enum)

Using the API, enable ParFollowSource if you want the service to use the pixel aspect ratio from the input. Using the console, do this by choosing Follow source for Pixel aspect ratio.

  INITIALIZE_FROM_SOURCE
  SPECIFIED

H265QualityTuningLevel (Enum)

Use Quality tuning level (H265QualityTuningLevel) to specify whether to use fast single-pass, high-quality singlepass, or high-quality multipass video encoding.

  SINGLE_PASS
  SINGLE_PASS_HQ
  MULTI_PASS_HQ

H265RateControlMode (Enum)

Rate control mode. CQ uses constant quantizer (qp), ABR (average bitrate) does not write HRD parameters.

  VBR
  CBR

H265SampleAdaptiveOffsetFilterMode (Enum)

Specify Sample Adaptive Offset (SAO) filter strength. Adaptive mode dynamically selects best strength based on content

  DEFAULT
  ADAPTIVE
H265SceneChangeDetect (Enum)

Scene change detection (inserts I-frames on scene changes).

- DISABLED
- ENABLED

H265Settings

slices

Number of slices per picture. Must be less than or equal to the number of macroblock rows for progressive pictures, and less than or equal to half the number of macroblock rows for interlaced pictures.

- Type: integer
- Required: False
- Minimum: 1
- Maximum: 32

minIInterval

Enforces separation between repeated (cadence) I-frames and I-frames inserted by Scene Change Detection. If a scene change I-frame is within I-interval frames of a cadence I-frame, the GOP is shrunk and/or stretched to the scene change I-frame. GOP stretch requires enabling lookahead as well as setting I-interval. The normal cadence resumes for the next GOP. This setting is only used when Scene Change Detect is enabled. Note: Maximum GOP stretch = GOP size + Min-I-interval - 1

- Type: integer
- Required: False
- Minimum: 0
- Maximum: 30

parNumerator

Pixel Aspect Ratio numerator.

- Type: integer
- Required: False
- Minimum: 1
- Maximum: 2147483647

flickerAdaptiveQuantization

- Type: H265FlickerAdaptiveQuantization (p. 235)
- Required: False

gopSizeUnits

- Type: H265GopSizeUnits (p. 235)
Required: False

hrdBufferSize

Size of buffer (HRD buffer model). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

qualityTuningLevel

Type: H265QualityTuningLevel (p. 236)
Required: False

maxBitrate

Maximum bitrate in bits/second (for VBR mode only). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

bitrate

Average bitrate in bits/second. Required for VBR, CBR, and ABR. Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.

Type: integer
Required: False
Minimum: 1000
Maximum: 2147483647

spatialAdaptiveQuantization

Type: H265SpatialAdaptiveQuantization (p. 242)
Required: False

sampleAdaptiveOffsetFilterMode

Type: H265SampleAdaptiveOffsetFilterMode (p. 236)
Required: False

temporalIds

Type: H265TemporalIds (p. 242)
Required: False
slowPal

Type: H26SSlowPal (p. 241)
Required: False

tiles

Type: H26STiles (p. 242)
Required: False

codecProfile

Type: H265CodecProfile (p. 235)
Required: False

alternateTransferFunctionSei

Type: H265AlternateTransferFunctionSei (p. 234)
Required: False

unregisteredSeiTimecode

Type: H265UnregisteredSeiTimecode (p. 242)
Required: False

framerateControl

Type: H265FramerateControl (p. 235)
Required: False

telecine

Type: H265Telecine (p. 242)
Required: False

framerateConversionAlgorithm

Type: H265FramerateConversionAlgorithm (p. 235)
Required: False

codecLevel

Type: H265CodecLevel (p. 234)
Required: False

numberReferenceFrames

Number of reference frames to use. The encoder may use more than requested if using B-frames and/or interlaced encoding.

Type: integer
Required: False
Minimum: 1
Maximum: 6

temporalAdaptiveQuantization

Type: H265TemporalAdaptiveQuantization (p. 242)
Required: False

hrdBufferInitialFillPercentage

Percentage of the buffer that should initially be filled (HRD buffer model).

Type: integer
Required: False
Minimum: 0
Maximum: 100

framerateNumerator

Framerate numerator - framerate is a fraction, e.g. 24000 / 1001 = 23.976 fps.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

numberBFramesBetweenReferenceFrames

Number of B-frames between reference frames.

Type: integer
Required: False
Minimum: 0
Maximum: 7

gopClosedCadence

Frequency of closed GOPs. In streaming applications, it is recommended that this be set to 1 so a decoder joining mid-stream will receive an IDR frame as quickly as possible. Setting this value to 0 will break output segmenting.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

framerateDenominator

Framerate denominator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647
adaptiveQuantization
Type: H265AdaptiveQuantization (p. 234)
Required: False

interlaceMode
Type: H265InterlaceMode (p. 236)
Required: False

gopSize
GOP Length (keyframe interval) in frames or seconds. Must be greater than zero.
Type: number
Required: False
Format: float
Minimum: 0.0

parDenominator
Pixel Aspect Ratio denominator.
Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

H265SlowPal (Enum)
Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.
H265SpatialAdaptiveQuantization (Enum)
Adjust quantization within each frame based on spatial variation of content complexity.

DISABLED
ENABLED

H265Telecine (Enum)
This field applies only if the Streams > Advanced > Framerate (framerate) field is set to 29.970. This field works with the Streams > Advanced > Preprocessors > Deinterlacer field (deinterlace_mode) and the Streams > Advanced > Interlaced Mode field (interlace_mode) to identify the scan type for the output: Progressive, Interlaced, Hard Telecine or Soft Telecine. - Hard: produces 29.97i output from 23.976 input. - Soft: produces 23.976; the player converts this output to 29.97i.

NONE
SOFT
HARD

H265TemporalAdaptiveQuantization (Enum)
Adjust quantization within each frame based on temporal variation of content complexity.

DISABLED
ENABLED

H265TemporalIds (Enum)
Enables temporal layer identifiers in the encoded bitstream. Up to 3 layers are supported depending on GOP structure: I- and P-frames form one layer, reference B-frames can form a second layer and non-reference b-frames can form a third layer. Decoders can optionally decode only the lower temporal layers to generate a lower frame rate output. For example, given a bitstream with temporal IDs and with b-frames = 1 (i.e. IbpPb display order), a decoder could decode all the frames for full frame rate output or only the I and P frames (lowest temporal layer) for a half frame rate output.

DISABLED
ENABLED

H265Tiles (Enum)
Enable use of tiles, allowing horizontal as well as vertical subdivision of the encoded pictures.

DISABLED
ENABLED

H265UnregisteredSeiTimecode (Enum)
Inserts timecode for each frame as 4 bytes of an unregistered SEI message.

DISABLED
ENABLED

Hdr10Metadata

redPrimaryY
HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

Type: integer
Required: False
Minimum: 0
Maximum: 50000

greenPrimaryY
HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

Type: integer
Required: False
Minimum: 0
Maximum: 50000

whitePointX
HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

Type: integer
Required: False
Minimum: 0
Maximum: 50000

maxLuminance
Nominal maximum mastering display luminance in units of 0.0001 candelas per square meter.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

greenPrimaryX
HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

Type: integer
Required: False
Minimum: 0
Maximum: 50000
whitePointY

HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

Type: integer
Required: False
Minimum: 0
Maximum: 50000

redPrimaryX

HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

Type: integer
Required: False
Minimum: 0
Maximum: 50000

bluePrimaryX

HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

Type: integer
Required: False
Minimum: 0
Maximum: 50000

maxFrameAverageLightLevel

Maximum average light level of any frame in the coded video sequence, in units of candelas per square meter.

Type: integer
Required: True
Minimum: 0
Maximum: 65535

bluePrimaryY

HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

Type: integer
Required: False
Minimum: 0
Maximum: 50000

maxContentLightLevel

Maximum light level among all samples in the coded video sequence, in units of candelas per square meter.
Properties

**Type**: integer
**Required**: True
**Minimum**: 0
**Maximum**: 65535

**minLuminance**
Nominal minimum mastering display luminance in units of 0.0001 candelas per square meter

**Type**: integer
**Required**: False
**Minimum**: 0
**Maximum**: 2147483647

**HlsAdMarkers (Enum)**

- ELEMENTAL
- ELEMENTAL_SCTE35

**HlsAudioTrackType (Enum)**

Four types of audio-only tracks are supported: Audio-Only Variant Stream The client can play back this audio-only stream instead of video in low-bandwidth scenarios. Represented as an EXT-X-STREAM-INF in the HLS manifest. Alternate Audio, Auto Select, Default Alternate rendition that the client should try to play back by default. Represented as an EXT-X-MEDIA in the HLS manifest with DEFAULT=YES, AUTOSELECT=YES, Alternate Audio, Auto Select, Not Default Alternate rendition that the client may try to play back by default. Represented as an EXT-X-MEDIA in the HLS manifest with DEFAULT=NO, AUTOSELECT=YES, Alternate Audio, not Auto Select Alternate rendition that the client will not try to play back by default. Represented as an EXT-X-MEDIA in the HLS manifest with DEFAULT=NO, AUTOSELECT=NO

- ALTERNATE_AUDIO_AUTO_SELECT_DEFAULT
- ALTERNATE_AUDIO_AUTO_SELECT
- ALTERNATE_AUDIO_NOT_AUTO_SELECT
- AUDIO_ONLY_VARIANT_STREAM

**HlsCaptionLanguageMapping**

**languageDescription**
Caption language description.

**Type**: string
**Required**: False

**captionChannel**
Caption channel.

**Type**: integer
**Required**: False
**Minimum**: -2147483648
**Maximum**: 2147483647
languageCode

Type: LanguageCode (p. 260)
Required: False

HlsCaptionLanguageSetting (Enum)

Applies only to 608 Embedded output captions. Insert: Include CLOSED-CAPTIONS lines in the manifest. Specify at least one language in the CC1 Language Code field. One CLOSED-CAPTION line is added for each Language Code you specify. Make sure to specify the languages in the order in which they appear in the original source (if the source is embedded format) or the order of the caption selectors (if the source is other than embedded). Otherwise, languages in the manifest will not match up properly with the output captions. None: Include CLOSED-CAPTIONS=NONE line in the manifest. Omit: Omit any CLOSED-CAPTIONS line from the manifest.

INSERT
OMIT
NONE

HlsClientCache (Enum)

When set to ENABLED, sets #EXT-X-ALLOW-CACHE:no tag, which prevents client from saving media segments for later replay.

DISABLED
ENABLED

HlsCodecSpecification (Enum)

Specification to use (RFC-6381 or the default RFC-4281) during m3u8 playlist generation.

RFC_6381
RFC_4281

HlsDirectoryStructure (Enum)

Indicates whether segments should be placed in subdirectories.

SINGLE_DIRECTORY
SUBDIRECTORY_PER_STREAM

HlsEncryptionSettings

initializationVectorInManifest

Type: HlsInitializationVectorInManifest (p. 250)
Required: False

constantInitializationVector

This is a 128-bit, 16-byte hex value represented by a 32-character text string. If this parameter is not set then the Initialization Vector will follow the segment number by default.

Type: string
Properties

**staticKeyProvider**
- **Type**: StaticKeyProvider (p. 294)
- **Required**: False

**type**
- **Type**: HlsKeyProviderType (p. 250)
- **Required**: True

**spekeKeyProvider**
- **Type**: SpekeKeyProvider (p. 293)
- **Required**: False

**encryptionMethod**
- **Type**: HlsEncryptionType (p. 247)
- **Required**: False

**HlsEncryptionType (Enum)**
Encrypts the segments with the given encryption scheme. Leave blank to disable. Selecting 'Disabled' in the web interface also disables encryption.

- AES128
- SAMPLE_AES

**HlsGroupSettings**

**segmentsPerSubdirectory**
Number of segments to write to a subdirectory before starting a new one. directoryStructure must be SINGLE_DIRECTORY for this setting to have an effect.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 2147483647

**streamInfResolution**
- **Type**: HlsStreamInfResolution (p. 252)
- **Required**: False

**timestampDeltaMilliseconds**
Provides an extra millisecond delta offset to fine tune the timestamps.

- **Type**: integer
Properties

outputSelection

Type: HlsOutputSelection (p. 251)
Required: False

captionLanguageMappings

Type: Array of type HlsCaptionLanguageMapping (p. 245)
Required: False

clientCache

Type: HlsClientCache (p. 246)
Required: False

codecSpecification

Type: HlsCodecSpecification (p. 246)
Required: False

destination

Use Destination (Destination) to specify the S3 output location and the output filename base. Destination accepts format identifiers. If you do not specify the base filename in the URI, the service will use the filename of the input file. If your job has multiple inputs, the service uses the filename of the first input file.

Type: string
Required: True
Pattern: ^s3: \/[\/

segmentControl

Type: HlsSegmentControl (p. 251)
Required: False

timedMetadatald3Frame

Type: HlsTimedMetadatald3Frame (p. 252)
Required: False

timedMetadatald3Period

Timed Metadata interval in seconds.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

captionLanguageSetting
Type: HlsCaptionLanguageSetting (p. 246)
Required: False

minSegmentLength
When set, Minimum Segment Size is enforced by looking ahead and back within the specified range for a nearby avail and extending the segment size if needed.
Type: integer
Required: True
Minimum: 0
Maximum: 2147483647

directoryStructure
Type: HlsDirectoryStructure (p. 246)
Required: False

programDateTime
Type: HlsProgramDateTime (p. 251)
Required: False

baseUrl
A partial URI prefix that will be prepended to each output in the media .m3u8 file. Can be used if base manifest is delivered from a different URL than the main .m3u8 file.
Type: string
Required: False

cipher
DRM settings.
Type: HlsEncryptionSettings (p. 246)
Required: False

adMarkers
Choose one or more ad marker types to pass SCTE35 signals through to this group of Apple HLS outputs.
Type: Array of type HlsAdMarkers (p. 245)
Required: False

programDateTimePeriod
Period of insertion of EXT-X-PROGRAM-DATE-TIME entry, in seconds.
**Type:** integer  
**Required:** False  
**Minimum:** 0  
**Maximum:** 3600

**manifestCompression**

- **Type:** HlsManifestCompression (p. 250)  
- **Required:** False

**segmentLength**

Length of MPEG-2 Transport Stream segments to create (in seconds). Note that segments will end on the next keyframe after this number of seconds, so actual segment length may be longer.

- **Type:** integer  
- **Required:** True  
- **Minimum:** 1  
- **Maximum:** 2147483647

**manifestDurationFormat**

- **Type:** HlsManifestDurationFormat (p. 251)  
- **Required:** False

**HlsIFrameOnlyManifest (Enum)**

When set to INCLUDE, writes I-Frame Only Manifest in addition to the HLS manifest

- INCLUDE  
- EXCLUDE

**HlsInitializationVectorInManifest (Enum)**

The Initialization Vector is a 128-bit number used in conjunction with the key for encrypting blocks. If set to INCLUDE, Initialization Vector is listed in the manifest. Otherwise Initialization Vector is not in the manifest.

- INCLUDE  
- EXCLUDE

**HlsKeyProviderType (Enum)**

Indicates which type of key provider is used for encryption.

- SPEKE  
- STATIC_KEY

**HlsManifestCompression (Enum)**

When set to GZIP, compresses HLS playlist.

- GZIP
NONE

**HlsManifestDurationFormat (Enum)**

Indicates whether the output manifest should use floating point values for segment duration.

- FLOATING_POINT
- INTEGER

**HlsOutputSelection (Enum)**

Indicates whether the .m3u8 manifest file should be generated for this HLS output group.

- MANIFESTS_AND_SEGMENTS
- SEGMENTS_ONLY

**HlsProgramDateTime (Enum)**

Includes or excludes EXT-X-PROGRAM-DATE-TIME tag in .m3u8 manifest files. The value is calculated as follows: either the program date and time are initialized using the input timecode source, or the time is initialized using the input timecode source and the date is initialized using the timestamp_offset.

- INCLUDE
- EXCLUDE

**HlsSegmentControl (Enum)**

When set to SINGLE_FILE, emits program as a single media resource (.ts) file, uses #EXT-X-BYTERANGE tags to index segment for playback.

- SINGLE_FILE
- SEGMENTED_FILES

**HlsSettings**

**iFrameOnlyManifest**

- **Type**: HlsIFrameOnlyManifest (p. 250)
- **Required**: False

**segmentModifier**

String concatenated to end of segment filenames. Accepts "Format Identifiers":#format_identifier_parameters.

- **Type**: string
- **Required**: False

**audioTrackType**

- **Type**: HlsAudioTrackType (p. 245)
- **Required**: False
**audioRenditionSets**

List all the audio groups that are used with the video output stream. Input all the audio GROUP-IDs that are associated to the video, separate by ','.

  Type: string
  Required: False

**audioGroupId**

Specifies the group to which the audio Rendition belongs.

  Type: string
  Required: False

**HlsStreamInfResolution (Enum)**

Include or exclude RESOLUTION attribute for video in EXT-X-STREAM-INF tag of variant manifest.

  INCLUDE
  EXCLUDE

**HlsTimedMetadataId3Frame (Enum)**

Indicates ID3 frame that has the timecode.

  NONE
  PRIV
  TDRL

**Id3Insertion**

**id3**

Use ID3 tag (Id3) to provide a tag value in base64-encode format.

  Type: string
  Required: True
  Pattern: /^[A-Za-z0-9+/\]+={0,2}$/

**timecode**

Provide a Timecode (TimeCode) in HH:MM:SS:FF or HH:MM:SS;FF format.

  Type: string
  Required: True
  Format: timecode
  Pattern: ^\([01][0-9][0-9]|[0-4]):[0-5][0-9][0-9]:[0-5][0-9][0-9]:[0-9][0-9][0-9][;][0-9][0-9]{2}$

**ImageInsert**

**insertableImages**

Image to insert. Must be 32 bit windows BMP, PNG, or TGA file. Must not be larger than the output frames.
**Properties**

**Type**: Array of type [InsertableImage](p. 256)  
**Required**: True

**InputClipping**

**startTimecode**

Set Start timecode (StartTimecode) to the beginning of the portion of the input you are clipping. The frame corresponding to the Start timecode value is included in the clip. Start timecode or End timecode may be left blank, but not both. When choosing this value, take into account your setting for Input timecode source. For example, if you have embedded timecodes that start at 01:00:00:00 and you want your clip to begin five minutes into the video, use 01:00:05:00.

**Type**: string  
**Required**: False  
**Format**: timecode  
**Pattern**: ^([01][0-9]|2[0-4]):[0-5][0-9]:[0-5][0-9]:[0-9]{2}$

**endTimecode**

Set End timecode (EndTimecode) to the end of the portion of the input you are clipping. The frame corresponding to the End timecode value is included in the clip. Start timecode or End timecode may be left blank, but not both. When choosing this value, take into account your setting for Input timecode source. For example, if you have embedded timecodes that start at 01:00:00:00 and you want your clip to begin five minutes into the video, use 01:00:05:00.

**Type**: string  
**Required**: False  
**Format**: timecode  
**Pattern**: ^([01][0-9]|2[0-4]):[0-5][0-9]:[0-5][0-9]:[0-9]{2}$

**InputDeblockFilter (Enum)**

Enable Deblock (InputDeblockFilter) to produce smoother motion in the output. Default is disabled. Only manually controllable for MPEG2 and uncompressed video inputs.

- ENABLED
- DISABLED

**InputDenoiseFilter (Enum)**

Enable Denoise (InputDenoiseFilter) to filter noise from the input. Default is disabled. Only applicable to MPEG2, H.264, H.265, and uncompressed video inputs.

- ENABLED
- DISABLED

**InputFilterEnable (Enum)**

Use Filter enable (InputFilterEnable) to specify how the transcoding service applies the denoise and deblock filters. You must also enable the filters separately, with Denoise (InputDenoiseFilter) and Deblock (InputDeblockFilter).

- * Auto - The transcoding service determines whether to apply filtering, depending on input type and quality.
Properties

* Disable - The input is not filtered. This is true even if you use the API to enable them in (InputDeblockFilter) and (InputDeblockFilter).

* Force - The input is filtered regardless of input type.

  AUTO
  DISABLE
  FORCE

InputPsiControl (Enum)

Set PSI control (InputPsiControl) for transport stream inputs to specify which data the demux process to scans.

* Ignore PSI - Scan all PIDs for audio and video.

* Use PSI - Scan only PSI data.

  IGNORE_PSI
  USE_PSI

InputTemplate

audioSelectors

Use Audio selectors (AudioSelectors) to specify a track or set of tracks from the input that you will use in your outputs. You can use multiple Audio selectors per input.

  Type: object
  Required: False

audioSelectorGroups

Specifies set of audio selectors within an input to combine. An input may have multiple audio selector groups. See "Audio Selector Group":#inputs-audio_selector_group for more information.

  Type: object
  Required: False

filterEnable

  Type: InputFilterEnable (p. 253)
  Required: False

deblockFilter

  Type: InputDeblockFilter (p. 253)
  Required: False

videoSelector

  Type: VideoSelector (p. 302)
  Required: False
filterStrength

Use Filter strength (FilterStrength) to adjust the magnitude the input filter settings (Deblock and Denoise). The range is -5 to 5. Default is 0.

Type: integer
Required: False
Minimum: -5
Maximum: 5

programNumber

Use Program (programNumber) to select a specific program from within a multi-program transport stream. Note that Quad 4K is not currently supported. Default is the first program within the transport stream. If the program you specify doesn't exist, the transcoding service will use this default.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

timecodeSource

Type: InputTimecodeSource (p. 256)
Required: False

captionSelectors

Use Captions selectors (CaptionSelectors) to specify the captions data from the input that you will use in your outputs. You can use mutiple captions selectors per input.

Type: object
Required: False

denoiseFilter

Type: InputDenoiseFilter (p. 253)
Required: False

psiControl

Type: InputPsiControl (p. 254)
Required: False

inputClippings

(InputClippings) contains sets of start and end times that together specify a portion of the input to be used in the outputs. If you provide only a start time, the clip will be the entire input from that point to the end. If you provide only an end time, it will be the entire input up to that point. When you specify more than one input clip, the transcoding service creates the job outputs by stringing the clips together in the order you specify them.

Type: Array of type InputClipping (p. 253)
InputTimecodeSource (Enum)

Use Timecode source (InputTimecodeSource) to specify how timecode information from your input is adjusted and encoded in all outputs for the job. Default is embedded. Set to Embedded (EMBEDDED) to use the timecode that is in the input video. If no embedded timecode is in the source, will set the timecode for the first frame to 00:00:00:00. Set to Start at 0 (ZEROBASED) to set the timecode of the initial frame to 00:00:00:00. Set to Specified start (SPECIFIEDSTART) to provide the initial timecode yourself the setting (Start).

- EMBEDDED
- ZEROBASED
- SPECIFIEDSTART

InsertableImage

duration

Use Duration (Duration) to set the time, in milliseconds, for the image to remain on the output video.

- Type: integer
- Required: False
- Minimum: -2147483648
- Maximum: 2147483647

fadeOut

Use Fade out (FadeOut) to set the length, in milliseconds, of the inserted image fade out. If you don't specify a value for Fade out, the image will disappear abruptly at the end of the inserted image duration.

- Type: integer
- Required: False
- Minimum: -2147483648
- Maximum: 2147483647

imageY

Use Top (ImageY) to set the distance, in pixels, between the inserted image and the top edge of the video frame. Required for BMP, PNG and TGA input.

- Type: integer
- Required: True
- Minimum: -2147483648
- Maximum: 2147483647

fadeIn

Use Fade in (FadeIn) to set the length, in milliseconds, of the inserted image fade in. If you don't specify a value for Fade in, the image will appear abruptly at the Start time.

- Type: integer
- Required: False
- Minimum: -2147483648
- Maximum: 2147483647
imageX

Use Left (ImageX) to set the distance, in pixels, between the inserted image and the left edge of the frame. Required for BMP, PNG and TGA input.

- **Type:** integer
- **Required:** True
- **Minimum:** -2147483648
- **Maximum:** 2147483647

width

Specify the Width (Width) of the inserted image. Use a value that is less than or equal to the video resolution width. Leave this setting blank to use the native width of the image.

- **Type:** integer
- **Required:** False
- **Minimum:** -2147483648
- **Maximum:** 2147483647

startTime

Use Start time (StartTime) to specify the video timecode when the image is inserted in the output. This must be in timecode format (HH:MM:SS:FF)

- **Type:** string
- **Required:** False

opacity

Use Opacity (Opacity) to specify how much of the underlying video shows through the inserted image. 0 is transparent and 100 is fully opaque. Default is 50.

- **Type:** integer
- **Required:** True
- **Minimum:** 0
- **Maximum:** 100

layer

Use Layer (Layer) to specify how overlapping inserted images appear. Images with higher values of layer appear on top of images with lower values of layer.

- **Type:** integer
- **Required:** True
- **Minimum:** 0
- **Maximum:** 7

height

Specify the Height (Height) of the inserted image. Use a value that is less than or equal to the video resolution height. Leave this setting blank to use the native height of the image.

- **Type:** integer
- **Required:** False
**Properties**

**Minimum**: -2147483648  
**Maximum**: 2147483647

**imageInserterInput**

Use Image location (imageInserterInput) to specify the Amazon S3 location of the image to be inserted into the output. Use a 32 bit BMP, PNG, or TGA file that fits inside the video frame.

  - **Type**: string  
  - **Required**: True  
  - **Pattern**: `^(s3://)(.*)\.(bmp|BMP|png|PNG|tga|TGA)$`

**JobTemplate**

**settings**

  - **Type**: JobTemplateSettings (p. 259)  
  - **Required**: True

**lastUpdated**

The timestamp in epoch seconds when the Job template was last updated.

  - **Type**: string  
  - **Required**: False  
  - **Format**: date-time

**createdAt**

The timestamp in epoch seconds for Job template creation.

  - **Type**: string  
  - **Required**: False  
  - **Format**: date-time

**name**

A name you create for each job template. Each name must be unique within your account.

  - **Type**: string  
  - **Required**: True

**description**

An optional description you create for each job template.

  - **Type**: string  
  - **Required**: False

**category**

An optional category you create to organize your job templates.

  - **Type**: string
**Required**: False

**type**

*Type:* Type (p. 298)
*Required*: False

**arn**

An identifier for this resource that is unique within all of AWS.

*Type*: string
*Required*: False

**queue**

Optional. The queue that jobs created from this template are assigned to. If you don't specify this, jobs will go to the default queue.

*Type*: string
*Required*: False

**JobTemplateSettings**

**timecodeConfig**

Contains settings used to acquire and adjust timecode information from inputs.

*Type*: TimecodeConfig (p. 296)
*Required*: False

**adAvailOffset**

When specified, this offset (in milliseconds) is added to the input Ad Avail PTS time.

*Type*: integer
*Required*: False
*Minimum*: -1000
*Maximum*: 1000

**nielsenConfiguration**

Nielsen configuration settings

*Type*: NielsenConfiguration (p. 284)
*Required*: False

**inputs**

Use Inputs (inputs) to define the source file used in the transcode job. There can only be one input in a job template. Using the API, you can include multiple inputs when referencing a job template.

*Type*: Array of type InputTemplate (p. 254)
**outputGroups**

(OutputGroups) contains one group of settings for each set of outputs that share a common package type. All unpackaged files (MPEG-4, MPEG-2 TS, Quicktime, MXF, and no container) are grouped in a single output group as well. Required in (OutputGroups) is a group of settings that apply to the whole group. This required object depends on the value you set for (Type) under (OutputGroups)> (OutputGroupSettings). Type, settings object pairs are as follows:

- *FILE_GROUP_SETTINGS, FileGroupSettings
- *HLS_GROUP_SETTINGS, HlsGroupSettings
- *DASH_ISO_GROUP_SETTINGS, DashIsoGroupSettings
- *MS_SMOOTH_GROUP_SETTINGS, MsSmoothGroupSettings

  **Type:** Array of type OutputGroup (p. 287)
  **Required:** True

**timedMetadataInsertion**

  **Type:** TimedMetadataInsertion (p. 297)
  **Required:** False

**availBlanking**

Settings for ad avail blanking. Video can be blanked or overlaid with an image, and audio muted during SCTE-35 triggered ad avails.

  **Type:** AvailBlanking (p. 200)
  **Required:** False

**LanguageCode (Enum)**

Selects a specific language code from within an audio source.
AAR
AFR
AKA
SQI
AMH
ARG
HYE
ASM
AVA
AVE
AYM
AZE
BAM
BAK
EUS
BEL
BEN
BIH
BIS
BOS
BRE
BUL
MYA
CAT
KHM
CHA
CHE
NYA
CHU
CHV
COR
COS
CRE
HRV
CES
DAN
DIV
NLD
DZO
ENM
EPO
EST
EWE
FAO
FIJ
FIN
FRM
FUL
GLA
GLG
LUG
KAT
ELL
GRN
<table>
<thead>
<tr>
<th>Properties</th>
</tr>
</thead>
<tbody>
<tr>
<td>GUJ</td>
</tr>
<tr>
<td>HAT</td>
</tr>
<tr>
<td>HAU</td>
</tr>
<tr>
<td>HEB</td>
</tr>
<tr>
<td>HER</td>
</tr>
<tr>
<td>HMO</td>
</tr>
<tr>
<td>HUN</td>
</tr>
<tr>
<td>ISL</td>
</tr>
<tr>
<td>IDO</td>
</tr>
<tr>
<td>IBO</td>
</tr>
<tr>
<td>IND</td>
</tr>
<tr>
<td>INA</td>
</tr>
<tr>
<td>ILE</td>
</tr>
<tr>
<td>IKU</td>
</tr>
<tr>
<td>IPK</td>
</tr>
<tr>
<td>GLE</td>
</tr>
<tr>
<td>JAV</td>
</tr>
<tr>
<td>KAL</td>
</tr>
<tr>
<td>KAN</td>
</tr>
<tr>
<td>KAU</td>
</tr>
<tr>
<td>KAS</td>
</tr>
<tr>
<td>KAZ</td>
</tr>
<tr>
<td>KIK</td>
</tr>
<tr>
<td>KIN</td>
</tr>
<tr>
<td>KIR</td>
</tr>
<tr>
<td>KOM</td>
</tr>
<tr>
<td>KON</td>
</tr>
<tr>
<td>KUA</td>
</tr>
<tr>
<td>KUR</td>
</tr>
<tr>
<td>LAO</td>
</tr>
<tr>
<td>LAT</td>
</tr>
<tr>
<td>LAV</td>
</tr>
<tr>
<td>LIM</td>
</tr>
<tr>
<td>LIN</td>
</tr>
<tr>
<td>LIT</td>
</tr>
<tr>
<td>LUB</td>
</tr>
<tr>
<td>LTZ</td>
</tr>
<tr>
<td>MKD</td>
</tr>
<tr>
<td>MLG</td>
</tr>
<tr>
<td>MSA</td>
</tr>
<tr>
<td>MAL</td>
</tr>
<tr>
<td>MLT</td>
</tr>
<tr>
<td>GLV</td>
</tr>
<tr>
<td>MRI</td>
</tr>
<tr>
<td>MAR</td>
</tr>
<tr>
<td>MAH</td>
</tr>
<tr>
<td>MON</td>
</tr>
<tr>
<td>NAU</td>
</tr>
<tr>
<td>NAV</td>
</tr>
<tr>
<td>NDE</td>
</tr>
<tr>
<td>NBL</td>
</tr>
<tr>
<td>NDO</td>
</tr>
<tr>
<td>NEP</td>
</tr>
<tr>
<td>SME</td>
</tr>
</tbody>
</table>
NOR
NOB
NNO
OCI
OJI
ORI
ORM
OSS
PLI
FAS
POL
PUS
QUE
QAA
RON
ROH
RUN
SMO
SAG
SAN
SRD
SRB
SNA
III
SND
SIN
SLK
SLV
SOM
SOT
SUN
SWA
SSW
SWE
TGL
TAH
TGK
TAM
TAT
TEL
THA
BOD
TIR
TON
TSO
TSN
TUR
TUK
TWI
UIG
UKR
UZB
VEN
VOL
M2tsAudioBufferModel (Enum)

Selects between the DVB and ATSC buffer models for Dolby Digital audio.

- DVB
- ATSC

M2tsBufferModel (Enum)

Controls what buffer model to use for accurate interleaving. If set to MULTIPLEX, use multiplex buffer model. If set to NONE, this can lead to lower latency, but low-memory devices may not be able to play back the stream without interruptions.

- MULTIPLEX
- NONE

M2tsEbpAudioInterval (Enum)

When set to VIDEO_AND_FIXED_INTERVALS, audio EBP markers will be added to partitions 3 and 4. The interval between these additional markers will be fixed, and will be slightly shorter than the video EBP marker interval. When set to VIDEO_INTERVAL, these additional markers will not be inserted. Only applicable when EBP segmentation markers are is selected (segmentationMarkers is EBP or EBP_LEGACY).

- VIDEO_AND_FIXED_INTERVALS
- VIDEO_INTERVAL

M2tsEbpPlacement (Enum)

Selects which PIDs to place EBP markers on. They can either be placed only on the video PID, or on both the video PID and all audio PIDs. Only applicable when EBP segmentation markers are is selected (segmentationMarkers is EBP or EBP_LEGACY).

- VIDEO_AND_AUDIO_PIDS
- VIDEO_PID

M2tsEsRateInPes (Enum)

Controls whether to include the ES Rate field in the PES header.

- INCLUDE
**M2tsPcrControl (Enum)**

When set to PCR_EVERY_PES_PACKET, a Program Clock Reference value is inserted for every Packetized Elementary Stream (PES) header. This is effective only when the PCR PID is the same as the video or audio elementary stream.

- PCR_EVERY_PES_PACKET
- CONFIGURED_PCR_PERIOD

**M2tsRateMode (Enum)**

When set to CBR, inserts null packets into transport stream to fill specified bitrate. When set to VBR, the bitrate setting acts as the maximum bitrate, but the output will not be padded up to that bitrate.

- VBR
- CBR

**M2tsScte35Source (Enum)**

Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output. This is only available for certain containers.

- PASSTHROUGH
- NONE

**M2tsSegmentationMarkers (Enum)**

Inserts segmentation markers at each segmentation_time period. rai_segstart sets the Random Access Indicator bit in the adaptation field. rai_adapt sets the RAI bit and adds the current timecode in the private data bytes. psi_segstart inserts PAT and PMT tables at the start of segments. ebp adds Encoder Boundary Point information to the adaptation field as per OpenCable specification OC-SP-EBP-I01-130118. ebp_legacy adds Encoder Boundary Point information to the adaptation field using a legacy proprietary format.

- NONE
- RAI_SEGSTART
- RAI_ADAPT
- PSI_SEGSTART
- EBP
- EBP_LEGACY

**M2tsSegmentationStyle (Enum)**

The segmentation style parameter controls how segmentation markers are inserted into the transport stream. With avails, it is possible that segments may be truncated, which can influence where future segmentation markers are inserted. When a segmentation style of "reset_cadence" is selected and a segment is truncated due to an avail, we will reset the segmentation cadence. This means the subsequent segment will have a duration of $segmentation_time seconds. When a segmentation style of "maintain_cadence" is selected and a segment is truncated due to an avail, we will not reset the segmentation cadence. This means the subsequent segment will likely be truncated as well. However, all segments after that will have a duration of $segmentation_time seconds. Note that EBP lookahead is a slight exception to this rule.
MAINTAIN_CADENCE
RESET_CADENCE

**M2tsSettings**

**dvbTeletextPid**
Packet Identifier (PID) for input source DVB Teletext data to this output. Can be entered as a decimal or hexadecimal value.

- **Type**: integer
- **Required**: False
- **Minimum**: 32
- **Maximum**: 8182

**bitrate**
The output bitrate of the transport stream in bits per second. Setting to 0 lets the muxer automatically determine the appropriate bitrate. Other common values are 3750000, 7500000, and 15000000.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 2147483647

**segmentationTime**
The length in seconds of each segment. Required unless markers is set to _none_.

- **Type**: number
- **Required**: False
- **Format**: float
- **Minimum**: 0.0

**audioPids**
Packet Identifier (PID) of the elementary audio stream(s) in the transport stream. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values.

- **Type**: Array of type integer
- **Required**: False

**rateMode**

- **Type**: M2tsRateMode (p. 265)
- **Required**: False

**ebpAudioInterval**

- **Type**: M2tsEbpAudioInterval (p. 264)
- **Required**: False
fragmentTime

The length in seconds of each fragment. Only used with EBP markers.

Type: number
Required: False
Format: float
Minimum: 0.0

audioFramesPerPes

The number of audio frames to insert for each PES packet.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

maxPcrInterval

Maximum time in milliseconds between Program Clock References (PCRs) inserted into the transport stream.

Type: integer
Required: False
Minimum: 0
Maximum: 500

scte35Pid

Packet Identifier (PID) of the SCTE-35 stream in the transport stream. Can be entered as a decimal or hexadecimal value.

Type: integer
Required: False
Minimum: 32
Maximum: 8182

privateMetadataPid

Packet Identifier (PID) of the private metadata stream in the transport stream. Can be entered as a decimal or hexadecimal value.

Type: integer
Required: False
Minimum: 32
Maximum: 8182

pmtInterval

The number of milliseconds between instances of this table in the output transport stream.

Type: integer
Required: False
Minimum: 0
Maximum: 1000

**segmentationStyle**

Type: `M2tsSegmentationStyle (p. 265)`
Required: False

**audioBufferModel**

Type: `M2tsAudioBufferModel (p. 264)`
Required: False

**programNumber**

The value of the program number field in the Program Map Table.

Type: integer
Required: False
Minimum: 0
Maximum: 65535

**dvbNitSettings**

Type: `DvbNitSettings (p. 212)`
Required: False

**scte35Source**

Type: `M2tsScte35Source (p. 265)`
Required: False

**pmtPid**

Packet Identifier (PID) for the Program Map Table (PMT) in the transport stream. Can be entered as a decimal or hexadecimal value.

Type: integer
Required: False
Minimum: 32
Maximum: 8182

**bufferModel**

Type: `M2tsBufferModel (p. 264)`
Required: False

**ebpPlacement**

Type: `M2tsEbpPlacement (p. 264)`
**Properties**

**Required**: False

**dvbSdtSettings**

**Type**: DvbSdtSettings (p. 213)
**Required**: False

**nullPacketBitrate**

Value in bits per second of extra null packets to insert into the transport stream. This can be used if a downstream encryption system requires periodic null packets.

**Type**: number
**Required**: False
**Format**: float
**Minimum**: 0.0

**pcrPid**

Packet Identifier (PID) of the Program Clock Reference (PCR) in the transport stream. When no value is given, the encoder will assign the same value as the Video PID. Can be entered as a decimal or hexadecimal value.

**Type**: integer
**Required**: False
**Minimum**: 32
**Maximum**: 8182

**minEbpInterval**

When set, enforces that Encoder Boundary Points do not come within the specified time interval of each other by looking ahead at input video. If another EBP is going to come in within the specified time interval, the current EBP is not emitted, and the segment is "stretched" to the next marker. The lookahead value does not add latency to the system. The Live Event must be configured elsewhere to create sufficient latency to make the lookahead accurate.

**Type**: integer
**Required**: False
**Minimum**: 0
**Maximum**: 10000

**transportStreamId**

The value of the transport stream ID field in the Program Map Table.

**Type**: integer
**Required**: False
**Minimum**: 0
**Maximum**: 65535

**pcrControl**

**Type**: M2tsPcrControl (p. 265)
**videoPid**

Packet Identifier (PID) of the elementary video stream in the transport stream. Can be entered as a decimal or hexadecimal value.

- **Type:** integer
- **Required:** False
- **Minimum:** 32
- **Maximum:** 8182

**esRateInPes**

- **Type:** M2tsEsRateInPes (p. 264)
- **Required:** False

**segmentationMarkers**

- **Type:** M2tsSegmentationMarkers (p. 265)
- **Required:** False

**dvbTdtSettings**

- **Type:** DvbTdtSettings (p. 217)
- **Required:** False

**patInterval**

The number of milliseconds between instances of this table in the output transport stream.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 1000

**dvbSubPids**

Packet Identifier (PID) for input source DVB Subtitle data to this output. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values.

- **Type:** Array of type integer
- **Required:** False

**M3u8PcrControl (Enum)**

When set to PCR_EVERY_PES_PACKET a Program Clock Reference value is inserted for every Packetized Elementary Stream (PES) header. This parameter is effective only when the PCR PID is the same as the video or audio elementary stream.

- **PCR_EVERY_PES_PACKET**
- **CONFIGURED_PCR_PERIOD**
M3u8Scte35Source (Enum)
Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output. This is only available for certain containers.

- PASSTHROUGH
- NONE

M3u8Settings

pmtPid
Packet Identifier (PID) for the Program Map Table (PMT) in the transport stream. Can be entered as a decimal or hexadecimal value.

- **Type**: integer
- **Required**: False
- **Minimum**: 32
- **Maximum**: 8182

pcrPid
Packet Identifier (PID) of the Program Clock Reference (PCR) in the transport stream. When no value is given, the encoder will assign the same value as the Video PID. Can be entered as a decimal or hexadecimal value.

- **Type**: integer
- **Required**: False
- **Minimum**: 32
- **Maximum**: 8182

audioPids
Packet Identifier (PID) of the elementary audio stream(s) in the transport stream. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values.

- **Type**: Array of type integer
- **Required**: False

audioFramesPerPes
The number of audio frames to insert for each PES packet.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 2147483647

scte35Pid
Packet Identifier (PID) of the SCTE-35 stream in the transport stream. Can be entered as a decimal or hexadecimal value.

- **Type**: integer
**transportStreamId**

The value of the transport stream ID field in the Program Map Table.

- **Type:** integer
- **Required:** False
- **Minimum:** 32
- **Maximum:** 8182

**videoPid**

Packet Identifier (PID) of the elementary video stream in the transport stream. Can be entered as a decimal or hexadecimal value.

- **Type:** integer
- **Required:** False
- **Minimum:** 32
- **Maximum:** 8182

**pcrControl**

- **Type:** M3u8PcrControl (p. 270)
- **Required:** False

**privateMetadataPid**

Packet Identifier (PID) of the private metadata stream in the transport stream. Can be entered as a decimal or hexadecimal value.

- **Type:** integer
- **Required:** False
- **Minimum:** 32
- **Maximum:** 8182

**pmtInterval**

The number of milliseconds between instances of this table in the output transport stream.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 1000

**patInterval**

The number of milliseconds between instances of this table in the output transport stream.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
**Maximum**: 1000

**programNumber**
The value of the program number field in the Program Map Table.

**Type**: integer
**Required**: False
**Minimum**: 0
**Maximum**: 65535

**timedMetadataPid**
Packet Identifier (PID) of the timed metadata stream in the transport stream. Can be entered as a decimal or hexadecimal value.

**Type**: integer
**Required**: False
**Minimum**: 32
**Maximum**: 8182

**timedMetadata**

**Type**: TimedMetadata (p. 297)
**Required**: False

**scte35Source**

**Type**: M3u8Scte35Source (p. 271)
**Required**: False

**MovClapAtom (Enum)**
When enabled, include 'clap' atom if appropriate for the video output settings.

- INCLUDE
- EXCLUDE

**MovCslgAtom (Enum)**
When enabled, file composition times will start at zero, composition times in the 'ctts' (composition time to sample) box for B-frames will be negative, and a 'cslg' (composition shift least greatest) box will be included per 14496-1 amendment 1. This improves compatibility with Apple players and tools.

- INCLUDE
- EXCLUDE

**MovMpeg2FourCCControl (Enum)**
When set to XDCAM, writes MPEG2 video streams into the QuickTime file using XDCAM fourcc codes. This increases compatibility with Apple editors and players, but may decrease compatibility with other players. Only applicable when the video codec is MPEG2.

- XDCAM
MPEG

MovPaddingControl (Enum)
If set to OMNEON, inserts Omneon-compatible padding

OMNEON
NONE

MovReference (Enum)
A value of 'external' creates separate media files and the wrapper file (.mov) contains references to these media files. A value of 'self_contained' creates only a wrapper (.mov) file and this file contains all of the media.

SELF_CONTAINED
EXTERNAL

MovSettings

reference
Type: MovReference (p. 274)
Required: False

paddingControl
Type: MovPaddingControl (p. 274)
Required: False

mpeg2FourCCControl
Type: MovMpeg2FourCCControl (p. 273)
Required: False

cslgAtom
Type: MovCslgAtom (p. 273)
Required: False

clapAtom
Type: MovClapAtom (p. 273)
Required: False

Mp2Settings

channels
Set Channels to specify the number of channels in this output audio track. Choosing Mono in the console will give you 1 output channel; choosing Stereo will give you 2. In the API, valid values are 1 and 2.
Properties

Type: integer
Required: False
Minimum: 1
Maximum: 2

**bitrate**

Average bitrate in bits/second.

Type: integer
Required: False
Minimum: 32000
Maximum: 384000

**sampleRate**

Sample rate in hz.

Type: integer
Required: False
Minimum: 32000
Maximum: 48000

**Mp4CslgAtom (Enum)**

When enabled, file composition times will start at zero, composition times in the 'ctts' (composition time to sample) box for B-frames will be negative, and a 'cslg' (composition shift least greatest) box will be included per 14496-1 amendment 1. This improves compatibility with Apple players and tools.

INCLUDE
EXCLUDE

**Mp4FreeSpaceBox (Enum)**

Inserts a free-space box immediately after the moov box.

INCLUDE
EXCLUDE

**Mp4MoovPlacement (Enum)**

If set to PROGRESSIVE_DOWNLOAD, the MOOV atom is relocated to the beginning of the archive as required for progressive downloading. Otherwise it is placed normally at the end.

PROGRESSIVE_DOWNLOAD
NORMAL

**Mp4Settings**

**mp4MajorBrand**

Overrides the "Major Brand" field in the output file. Usually not necessary to specify.

Type: string
Required: False
**moovPlacement**

Type: Mp4MoovPlacement (p. 275)  
Required: False

**csigAtom**

Type: Mp4CslgAtom (p. 275)  
Required: False

**freeSpaceBox**

Type: Mp4FreeSpaceBox (p. 275)  
Required: False

**Mpeg2AdaptiveQuantization (Enum)**

Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

- OFF
- LOW
- MEDIUM
- HIGH

**Mpeg2CodecLevel (Enum)**

Use Level (Mpeg2CodecLevel) to set the MPEG-2 level for the video output.

- AUTO
- LOW
- MAIN
- HIGH1440
- HIGH

**Mpeg2CodecProfile (Enum)**

Use Profile (Mpeg2CodecProfile) to set the MPEG-2 profile for the video output.

- MAIN
- PROFILE_422

**Mpeg2FramerateControl (Enum)**

Using the API, set FramerateControl to INITIALIZE_FROM_SOURCE if you want the service to use the framerate from the input. Using the console, do this by choosing INITIALIZE_FROM_SOURCE for Framerate.

- INITIALIZE_FROM_SOURCE
- SPECIFIED

**Mpeg2FramerateConversionAlgorithm (Enum)**

When set to INTERPOLATE, produces smoother motion during framerate conversion.
DUPLICATE_DROP
INTERPOLATE

**Mpeg2GopSizeUnits (Enum)**

Indicates if the GOP Size in MPEG2 is specified in frames or seconds. If seconds the system will convert the GOP Size into a frame count at run time.

- FRAMES
- SECONDS

**Mpeg2InterlaceMode (Enum)**

Use Interlace mode (InterlaceMode) to choose the scan line type for the output.

* Top Field First (TOP_FIELD) and Bottom Field First (BOTTOM_FIELD) produce interlaced output with the entire output having the same field polarity (top or bottom first).

* Follow, Default Top (FOLLOW_TOP_FIELD) and Follow, Default Bottom (FOLLOW_BOTTOM_FIELD) use the same field polarity as the source. Therefore, behavior depends on the input scan type. If the source is interlaced, the output will be interlaced with the same polarity as the source (it will follow the source). The output could therefore be a mix of "top field first" and "bottom field first". If the source is progressive, the output will be interlaced with "top field first" or "bottom field first" polarity, depending on which of the Follow options you chose.

- PROGRESSIVE
- TOP_FIELD
- BOTTOM_FIELD
- FOLLOW_TOP_FIELD
- FOLLOW_BOTTOM_FIELD

**Mpeg2IntraDcPrecision (Enum)**

Use Intra DC precision (Mpeg2IntraDcPrecision) to set quantization precision for intra-block DC coefficients. If you choose the value auto, the service will automatically select the precision based on the per-frame compression ratio.

- AUTO
- INTRA_DC_PRECISION_8
- INTRA_DC_PRECISION_9
- INTRA_DC_PRECISION_10
- INTRA_DC_PRECISION_11

**Mpeg2ParControl (Enum)**

Using the API, enable ParFollowSource if you want the service to use the pixel aspect ratio from the input. Using the console, do this by choosing Follow source for Pixel aspect ratio.

- INITIALIZE_FROM_SOURCE
- SPECIFIED

**Mpeg2QualityTuningLevel (Enum)**

Use Quality tuning level (Mpeg2QualityTuningLevel) to specify whether to use single-pass or multipass video encoding.
SINGLE_PASS
MULTI_PASS

Mpeg2RateControlMode (Enum)

Use Rate control mode (Mpeg2RateControlMode) to specify whether the bitrate is variable (vbr) or constant (cbr).

VBR
CBR

Mpeg2SceneChangeDetect (Enum)

Scene change detection (inserts I-frames on scene changes).

DISABLED
ENABLED

Mpeg2Settings

mIInterval

Enforces separation between repeated (cadence) I-frames and I-frames inserted by Scene Change Detection. If a scene change I-frame is within I-interval frames of a cadence I-frame, the GOP is shrunk and/or stretched to the scene change I-frame. GOP stretch requires enabling lookahead as well as setting I-interval. The normal cadence resumes for the next GOP. This setting is only used when Scene Change Detect is enabled. Note: Maximum GOP stretch = GOP size + Min-I-interval - 1

Type: integer
Required: False
Minimum: 0
Maximum: 30

parNumerator

Pixel Aspect Ratio numerator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

gopSizeUnits

Type: Mpeg2GopSizeUnits (p. 277)
Required: False

hrdBufferSize

Size of buffer (HRD buffer model). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

**qualityTuningLevel**

Type: Mpeg2QualityTuningLevel (p. 277)
Required: False

**maxBitrate**

Maximum bitrate in bits/second (for VBR mode only). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

**bitrate**

Average bitrate in bits/second. Required for VBR, CBR, and ABR. Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.

Type: integer
Required: False
Minimum: 1000
Maximum: 2147483647

**spatialAdaptiveQuantization**

Type: Mpeg2SpatialAdaptiveQuantization (p. 282)
Required: False

**slowPal**

Type: Mpeg2SlowPal (p. 282)
Required: False

**codecProfile**

Type: Mpeg2CodecProfile (p. 276)
Required: False

**intraDcPrecision**

Type: Mpeg2IntraDcPrecision (p. 277)
Required: False

**softness**

Softness. Selects quantizer matrix, larger values reduce high-frequency content in the encoded image.
Type: integer
Required: False
Minimum: 0
Maximum: 128

framerateControl
Type: Mpeg2FramerateControl (p. 276)
Required: False

telecine
Type: Mpeg2Telecine (p. 282)
Required: False

framerateConversionAlgorithm
Type: Mpeg2FramerateConversionAlgorithm (p. 276)
Required: False

codecLevel
Type: Mpeg2CodecLevel (p. 276)
Required: False

temporalAdaptiveQuantization
Type: Mpeg2TemporalAdaptiveQuantization (p. 282)
Required: False

hrdBufferInitialFillPercentage
Percentage of the buffer that should initially be filled (HRD buffer model).
Type: integer
Required: False
Minimum: 0
Maximum: 100

framerateNumerator
Framerate numerator - framerate is a fraction, e.g. 24000 / 1001 = 23.976 fps.
Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

numberBFramesBetweenReferenceFrames
Number of B-frames between reference frames.
Type: integer
Properties

**Required**: False
**Minimum**: 0
**Maximum**: 7

**gopClosedCadence**
Frequency of closed GOPs. In streaming applications, it is recommended that this be set to 1 so a decoder joining mid-stream will receive an IDR frame as quickly as possible. Setting this value to 0 will break output segmenting.

**Type**: integer
**Required**: False
**Minimum**: 0
**Maximum**: 2147483647

**framerateDenominator**
Framerate denominator.

**Type**: integer
**Required**: False
**Minimum**: 1
**Maximum**: 2147483647

**adaptiveQuantization**

**Type**: Mpeg2AdaptiveQuantization (p. 276)
**Required**: False

**interlaceMode**

**Type**: Mpeg2InterlaceMode (p. 277)
**Required**: False

**gopSize**
GOP Length (keyframe interval) in frames or seconds. Must be greater than zero.

**Type**: number
**Required**: False
**Format**: float
**Minimum**: 0.0

**sceneChangeDetect**

**Type**: Mpeg2SceneChangeDetect (p. 278)
**Required**: False

**parDenominator**
Pixel Aspect Ratio denominator.

**Type**: integer
**Required**: False
**Properties**

**Minimum**: 1  
**Maximum**: 2147483647

**parControl**

Type: [Mpeg2ParControl](#)  
Required: False

**syntax**

Type: [Mpeg2Syntax](#)  
Required: False

**rateControlMode**

Type: [Mpeg2RateControlMode](#)  
Required: False

**Mpeg2SlowPal (Enum)**

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

- DISABLED  
- ENABLED

**Mpeg2SpatialAdaptiveQuantization (Enum)**

Adjust quantization within each frame based on spatial variation of content complexity.

- DISABLED  
- ENABLED

**Mpeg2Syntax (Enum)**

Produces a Type D-10 compatible bitstream (SMPTE 356M-2001).

- DEFAULT  
- D_10

**Mpeg2Telecine (Enum)**

Only use Telecine (Mpeg2Telecine) when you set Framerate (Framerate) to 29.970. Set Telecine (Mpeg2Telecine) to Hard (hard) to produce a 29.97i output from a 23.976 input. Set it to Soft (soft) to produce 23.976 output and leave conversion to the player.

- NONE  
- SOFT  
- HARD

**Mpeg2TemporalAdaptiveQuantization (Enum)**

Adjust quantization within each frame based on temporal variation of content complexity.
DISABLED
ENABLED

**MsSmoothAudioDeduplication (Enum)**

COMBINE_DUPLICATE_STREAMS combines identical audio encoding settings across a Microsoft Smooth output group into a single audio stream.

- COMBINE_DUPLICATE_STREAMS
- NONE

**MsSmoothEncryptionSettings**

**spekeKeyProvider**

- **Type:** SpekeKeyProvider (p. 293)
- **Required:** True

**MsSmoothGroupSettings**

**fragmentLength**

Use Fragment length (FragmentLength) to specify the mp4 fragment sizes in seconds. Fragment length must be compatible with GOP size and framerate.

- **Type:** integer
- **Required:** True
- **Minimum:** 1
- **Maximum:** 2147483647

**encryption**

- **Type:** MsSmoothEncryptionSettings (p. 283)
- **Required:** False

**audioDeduplication**

- **Type:** MsSmoothAudioDeduplication (p. 283)
- **Required:** False

**manifestEncoding**

- **Type:** MsSmoothManifestEncoding (p. 284)
- **Required:** False

**destination**

Use Destination (Destination) to specify the S3 output location and the output filename base. Destination accepts format identifiers. If you do not specify the base filename in the URI, the service will use the filename of the input file. If your job has multiple inputs, the service uses the filename of the first input file.

- **Type:** string
**MsSmoothManifestEncoding (Enum)**

Use Manifest encoding (MsSmoothManifestEncoding) to specify the encoding format for the server and client manifest. Valid options are utf8 and utf16.

- UTF8
- UTF16

**NielsenConfiguration**

**distributorId**

Use Distributor ID (DistributorID) to specify the distributor ID that is assigned to your organization by Neilsen.

- **Type:** string
- **Required:** False

**breakoutCode**

Use Nielsen Configuration (NielsenConfiguration) to set the Nielsen measurement system breakout code. Supported values are 0, 3, 7, and 9.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 9

**NoiseReducer**

**filter**

- **Type:** NoiseReducerFilter (p. 284)
- **Required:** True

**filterSettings**

- **Type:** NoiseReducerFilterSettings (p. 285)
- **Required:** False

**spatialFilterSettings**

- **Type:** NoiseReducerSpatialFilterSettings (p. 285)
- **Required:** False

**NoiseReducerFilter (Enum)**

Use Noise reducer filter (NoiseReducerFilter) to select one of the following spatial image filtering functions. To use this setting, you must also enable Noise reducer (NoiseReducer).
* Bilateral is an edge preserving noise reduction filter

* Mean (softest), Gaussian, Lanczos, and Sharpen (sharpest) are convolution filters

* Conserve is a min/max noise reduction filter

* Spatial is frequency-domain filter based on JND principles.

BILATERAL
MEAN
GAUSSIAN
LANCZOS
SHARPEN
CONSERVE
SPATIAL

**NoiseReducerFilterSettings**

**strength**
Relative strength of noise reducing filter. Higher values produce stronger filtering.

* Type: integer
  * Required: False
  * Minimum: 0
  * Maximum: 3

**NoiseReducerSpatialFilterSettings**

**strength**
Relative strength of noise reducing filter. Higher values produce stronger filtering.

* Type: integer
  * Required: False
  * Minimum: 0
  * Maximum: 16

**postFilterSharpenStrength**
Specify strength of post noise reduction sharpening filter, with 0 disabling the filter and 3 enabling it at maximum strength.

* Type: integer
  * Required: False
  * Minimum: 0
  * Maximum: 3

**speed**
The speed of the filter, from -2 (lower speed) to 3 (higher speed), with 0 being the nominal value.

* Type: integer
  * Required: False
  * Minimum: -2
  * Maximum: 3
Output

extension

Use Extension (Extension) to specify the file extension for outputs in File output groups. If you do not specify a value, the service will use default extensions by container type as follows:

* MPEG-2 transport stream, m2ts
* Quicktime, mov
* MXF container, mxf
* MPEG-4 container, mp4
* No Container, the service will use codec extensions (e.g. AAC, H265, H265, AC3)

  Type: string
  Required: False

videoDescription

(VideoDescription) contains a group of video encoding settings. The specific video settings depend on the video codec you choose when you specify a value for Video codec (codec). Include one instance of (VideoDescription) per output.

  Type: VideoDescription (p. 299)
  Required: False

audioDescriptions

(AudioDescriptions) contains groups of audio encoding settings organized by audio codec. Include one instance of (AudioDescriptions) per output. (AudioDescriptions) can contain multiple groups of encoding settings.

  Type: Array of type AudioDescription (p. 195)
  Required: False

containerSettings

  Type: ContainerSettings (p. 208)
  Required: False

preset

Use Preset (Preset) to specify a preset for your transcoding settings. Provide the system or custom preset name. You can specify either Preset (Preset) or Container settings (ContainerSettings), but not both.

  Type: string
  Required: False

outputSettings

  Type: OutputSettings (p. 288)
  Required: False
captionDescriptions

(CaptionDescriptions) contains groups of captions settings. For each output that has captions, include one instance of (CaptionDescriptions). (CaptionDescriptions) can contain multiple groups of captions settings.

Type: Array of type CaptionDescription (p. 204)
Required: False

nameModifier

Use Name modifier (NameModifier) to have the service add a string to the end of each output filename. You specify the base filename as part of your destination URI. When you create multiple outputs in the same output group, Name modifier is required. Name modifier also accepts format identifiers. For DASH ISO outputs, if you use the format identifiers $Number$ or $Time$ in one output, you must use them in the same way in all outputs of the output group.

Type: string
Required: False

OutputChannelMapping

inputChannels

Type: Array of type integer
Required: True

OutputGroup

outputs

This object holds groups of encoding settings, one group of settings per output.

Type: Array of type Output (p. 286)
Required: True

outputGroupSettings

Type: OutputGroupSettings (p. 288)
Required: True

name

Type: string
Required: False

customName

Use Custom Group Name (CustomName) to specify a name for the output group. This value is displayed on the console and can make your job settings JSON more human-readable. It does not affect your outputs. Use up to twelve characters that are either letters, numbers, spaces, or underscores.

Type: string
Required: False
OutputGroupSettings

dashIsoGroupSettings
  Type: DashIsoGroupSettings (p. 209)
  Required: False

fileGroupSettings
  Type: FileGroupSettings (p. 223)
  Required: False

msSmoothGroupSettings
  Type: MsSmoothGroupSettings (p. 283)
  Required: False

type
  Type: OutputGroupType (p. 288)
  Required: True

hlsGroupSettings
  Type: HlsGroupSettings (p. 247)
  Required: False

OutputGroupType (Enum)
  HLS_GROUP_SETTINGS
  DASH_ISO_GROUP_SETTINGS
  FILE_GROUP_SETTINGS
  MS_SMOOTH_GROUP_SETTINGS

OutputSdt (Enum)
  Selects method of inserting SDT information into output stream. "Follow input SDT" copies SDT information from input stream to output stream. "Follow input SDT if present" copies SDT information from input stream to output stream if SDT information is present in the input, otherwise it will fall back on the user-defined values. Enter "SDT Manually" means user will enter the SDT information. "No SDT" means output stream will not contain SDT information.
  SDT_FOLLOW
  SDT_FOLLOW_IF_PRESENT
  SDT_MANUAL
  SDT_NONE

OutputSettings

hlsSettings
  Type: HlsSettings (p. 251)
**Required:** False

**ProResCodecProfile (Enum)**

Use Profile (ProResCodecProfile) to specify the type of Apple ProRes codec to use for this output.

- APPLE_PRORES_422
- APPLE_PRORES_422_HQ
- APPLE_PRORES_422_LT
- APPLE_PRORES_422_PROXY

**ProResFramerateControl (Enum)**

Using the API, set FramerateControl to INITIALIZE_FROM_SOURCE if you want the service to use the framerate from the input. Using the console, do this by choosing INITIALIZE_FROM_SOURCE for Framerate.

- INITIALIZE_FROM_SOURCE
- SPECIFIED

**ProResFramerateConversionAlgorithm (Enum)**

When set to INTERPOLATE, produces smoother motion during framerate conversion.

- DUPLICATE_DROP
- INTERPOLATE

**ProResInterlaceMode (Enum)**

Use Interlace mode (InterlaceMode) to choose the scan line type for the output.

* Top Field First (TOP_FIELD) and Bottom Field First (BOTTOM_FIELD) produce interlaced output with the entire output having the same field polarity (top or bottom first).

* Follow, Default Top (FOLLOW_TOP_FIELD) and Follow, Default Bottom (FOLLOW_BOTTOM_FIELD) use the same field polarity as the source. Therefore, behavior depends on the input scan type. If the source is interlaced, the output will be interlaced with the same polarity as the source (it will follow the source). The output could therefore be a mix of "top field first" and "bottom field first". If the source is progressive, the output will be interlaced with "top field first" or "bottom field first" polarity, depending on which of the Follow options you chose.

- PROGRESSIVE
- TOP_FIELD
- BOTTOM_FIELD
- FOLLOW_TOP_FIELD
- FOLLOW_BOTTOM_FIELD

**ProResParControl (Enum)**

Use (ProResParControl) to specify how the service determines the pixel aspect ratio. Set to Follow source (INITIALIZE_FROM_SOURCE) to use the pixel aspect ratio from the input. To specify a different pixel aspect ratio: Using the console, choose it from the dropdown menu. Using the API, set ProResParControl to (SPECIFIED) and provide for (ParNumerator) and (ParDenominator).
**ProresSettings**

**slowPal**
- Type: ProresSlowPal (p. 291)
- Required: False

**framerateControl**
- Type: ProresFramerateControl (p. 289)
- Required: False

**telecine**
- Type: ProresTelecine (p. 291)
- Required: False

**framerateDenominator**
Framerate denominator.
- Type: integer
- Required: False
- Minimum: 1
- Maximum: 2147483647

**framerateConversionAlgorithm**
- Type: ProresFramerateConversionAlgorithm (p. 289)
- Required: False

**interlaceMode**
- Type: ProresInterlaceMode (p. 289)
- Required: False

**codecProfile**
- Type: ProresCodecProfile (p. 289)
- Required: False

**parNumerator**
Pixel Aspect Ratio numerator.
- Type: integer
- Required: False
- Minimum: 1
Maximum: 2147483647

parControl

Type: ProresParControl (p. 289)
Required: False

parDenominator

Pixel Aspect Ratio denominator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

framerateNumerator

When you use the API for transcode jobs that use framerate conversion, specify the framerate as a fraction. For example, 24000 / 1001 = 23.976 fps. Use FramerateNumerator to specify the numerator of this fraction. In this example, use 24000 for the value of FramerateNumerator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

ProresSlowPal (Enum)

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

DISABLED
ENABLED

ProresTelecine (Enum)

Only use Telecine (ProresTelecine) when you set Framerate (Framerate) to 29.970. Set Telecine (ProresTelecine) to Hard (hard) to produce a 29.97i output from a 23.976 input. Set it to Soft (soft) to produce 23.976 output and leave conversion to the player.

NONE
HARD

Rectangle

width

Width of rectangle in pixels.

Type: integer
Required: True
Minimum: -2147483648
Properties

Maximum: 2147483647

x
The distance, in pixels, between the rectangle and the left edge of the video frame.

Type: integer
Required: True
Minimum: -2147483648
Maximum: 2147483647

y
The distance, in pixels, between the rectangle and the top edge of the video frame.

Type: integer
Required: True
Minimum: -2147483648
Maximum: 2147483647

height
Height of rectangle in pixels.

Type: integer
Required: True
Minimum: -2147483648
Maximum: 2147483647

RemixSettings

channelsOut
Specify the number of channels in this output after remixing. Valid values: 1, 2, 4, 6, 8

Type: integer
Required: True
Minimum: 1
Maximum: 8

channelMapping

Type: ChannelMapping (p. 206)
Required: True

channelsIn
Specify the number of audio channels from your input that you want to use in your output. With remixing, you might combine or split the data in these channels, so the number of channels in your final output might be different.

Type: integer
Required: True
Minimum: 1
Maximum: 16

**RespondToAfd (Enum)**

Use Respond to AFD (RespondToAfd) to specify how the service changes the video itself in response to AFD values in the input.

* Choose Respond to clip the input video frame according to the AFD value, input display aspect ratio, and output display aspect ratio.

* Choose Passthrough to include the input AFD values. Do not choose this when AfdSignaling is set to (NONE). A preferred implementation of this workflow is to set RespondToAfd to (NONE) and set AfdSignaling to (AUTO).

* Choose None to remove all input AFD values from this output.

    NONE
    RESPOND
    PASSTHROUGH

**ScalingBehavior (Enum)**

Applies only if your input aspect ratio is different from your output aspect ratio. Enable Stretch to output (StretchToOutput) to have the service stretch your video image to fit. Leave this setting disabled to allow the service to letterbox your video instead. This setting overrides any positioning value you specify elsewhere in the job.

    DEFAULT
    STRETCH_TO_OUTPUT

**SccDestinationFramerate (Enum)**

Set Framerate (SccDestinationFramerate) to make sure that the captions and the video are synchronized in the output. Specify a framerate that matches the framerate of the associated video. If the video framerate is 29.97, choose 29.97 dropframe (FRAMERATE_29_97_DROPFRAME) only if the video has video_insertion=true and drop_frame_timecode=true; otherwise, choose 29.97 non-dropframe (FRAMERATE_29_97_NON_DROPFRAME).

    FRAMERATE_23_97
    FRAMERATE_24
    FRAMERATE_29_97_DROPFRAME
    FRAMERATE_29_97_NON_DROPFRAME

**SccDestinationSettings**

framerate

    Type: SccDestinationFramerate (p. 293)
    Required: False

**SpekeKeyProvider**

resourceId

The SPEKE-compliant server uses Resource ID (ResourceId) to identify content.
Properties

systemIds

Relates to SPEKE implementation. DRM system identifiers. DASH output groups support a max of two system ids. Other group types support one system id.

- **Type**: Array of type string
- **Required**: True

url

Use URL (Url) to specify the SPEKE-compliant server that will provide keys for content.

- **Type**: string
- **Required**: True
- **Format**: uri
- **Pattern**: ^https:/\/

StaticKeyProvider

keyFormatVersions

Relates to DRM implementation. Either a single positive integer version value or a slash delimited list of version values (1/2/3).

- **Type**: string
- **Required**: False
- **Pattern**: ^(\d+(\d+)*)$

keyFormat

Relates to DRM implementation. Sets the value of the KEYFORMAT attribute. Must be 'identity' or a reverse DNS string. May be omitted to indicate an implicit value of 'identity'.

- **Type**: string
- **Required**: False
- **Pattern**: ^(identity|[A-Za-z]{2,6}([.][A-Za-z0-9-]{1,63})+)$

staticKeyValue

Relates to DRM implementation. Use a 32-character hexadecimal string to specify Key Value (StaticKeyValue).

- **Type**: string
- **Required**: True
- **Pattern**: ^[A-Za-z0-9]{32}$

url

Relates to DRM implementation. The location of the license server used for protecting content.
**TeletextDestinationSettings**

`pageNumber`

Set `pageNumber` to the Teletext page number for the destination captions for this output. This value must be a three-digit hexadecimal string; strings ending in -FF are invalid. If you are passing through the entire set of Teletext data, do not use this field.

```
Type: string
Required: True
Format: uri
```

**TeletextSourceSettings**

`pageNumber`

Use `Page Number` (PageNumber) to specify the three-digit hexadecimal page number that will be used for Teletext captions. Do not use this setting if you are passing through teletext from the input source to output.

```
Type: string
Required: False
Pattern: ^[1-8][0-9a-fA-F][0-9a-eA-E]$
```

**TimecodeBurnin**

`prefix`

Use `Prefix` (Prefix) to place ASCII characters before any burned-in timecode. For example, a prefix of "EZ-" will result in the timecode "EZ-00:00:00:00". Provide either the characters themselves or the ASCII code equivalents. The supported range of characters is 0x20 through 0x7e. This includes letters, numbers, and all special characters represented on a standard English keyboard.

```
Type: string
Required: False
Pattern: ^[- ]+$
```

`fontSize`

Use `Font Size` (FontSize) to set the font size of any burned-in timecode. Valid values are 10, 16, 32, 48.

```
Type: integer
Required: False
Minimum: 10
Maximum: 48
```

`position`

```
Type: TimecodeBurninPosition (p. 296)
```
**Required:** False

### TimecodeBurninPosition (Enum)

Use **Position (Position)** under **Timecode burn-in (TimecodeBurnIn)** to specify the location the burned-in timecode on output video.

- TOP_CENTER
- TOP_LEFT
- TOP_RIGHT
- MIDDLE_LEFT
- MIDDLE_CENTER
- MIDDLE_RIGHT
- BOTTOM_LEFT
- BOTTOM_CENTER
- BOTTOM_RIGHT

### TimecodeConfig

**timestampOffset**

Only applies to outputs that support program-date-time stamp. Use **Time stamp offset (TimestampOffset)** to overwrite the timecode date without affecting the time and frame number. To use this, you must also enable **Insert program-date-time (InsertProgramDateTime)** in the output settings.

- **Type:** string
- **Required:** False
- **Pattern:** ^([0-9]{4})-(0[1-9]|1[0-2])-(0[1-9]|1[2][0-9]|3[01])$

**anchor**

If you use an editing platform that relies on an anchor timecode, use **Anchor Timecode (Anchor)** to specify a timecode that will match the input video frame to the output video frame. Use 24-hour format with frame number, (HH:MM:SS:FF) or (HH:MM:SS;FF). This setting ignores framerate conversion. System behavior for Anchor Timecode varies depending on your setting for **Timecode source (TimecodeSource)**.

* If Timecode source (TimecodeSource) is set to Specified Start (specifiedstart), the first input frame is the specified value in Start Timecode (Start). Anchor Timecode (Anchor) and Start Timecode (Start) are used calculate output timecode.

* If Timecode source (TimecodeSource) is set to Start at 0 (zerobased) the first frame is 00:00:00:00.

* If Timecode source (TimecodeSource) is set to Embedded (embedded), the first frame is the timecode value on the first input frame of the input.

- **Type:** string
- **Required:** False
- **Format:** timecode
- **Pattern:** ^([01][0-9]|2[0-4]):[0-5][0-9]:[0-5][0-9]:[0-9]{2}$

**start**

Only use when you set Timecode Source (TimecodeSource) to Specified Start (SPECIFIEDSTART). Use **Start timecode (Start)** to specify the timecode for the initial frame. Use 24-hour format with frame number, (HH:MM:SS:FF) or (HH:MM:SS;FF).
**Type**: string  
**Required**: False  
**Format**: timecode  
**Pattern**: ^([01][0-9]|2[0-4]):[0-5][0-9]:[0-5][0-9][;][0-9]{2}\$  

### source

**Type**: TimecodeSource (p. 297)  
**Required**: False

**TimecodeSource (Enum)**

Use Timecode source (TimecodeSource) to set how timecodes are handled within this input. To make sure that your video, audio, captions, and markers are synchronized and that time-based features, such as image inserter, work correctly, choose the Timecode source option that matches your assets. All timecodes are in a 24-hour format with frame number (HH:MM:SS:FF).

* Embedded (EMBEDDED) - Use the timecode that is in the input video. If no embedded timecode is in the source, the service will use Start at 0 (ZEROBASED) instead.

* Start at 0 (ZEROBASED) - Set the timecode of the initial frame to 00:00:00:00.

* Specified Start (SPECIFIEDSTART) - Set the timecode of the initial frame to a value other than zero. You use Start timecode (Start) to provide this value.

  EMBEDDED
  ZEROBASED
  SPECIFIEDSTART

**TimedMetadata (Enum)**

If PASSTHROUGH, inserts ID3 timed metadata from the timed_metadata REST command into this output. Only available for certain containers.

  PASSTHROUGH
  NONE

**TimedMetadataInsertion**

**id3Insertions**

Id3Insertions contains the array of Id3Insertion instances.

  **Type**: Array of type Id3Insertion (p. 252)  
  **Required**: True

**TtmlDestinationSettings**

**stylePassthrough**

  **Type**: TtmlStylePassthrough (p. 298)  
  **Required**: False
**TtmlStylePassthrough (Enum)**

Pass through style and position information from a TTML-like input source (TTML, SMPTE-TT, CFF-TT) to the CFF-TT output or TTML output.

- ENABLED
- DISABLED

**Type (Enum)**

- SYSTEM
- CUSTOM

**UpdateJobTemplateRequest**

**settings**

- **Type**: JobTemplateSettings (p. 259)
- **Required**: False

**name**

The name of the job template you are modifying

- **Type**: string
- **Required**: True

**description**

The new description for the job template, if you are changing it.

- **Type**: string
- **Required**: False

**category**

The new category for the job template, if you are changing it.

- **Type**: string
- **Required**: False

**queue**

The new queue for the job template, if you are changing it.

- **Type**: string
- **Required**: False

**UpdateJobTemplateResponse**

**jobTemplate**

- **Type**: JobTemplate (p. 258)
**Required**: False

**VideoCodec (Enum)**

Type of video codec

- FRAME_CAPTURE
- H_264
- H_265
- MPEG2
- PRORES

**VideoCodecSettings**

**h265Settings**

- **Type**: H265Settings (p. 237)
- **Required**: False

**codec**

- **Type**: VideoCodec (p. 299)
- **Required**: True

**proresSettings**

- **Type**: ProresSettings (p. 290)
- **Required**: False

**mpeg2Settings**

- **Type**: Mpeg2Settings (p. 278)
- **Required**: False

**h264Settings**

- **Type**: H264Settings (p. 228)
- **Required**: False

**frameCaptureSettings**

- **Type**: FrameCaptureSettings (p. 224)
- **Required**: False

**VideoDescription**

**fixedAfd**

Applies only if you set AFD Signaling(AfdSignaling) to Fixed (FIXED). Use Fixed (FixedAfd) to specify a four-bit AFD value which the service will write on all frames of this video output.

- **Type**: integer
Required: False
Minimum: 0
Maximum: 15

scalingBehavior
  Type: ScalingBehavior (p. 293)
  Required: False

respondToAfd
  Type: RespondToAfd (p. 293)
  Required: False

codecSettings
  Type: VideoCodecSettings (p. 299)
  Required: True

afdSignaling
  Type: AfdSignaling (p. 193)
  Required: False

colorMetadata
  Type: ColorMetadata (p. 207)
  Required: False

timecodeInsertion
  Type: VideoTimecodeInsertion (p. 303)
  Required: False

width
Use Width (Width) to define the video resolution width, in pixels, for this output. If you don't provide a value here, the service will use the input width.

  Type: integer
  Required: False
  Minimum: 32
  Maximum: 4096

sharpness
Use Sharpness (Sharpness) setting to specify the strength of anti-aliasing. This setting changes the width of the anti-alias filter kernel used for scaling. Sharpness only applies if your output resolution is different from your input resolution, and if you set Anti-alias (AntiAlias) to ENABLED. 0 is the softest setting, 100 the sharpest, and 50 recommended for most content.

  Type: integer
  Required: False
Properties

Minimum: 0
Maximum: 100

antiAlias

Type: AntiAlias (p. 194)
Required: False

videoPreprocessors

Find additional transcoding features under Preprocessors (VideoPreprocessors). Enable the features at each output individually. These features are disabled by default.

Type: VideoPreprocessor (p. 301)
Required: False

position

Use Position (Position) to point to a rectangle object to define your position. This setting overrides any other aspect ratio.

Type: Rectangle (p. 291)
Required: False

dropFrameTimecode

Type: DropFrameTimecode (p. 212)
Required: False

crop

Applies only if your input aspect ratio is different from your output aspect ratio. Use Input cropping rectangle (Crop) to specify the video area the service will include in the output. This will crop the input source, causing video pixels to be removed on encode. Do not use this setting if you have enabled Stretch to output (stretchToOutput) in your output settings.

Type: Rectangle (p. 291)
Required: False

height

Use the Height (Height) setting to define the video resolution height for this output. Specify in pixels. If you don't provide a value here, the service will use the input height.

Type: integer
Required: False
Minimum: 32
Maximum: 2160

VideoPreprocessor

timecodeBurnin

Timecode burn-in (TimecodeBurnIn)—Burns the output timecode and specified prefix into the output.
Properties

Type: TimecodeBurnin (p. 295)
Required: False

noiseReducer

Enable the Noise reducer (NoiseReducer) feature to remove noise from your video output if necessary. Enable or disable this feature for each output individually. This setting is disabled by default.

Type: NoiseReducer (p. 284)
Required: False

colorCorrector

Enable the Color corrector (ColorCorrector) feature if necessary. Enable or disable this feature for each output individually. This setting is disabled by default.

Type: ColorCorrector (p. 206)
Required: False

imageInserter

Enable the Image inserter (ImageInserter) feature to include a graphic overlay on your video. Enable or disable this feature for each output individually. This setting is disabled by default.

Type: ImageInserter (p. 252)
Required: False

deinterlacer

Use Deinterlacer (Deinterlacer) to produce smoother motion and a clearer picture.

Type: Deinterlacer (p. 211)
Required: False

VideoSelector

colorSpace

Type: ColorSpace (p. 207)
Required: False

hdr10Metadata

Type: Hdr10Metadata (p. 243)
Required: False

programNumber

Selects a specific program from within a multi-program transport stream. Note that Quad 4K is not currently supported.

Type: integer
Required: False
**Properties**

- **pid**
  
  Use PID (Pid) to select specific video data from an input file. Specify this value as an integer; the system automatically converts it to the hexadecimal value. For example, 257 selects PID 0x101. A PID, or packet identifier, is an identifier for a set of data in an MPEG-2 transport stream container.

  - **Type:** integer
  - **Required:** False
  - **Minimum:** 1
  - **Maximum:** 2147483647

- **colorSpaceUsage**
  
  - **Type:** ColorSpaceUsage (p. 208)
  - **Required:** False

- **VideoTimecodeInsertion (Enum)**
  
  Enable Timecode insertion to include timecode information in this output. Do this in the API by setting (VideoTimecodeInsertion) to (PIC_TIMING_SEI). To get timecodes to appear correctly in your output, also set up the timecode configuration for your job in the input settings. Only enable Timecode insertion when the input framerate is identical to output framerate. Disable this setting to remove the timecode from the output. Default is disabled.

  - **DISABLED**
  - **PIC_TIMING_SEI**

- **WavSettings**

  - **channels**
    
    Set Channels to specify the number of channels in this output audio track. With WAV, valid values 1, 2, 4, and 8. In the console, these values are Mono, Stereo, 4-Channel, and 8-Channel, respectively.

    - **Type:** integer
    - **Required:** False
    - **Minimum:** 1
    - **Maximum:** 8

  - **bitDepth**
    
    Specify Bit depth (BitDepth), in bits per sample, to choose the encoding quality for this audio track.

    - **Type:** integer
    - **Required:** False
    - **Minimum:** 16
    - **Maximum:** 24

  - **sampleRate**
    
    Sample rate in Hz.
Jobs

URI
/jobs

HTTP Methods

GET

Operation ID: ListJobs

Retrieve a JSON array of up to twenty of your most recently created jobs. This array includes in-process, completed, and errored jobs. This will return the jobs themselves, not just a list of the jobs. To retrieve the twenty next most recent jobs, use the nextToken string returned with the array.

Query Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>status</td>
<td>String</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>nextToken</td>
<td>String</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>maxResults</td>
<td>String</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>order</td>
<td>String</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>queue</td>
<td>String</td>
<td>False</td>
<td></td>
</tr>
</tbody>
</table>

Responses

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>ListJobsResponse (p. 316)</td>
<td>200: OkResponse</td>
</tr>
<tr>
<td>400</td>
<td>ExceptionBody (p. 337)</td>
<td>400: BadRequestException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The conditional request failed. The service can't process your request because of a problem in the request. Please check your request form and syntax.</td>
</tr>
<tr>
<td>500</td>
<td>ExceptionBody (p. 337)</td>
<td>500: InternalServiceException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The service encountered an unexpected condition and cannot fulfill your request.</td>
</tr>
</tbody>
</table>
HTTP Methods

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>403</td>
<td>ExceptionBody (p. 337)</td>
<td>403: AccessDeniedException You don't have permissions for this action with the credentials you sent. Please check your authorization credentials. You should be sending credentials using the AWS Signature Version 4 signing process.</td>
</tr>
<tr>
<td>404</td>
<td>ExceptionBody (p. 337)</td>
<td>404: ResourceNotFoundException The resource you requested does not exist.</td>
</tr>
<tr>
<td>429</td>
<td>ExceptionBody (p. 337)</td>
<td>429: LimitExceededException Too many requests have been sent in too short of a time. The service limits the rate at which it will accept requests. For example, you may be hitting your account limits for preset creation or job submission.</td>
</tr>
<tr>
<td>409</td>
<td>ExceptionBody (p. 337)</td>
<td>409: ResourceInUseException The service could not complete your request because there is a conflict with the current state of the resource. For example, you may be trying to delete a Queue that has jobs processing.</td>
</tr>
</tbody>
</table>

POST

Operation ID: CreateJob

Create a new transcoding job. For information about jobs and job settings, see the User Guide at http://docs.aws.amazon.com/mediaconvert/latest/ug/what-is.html

Responses

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>201</td>
<td>CreateJobResponse (p. 327)</td>
<td>201: CreatedResponse Your resource has been successfully created.</td>
</tr>
<tr>
<td>400</td>
<td>ExceptionBody (p. 337)</td>
<td>400: BadRequestException The conditional request failed. The service can't process your</td>
</tr>
<tr>
<td>Status Code</td>
<td>Response Model</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>----------------</td>
<td>-------------</td>
</tr>
<tr>
<td>500</td>
<td>ExceptionBody (p. 337)</td>
<td>500: InternalServiceException. The service encountered an unexpected condition and cannot fulfill your request.</td>
</tr>
<tr>
<td>403</td>
<td>ExceptionBody (p. 337)</td>
<td>403: AccessDeniedException. You don't have permissions for this action with the credentials you sent. Please check your authorization credentials. You should be sending credentials using the AWS Signature Version 4 signing process.</td>
</tr>
<tr>
<td>404</td>
<td>ExceptionBody (p. 337)</td>
<td>404: ResourceNotFoundException. The resource you requested does not exist.</td>
</tr>
<tr>
<td>429</td>
<td>ExceptionBody (p. 337)</td>
<td>429: LimitExceededException. Too many requests have been sent in too short of a time. The service limits the rate at which it will accept requests. For example, you may be hitting your account limits for preset creation or job submission.</td>
</tr>
<tr>
<td>409</td>
<td>ExceptionBody (p. 337)</td>
<td>409: ResourceInUseException. The service could not complete your request because there is a conflict with the current state of the resource. For example, you may be trying to delete a Queue that has jobs processing.</td>
</tr>
</tbody>
</table>

**Schemas**

**Request Bodies**

**Example GET**

```json
{}
```
"nextToken (p. 414)": "string",
"maxResults (p. 414)": integer,
"queue (p. 414)": "string",
"status (p. 414)": enum,
"order (p. 414)": enum
}

Example POST

```
{
  "settings (p. 358)": {
    "timecodeConfig (p. 409)": {
      "timestampOffset (p. 447)": "string",
      "anchor (p. 447)": "string",
      "start (p. 448)": "string",
      "source (p. 448)": enum
    },
    "adAvailOffset (p. 409)": integer,
    "nielsenConfiguration (p. 409)": {
      "distributorId (p. 435)": "string",
      "breakoutCode (p. 435)": integer
    },
    "inputs (p. 409)": [
      {
        "audioSelectors (p. 402)": {
        },
        "audioSelectorGroups (p. 402)": {
        },
        "videoSelector (p. 402)": {
          "colorSpace (p. 453)": enum,
          "hdr10Metadata (p. 454)": {
            "redPrimaryY (p. 392)": integer,
            "greenPrimaryY (p. 392)": integer,
            "whitePointX (p. 392)": integer,
            "maxLuminance (p. 392)": integer,
            "greenPrimaryX (p. 393)": integer,
            "whitePointY (p. 393)": integer,
            "redPrimaryX (p. 393)": integer,
            "bluePrimaryX (p. 393)": integer,
            "maxFrameAverageLightLevel (p. 393)": integer,
            "bluePrimaryY (p. 393)": integer,
            "maxContentLightLevel (p. 394)": integer,
            "minLuminance (p. 394)": integer
          },
          "programNumber (p. 454)": integer,
          "pid (p. 454)": integer,
          "colorSpaceUsage (p. 454)": enum
        },
        "denoiseFilter (p. 402)": enum,
        "filterEnable (p. 402)": enum,
        "deblockFilter (p. 402)": enum,
        "filterStrength (p. 402)": integer,
        "programNumber (p. 403)": integer,
        "timecodeSource (p. 403)": enum,
        "captionSelectors (p. 403)": {
        },
        "fileInput (p. 403)": "string",
        "inputClippings (p. 403)": [
          {
            "startTimecode (p. 404)": "string",
            "endTimecode (p. 404)": "string"
          }
        ],
        "psiControl (p. 404)": enum
      }
    ]
  }
```
"outputGroups (p. 409)": [
  "outputs (p. 438)": [
    "extension (p. 437)": "string",
    "videoDescription (p. 437)": {
      "fixedAfD (p. 450)": integer,
      "scalingBehavior (p. 451)": enum,
      "respondToAfD (p. 451)": enum,
      "codecSettings (p. 451)": {
        "h265Settings (p. 450)": {
          "slices (p. 386)": integer,
          "minIInterval (p. 386)": integer,
          "parNumerator (p. 386)": integer,
          "flickerAdaptiveQuantization (p. 387)": enum,
          "gopSizeUnits (p. 387)": enum,
          "hrdBufferSize (p. 387)": integer,
          "qualityTuningLevel (p. 387)": enum,
          "maxBitrate (p. 387)": integer,
          "bitrate (p. 387)": integer,
          "spatialAdaptiveQuantization (p. 387)": enum,
          "sampleAdaptiveOffsetFilterMode (p. 388)": enum,
          "temporalIds (p. 388)": enum,
          "slowPal (p. 388)": enum,
          "tiles (p. 388)": enum,
          "codecProfile (p. 388)": enum,
          "alternateTransferFunctionSei (p. 388)": enum,
          "unregisteredSeiTimecode (p. 388)": enum,
          "framerateControl (p. 388)": enum,
          "telecine (p. 388)": enum,
          "framerateConversionAlgorithm (p. 388)": enum,
          "codecLevel (p. 389)": enum,
          "numberReferenceFrames (p. 389)": integer,
          "temporalAdaptiveQuantization (p. 389)": enum,
          "hrdBufferInitialFillPercentage (p. 389)": integer,
          "framerateNumerator (p. 389)": integer,
          "numberBFramesBetweenReferenceFrames (p. 389)": integer,
          "gopClosedCadence (p. 389)": integer,
          "framerateDenominator (p. 390)": integer,
          "adaptiveQuantization (p. 390)": enum,
          "interlaceMode (p. 390)": enum,
          "gopSize (p. 390)": number,
          "gopBReference (p. 390)": enum,
          "sceneChangeDetect (p. 390)": enum,
          "parDenominator (p. 390)": integer,
          "parControl (p. 391)": enum,
          "rateControlMode (p. 391)": enum
        },
        "proresSettings (p. 450)": {
          "slowPal (p. 441)": enum,
          "framerateControl (p. 441)": enum,
          "telecine (p. 441)": enum,
          "framerateDenominator (p. 441)": integer,
          "framerateConversionAlgorithm (p. 441)": enum,
          "interlaceMode (p. 442)": enum,
          "codecProfile (p. 442)": enum,
          "parNumerator (p. 442)": integer,
          "parControl (p. 442)": enum,
          "parDenominator (p. 442)": integer,
          "framerateNumerator (p. 442)": integer
        },
        "mpeg2Settings (p. 450)": {
          "minIInterval (p. 429)": integer,
"parNumerator (p. 429)": integer,
"gopSizeUnits (p. 429)": enum,
"hrdBuffersize (p. 429)": integer,
"qualityTuningLevel (p. 429)": enum,
"maxBitrate (p. 430)": integer,
"bitrate (p. 430)": integer,
"spatialAdaptiveQuantization (p. 430)": enum,
"slowPal (p. 430)": enum,
"codecProfile (p. 430)": enum,
"intraDcPrecision (p. 430)": enum,
"softness (p. 430)": integer,
"framerateControl (p. 430)": enum,
"telecine (p. 431)": enum,
"framerateConversionAlgorithm (p. 431)": enum,
"codecLevel (p. 431)": enum,
"temporalAdaptiveQuantization (p. 431)": enum,
"hrdBuffersizePercentage (p. 431)": integer,
"framerateNumerator (p. 431)": integer,
"numberBFramesBetweenReferenceFrames (p. 431)": integer,
"gopClosedCadence (p. 431)": integer,
"framerateDenominator (p. 431)": integer,
"numberReferenceFrames (p. 431)": integer,
"temporalAdaptiveQuantization (p. 431)": enum,
"repeatPps (p. 380)": integer,
"hrdBuffersizePercentage (p. 380)": integer,
"framerateNumerator (p. 380)": integer,
"numberBFramesBetweenReferenceFrames (p. 380)": integer,
"gopClosedCadence (p. 381)": integer,
"fieldEncoding (p. 381)": integer,
"entropyEncoding (p. 381)": integer,
"framerateDenominator (p. 381)": integer,
"adaptiveQuantization (p. 381)": enum,
"interlaceMode (p. 381)": enum,
"gopSize (p. 381)": number,
"gopReference (p. 382)": enum,
"syntax (p. 382)": enum,
"rateControlMode (p. 382)": enum
},
"frameCaptureSettings (p. 450)": {
"framerateDenominator (p. 374)": integer,
"maxCaptures (p. 374)": integer,
"framerateNumerator (p. 374)": integer,
"quality (p. 374)": integer
}
},
"afdSignaling (p. 451)": enum,
"colorMetadata (p. 451)": enum,
"timecodeInsertion (p. 451)": enum,
"width (p. 451)": integer,
"sharpness (p. 451)": integer,
"antiAlias (p. 452)": enum,
"videoPreprocessors (p. 452)": {
"timecodeBurnin (p. 453)": {
"prefix (p. 446)": "string",
"fontSize (p. 447)": integer,
"position (p. 447)": enum
}
},
"noiseReducer (p. 453)": {
"filter (p. 435)": enum,
"filterSettings (p. 435)": {
"strength (p. 436)": integer
}
},
"spatialFilterSettings (p. 435)": {
"strength (p. 436)": integer,
"postFilterSharpenStrength (p. 436)": integer,
"speed (p. 436)": integer
}
},
"colorCorrector (p. 453)": {
"saturation (p. 355)": integer,
"brightness (p. 355)": integer,
"hdr10Metadata (p. 355)": {
"redPrimaryY (p. 392)": integer,
"greenPrimaryY (p. 392)": integer,
"whitePointX (p. 392)": integer,
"maxLuminance (p. 392)": integer,
"greenPrimaryX (p. 393)": integer,
"whitePointY (p. 393)": integer,
"redPrimaryX (p. 393)": integer,
"bluePrimaryX (p. 393)": integer,
"maxFrameAverageLightLevel (p. 393)": integer,
"bluePrimaryY (p. 393)": integer,
"maxContentLightLevel (p. 394)": integer,
"minLuminance (p. 394)": integer
}
},
"contrast (p. 356)": integer,
"hue (p. 356)": integer,
"colorSpaceConversion (p. 356)": enum
},
"imageInserter (p. 453)": {
"insertableImages (p. 402)": [
"duration (p. 405)": integer,
"fadeOut (p. 405)": integer,
"imageY (p. 406)": integer,
"fadeIn (p. 406)": integer,
"imageX (p. 406)": integer,
"width (p. 406)": integer,
"startTime (p. 406)": "string",
"opacity (p. 406)": integer,
"layer (p. 407)": integer,
"height (p. 407)": integer,
"imageInserterInput (p. 407)": "string"
}
],
"deinterlacer (p. 453)": {
"mode (p. 361)": enum,
"control (p. 361)": enum,
"algorithm (p. 361)": enum
}
],
"position (p. 452)": {
"width (p. 443)": integer,
"x (p. 443)": integer,
"y (p. 443)": integer,
"height (p. 443)": integer
},
"dropFrameTimecode (p. 452)": enum,
"crop (p. 452)": {
"width (p. 443)": integer,
"x (p. 443)": integer,
"y (p. 443)": integer,
"height (p. 443)": integer
},
"height (p. 452)": integer
],
"audioDescriptions (p. 437)": [
{
"languageCodeControl (p. 343)": enum,
"audioTypeControl (p. 343)": enum,
"remixSettings (p. 344)": {
"channelsOut (p. 443)": integer,
"channelMapping (p. 444)": {
"outputChannels (p. 355)": [
{"inputChannels (p. 438)": [integer
]
}
],
"channelsIn (p. 444)": integer
},
"audioType (p. 344)": integer,
"audioSourceName (p. 344)": "string",
"codecSettings (p. 344)": {
"codec (p. 342)": enum,
"wavSettings (p. 343)": {
"channels (p. 454)": integer,
"bitDepth (p. 455)": integer,
"sampleRate (p. 455)": integer
},
"ac3Settings (p. 343)": {
"dynamicRangeCompressionProfile (p. 340)": enum,
"dialnorm (p. 340)": integer,
"codingMode (p. 340)": enum,
"metadataControl (p. 340)": enum,
"lfeFilter (p. 341)": enum,
"bitrate (p. 341)": integer,
"bitstreamMode (p. 341)": enum,
"sampleRate (p. 341)": integer
},
"aacSettings (p. 343)": {
"vbrQuality (p. 338)": enum,
"codecProfile (p. 338)": enum,
"codingMode (p. 338)": enum,
"specification (p. 338)": enum,
"bitrate (p. 338)": integer,
"rawFormat (p. 338)": enum,
"rateControlMode (p. 339)": enum,
"sampleRate (p. 339)": integer,
"audioDescriptionBroadcasterMix (p. 339)": enum
},
"aiffSettings (p. 343)": {
  "channels (p. 341)": integer,
  "bitDepth (p. 342)": integer,
  "sampleRate (p. 342)": integer
},
"eac3Settings (p. 343)": {
  "dialnorm (p. 369)": integer,
  "passthroughControl (p. 369)": enum,
  "metadataControl (p. 369)": enum,
  "bitrate (p. 369)": integer,
  "dynamicRangeCompressionRf (p. 369)": enum,
  "sampleRate (p. 369)": integer,
  "1LtRtSurroundMixLevel (p. 369)": number,
  "surroundExMode (p. 370)": enum,
  "dynamicRangeCompressionLine (p. 370)": enum,
  "lfeControl (p. 370)": enum,
  "codingMode (p. 370)": enum,
  "surroundMode (p. 370)": enum,
  "attenuationControl (p. 370)": enum,
  "lfeFilter (p. 370)": enum,
  "phaseControl (p. 370)": enum,
  "1LtRtCenterMixLevel (p. 370)": number,
  "dcFilter (p. 371)": enum,
  "stereoDownmix (p. 371)": enum,
  "bitstreamMode (p. 371)": enum,
  "loRoSurroundMixLevel (p. 371)": number,
  "loRoCenterMixLevel (p. 371)": number
},
"mp2Settings (p. 343)": {
  "channels (p. 425)": integer,
  "bitrate (p. 425)": integer,
  "sampleRate (p. 426)": integer
},
"languageCode (p. 344)": enum,
"streamName (p. 344)": "string",
"audioNormalizationSettings (p. 345)": {
  "targetLkfs (p. 345)": number,
  "algorithmControl (p. 346)": enum,
  "peakCalculation (p. 346)": enum,
  "loudnessLogging (p. 346)": enum,
  "correctionGateLevel (p. 346)": integer,
  "algorithm (p. 346)": enum
},
"containerSettings (p. 437)": {
  "container (p. 357)": enum,
  "mp4Settings (p. 357)": {
    "mp4MajorBrand (p. 426)": "string",
    "moovPlacement (p. 426)": enum,
    "csrlAtom (p. 426)": enum,
    "freeSpaceBox (p. 427)": enum
  },
  "m3u8Settings (p. 357)": {
    "pmtPid (p. 422)": integer,
    "pcrPid (p. 422)": integer,
    "audioPids (p. 422)": [ integer
AWS Elemental MediaConvert API Reference
Schemas

`m2tsSettings (p. 357)`: {
    "dvbTeletextPid (p. 417)": integer,
    "bitrate (p. 417)": integer,
    "segmentationTime (p. 417)": number,
    "audioPids (p. 417)": [
        integer
    ],
    "rateMode (p. 417)": enum,
    "ebpAudioInterval (p. 417)": enum,
    "fragmentTime (p. 418)": number,
    "audioFramesPerPes (p. 418)": integer,
    "maxPcrInterval (p. 418)": integer,
    "scte35Pid (p. 418)": integer,
    "privateMetadataPid (p. 418)": integer,
    "pmtInterval (p. 418)": integer,
    "segmentationStyle (p. 419)": enum,
    "audioBufferModel (p. 419)": enum,
    "programNumber (p. 419)": integer,
    "dvbNitSettings (p. 419)": {
        "networkName (p. 361)": "string",
        "networkId (p. 362)": integer,
        "nitInterval (p. 362)": integer
    },
    "nullPacketBitrate (p. 420)": number,
    "pcrPid (p. 420)": integer,
    "minEbpInterval (p. 420)": integer,
    "transportStreamId (p. 420)": integer,
    "pcrControl (p. 420)": enum,
    "videoPid (p. 420)": integer,
    "esRateInPes (p. 421)": enum,
    "segmentationMarkers (p. 421)": enum,
    "dvbTdtSettings (p. 421)": {
        "tdtInterval (p. 367)": integer
    },
    "patInterval (p. 421)": integer,
    "dvbSubPids (p. 421)": [
        integer
    ],
    "movSettings (p. 357)": {
        "reference (p. 425)": enum,
        "paddingControl (p. 425)": enum,
        "scte35Source (p. 424)": enum
    }
}

"audioFramesPerPes (p. 422)": integer,
"scte35Pid (p. 422)": integer,
"transportStreamId (p. 422)": integer,
"videoPid (p. 423)": integer,
"pcrControl (p. 423)": enum,
"privateMetadataPid (p. 423)": integer,
"pmtInterval (p. 423)": integer,
"patInterval (p. 423)": integer,
"programNumber (p. 423)": integer,
"timedMetadataPid (p. 424)": integer,
"timedMetadata (p. 424)": enum,
"scte35Source (p. 424)": enum

"dvbTeletextPid (p. 417)": integer,
"bitrate (p. 417)": integer,
"segmentationTime (p. 417)": number,
"audioPids (p. 417)": [
    integer
],
"rateMode (p. 417)": enum,
"ebpAudioInterval (p. 417)": enum,
"fragmentTime (p. 418)": number,
"audioFramesPerPes (p. 418)": integer,
"maxPcrInterval (p. 418)": integer,
"scte35Pid (p. 418)": integer,
"privateMetadataPid (p. 418)": integer,
"pmtInterval (p. 418)": integer,
"segmentationStyle (p. 419)": enum,
"audioBufferModel (p. 419)": enum,
"programNumber (p. 419)": integer,
"dvbNitSettings (p. 419)": {
    "networkName (p. 361)": "string",
    "networkId (p. 362)": integer,
    "nitInterval (p. 362)": integer
},
"nullPacketBitrate (p. 420)": number,
"pcrPid (p. 420)": integer,
"minEbpInterval (p. 420)": integer,
"transportStreamId (p. 420)": integer,
"pcrControl (p. 420)": enum,
"videoPid (p. 420)": integer,
"esRateInPes (p. 421)": enum,
"segmentationMarkers (p. 421)": enum,
"dvbTdtSettings (p. 421)": {
    "tdtInterval (p. 367)": integer
},
"patInterval (p. 421)": integer,
"dvbSubPids (p. 421)": [
    integer
]

"reference (p. 425)": enum,
"paddingControl (p. 425)": enum,
"mpeg2FourCCControl (p. 425)": enum,
"cslgAtom (p. 425)": enum,
"clapAtom (p. 425)": enum
},
"f4vSettings (p. 357)": {
  "moovPlacement (p. 373)": enum
},
"preset (p. 437)": "string",
"outputSettings (p. 437)": {
  "hlsSettings (p. 440)": {
    "iFrameOnlyManifest (p. 400)": enum,
    "segmentModifier (p. 400)": "string",
    "audioTrackType (p. 401)": enum,
    "audioRenditionSets (p. 401)": "string",
    "audioGroupId (p. 401)": "string"
  }
},
"captionDescriptions (p. 438)": [
  {
    "captionSelectorName (p. 352)": "string",
    "languageDescription (p. 352)": "string",
    "languageCode (p. 353)": enum,
    "destinationSettings (p. 353)": {
      "burninDestinationSettings (p. 353)": {
        "xPosition (p. 348)": integer,
        "backgroundColor (p. 349)": enum,
        "teletextSpacing (p. 349)": enum,
        "yPosition (p. 349)": integer,
        "backgroundOpacity (p. 349)": integer,
        "fontOpacity (p. 349)": integer,
        "shadowOpacity (p. 350)": integer,
        "fontResolution (p. 350)": integer,
        "shadowYOffset (p. 350)": integer,
        "outlineSize (p. 350)": integer,
        "fontSize (p. 350)": integer,
        "shadowXOffset (p. 351)": integer,
        "alignment (p. 351)": enum,
        "shadowColor (p. 351)": enum,
        "fontColor (p. 351)": enum
      },
      "teletextDestinationSettings (p. 353)": {
        "pageNumber (p. 446)": "string"
      },
      "ttmlDestinationSettings (p. 353)": {
        "stylePassthrough (p. 449)": enum
      },
      "destinationType (p. 353)": enum,
      "dvbSubDestinationSettings (p. 353)": {
        "xPosition (p. 363)": integer,
        "backgroundColor (p. 363)": enum,
        "teletextSpacing (p. 363)": enum,
        "yPosition (p. 363)": integer,
        "backgroundOpacity (p. 363)": integer,
        "fontOpacity (p. 363)": integer,
        "shadowOpacity (p. 364)": integer,
        "fontResolution (p. 364)": integer,
        "shadowYOffset (p. 364)": integer,
        "outlineSize (p. 364)": integer,
        "outlineColor (p. 364)": enum,
        "fontSize (p. 364)": integer,
        "shadowXOffset (p. 365)": integer,
        "alignment (p. 365)": enum,
        "shadowColor (p. 365)": enum,
        "fontColor (p. 365)": enum
      }
    }
  }
]
"sccDestinationSettings (p. 353)": {
  "framerate (p. 445)": enum
}
,"nameModifier (p. 438)": "string"
],
"outputGroupSettings (p. 438)": {
  "dashIsoGroupSettings (p. 439)": {
    "fragmentLength (p. 359)": integer,
    "baseUrl (p. 359)": "string",
    "minBufferTime (p. 359)": integer,
    "encryption (p. 359)": {
      "spekeKeyProvider (p. 359)": {
        "resourceId (p. 445)": "string",
        "systemIds (p. 445)": [ "string"
      ],
      "url (p. 445)": "string"
    }
  }
},
  "destination (p. 359)": "string",
  "segmentLength (p. 360)": integer,
  "segmentControl (p. 360)": enum,
  "hbbtvCompliance (p. 360)": enum
},
"fileGroupSettings (p. 439)": {
  "destination (p. 373)": "string"
},
"msSmoothGroupSettings (p. 439)": {
  "fragmentLength (p. 434)": integer,
  "encryption (p. 434)": {
    "spekeKeyProvider (p. 434)": {
      "resourceId (p. 445)": "string",
      "systemIds (p. 445)": [ "string"
    ],
    "url (p. 445)": "string"
  }
},
  "audioDeduplication (p. 434)": enum,
  "manifestEncoding (p. 434)": enum,
  "destination (p. 434)": "string"
},
  "type (p. 439)": enum,
  "hlsGroupSettings (p. 439)": {
    "segmentsPerSubdirectory (p. 396)": integer,
    "streamInfResolution (p. 397)": enum,
    "timestampDeltaMilliseconds (p. 397)": integer,
    "outputSelection (p. 397)": enum,
    "captionLanguageMappings (p. 397)": [ {
      "languageDescription (p. 394)": "string",
      "captionChannel (p. 395)": integer,
      "languageCode (p. 395)": enum
    }
  ],
  "clientCache (p. 397)": enum,
  "codecSpecification (p. 397)": enum,
  "destination (p. 397)": "string",
  "segmentControl (p. 397)": enum,
  "timedMetadataId3Frame (p. 397)": enum,
  "timedMetadataId3Period (p. 398)": integer,
"captionLanguageSetting (p. 398)": enum,
"minSegmentLength (p. 398)": integer,
"directoryStructure (p. 398)": enum,
"programDateTime (p. 398)": enum,
"baseUrl (p. 398)": "string",
"encryption (p. 398)": {
  "initializationVectorInManifest (p. 395)": enum,
  "constantInitializationVector (p. 396)": "string",
  "staticKeyProvider (p. 396)": {
    "keyFormatVersions (p. 445)": "string",
    "keyFormat (p. 445)": "string",
    "staticKeyValue (p. 446)": "string",
    "url (p. 446)": "string"
  },
  "type (p. 396)": enum,
  "spekeKeyProvider (p. 396)": {
    "resourceId (p. 445)": "string",
    "systemIds (p. 445)": [
      "string"
    ],
    "url (p. 445)": "string"
  },
  "encryptionMethod (p. 396)": enum,
  "adMarkers (p. 398)": [
    enum
  ],
  "programDateTimePeriod (p. 399)": integer,
  "manifestCompression (p. 399)": enum,
  "segmentLength (p. 399)": integer,
  "manifestDurationFormat (p. 399)": enum
},
"name (p. 439)": "string",
"customName (p. 439)": "string"
},
"timedMetadataInsertion (p. 410)": {
  "id3Insertions (p. 449)": [
    {"id3 (p. 401)": "string",
     "timecode (p. 401)": "string"
    }
  ],
  "availableBlanking (p. 410)": {
    "availableBlankingImage (p. 348)": "string"
  },
  "role (p. 358)": "string",
  "clientRequestToken (p. 358)": "string",
  "jobTemplate (p. 358)": "string",
  "userMetadata (p. 358)": {
  },
  "queue (p. 358)": "string"
}

Response Bodies

Example ListJobsResponse

{ "nextToken (p. 415)": "string",
  "jobs (p. 415)": [ "string" ]
}
{ "settings (p. 407)": {
    "timecodeConfig (p. 409)": {
        "timestampOffset (p. 447)": "string",
        "anchor (p. 447)": "string",
        "start (p. 448)": "string",
        "source (p. 448)": enum
    },
    "adAvailOffset (p. 409)": integer,
    "nielsenConfiguration (p. 409)": {
        "distributorId (p. 435)": "string",
        "breakoutCode (p. 435)": integer
    },
    "inputs (p. 409)": [
        {
            "audioSelectors (p. 402)": {
            },
            "audioSelectorGroups (p. 402)": {
            },
            "videoSelector (p. 402)": {
                "colorSpace (p. 453)": enum,
                "hdr10Metadata (p. 454)": {
                    "redPrimaryY (p. 392)": integer,
                    "greenPrimaryY (p. 392)": integer,
                    "whitePointX (p. 392)": integer,
                    "maxLuminance (p. 392)": integer,
                    "greenPrimaryX (p. 393)": integer,
                    "whitePointY (p. 393)": integer,
                    "redPrimaryX (p. 393)": integer,
                    "bluePrimaryX (p. 393)": integer,
                    "maxFrameAverageLightLevel (p. 393)": integer,
                    "bluePrimaryY (p. 393)": integer,
                    "maxContentLightLevel (p. 394)": integer,
                    "minLuminance (p. 394)": integer
                },
                "programNumber (p. 454)": integer,
                "pid (p. 454)": integer,
                "colorSpaceUsage (p. 454)": enum
            },
            "denoiseFilter (p. 402)": enum,
            "filterEnable (p. 402)": enum,
            "deblockFilter (p. 402)": enum,
            "filterStrength (p. 402)": integer,
            "programNumber (p. 403)": integer,
            "timecodeSource (p. 403)": enum,
            "captionSelectors (p. 403)": {
            },
            "fileInput (p. 403)": "string",
            "inputClippings (p. 403)": [
                {
                    "startTimecode (p. 404)": "string",
                    "endTimecode (p. 404)": "string"
                }
            ],
            "psiControl (p. 404)": enum
        }
    ],
    "outputGroups (p. 409)": [
        {
            "outputs (p. 438)": [
                {
                    "extension (p. 437)": "string",
                    "videoDescription (p. 437)": {
                        "fixedAfd (p. 450)": integer,
                        "scalingBehavior (p. 451)": enum,
                        "respondToAfd (p. 451)": enum,
                        "width (p. 451)": integer,
                        "height (p. 451)": integer,
                        "aspect (p. 451)": enum,
                        "numFrame (p. 451)": integer
                    },
                    "videoSelector (p. 453)": {
                        "colorSpace (p. 453)": enum,
                        "colorSpaceUsage (p. 454)": enum,
                        "hdr10Metadata (p. 454)": {
                            "redPrimaryY (p. 392)": integer,
                            "greenPrimaryY (p. 392)": integer,
                            "whitePointX (p. 392)": integer,
                            "maxLuminance (p. 392)": integer,
                            "greenPrimaryX (p. 393)": integer,
                            "whitePointY (p. 393)": integer,
                            "redPrimaryX (p. 393)": integer,
                            "bluePrimaryX (p. 393)": integer,
                            "maxFrameAverageLightLevel (p. 393)": integer,
                            "bluePrimaryY (p. 393)": integer,
                            "maxContentLightLevel (p. 394)": integer,
                            "minLuminance (p. 394)": integer
                        }
                    }
                }
            ]
        }
    ]
}
"codecSettings (p. 451)": {
  "h265Settings (p. 450)": {
    "slices (p. 386)": integer,
    "minIInterval (p. 386)": integer,
    "parNumerator (p. 386)": integer,
    "flickerAdaptiveQuantization (p. 387)": enum,
    "gopSizeUnits (p. 387)": enum,
    "hrdBufferSize (p. 387)": integer,
    "qualityTuningLevel (p. 387)": enum,
    "maxBitrate (p. 387)": integer,
    "bitrate (p. 387)": integer,
    "spatialAdaptiveQuantization (p. 387)": enum,
    "sampleAdaptiveOffsetFilterMode (p. 388)": enum,
    "temporalIds (p. 388)": enum,
    "slowPal (p. 388)": enum,
    "tiles (p. 388)": enum,
    "codecProfile (p. 388)": enum,
    "alternateTransferFunctionSei (p. 388)": enum,
    "unregisteredSeiTimecode (p. 388)": enum,
    "framerateControl (p. 388)": enum,
    "telecine (p. 388)": enum,
    "framerateConversionAlgorithm (p. 388)": enum,
    "codecLevel (p. 389)": enum,
    "numberReferenceFrames (p. 389)": integer,
    "temporalAdaptiveQuantization (p. 389)": enum,
    "hrdBufferInitialFillPercentage (p. 389)": integer,
    "framerateNumerator (p. 389)": integer,
    "framerateDenominator (p. 390)": integer,
    "adaptiveQuantization (p. 390)": enum,
    "interlaceMode (p. 390)": enum,
    "gopSize (p. 390)": number,
    "gopBReference (p. 390)": enum,
    "parNumerator (p. 390)": integer,
    "parControl (p. 391)": enum,
    "parDenominator (p. 391)": integer,
    "framerateControl (p. 391)": enum,
    "telecine (p. 391)": enum,
    "framerateConversionAlgorithm (p. 391)": enum,
    "framerateDenominator (p. 392)": integer,
    "adapti...
"framerateControl (p. 430)": enum,
"telecine (p. 431)": enum,
"framerateConversionAlgorithm (p. 431)": enum,
"codecLevel (p. 431)": enum,
"temporalAdaptiveQuantization (p. 431)": enum,
"hrdBufferInitialFillPercentage (p. 431)": integer,
"framerateNumerator (p. 431)": integer,
"numberBFramesBetweenReferenceFrames (p. 431)": integer,
"gopClosedCadence (p. 431)": integer,
"framerateDenominator (p. 432)": integer,
"adaptiveQuantization (p. 432)": enum,
"interlaceMode (p. 432)": enum,
"gopSize (p. 432)": number,
"sceneChangeDetect (p. 432)": enum,
"parDenominator (p. 432)": integer,
"parControl (p. 432)": enum,
"syntax (p. 433)": enum,
"rateControlMode (p. 433)": enum,
},
"h264Settings (p. 450)": {
"slices (p. 377)": integer,
"minIInterval (p. 378)": integer,
"parNumerator (p. 378)": integer,
"flickerAdaptiveQuantization (p. 378)": enum,
"gopSizeUnits (p. 378)": enum,
"hrdBufferSize (p. 378)": integer,
"qualityTuningLevel (p. 378)": enum,
"maxBitrate (p. 378)": integer,
"bitrate (p. 379)": integer,
"spatialAdaptiveQuantization (p. 379)": enum,
"slowPal (p. 379)": enum,
"codecProfile (p. 379)": enum,
"unregisteredSeiTimecode (p. 379)": enum,
"softness (p. 379)": integer,
"framerateControl (p. 379)": enum,
"telecine (p. 379)": enum,
"framerateConversionAlgorithm (p. 380)": enum,
"codecLevel (p. 380)": enum,
"numberReferenceFrames (p. 380)": integer,
"temporalAdaptiveQuantization (p. 380)": enum,
"repeatPps (p. 380)": enum,
"hrdBufferInitialFillPercentage (p. 380)": integer,
"framerateNumerator (p. 380)": integer,
"numberBFramesBetweenReferenceFrames (p. 380)": integer,
"gopClosedCadence (p. 381)": integer,
"fieldEncoding (p. 381)": enum,
"entropyEncoding (p. 381)": enum,
"framerateDenominator (p. 381)": integer,
"adaptiveQuantization (p. 381)": enum,
"interlaceMode (p. 381)": enum,
"gopSize (p. 381)": number,
"gopBReference (p. 382)": enum,
"sceneChangeDetect (p. 382)": enum,
"parDenominator (p. 382)": integer,
"parControl (p. 382)": enum,
"syntax (p. 382)": enum,
"rateControlMode (p. 382)": enum,
},
"frameCaptureSettings (p. 450)": {
"framerateDenominator (p. 374)": integer,
"maxCaptures (p. 374)": integer,
"framerateNumerator (p. 374)": integer,
"quality (p. 374)": integer,
},
"afdSignaling (p. 451)": enum,
"colorMetadata (p. 451)": enum,
"timecodeInsertion (p. 451)": enum,
"width (p. 451)": integer,
"sharpness (p. 451)": integer,
"antiAlias (p. 452)": enum,
"videoPreprocessors (p. 452)": {
  "timecodeBurnin (p. 453)": {
    "prefix (p. 446)": "string",
    "fontSize (p. 447)": integer,
    "position (p. 447)": enum
  },
  "noiseReducer (p. 453)": {
    "filter (p. 435)": enum,
    "filterSettings (p. 435)": {
      "strength (p. 436)": integer
    },
    "spatialFilterSettings (p. 435)": {
      "strength (p. 436)": integer,
      "postFilterSharpenStrength (p. 436)": integer,
      "speed (p. 436)": integer
    }
  },
  "colorCorrector (p. 453)": {
    "saturation (p. 355)": integer,
    "brightness (p. 355)": integer,
    "hdr10Metadata (p. 355)": {
      "redPrimaryY (p. 392)": integer,
      "greenPrimaryY (p. 392)": integer,
      "whitePointX (p. 392)": integer,
      "maxLuminance (p. 392)": integer,
      "greenPrimaryX (p. 393)": integer,
      "whitePointY (p. 393)": integer,
      "redPrimaryX (p. 393)": integer,
      "bluePrimaryX (p. 393)": integer,
      "maxFrameAverageLightLevel (p. 393)": integer,
      "bluePrimaryY (p. 393)": integer,
      "maxContentLightLevel (p. 394)": integer,
      "minLuminance (p. 394)": integer
    },
    "contrast (p. 356)": integer,
    "hue (p. 356)": integer,
    "colorSpaceConversion (p. 356)": enum
  },
  "imageInserter (p. 453)": {
    "insertableImages (p. 402)": [ 
      { "duration (p. 405)": integer,
        "fadeOut (p. 405)": integer,
        "imageY (p. 406)": integer,
        "fadeOut (p. 406)": integer,
        "imageX (p. 406)": integer,
        "width (p. 406)": integer,
        "startTime (p. 406)": "string",
        "opacity (p. 406)": integer,
        "layer (p. 407)": integer,
        "height (p. 407)": integer,
        "imageInserterInput (p. 407)": "string"
      }
    ],
    "deinterlacer (p. 453)": {
      "mode (p. 361)": enum,
      "control (p. 361)": enum,
      "algorithm (p. 361)": enum
    }
  }
},
"position (p. 452)": {
  "width (p. 443)": integer,
  "x (p. 443)": integer,
  "y (p. 443)": integer,
  "height (p. 443)": integer
},
"dropFrameTimecode (p. 452)": enum,
"crop (p. 452)": {
  "width (p. 443)": integer,
  "x (p. 443)": integer,
  "y (p. 443)": integer,
  "height (p. 443)": integer
},
"height (p. 452)": integer
},
"audioDescriptions (p. 437)": [ 
  {
    "languageCodeControl (p. 343)": enum,
    "audioTypeControl (p. 343)": enum,
    "remixSettings (p. 344)": {
      "channelsOut (p. 443)": integer,
      "channelMapping (p. 444)": {
        "outputChannels (p. 355)": [ 
          {
            "inputChannels (p. 438)": [ integer
          ]
        }
      },
      "channelsIn (p. 444)": integer
    },
    "audioType (p. 344)": integer,
    "audioSourceName (p. 344)": "string",
    "codecSettings (p. 344)": {
      "codec (p. 342)": enum,
      "wavSettings (p. 343)": {
        "channels (p. 454)": integer,
        "bitDepth (p. 455)": integer,
        "sampleRate (p. 455)": integer
      },
      "ac3Settings (p. 343)": {
        "dynamicRangeCompressionProfile (p. 340)": enum,
        "dialnorm (p. 340)": integer,
        "codingMode (p. 340)": enum,
        "metadataControl (p. 340)": enum,
        "lfeFilter (p. 341)": enum,
        "bitrate (p. 341)": integer,
        "bitstreamMode (p. 341)": enum,
        "sampleRate (p. 341)": integer
      },
      "aacSettings (p. 343)": {
        "vbrQuality (p. 338)": enum,
        "codecProfile (p. 338)": enum,
        "codingMode (p. 338)": enum,
        "specification (p. 338)": enum,
        "bitrate (p. 338)": integer,
        "rawFormat (p. 338)": enum,
        "rateControlMode (p. 339)": enum,
        "sampleRate (p. 339)": integer,
        "audioDescriptionBroadcasterMix (p. 339)": enum
      },
      "aiffSettings (p. 343)": {
        "channels (p. 341)": integer,
        "bitDepth (p. 342)": integer,
        "sampleRate (p. 342)": integer
      }
    }
  }
]
null
"timedMetadata (p. 424)" : enum,
"scte35Source (p. 424)" : enum
},
"m2tsSettings (p. 357)" : {
"dvbTeletextPid (p. 417)" : integer,
"bitrate (p. 417)" : integer,
"segmentationTime (p. 417)" : number,
"audioPids (p. 417)" : [
  integer
],
"rateMode (p. 417)" : enum,
"ebpAudioInterval (p. 417)" : enum,
"fragmentTime (p. 418)" : number,
"audioFramesPerPesc (p. 418)" : integer,
"maxPcrInterval (p. 418)" : integer,
"scte35Pid (p. 418)" : integer,
"privateMetadataPid (p. 418)" : integer,
"pmtInterval (p. 418)" : integer,
"segmentationStyle (p. 419)" : enum,
"audioBufferModel (p. 419)" : enum,
"programNumber (p. 419)" : integer,
"dvbNitSettings (p. 419)" : {
  "networkName (p. 361)" : "string",
  "networkId (p. 362)" : integer,
  "nitInterval (p. 362)" : integer
},
"scte35Source (p. 419)" : enum,
"pmtPid (p. 419)" : integer,
"bufferModel (p. 419)" : enum,
"ebpPlacement (p. 419)" : enum,
"dvbSdtSettings (p. 419)" : {
  "sdtInterval (p. 362)" : integer,
  "serviceName (p. 362)" : "string",
  "serviceProviderName (p. 362)" : "string",
  "outputSdt (p. 362)" : enum
},
"nullPacketBitrate (p. 420)" : number,
"pcrPid (p. 420)" : integer,
"minEbpInterval (p. 420)" : integer,
"transportStreamId (p. 420)" : integer,
"pcrControl (p. 420)" : enum,
"videoPid (p. 420)" : integer,
"esRateInPes (p. 421)" : enum,
"segmentationMarkers (p. 421)" : enum,
"dvbTdtSettings (p. 421)" : {
  "tdtInterval (p. 367)" : integer
},
"patInterval (p. 421)" : integer,
"dvbSubPids (p. 421)" : [
  integer
]
},
"movSettings (p. 357)" : {
  "reference (p. 425)" : enum,
  "paddingControl (p. 425)" : enum,
  "mpeg2FourCCControl (p. 425)" : enum,
  "csigAtom (p. 425)" : enum,
  "clapAtom (p. 425)" : enum
},
"f4vSettings (p. 357)" : {
  "moovPlacement (p. 373)" : enum
}
},
"preset (p. 437)" : "string",
"outputSettings (p. 437)" : {
  "hlsSettings (p. 440)" : {
"iFrameOnlyManifest (p. 400)" : enum,
"segmentModifier (p. 400)" : "string",
"audioTrackType (p. 401)" : enum,
"audioRenditionSets (p. 401)" : "string",
"audioGroupId (p. 401)" : "string"
}

,"captionDescriptions (p. 438)" : [
  {
    "captionSelectorName (p. 352)" : "string",
    "languageDescription (p. 352)" : "string",
    "languageCode (p. 353)" : enum,
    "destinationSettings (p. 353)" : {
      "burninDestinationSettings (p. 353)" : {
        "xPosition (p. 348)" : integer,
        "backgroundColor (p. 349)" : enum,
        "teletextSpacing (p. 349)" : enum,
        "yPosition (p. 349)" : integer,
        "backgroundColorOpacity (p. 349)" : integer,
        "fontOpacity (p. 349)" : integer,
        "shadowOpacity (p. 350)" : integer,
        "fontResolution (p. 350)" : integer,
        "shadowYOffset (p. 350)" : integer,
        "outlineSize (p. 350)" : integer,
        "outlineColor (p. 350)" : enum,
        "fontSize (p. 350)" : integer,
        "shadowXOffset (p. 351)" : integer,
        "alignment (p. 351)" : enum,
        "shadowColor (p. 351)" : enum,
        "fontColor (p. 351)" : enum
      },
      "teletextDestinationSettings (p. 353)" : {
        "pageNumber (p. 446)" : "string"
      },
      "ttmlDestinationSettings (p. 353)" : {
        "stylePassthrough (p. 449)" : enum
      },
      "destinationType (p. 353)" : enum,
      "dvbSubDestinationSettings (p. 353)" : {
        "xPosition (p. 363)" : integer,
        "backgroundColor (p. 363)" : enum,
        "teletextSpacing (p. 363)" : enum,
        "yPosition (p. 363)" : integer,
        "backgroundColorOpacity (p. 363)" : integer,
        "fontOpacity (p. 363)" : integer,
        "shadowOpacity (p. 364)" : integer,
        "fontResolution (p. 364)" : integer,
        "shadowYOffset (p. 364)" : integer,
        "outlineSize (p. 364)" : integer,
        "outlineColor (p. 364)" : enum,
        "fontSize (p. 364)" : integer,
        "shadowXOffset (p. 365)" : integer,
        "alignment (p. 365)" : enum,
        "shadowColor (p. 365)" : enum,
        "fontColor (p. 365)" : enum
      },
      "sccDestinationSettings (p. 353)" : {
        "framerate (p. 445)" : enum
      }
    }
  },
  {
    "nameModifier (p. 438)" : "string"
  }
],
"outputGroupSettings (p. 438)" : {
}
"dashIsoGroupSettings (p. 439)": {
  "fragmentLength (p. 359)" : integer,
  "baseUrl (p. 359)" : "string",
  "minBufferTime (p. 359)" : integer,
  "encryption (p. 359)" : {
    "spekeKeyProvider (p. 359)" : {
      "resourceId (p. 445)" : "string",
      "systemIds (p. 445)" : [
        "string"
      ],
      "url (p. 445)" : "string"
    }
  },
  "destination (p. 359)" : "string",
  "segmentLength (p. 360)" : integer,
  "segmentControl (p. 360)" : enum,
  "hbbtvCompliance (p. 360)" : enum
},
"fileGroupSettings (p. 439)" : {
  "destination (p. 373)" : "string"
},
"msSmoothGroupSettings (p. 439)" : {
  "fragmentLength (p. 434)" : integer,
  "encryption (p. 434)" : {
    "spekeKeyProvider (p. 434)" : {
      "resourceId (p. 445)" : "string",
      "systemIds (p. 445)" : [
        "string"
      ],
      "url (p. 445)" : "string"
    }
  },
  "audioDeduplication (p. 434)" : enum,
  "manifestEncoding (p. 434)" : enum,
  "destination (p. 434)" : "string"
},
"type (p. 439)" : enum,
"hlsGroupSettings (p. 439)" : {
  "segmentsPerSubdirectory (p. 396)" : integer,
  "streamInfResolution (p. 397)" : enum,
  "timestampDeltaMilliseconds (p. 397)" : integer,
  "outputSelection (p. 397)" : enum,
  "captionLanguageMappings (p. 397)" : [
    {"languageDescription (p. 394)" : "string",
     "captionChannel (p. 395)" : integer,
     "languageCode (p. 395)" : enum
    }
  ],
  "clientCache (p. 397)" : enum,
  "codecSpecification (p. 397)" : enum,
  "destination (p. 397)" : "string",
  "segmentControl (p. 397)" : enum,
  "timedMetadataId3Frame (p. 397)" : enum,
  "timedMetadataId3Period (p. 398)" : integer,
  "captionLanguageSetting (p. 398)" : enum,
  "minSegmentLength (p. 398)" : integer,
  "directoryStructure (p. 398)" : enum,
  "programDateTime (p. 398)" : enum,
  "baseUrl (p. 398)" : "string",
  "encryption (p. 398)" : {
    "initializationVectorInManifest (p. 395)" : enum,
    "constantInitializationVector (p. 396)" : "string",
    "staticKeyProvider (p. 396)" : {
      "keyFormatVersions (p. 445)" : "string",
      "keyFormat (p. 445)" : "string"
    }
  }
"staticKeyValue (p. 446)": "string",
"url (p. 446)": "string"
},
"type (p. 396)": enum,
"spekeKeyProvider (p. 396)": {
"resourceId (p. 445)": "string",
"systemIds (p. 445)": [
"string"
],
"url (p. 445)": "string"
},
"encryptionMethod (p. 396)": enum
},
"adMarkers (p. 398)": [
enum
],
"programDateTimePeriod (p. 399)": integer,
"manifestCompression (p. 399)": enum,
"segmentLength (p. 399)": integer,
"manifestDurationFormat (p. 399)": enum
},
"name (p. 439)": "string",
"customName (p. 439)": "string"
],
"timedMetadataInsertion (p. 410)": {
"id3Insertions (p. 449)": [
{ "id3 (p. 401)": "string",
"timecode (p. 401)": "string"
}
],
"availBlanking (p. 410)": {
"availBlankingImage (p. 348)": "string"
}
],
"outputGroupDetails (p. 407)": [
{ "outputDetails (p. 439)": [
{ "durationInMs (p. 438)": integer,
"videoDetails (p. 438)": {
"heightInPx (p. 452)": integer,
"widthInPx (p. 453)": integer
}
}
]
],
"role (p. 407)": "string",
"jobTemplate (p. 408)": "string",
"timing (p. 408)": {
"finishTime (p. 449)": "string",
"submitTime (p. 449)": "string",
"startTime (p. 449)": "string"
},
"errorMessage (p. 408)": "string",
"errorCode (p. 408)": integer,
"createdAt (p. 408)": "string",
"id (p. 408)": "string",
"arn (p. 408)": "string",
"userMetadata (p. 408)": {
},
"queue (p. 409)": "string"
Example CreateJobResponse

```json
{
  "job (p. 358)": {
    "settings (p. 407)": {
      "timecodeConfig (p. 409)": {
        "timestampOffset (p. 447)": "string",
        "anchor (p. 447)": "string",
        "start (p. 448)": "string",
        "source (p. 448)": enum
      },
      "adAvailOffset (p. 409)": integer,
      "nielsenConfiguration (p. 409)": {
        "distributorId (p. 435)": "string",
        "breakoutCode (p. 435)": integer
      },
      "inputs (p. 409)": [
        {
          "audioSelectors (p. 402)": {
          },
          "audioSelectorGroups (p. 402)": {
          },
          "videoSelector (p. 402)": {
            "colorSpace (p. 453)": enum,
            "hdr10Metadata (p. 454)": {
              "redPrimaryY (p. 392)": integer,
              "greenPrimaryY (p. 392)": integer,
              "whitePointX (p. 392)": integer,
              "maxLuminance (p. 392)": integer,
              "greenPrimaryX (p. 393)": integer,
              "whitePointY (p. 393)": integer,
              "redPrimaryX (p. 393)": integer,
              "bluePrimaryX (p. 393)": integer,
              "maxFrameAverageLightLevel (p. 393)": integer,
              "bluePrimaryY (p. 393)": integer,
              "maxContentLightLevel (p. 394)": integer,
              "minLuminance (p. 394)": integer
            },
            "programNumber (p. 454)": integer,
            "pid (p. 454)": integer,
            "colorSpaceUsage (p. 454)": enum
          },
          "denoiseFilter (p. 402)": enum,
          "filterEnable (p. 402)": enum,
          "deblockFilter (p. 402)": enum,
          "filterStrength (p. 402)": integer,
          "programNumber (p. 403)": integer,
          "timecodeSource (p. 403)": enum,
          "captionSelectors (p. 403)": {
          },
          "fileInput (p. 403)": "string",
          "inputClippings (p. 403)": [
            {
              "startTimecode (p. 404)": "string",
              "endTimecode (p. 404)": "string"
            }
          ],
          "psiControl (p. 404)": enum
        }
      }
    }
  }
}
```
Schemas

"outputGroups (p. 409)": [
  "outputs (p. 438)": [
    "extension (p. 437)": "string",
    "videoDescription (p. 437)": {
      "fixedAfd (p. 450)": integer,
      "scalingBehavior (p. 451)": enum,
      "respondToAfd (p. 451)": enum,
      "codecSettings (p. 451)": {
        "h265Settings (p. 450)": {
          "slices (p. 386)": integer,
          "minInterval (p. 386)": integer,
          "parNumerator (p. 386)": integer,
          "flickerAdaptiveQuantization (p. 387)": enum,
          "gopSizeUnits (p. 387)": enum,
          "hrdBufferSize (p. 387)": integer,
          "qualityTuningLevel (p. 387)": enum,
          "maxBitrate (p. 387)": integer,
          "bitrate (p. 387)": integer,
          "spatialAdaptiveQuantization (p. 387)": enum,
          "unregisteredSeiTimecode (p. 388)": enum,
          "temporalIds (p. 388)": enum,
          "slowPal (p. 388)": enum,
          "tiles (p. 388)": enum,
          "codecProfile (p. 388)": enum,
          "alternateTransferFunctionSei (p. 388)": enum,
          "framerateControl (p. 388)": enum,
          "telecine (p. 388)": enum,
          "framerateConversionAlgorithm (p. 388)": enum,
          "codecLevel (p. 389)": enum,
          "numberReferenceFrames (p. 389)": integer,
          "temporalAdaptiveQuantization (p. 389)": enum,
          "hrdBuffInitialFillPercentage (p. 389)": integer,
          "framerateNumerator (p. 389)": integer,
          "framerateDenominator (p. 390)": integer,
          "adaptiveQuantization (p. 390)": enum,
          "gopSize (p. 390)": number,
          "gopReference (p. 390)": enum,
          "sceneChangeDetect (p. 390)": enum,
          "parDenominator (p. 390)": integer,
          "rateControlMode (p. 391)": enum
        }
      }
    }},
    "proresSettings (p. 450)": {
      "slowPal (p. 441)": enum,
      "framerateControl (p. 441)": enum,
      "telecine (p. 441)": enum,
      "framerateDenominator (p. 441)": integer,
      "framerateConversionAlgorithm (p. 441)": enum,
      "interlaceMode (p. 442)": enum,
      "codecProfile (p. 442)": enum,
      "parNumerator (p. 442)": integer,
      "parControl (p. 442)": enum,
      "parDenominator (p. 442)": integer,
      "framerateNumerator (p. 442)": integer
    }
  },
  "mpeg2Settings (p. 450)": {
    "minInterval (p. 429)": integer,
    "parNumerator (p. 429)": integer,
"gopSizeUnits (p. 429)": enum,
"hrdBufferSize (p. 429)": integer,
"qualityTuningLevel (p. 429)": enum,
"maxBitrate (p. 430)": integer,
"bitrate (p. 430)": integer,
"spatialAdaptiveQuantization (p. 430)": enum,
"slowPal (p. 430)": enum,
"codecProfile (p. 430)": enum,
"intraDcPrecision (p. 430)": enum,
"softness (p. 430)": integer,
"framerateControl (p. 430)": enum,
"telecine (p. 431)": enum,
"framerateConversionAlgorithm (p. 431)": enum,
"codecLevel (p. 431)": enum,
"temporalAdaptiveQuantization (p. 431)": enum,
"hrdBufferInitialFillPercentage (p. 431)": integer,
"framerateNumerator (p. 431)": integer,
"framerateDenominator (p. 432)": integer,
"adaptiveQuantization (p. 432)": enum,
"interlaceMode (p. 432)": enum,
"gopSize (p. 432)": number,
"sceneChangeDetect (p. 432)": enum,
"parDenominator (p. 432)": integer,
"parControl (p. 432)": enum,
"syntax (p. 433)": enum,
"rateControlMode (p. 433)": enum,
"h264Settings (p. 450)": {
  "slices (p. 377)": integer,
  "minIInterval (p. 378)": integer,
  "parNumerator (p. 378)": integer,
  "flickerAdaptiveQuantization (p. 378)": enum,
  "gopSizeUnits (p. 378)": enum,
  "hrdBufferSize (p. 378)": integer,
  "qualityTuningLevel (p. 378)": enum,
  "maxBitrate (p. 378)": integer,
  "bitrate (p. 379)": integer,
  "spatialAdaptiveQuantization (p. 379)": enum,
  "slowPal (p. 379)": enum,
  "codecProfile (p. 379)": enum,
  "unregisteredSeiTimecode (p. 379)": enum,
  "softness (p. 379)": integer,
  "framerateControl (p. 379)": enum,
  "telecine (p. 379)": enum,
  "framerateConversionAlgorithm (p. 380)": enum,
  "codecLevel (p. 380)": enum,
  "numberReferenceFrames (p. 380)": integer,
  "temporalAdaptiveQuantization (p. 380)": enum,
  "repeatPps (p. 380)": enum,
  "hrdBufferInitialFillPercentage (p. 380)": integer,
  "framerateNumerator (p. 380)": integer,
  "framerateDenominator (p. 380)": integer,
  "gopClosedCadence (p. 381)": integer,
  "fieldEncoding (p. 381)": enum,
  "entropyEncoding (p. 381)": enum,
  "framerateDenominator (p. 381)": integer,
  "adaptiveQuantization (p. 381)": enum,
  "interlaceMode (p. 381)": enum,
  "gopSize (p. 381)": number,
  "gopBReference (p. 382)": enum,
  "sceneChangeDetect (p. 382)": enum,
  "parDenominator (p. 382)": integer,
  "parControl (p. 382)": enum,
  "syntax (p. 382)": enum,
"rateControlMode (p. 382)": enum
},
"frameCaptureSettings (p. 450)": {
  "framerateDenominator (p. 374)": integer,
  "maxCaptures (p. 374)": integer,
  "framerateNumerator (p. 374)": integer,
  "quality (p. 374)": integer
}
},
"afdSignaling (p. 451)": enum,
"colorMetadata (p. 451)": enum,
"timecodeInsertion (p. 451)": enum,
"width (p. 451)": integer,
"sharpness (p. 451)": integer,
"antiAlias (p. 452)": enum,
"videoPreprocessors (p. 452)": {
  "timecodeBurnin (p. 453)": {
    "prefix (p. 446)": "string",
    "fontSize (p. 447)": integer,
    "position (p. 447)": enum
  },
  "noiseReducer (p. 453)": {
    "filter (p. 435)": enum,
    "filterSettings (p. 435)": {
      "strength (p. 436)": integer
    },
    "spatialFilterSettings (p. 435)": {
      "strength (p. 436)": integer,
      "postFilterSharpenStrength (p. 436)": integer,
      "speed (p. 436)": integer
    }
  },
  "colorCorrector (p. 453)": {
    "saturation (p. 355)": integer,
    "brightness (p. 355)": integer,
    "hdr10Metadata (p. 355)": {
      "redPrimaryY (p. 392)": integer,
      "greenPrimaryY (p. 392)": integer,
      "whitePointX (p. 392)": integer,
      "maxLuminance (p. 392)": integer,
      "greenPrimaryX (p. 393)": integer,
      "whitePointY (p. 393)": integer,
      "redPrimaryX (p. 393)": integer,
      "bluePrimaryX (p. 393)": integer,
      "maxFrameAverageLightLevel (p. 393)": integer,
      "bluePrimaryY (p. 393)": integer,
      "maxContentLightLevel (p. 394)": integer,
      "minLuminance (p. 394)": integer
    },
    "contrast (p. 356)": integer,
    "hue (p. 356)": integer,
    "colorSpaceConversion (p. 356)": enum
  },
  "imageInserter (p. 453)": {
    "insertableImages (p. 402)": [
      {
        "duration (p. 405)": integer,
        "fadeOut (p. 405)": integer,
        "imageY (p. 406)": integer,
        "fadeIn (p. 406)": integer,
        "imageX (p. 406)": integer,
        "width (p. 406)": integer,
        "startTime (p. 406)": "string",
        "opacity (p. 406)": integer,
        "layer (p. 407)": integer,
        "height (p. 407)": integer,
        "width (p. 408)": integer,
        "minValue (p. 409)": integer,
        "maxValue (p. 409)": integer,
        "primary (p. 409)": integer,
        "secondary (p. 409)": integer,
        "offset (p. 410)": integer,
        "primaryOffset (p. 410)": integer,
        "secondaryOffset (p. 410)": integer,
        "position (p. 410)": integer,
        "opacity (p. 411)": integer,
        "layer (p. 411)": integer,
        "width (p. 412)": integer,
        "height (p. 412)": integer,
        "imageX (p. 413)": integer,
        "imageY (p. 413)": integer
      }
    ]
  }
}
"imageInserterInput (p. 407)": "string"
],
},
"deinterlacer (p. 453)": {
"mode (p. 361)": enum,
"control (p. 361)": enum,
"algorithm (p. 361)": enum
},
"position (p. 452)": {
"width (p. 443)": integer,
"x (p. 443)": integer,
"y (p. 443)": integer,
"height (p. 443)": integer
},
"dropFrameTimecode (p. 452)": enum,
"crop (p. 452)": {
"width (p. 443)": integer,
"x (p. 443)": integer,
"y (p. 443)": integer,
"height (p. 443)": integer
},
"height (p. 452)": integer
],
"audioDescriptions (p. 437)": [
{
"languageCodeControl (p. 343)": enum,
"audioTypeControl (p. 343)": enum,
"remixSettings (p. 344)": {
"channelOut (p. 443)": integer,
"channelMapping (p. 444)": {
"outputChannels (p. 355)": [
"inputChannels (p. 438)": [integer
]
}
},
"channelsIn (p. 444)": integer
},
"audioType (p. 344)": integer,
"audioSourceName (p. 344)": "string",
"codecSettings (p. 344)": {
"codec (p. 342)": enum,
"wavSettings (p. 343)": {
"channels (p. 454)": integer,
"bitDepth (p. 455)": integer,
"sampleRate (p. 455)": integer
},
"ac3Settings (p. 343)": {
"dynamicRangeCompressionProfile (p. 340)": enum,
"dialnorm (p. 340)": integer,
"codingMode (p. 340)": enum,
"metadataControl (p. 340)": enum,
"lfeFilter (p. 341)": enum,
"bitrate (p. 341)": integer,
"bitstreamMode (p. 341)": enum,
"sampleRate (p. 341)": integer
},
"aacSettings (p. 343)": {
"vbrQuality (p. 338)": enum,
"codecProfile (p. 338)": enum,
"codingMode (p. 338)": enum,
"specification (p. 338)": enum,
"bitrate (p. 338)": integer,
"rawFormat (p. 338)": enum,
"rateControlMode (p. 339)": enum,
"sampleRate (p. 339)": integer,
"audioDescriptionBroadcasterMix (p. 339)": enum
},
"aiffSettings (p. 343)": {
  "channels (p. 341)": integer,
  "bitDepth (p. 342)": integer,
  "sampleRate (p. 342)": integer
},
"eac3Settings (p. 343)": {
  "dialnorm (p. 369)": integer,
  "passthroughControl (p. 369)": enum,
  "metadataControl (p. 369)": enum,
  "bitrate (p. 369)": integer,
  "dynamicRangeCompressionRf (p. 369)": enum,
  "sampleRate (p. 369)": integer,
  "lRtSurroundMixLevel (p. 369)": number,
  "surroundExMode (p. 370)": enum,
  "dynamicRangeCompressionLine (p. 370)": enum,
  "lifeControl (p. 370)": enum,
  "codingMode (p. 370)": enum,
  "surroundMode (p. 370)": enum,
  "attenuationControl (p. 370)": enum,
  "lifeFilter (p. 370)": enum,
  "phaseControl (p. 370)": enum,
  "lRtCenterMixLevel (p. 370)": number,
  "dcFilter (p. 371)": enum,
  "stereoDownmix (p. 371)": enum,
  "bitstreamMode (p. 371)": enum,
  "loRoSurroundMixLevel (p. 371)": number,
  "loRoCenterMixLevel (p. 371)": number
},
"mp2Settings (p. 343)": {
  "channels (p. 425)": integer,
  "bitrate (p. 425)": integer,
  "sampleRate (p. 426)": integer
},
"languageCode (p. 344)": enum,
"streamName (p. 344)": "string",
"audioNormalizationSettings (p. 345)": {
  "targetLkfs (p. 345)": number,
  "algorithmControl (p. 346)": enum,
  "peakCalculation (p. 346)": enum,
  "loudnessLogging (p. 346)": enum,
  "correctionGateLevel (p. 346)": integer,
  "algorithm (p. 346)": enum
}
],
"containerSettings (p. 437)": {
  "container (p. 357)": enum,
  "mp4Settings (p. 357)": {
    "mp4MajorBrand (p. 426)": "string",
    "moovPlacement (p. 426)": enum,
    "cslgAtom (p. 426)": enum,
    "freeSpaceBox (p. 427)": enum
  },
  "m3u8Settings (p. 357)": {
    "pmtPid (p. 422)": integer,
    "pcrPid (p. 422)": integer,
    "audioPids (p. 422)": [integer
    ]
  }
};
"audioFramesPerPes (p. 422)": integer,
"scte35Pid (p. 422)": integer,
"transportStreamId (p. 422)": integer,
"videoPid (p. 423)": integer,
"pcrControl (p. 423)": enum,
"privateMetadataPid (p. 423)": integer,
"pmtInterval (p. 423)": integer,
"patInterval (p. 423)": integer,
"programNumber (p. 423)": integer,
"timedMetadataPid (p. 424)": integer,
"timedMetadata (p. 424)": enum,
"scte35Source (p. 424)": enum
},
"m2tsSettings (p. 357)": {
  "dvbTeletextPid (p. 417)": integer,
  "bitrate (p. 417)": integer,
  "segmentationTime (p. 417)": number,
  "audioPids (p. 417)": [ integer ]
},
"rateMode (p. 417)": enum,
"ebpAudioInterval (p. 417)": enum,
"fragmentTime (p. 418)": number,
"audioFramesPerPes (p. 418)": integer,
"maxPcrInterval (p. 418)": integer,
"scte35Pid (p. 418)": integer,
"privateMetadataPid (p. 418)": integer,
"pmtInterval (p. 418)": integer,
"segmentationStyle (p. 419)": enum,
"audioBufferModel (p. 419)": enum,
"programNumber (p. 419)": integer,
"dvbNitSettings (p. 419)": {
  "networkName (p. 361)": "string",
  "networkId (p. 362)": integer,
  "nitInterval (p. 362)": integer
},
"scte35Source (p. 419)": enum,
"pmtPid (p. 419)": integer,
"bufferModel (p. 419)": enum,
"ebpPlacement (p. 419)": enum,
"dvbSdtSettings (p. 419)": {
  "sdtInterval (p. 362)": integer,
  "serviceName (p. 362)": "string",
  "serviceProviderName (p. 362)": "string",
  "outputSdt (p. 362)": enum
},
"nullPacketBitrate (p. 420)": number,
"pcrPId (p. 420)": integer,
"minEbpInterval (p. 420)": integer,
"transportStreamId (p. 420)": integer,
"pcrControl (p. 420)": enum,
"videoPid (p. 420)": integer,
"esRateInPes (p. 421)": enum,
"segmentationMarkers (p. 421)": enum,
"dvbTdtSettings (p. 421)": {
  "tdtInterval (p. 367)": integer
},
"patInterval (p. 421)": integer,
"dvbSubPids (p. 421)": [ integer ]
},
"movSettings (p. 357)": {
  "reference (p. 425)": enum,
  "paddingControl (p. 425)": enum,
  "mpeg2FourCCControl (p. 425)": enum,
"cslgAtom (p. 425)": enum,
"clapAtom (p. 425)": enum
},
"f4vSettings (p. 357)": {
  "moovPlacement (p. 373)": enum
},
"preset (p. 437)": "string",
"outputSettings (p. 437)": {
  "hlsSettings (p. 440)": {
    "iFrameOnlyManifest (p. 400)": enum,
    "segmentModifier (p. 400)": "string",
    "audioTrackType (p. 401)": enum,
    "audioRenditionSets (p. 401)": "string",
    "audioGroupId (p. 401)": "string"
  }
},
"captionDescriptions (p. 438)": [
  {
    "captionSelectorName (p. 352)": "string",
    "languageDescription (p. 352)": "string",
    "languageCode (p. 353)": enum,
    "destinationSettings (p. 353)": {
      "burninDestinationSettings (p. 353)": {
        "xPosition (p. 348)": integer,
        "backgroundColor (p. 349)": enum,
        "teletextSpacing (p. 349)": enum,
        "yPosition (p. 349)": integer,
        "backgroundOpacity (p. 349)": integer,
        "fontOpacity (p. 349)": integer,
        "shadowOpacity (p. 350)": integer,
        "fontResolution (p. 350)": integer,
        "shadowYOffset (p. 350)": integer,
        "outlineSize (p. 350)": integer,
        "outlineColor (p. 350)": enum,
        "fontSize (p. 350)": integer,
        "shadowXOffset (p. 351)": integer,
        "alignment (p. 351)": enum,
        "shadowColor (p. 351)": enum,
        "fontColor (p. 351)": enum
      },
      "teletextDestinationSettings (p. 353)": {
        "pageNumber (p. 446)": "string"
      },
      "ttmlDestinationSettings (p. 353)": {
        "stylePassthrough (p. 449)": enum
      },
      "destinationType (p. 353)": enum,
      "dvbSubDestinationSettings (p. 353)": {
        "xPosition (p. 363)": integer,
        "backgroundColor (p. 363)": enum,
        "teletextSpacing (p. 363)": enum,
        "yPosition (p. 363)": integer,
        "backgroundOpacity (p. 363)": integer,
        "fontOpacity (p. 363)": integer,
        "shadowOpacity (p. 364)": integer,
        "fontResolution (p. 364)": integer,
        "shadowYOffset (p. 364)": integer,
        "outlineSize (p. 364)": integer,
        "outlineColor (p. 364)": enum,
        "fontSize (p. 364)": integer,
        "shadowXOffset (p. 365)": integer,
        "alignment (p. 365)": enum,
        "shadowColor (p. 365)": enum,
        "fontColor (p. 365)": enum
      }
    }
  }
]
"sccDestinationSettings (p. 353)": {
  "framerate (p. 445)": enum
}
],
"nameModifier (p. 438)": "string"
},
"outputGroupSettings (p. 438)": {
  "dashIsoGroupSettings (p. 439)": {
    "fragmentLength (p. 359)": integer,
    "baseUrl (p. 359)": "string",
    "minBufferTime (p. 359)": integer,
    "encryption (p. 359)": {
      "spekeKeyProvider (p. 359)": {
        "resourceId (p. 445)": "string",
        "systemIds (p. 445)": ["string"]
      },
      "url (p. 445)": "string"
    },
    "destination (p. 359)": "string",
    "segmentLength (p. 360)": integer,
    "segmentControl (p. 360)": enum,
    "hbbtvCompliance (p. 360)": enum
  },
  "fileGroupSettings (p. 439)": {
    "destination (p. 373)": "string"
  },
  "msSmoothGroupSettings (p. 439)": {
    "fragmentLength (p. 434)": integer,
    "encryption (p. 434)": {
      "spekeKeyProvider (p. 434)": {
        "resourceId (p. 445)": "string",
        "systemIds (p. 445)": ["string"]
      },
      "url (p. 445)": "string"
    },
    "audioDeduplication (p. 434)": enum,
    "manifestEncoding (p. 434)": enum,
    "destination (p. 434)": "string"
  },
  "type (p. 439)": enum,
  "hlsGroupSettings (p. 439)": {
    "segmentsPerSubdirectory (p. 396)": integer,
    "streamInfResolution (p. 397)": enum,
    "timestampDeltaMilliseconds (p. 397)": integer,
    "outputSelection (p. 397)": enum,
    "captionLanguageMappings (p. 397)": [
      {
        "languageDescription (p. 394)": "string",
        "captionChannel (p. 395)": integer,
        "languageCode (p. 395)": enum
      }
    ],
    "clientCache (p. 397)": enum,
    "codecSpecification (p. 397)": enum,
    "destination (p. 397)": "string",
    "segmentControl (p. 397)": enum,
    "timedMetadataId3Frame (p. 397)": enum,
    "timedMetadataId3Period (p. 398)": integer,
    "captionLanguageSetting (p. 398)": enum,
  },
  "sccDestinationSettings (p. 353)": {
    "framerate (p. 445)": enum
  }
}
"minSegmentLength (p. 398)" : integer,
"directoryStructure (p. 398)" : enum,
"programDateTime (p. 398)" : enum,
"baseUrl (p. 398)" : "string",
"encryption (p. 398)" : {
  "initializationVectorInManifest (p. 395)" : enum,
  "constantInitializationVector (p. 396)" : "string",
  "staticKeyProvider (p. 396)" : {
    "keyFormatVersions (p. 445)" : "string",
    "keyFormat (p. 445)" : "string",
    "staticKeyValue (p. 446)" : "string",
    "url (p. 446)" : "string"
  },
  "type (p. 396)" : enum,
  "spekeKeyProvider (p. 396)" : {
    "resourceId (p. 445)" : "string",
    "systemIds (p. 445)" : [ "string" ],
    "url (p. 445)" : "string"
  },
  "encryptionMethod (p. 396)" : enum
},
"adMarkers (p. 398)" : [ enum ],
"programDateTimePeriod (p. 399)" : integer,
"manifestCompression (p. 399)" : enum,
"segmentLength (p. 399)" : integer,
"manifestDurationFormat (p. 399)" : enum
},
"name (p. 439)" : "string",
"customName (p. 439)" : "string"
}],
"timedMetadataInsertion (p. 410)" : {
  "id3Insertions (p. 449)" : [ {
    "id3 (p. 401)" : "string",
    "timecode (p. 401)" : "string"
  } ],
  "availBlanking (p. 410)" : {
    "availBlankingImage (p. 348)" : "string"
  }
},
"outputGroupDetails (p. 407)" : [ {
    "outputDetails (p. 439)" : [ {
      "durationInMs (p. 438)" : integer,
      "videoDetails (p. 438)" : {
        "heightInPx (p. 452)" : integer,
        "widthInPx (p. 453)" : integer
      }
    } ]
  },
  "role (p. 407)" : "string",
  "jobTemplate (p. 408)" : "string",
  "timing (p. 408)" : {
    "finishTime (p. 449)" : "string",
    "submitTime (p. 449)" : "string",
  }
}
Properties

**AacAudioDescriptionBroadcasterMix (Enum)**

Choose BROADCASTER_MIXED_AD when the input contains pre-mixed main audio + audio description (AD) as a stereo pair. The value for AudioType will be set to 3, which signals to downstream systems that this stream contains "broadcaster mixed AD". Note that the input received by the encoder must contain pre-mixed audio; the encoder does not perform the mixing. When you choose BROADCASTER_MIXED_AD, the encoder ignores any values you provide in AudioType and FollowInputAudioType. Choose NORMAL when the input does not contain pre-mixed audio + audio description (AD). In this case, the encoder will use any values you provide for AudioType and FollowInputAudioType.

- BROADCASTER_MIXED_AD
- NORMAL

**AacCodecProfile (Enum)**

AAC Profile.

- LC
- HEV1
- HEV2

**AacCodingMode (Enum)**

Mono (Audio Description), Mono, Stereo, or 5.1 channel layout. Valid values depend on rate control mode and profile. "1.0 - Audio Description (Receiver Mix)" setting receives a stereo description plus control track and emits a mono AAC encode of the description track, with control data emitted in the PES header as per ETSI TS 101 154 Annex E.

- AD_RECEIVER_MIX
- CODING_MODE_1_0
- CODING_MODE_1_1
- CODING_MODE_2_0
CODING_MODE_5_1

**AacRateControlMode (Enum)**

Rate Control Mode.

- CBR
- VBR

**AacRawFormat (Enum)**

Enables LATM/LOAS AAC output. Note that if you use LATM/LOAS AAC in an output, you must choose "No container" for the output container.

- LATM_LOAS
- NONE

**AacSettings**

**vbrQuality**

- Type: `AacVbrQuality (p. 339)`
- Required: False

**codecProfile**

- Type: `AacCodecProfile (p. 337)`
- Required: False

**codingMode**

- Type: `AacCodingMode (p. 337)`
- Required: True

**specification**

- Type: `AacSpecification (p. 339)`
- Required: False

**bitrate**

Average bitrate in bits/second. Valid values depend on rate control mode and profile.

- Type: integer
- Required: False
- Minimum: 6000
- Maximum: 1024000

**rawFormat**

- Type: `AacRawFormat (p. 338)`
- Required: False
rateControlMode

Type: AacRateControlMode (p. 338)
Required: False

datasampleRate

Sample rate in Hz. Valid values depend on rate control mode and profile.

Type: integer
Required: True
Minimum: 8000
Maximum: 96000

audioDescriptionBroadcasterMix

Type: AacAudioDescriptionBroadcasterMix (p. 337)
Required: False

AacSpecification (Enum)

Use MPEG-2 AAC instead of MPEG-4 AAC audio for raw or MPEG-2 Transport Stream containers.

MPEG2
MPEG4

AacVbrQuality (Enum)

VBR quality level. Only used if the rate control mode (AacRateControlMode) is VBR.

LOW
MEDIUM_LOW
MEDIUM_HIGH
HIGH

Ac3BitstreamMode (Enum)

Specifies the "Bitstream Mode" (bsmod) for the emitted AC-3 stream. See ATSC A/52-2012 for background on these values.

COMPLETE_MAIN
COMMENTARY
DIALOGUE
EMERGENCY
HEARING_IMPAIRED
MUSIC_AND_EFFECTS
VISUALLY_IMPAIRED
VOICE_OVER

Ac3CodingMode (Enum)

Dolby Digital coding mode. Determines number of channels.

CODING_MODE_1_0
CODING_MODE_1_1
CODING_MODE_2_0
CODING_MODE_3_2_LFE

**Ac3DynamicRangeCompressionProfile (Enum)**
If set to FILM_STANDARD, adds dynamic range compression signaling to the output bitstream as defined in the Dolby Digital specification.

FILM_STANDARD
NONE

**Ac3LfeFilter (Enum)**
Applies a 120Hz lowpass filter to the LFE channel prior to encoding. Only valid with 3_2_LFE coding mode.

ENABLED
DISABLED

**Ac3MetadataControl (Enum)**
When set to FOLLOW_INPUT, encoder metadata will be sourced from the DD, DD+, or DolbyE decoder that supplied this audio data. If audio was not supplied from one of these streams, then the static metadata settings will be used.

FOLLOW_INPUT
USE_CONFIGURED

**Ac3Settings**

**dynamicRangeCompressionProfile**

Type: Ac3DynamicRangeCompressionProfile (p. 340)
Required: False

**dialnorm**
Sets the dialnorm for the output. If blank and input audio is Dolby Digital, dialnorm will be passed through.

Type: integer
Required: False
Minimum: 1
Maximum: 31

**codingMode**

Type: Ac3CodingMode (p. 339)
Required: False

**metadataControl**

Type: Ac3MetadataControl (p. 340)
Required: False
**Properties**

**lfeFilter**

*Type: Ac3LfeFilter (p. 340)*  
*Required: False*

**bitrate**

Average bitrate in bits/second. Valid bitrates depend on the coding mode.

*Type: integer*  
*Required: False*  
*Minimum: 64000*  
*Maximum: 640000*

**bitstreamMode**

*Type: Ac3BitstreamMode (p. 339)*  
*Required: False*

**sampleRate**

Sample rate in hz. Sample rate is always 48000.

*Type: integer*  
*Required: False*  
*Minimum: 48000*  
*Maximum: 48000*

**AfdSignaling (Enum)**

This setting only applies to H.264 and MPEG2 outputs. Use Insert AFD signaling (AfdSignaling) to whether there are AFD values in the output video data and what those values are.

* Choose None to remove all AFD values from this output.
* Choose Fixed to ignore input AFD values and instead encode the value specified in the job.
* Choose Auto to calculate output AFD values based on the input AFD scaler data.

NONE  
AUTO  
FIXED

**AiffSettings**

**channels**

Set Channels to specify the number of channels in this output audio track. Choosing Mono in the console will give you 1 output channel; choosing Stereo will give you 2. In the API, valid values are 1 and 2.

*Type: integer*  
*Required: False*  
*Minimum: 1*  
*Maximum: 2*
bitDepth

Specify Bit depth (BitDepth), in bits per sample, to choose the encoding quality for this audio track.

Type: integer
Required: False
Minimum: 16
Maximum: 24

sampleRate

Sample rate in hertz.

Type: integer
Required: False
Minimum: 8000
Maximum: 192000

AncillarySourceSettings

sourceAncillaryChannelNumber

Specifies the 608 channel number in the ancillary data track from which to extract captions. Unused for passthrough.

Type: integer
Required: False
Minimum: 1
Maximum: 4

AntiAlias (Enum)

Enable Anti-alias (AntiAlias) to enhance sharp edges in video output when your input resolution is much larger than your output resolution. Default is enabled.

DISABLED
ENABLED

AudioCodec (Enum)

Type of Audio codec.

AAC
MP2
WAV
AIFF
AC3
EAC3
PASSTHROUGH

AudioCodecSettings

codec

Type: AudioCodec (p. 342)
Required: True

**wavSettings**

Type: WavSettings (p. 454)  
Required: False

**ac3Settings**

Type: Ac3Settings (p. 340)  
Required: False

**aacSettings**

Type: AacSettings (p. 338)  
Required: False

**aiffSettings**

Type: AiffSettings (p. 341)  
Required: False

**eac3Settings**

Type: Eac3Settings (p. 369)  
Required: False

**mp2Settings**

Type: Mp2Settings (p. 425)  
Required: False

**AudioDefaultSelection (Enum)**

When an "Audio Description":#audio_description specifies an AudioSelector or AudioSelectorGroup for which no matching source is found in the input, then the audio selector marked as DEFAULT will be used. If none are marked as default, silence will be inserted for the duration of the input.

- DEFAULT
- NOT_DEFAULT

**AudioDescription**

**languageCodeControl**

Type: AudioLanguageCodeControl (p. 345)  
Required: False

**audioTypeControl**

Type: AudioTypeControl (p. 348)  
Required: False
**remixSettings**

Advanced audio remixing settings.

*Type: RemixSettings (p. 443)*

*Required: False*

**audioType**

Applies only if Follow Input Audio Type is unchecked (false). A number between 0 and 255. The following are defined in ISO-IEC 13818-1: 0 = Undefined, 1 = Clean Effects, 2 = Hearing Impaired, 3 = Visually Impaired Commentary, 4-255 = Reserved.

*Type: integer*

*Required: False*

*Minimum: 0*

*Maximum: 255*

**audioSourceName**

Specifies which audio data to use from each input. In the simplest case, specify an "Audio Selector":#inputs-audio_selector by name based on its order within each input. For example if you specify "Audio Selector 3", then the third audio selector will be used from each input. If an input does not have an "Audio Selector 3", then the audio selector marked as "default" in that input will be used. If there is no audio selector marked as "default", silence will be inserted for the duration of that input. Alternatively, an "Audio Selector Group":#inputs-audio_selector_group name may be specified, with similar default/silence behavior. If no audio_source_name is specified, then "Audio Selector 1" will be chosen automatically.

*Type: string*

*Required: False*

**codecSettings**

*Type: AudioCodecSettings (p. 342)*

*Required: True*

**languageCode**

Indicates the language of the audio output track. The ISO 639 language specified in the 'Language Code' drop down will be used when 'Follow Input Language Code' is not selected or when 'Follow Input Language Code' is selected but there is no ISO 639 language code specified by the input.

*Type: LanguageCode (p. 410)*

*Required: False*

**streamName**

Used for Microsoft Smooth Streaming and Apple HLS outputs. Indicates the name displayed by the player (eg. English, or Director Commentary). Alphanumeric characters, spaces, and underscore are legal.

*Type: string*

*Required: False*

*Pattern: ^[^\w\s]*$*
**audioNormalizationSettings**

Advanced audio normalization settings.

_Type: AudioNormalizationSettings (p. 345)  
Required: False

**AudioLanguageCodeControl (Enum)**

Choosing FOLLOW_INPUT will cause the ISO 639 language code of the output to follow the ISO 639 language code of the input. The language specified for languageCode' will be used when USE_CONFIGURED is selected or when FOLLOW_INPUT is selected but there is no ISO 639 language code specified by the input.

FOLLOW_INPUT  
USE_CONFIGURED

**AudioNormalizationAlgorithm (Enum)**

Audio normalization algorithm to use. 1770-1 conforms to the CALM Act specification, 1770-2 conforms to the EBU R-128 specification.

ITU_BS_1770_1  
ITU_BS_1770_2

**AudioNormalizationAlgorithmControl (Enum)**

When enabled the output audio is corrected using the chosen algorithm. If disabled, the audio will be measured but not adjusted.

CORRECT_AUDIO  
MEASURE_ONLY

**AudioNormalizationLoudnessLogging (Enum)**

If set to LOG, log each output's audio track loudness to a CSV file.

LOG  
DONT_LOG

**AudioNormalizationPeakCalculation (Enum)**

If set to TRUE_PEAK, calculate and log the TruePeak for each output's audio track loudness.

TRUE_PEAK  
NONE

**AudioNormalizationSettings**

targetLkfs

Target LKFS(loudness) to adjust volume to. If no value is entered, a default value will be used according to the chosen algorithm. The CALM Act (1770-1) recommends a target of -24 LKFS. The EBU R-128 specification (1770-2) recommends a target of -23 LKFS.
Properties

- **Type**: number
  - **Required**: False
  - **Format**: float
  - **Minimum**: -59.0
  - **Maximum**: 0.0

### algorithmControl
- **Type**: AudioNormalizationAlgorithmControl (p. 345)
- **Required**: False

### peakCalculation
- **Type**: AudioNormalizationPeakCalculation (p. 345)
- **Required**: False

### loudnessLogging
- **Type**: AudioNormalizationLoudnessLogging (p. 345)
- **Required**: False

### correctionGateLevel
Content measuring above this level will be corrected to the target level. Content measuring below this level will not be corrected. Gating only applies when not using `real_time_correction`.

- **Type**: integer
  - **Required**: False
  - **Minimum**: -70
  - **Maximum**: 0

### algorithm
- **Type**: AudioNormalizationAlgorithm (p. 345)
- **Required**: False

### AudioSelector

#### remixSettings
Advanced audio remixing settings.

- **Type**: RemixSettings (p. 443)
  - **Required**: False

#### programSelection
Applies only when input streams contain Dolby E. Enter the program ID (according to the metadata in the audio) of the Dolby E program to extract from the specified track. One program extracted per audio selector. To select multiple programs, create multiple selectors with the same Track and different Program numbers. "All channels" means to ignore the program IDs and include all the channels in this selector; useful if metadata is known to be incorrect.
Properties

offset

Specifies a time delta in milliseconds to offset the audio from the input video.

Type: integer
Required: False
Minimum: 0
Maximum: 8

defaultSelection

Type: AudioDefaultSelection (p. 343)
Required: False

selectorType

Type: AudioSelectorType (p. 348)
Required: False

languageCode

Selects a specific language code from within an audio source.

Type: LanguageCode (p. 410)
Required: False

pids

Selects a specific PID from within an audio source (e.g. 257 selects PID 0x101).

Type: Array of type integer
Required: False

externalAudioFileInput

Specifies audio data from an external file source. Auto populated when Infer External Filename is checked

Type: string
Required: False
Pattern: ^(s3://[^/]+/)+([a-zA-Z0-9_]+:)?([a-zA-Z0-9_]+\.[a-zA-Z0-9_]+|\[a-zA-Z0-9_]+\.[a-zA-Z0-9_]+)\d+$
tracks

Identify the channel to include in this selector by entering the 1-based track index. To combine several tracks, enter a comma-separated list, e.g. "1,2,3" for tracks 1-3.

    Type: Array of type integer
    Required: False

AudioSelectorGroup

audioSelectorNames

Name of an "Audio Selector":#inputs-audio_selector within the same input to include in the group. Audio selector names are standardized, based on their order within the input (e.g. "Audio Selector 1"). The audio_selector_name parameter can be repeated to add any number of audio selectors to the group.

    Type: Array of type string
    Required: True

AudioSelectorType (Enum)

Specifies the type of the audio selector.

    PID
    TRACK
    LANGUAGE_CODE

AudioTypeControl (Enum)

When set to FOLLOW_INPUT, if the input contains an ISO 639 audio_type, then that value is passed through to the output. If the input contains no ISO 639 audio_type, the value in Audio Type is included in the output. Otherwise the value in Audio Type is included in the output. Note that this field and audioType are both ignored if audioDescriptionBroadcasterMix is set to BROADCASTER_MIXED_AD.

    FOLLOW_INPUT
    USE_CONFIGURED

AvailBlanking

availBlankingImage

Blanking image to be used. Leave empty for solid black. Only bmp and png images are supported.

    Type: string
    Required: False
    Pattern: ^s3://\/[\w\./]+\.(bmp|BMP|png|PNG)$

BurninDestinationSettings

xPosition

Specifies the horizontal position of the caption relative to the left side of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the left of the output. If no explicit x_position is
provided, the horizontal caption position will be determined by the alignment parameter. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

**Type:** integer  
**Required:** False  
**Minimum:** 0  
**Maximum:** 2147483647

### backgroundColor

**Type:** BurninSubtitleBackgroundColor (p. 351)  
**Required:** False

### teletextSpacing

**Type:** BurninSubtitleTeletextSpacing (p. 352)  
**Required:** False

### yPosition

Specifies the vertical position of the caption relative to the top of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the top of the output. If no explicit y_position is provided, the caption will be positioned towards the bottom of the output. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

**Type:** integer  
**Required:** False  
**Minimum:** 0  
**Maximum:** 2147483647

### backgroundOpacity

Specifies the opacity of the background rectangle. 255 is opaque; 0 is transparent. Leaving this parameter blank is equivalent to setting it to 0 (transparent). All burn-in and DVB-Sub font settings must match.

**Type:** integer  
**Required:** False  
**Minimum:** 0  
**Maximum:** 255

### fontOpacity

Specifies the opacity of the burned-in captions. 255 is opaque; 0 is transparent. All burn-in and DVB-Sub font settings must match.

**Type:** integer  
**Required:** True  
**Minimum:** 0  
**Maximum:** 255
shadowOpacity

Specifies the opacity of the shadow. 255 is opaque; 0 is transparent. Leaving this parameter blank is equivalent to setting it to 0 (transparent). All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: 0
Maximum: 255

fontResolution

Font resolution in DPI (dots per inch); default is 96 dpi. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: 96
Maximum: 600

shadowYOffset

Specifies the vertical offset of the shadow relative to the captions in pixels. A value of -2 would result in a shadow offset 2 pixels above the text. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

outlineSize

Specifies font outline size in pixels. This option is not valid for source captions that are either 608/ embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: True
Minimum: 0
Maximum: 10

outlineColor

Type: BurninSubtitleOutlineColor (p. 352)
Required: True

fontSize

A positive integer indicates the exact font size in points. Set to 0 for automatic font size selection. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: 0
Maximum: 96

**shadowXOffset**

Specifies the horizontal offset of the shadow relative to the captions in pixels. A value of -2 would result in a shadow offset 2 pixels to the left. All burn-in and DVB-Sub font settings must match.

- **Type**: integer
- **Required**: False
- **Minimum**: -2147483648
- **Maximum**: 2147483647

**alignment**

- **Type**: BurninSubtitleAlignment (p. 351)
- **Required**: True

**shadowColor**

- **Type**: BurninSubtitleShadowColor (p. 352)
- **Required**: False

**fontColor**

- **Type**: BurninSubtitleFontColor (p. 351)
- **Required**: False

**BurninSubtitleAlignment (Enum)**

If no explicit x_position or y_position is provided, setting alignment to centered will place the captions at the bottom center of the output. Similarly, setting a left alignment will align captions to the bottom left of the output. If x and y positions are given in conjunction with the alignment parameter, the font will be justified (either left or centered) relative to those coordinates. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

- CENTERED
- LEFT

**BurninSubtitleBackgroundColor (Enum)**

Specifies the color of the rectangle behind the captions. All burn-in and DVB-Sub font settings must match.

- NONE
- BLACK
- WHITE

**BurninSubtitleFontColor (Enum)**

Specifies the color of the burned-in captions. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.
WHITE
BLACK
YELLOW
RED
GREEN
BLUE

**BurninSubtitleOutlineColor (Enum)**

Specifies font outline color. This option is not valid for source captions that are either 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

BLACK
WHITE
YELLOW
RED
GREEN
BLUE

**BurninSubtitleShadowColor (Enum)**

Specifies the color of the shadow cast by the captions. All burn-in and DVB-Sub font settings must match.

NONE
BLACK
WHITE

**BurninSubtitleTeletextSpacing (Enum)**

Controls whether a fixed grid size or proportional font spacing will be used to generate the output subtitles bitmap. Only applicable for Teletext inputs and DVB-Sub/Burn-in outputs.

FIXED_GRID
PROPORTIONAL

**CaptionDescription**

**captionSelectorName**

Specifies which "Caption Selector":#inputs-caption_selector to use from each input when generating captions. The name should be of the format "Caption Selector <N>", which denotes that the Nth Caption Selector will be used from each input.

**Type**: string

**Required**: True

**languageDescription**

Human readable information to indicate captions available for players (eg, English, or Spanish). Alphanumeric characters, spaces, and underscore are legal.
**Type**: string  
**Required**: False  
**Pattern**: `^[\w\ ]*$`

### languageCode

Indicates the language of the caption output track.

**Type**: LanguageCode (p. 410)  
**Required**: False

### destinationSettings

**Type**: CaptionDestinationSettings (p. 353)  
**Required**: False

#### CaptionDestinationSettings

**burninDestinationSettings**

**Type**: BurninDestinationSettings (p. 348)  
**Required**: False

**teletextDestinationSettings**

**Type**: TeletextDestinationSettings (p. 446)  
**Required**: False

**ttmILDestinationSettings**

**Type**: TtmlDestinationSettings (p. 449)  
**Required**: False

#### destinationType

**Type**: CaptionDestinationType (p. 353)  
**Required**: True

**dvbSubDestinationSettings**

**Type**: DvbSubDestinationSettings (p. 363)  
**Required**: False

**sccDestinationSettings**

**Type**: SccDestinationSettings (p. 445)  
**Required**: False

**CaptionDestinationType (Enum)**

BURN_IN
CaptionSelector

sourceSettings

Type: CaptionSourceSettings (p. 354)
Required: True

languageCode

The specific language to extract from source. If input is SCTE-27, complete this field and/or PID to select the caption language to extract. If input is DVB-Sub and output is Burn-in or SMPTE-TT, complete this field and/or PID to select the caption language to extract. If input is DVB-Sub that is being passed through, omit this field (and PID field); there is no way to extract a specific language with pass-through captions.

Type: LanguageCode (p. 410)
Required: False

CaptionSourceSettings

fileSourceSettings

Type: FileSourceSettings (p. 373)
Required: False

ancillarySourceSettings

Type: AncillarySourceSettings (p. 342)
Required: False

embeddedSourceSettings

Type: EmbeddedSourceSettings (p. 372)
Required: False

sourceType

Type: CaptionSourceType (p. 355)
Required: True

dvbSubSourceSettings

Type: DvbSubSourceSettings (p. 365)
Required: False

teletextSourceSettings

Type: TeletextSourceSettings (p. 446)
Required: False

CaptionSourceType (Enum)

Use Source (SourceType) to identify the format of your input captions. The service cannot auto-detect caption format.

ANCILLARY
DVB_SUB
EMBEDDED
SCC
TTML
STL
SRT
TELETEXT
NULL_SOURCE

ChannelMapping

outputChannels

Type: Array of type OutputChannelMapping (p. 438)
Required: True

ColorCorrector

saturation

Saturation level.

Type: integer
Required: False
Minimum: 1
Maximum: 100

brightness

Brightness level.

Type: integer
Required: False
Minimum: 1
Maximum: 100

hdr10Metadata

Type: Hdr10Metadata (p. 392)
Properties

Required: False

**contrast**
Contrast level.

*Type: integer*
Required: False
Minimum: 1
Maximum: 100

**hue**
Hue in degrees.

*Type: integer*
Required: False
Minimum: -180
Maximum: 180

**colorSpaceConversion**

*Type: ColorSpaceConversion (p. 356)*
Required: False

**ColorMetadata (Enum)**
Enable Insert color metadata (ColorMetadata) to include color metadata in this output. This setting is enabled by default.

IGNORE
INSERT

**ColorSpace (Enum)**
Specifies the colorspace of an input. This setting works in tandem with "Color Corrector":#color_corrector > color_space_conversion to determine if any conversion will be performed.

FOLLOW
REC_601
REC_709
HDR10
HLG_2020

**ColorSpaceConversion (Enum)**
Determines if colorspace conversion will be performed. If set to _None_, no conversion will be performed. If _Force 601_ or _Force 709_ are selected, conversion will be performed for inputs with differing colorspaces. An input's colorspace can be specified explicitly in the "Video Selector":#inputs-video_selector if necessary.

NONE
FORCE_601
FORCE_709
FORCE_HDR10
FORCE_HLG_2020

ColorSpaceUsage (Enum)

There are two sources for color metadata, the input file and the job configuration. This enum controls which takes precedence. FORCE: System will use color metadata supplied by user, if any. If the user does not supply color metadata the system will use data from the source. FALLBACK: System will use color metadata from the source. If source has no color metadata, the system will use user-supplied color metadata values if available.

FORCE
FALLBACK

ContainerSettings

container

Type: ContainerType (p. 357)
Required: True

mp4Settings

Type: Mp4Settings (p. 426)
Required: False

m3u8Settings

Type: M3u8Settings (p. 422)
Required: False

m2tsSettings

Type: M2tsSettings (p. 417)
Required: False

movSettings

Type: MovSettings (p. 425)
Required: False

f4vSettings

Type: F4vSettings (p. 373)
Required: False

ContainerType (Enum)

Container for this output. Some containers require a container settings object. If not specified, the default object will be created.

F4V
ISMV
CreateJobRequest

settings

Type: JobSettings (p. 409)
Required: True

role

Required. The IAM role you use for creating this job. For details about permissions, see the User Guide topic at the User Guide at http://docs.aws.amazon.com/mediaconvert/latest/ug/iam-role.html.

Type: string
Required: True

clientRequestToken

Idempotency token for CreateJob operation.

Type: string
Required: False

jobTemplate

When you create a job, you can either specify a job template or specify the transcoding settings individually

Type: string
Required: False

userMetadata

Type: object
Required: False

queue

Type: string
Required: False

CreateJobResponse

job

Type: Job (p. 407)


**Required**: False

**DashIsoEncryptionSettings**

**spekeKeyProvider**

*Type*: SpekeKeyProvider (p. 445)

*Required*: True

**DashIsoGroupSettings**

**fragmentLength**

Length of fragments to generate (in seconds). Fragment length must be compatible with GOP size and framerate. Note that fragments will end on the next keyframe after this number of seconds, so actual fragment length may be longer. When Emit Single File is checked, the fragmentation is internal to a single output file and it does not cause the creation of many output files as in other output types.

*Type*: integer

*Required*: True

*Minimum*: 1

*Maximum*: 2147483647

**baseUrl**

A partial URI prefix that will be put in the manifest (.mpd) file at the top level BaseURL element. Can be used if streams are delivered from a different URL than the manifest file.

*Type*: string

*Required*: False

**minBufferTime**

Minimum time of initially buffered media that is needed to ensure smooth playout.

*Type*: integer

*Required*: False

*Minimum*: 0

*Maximum*: 2147483647

**encryption**

DRM settings.

*Type*: DashIsoEncryptionSettings (p. 359)

*Required*: False

**destination**

Use Destination (Destination) to specify the S3 output location and the output filename base. Destination accepts format identifiers. If you do not specify the base filename in the URI, the service will use the filename of the input file. If your job has multiple inputs, the service uses the filename of the first input file.
Properties

Type: string  
Required: True  
Pattern: ^s3:\/\/

segmentLength

Length of mpd segments to create (in seconds). Note that segments will end on the next keyframe after this number of seconds, so actual segment length may be longer. When Emit Single File is checked, the segmentation is internal to a single output file and it does not cause the creation of many output files as in other output types.

Type: integer  
Required: True  
Minimum: 1  
Maximum: 2147483647

segmentControl

Type: DashIsoSegmentControl (p. 360)  
Required: False

hbbtvCompliance

Type: DashIsoHbbtvCompliance (p. 360)  
Required: False

DashIsoHbbtvCompliance (Enum)

Supports HbbTV specification as indicated

HBBTV_1_5  
NONE

DashIsoSegmentControl (Enum)

When set to SINGLE_FILE, a single output file is generated, which is internally segmented using the Fragment Length and Segment Length. When set to SEGMENTED_FILES, separate segment files will be created.

SINGLE_FILE  
SEGMENTED_FILES

DeinterlaceAlgorithm (Enum)

Only applies when you set Deinterlacer (DeinterlaceMode) to Deinterlace (DEINTERLACE) or Adaptive (ADAPTIVE). Motion adaptive interpolate (INTERPOLATE) produces sharper pictures, while blend (BLEND) produces smoother motion. Use (INTERPOLATE_TICKER) OR (BLEND_TICKER) if your source file includes a ticker, such as a scrolling headline at the bottom of the frame.

INTERPOLATE  
INTERPOLATE_TICKER  
BLEND  
BLEND_TICKER
**Deinterlacer**

*mode*

Type: `DeinterlacerMode` (p. 361)  
Required: False

*control*

Type: `DeinterlacerControl` (p. 361)  
Required: False

*algorithm*

Type: `DeinterlaceAlgorithm` (p. 360)  
Required: False

**DeinterlacerControl (Enum)**

- When set to NORMAL (default), the deinterlacer does not convert frames that are tagged in metadata as progressive. It will only convert those that are tagged as some other type.  
- When set to FORCE_ALL_FRAMES, the deinterlacer converts every frame to progressive - even those that are already tagged as progressive. Turn Force mode on only if there is a good chance that the metadata has tagged frames as progressive when they are not progressive. Do not turn on otherwise; processing frames that are already progressive into progressive will probably result in lower quality video.

  *FORCE_ALL_FRAMES*  
  *NORMAL*

**DeinterlacerMode (Enum)**

Use Deinterlacer (`DeinterlaceMode`) to choose how the service will do deinterlacing. Default is Deinterlace.  
- Deinterlace converts interlaced to progressive.  
- Inverse telecine converts Hard Telecine 29.97i to progressive 23.976p.  
- Adaptive auto-detects and converts to progressive.

  *DEINTERLACE*  
  *INVERSE_TELECINE*  
  *ADAPTIVE*

**DropFrameTimecode (Enum)**

Applies only to 29.97 fps outputs. When this feature is enabled, the service will use drop-frame timecode on outputs. If it is not possible to use drop-frame timecode, the system will fall back to non-drop-frame. This setting is enabled by default when Timecode insertion (TimecodeInsertion) is enabled.

  *DISABLED*  
  *ENABLED*

**DvbNitSettings**

*networkName*

The network name text placed in the network_name_descriptor inside the Network Information Table. Maximum length is 256 characters.
**Properties**

**Type**
- string
- **Required**: True

**networkId**
The numeric value placed in the Network Information Table (NIT).
- **Type**: integer
- **Required**: True
- **Minimum**: 0
- **Maximum**: 65535

**nitInterval**
The number of milliseconds between instances of this table in the output transport stream.
- **Type**: integer
- **Required**: True
- **Minimum**: 25
- **Maximum**: 10000

**DvbSdtSettings**

**sdtInterval**
The number of milliseconds between instances of this table in the output transport stream.
- **Type**: integer
- **Required**: False
- **Minimum**: 25
- **Maximum**: 2000

**serviceName**
The service name placed in the service_descriptor in the Service Description Table. Maximum length is 256 characters.
- **Type**: string
- **Required**: False

**serviceProviderName**
The service provider name placed in the service_descriptor in the Service Description Table. Maximum length is 256 characters.
- **Type**: string
- **Required**: False

**outputSdt**
- **Type**: OutputSdt (p. 440)
- **Required**: False
**DvbSubDestinationSettings**

**xPosition**

Specifies the horizontal position of the caption relative to the left side of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the left of the output. If no explicit x_position is provided, the horizontal caption position will be determined by the alignment parameter. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

- **Type:** integer
  - **Required:** False
  - **Minimum:** 0
  - **Maximum:** 2147483647

**backgroundColor**

- **Type:** DvbSubtitleBackgroundColor (p. 366)
- **Required:** False

**teletextSpacing**

- **Type:** DvbSubtitleTeletextSpacing (p. 366)
- **Required:** False

**yPosition**

Specifies the vertical position of the caption relative to the top of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the top of the output. If no explicit y_position is provided, the caption will be positioned towards the bottom of the output. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

- **Type:** integer
  - **Required:** False
  - **Minimum:** 0
  - **Maximum:** 2147483647

**backgroundOpacity**

Specifies the opacity of the background rectangle. 255 is opaque; 0 is transparent. Leaving this parameter blank is equivalent to setting it to 0 (transparent). All burn-in and DVB-Sub font settings must match.

- **Type:** integer
  - **Required:** False
  - **Minimum:** 0
  - **Maximum:** 255

**fontOpacity**

Specifies the opacity of the burned-in captions. 255 is opaque; 0 is transparent. All burn-in and DVB-Sub font settings must match.

- **Type:** integer
Properties

shadowOpacity

Specifies the opacity of the shadow. 255 is opaque; 0 is transparent. Leaving this parameter blank is equivalent to setting it to 0 (transparent). All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: 0
Maximum: 255

fontResolution

Font resolution in DPI (dots per inch); default is 96 dpi. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: 96
Maximum: 600

shadowYOffset

 Specifies the vertical offset of the shadow relative to the captions in pixels. A value of -2 would result in a shadow offset 2 pixels above the text. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

outlineSize

Specifies font outline size in pixels. This option is not valid for source captions that are either 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: True
Minimum: 0
Maximum: 10

outlineColor

Type: DvbSubtitleOutlineColor (p. 366)
Required: True

fontSize

A positive integer indicates the exact font size in points. Set to 0 for automatic font size selection. All burn-in and DVB-Sub font settings must match.
Properties

**Type**
- integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 96

**shadowXOffset**
Specifies the horizontal offset of the shadow relative to the captions in pixels. A value of -2 would result in a shadow offset 2 pixels to the left. All burn-in and DVB-Sub font settings must match.
- **Type**: integer
- **Required**: False
- **Minimum**: -2147483648
- **Maximum**: 2147483647

**alignment**
- **Type**: DvbSubtitleAlignment (p. 365)
- **Required**: True

**shadowColor**
- **Type**: DvbSubtitleShadowColor (p. 366)
- **Required**: False

**fontColor**
- **Type**: DvbSubtitleFontColor (p. 366)
- **Required**: False

**DvbSubSourceSettings**

**pid**
When using DVB-Sub with Burn-In or SMPTE-TT, use this PID for the source content. Unused for DVB-Sub passthrough. All DVB-Sub content is passed through, regardless of selectors.
- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 2147483647

**DvbSubtitleAlignment (Enum)**
If no explicit x_position or y_position is provided, setting alignment to centered will place the captions at the bottom center of the output. Similarly, setting a left alignment will align captions to the bottom left of the output. If x and y positions are given in conjunction with the alignment parameter, the font will be justified (either left or centered) relative to those coordinates. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.
- CENTERED
Properties

**DvbSubtitleBackgroundColor (Enum)**

Specifies the color of the rectangle behind the captions. All burn-in and DVB-Sub font settings must match.

- NONE
- BLACK
- WHITE

**DvbSubtitleFontColor (Enum)**

Specifies the color of the burned-in captions. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

- WHITE
- BLACK
- YELLOW
- RED
- GREEN
- BLUE

**DvbSubtitleOutlineColor (Enum)**

Specifies font outline color. This option is not valid for source captions that are either 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

- BLACK
- WHITE
- YELLOW
- RED
- GREEN
- BLUE

**DvbSubtitleShadowColor (Enum)**

Specifies the color of the shadow cast by the captions. All burn-in and DVB-Sub font settings must match.

- NONE
- BLACK
- WHITE

**DvbSubtitleTeletextSpacing (Enum)**

Controls whether a fixed grid size or proportional font spacing will be used to generate the output subtitles bitmap. Only applicable for Teletext inputs and DVB-Sub/Burn-in outputs.

- FIXED_GRID
- PROPORTIONAL
DvbTdtSettings

tdtInterval

The number of milliseconds between instances of this table in the output transport stream.

  Type: integer
  Required: True
  Minimum: 1000
  Maximum: 30000

Eac3AttenuationControl (Enum)

If set to ATTENUATE_3_DB, applies a 3 dB attenuation to the surround channels. Only used for 3/2 coding mode.

  ATTENUATE_3_DB
  NONE

Eac3BitstreamMode (Enum)

Specifies the "Bitstream Mode" (bsmod) for the emitted E-AC-3 stream. See ATSC A/52-2012 (Annex E) for background on these values.

  COMPLETE_MAIN
  COMMENTARY
  EMERGENCY
  HEARING_IMPAIRED
  VISUALLY_IMPAIRED

Eac3CodingMode (Enum)

Dolby Digital Plus coding mode. Determines number of channels.

  CODING_MODE_1_0
  CODING_MODE_2_0
  CODING_MODE_3_2

Eac3DcFilter (Enum)

Activates a DC highpass filter for all input channels.

  ENABLED
  DISABLED

Eac3DynamicRangeCompressionLine (Enum)

Enables Dynamic Range Compression that restricts the absolute peak level for a signal.

  NONE
  FILM_STANDARD
  FILM_LIGHT
  MUSIC_STANDARD
MUSIC_LIGHT
SPEECH

**Eac3DynamicRangeCompressionRf (Enum)**

Enables Heavy Dynamic Range Compression, ensures that the instantaneous signal peaks do not exceed specified levels.

NONE
FILM_STANDARD
FILM_LIGHT
MUSIC_STANDARD
MUSIC_LIGHT
SPEECH

**Eac3LfeControl (Enum)**

When encoding 3/2 audio, controls whether the LFE channel is enabled.

LFE
NO_LFE

**Eac3LfeFilter (Enum)**

Applies a 120Hz lowpass filter to the LFE channel prior to encoding. Only valid with 3_2_LFE coding mode.

ENABLED
DISABLED

**Eac3MetadataControl (Enum)**

When set to FOLLOW_INPUT, encoder metadata will be sourced from the DD, DD+, or DolbyE decoder that supplied this audio data. If audio was not supplied from one of these streams, then the static metadata settings will be used.

FOLLOW_INPUT
USE_CONFIGURED

**Eac3PassthroughControl (Enum)**

When set to WHEN_POSSIBLE, input DD+ audio will be passed through if it is present on the input. This detection is dynamic over the life of the transcode. Inputs that alternate between DD+ and non-DD+ content will have a consistent DD+ output as the system alternates between passthrough and encoding.

WHEN_POSSIBLE
NO_PASSTHROUGH

**Eac3PhaseControl (Enum)**

Controls the amount of phase-shift applied to the surround channels. Only used for 3/2 coding mode.

SHIFT_90_DEGREES
NO_SHIFT
**Eac3Settings**

**dialnorm**

Sets the dialnorm for the output. If blank and input audio is Dolby Digital Plus, dialnorm will be passed through.

- **Type:** integer
- **Required:** False
- **Minimum:** 1
- **Maximum:** 31

**passthroughControl**

- **Type:** Eac3PassthroughControl (p. 368)
- **Required:** False

**metadataControl**

- **Type:** Eac3MetadataControl (p. 368)
- **Required:** False

**bitrate**

Average bitrate in bits/second. Valid bitrates depend on the coding mode.

- **Type:** integer
- **Required:** False
- **Minimum:** 64000
- **Maximum:** 640000

**dynamicRangeCompressionRf**

- **Type:** Eac3DynamicRangeCompressionRf (p. 368)
- **Required:** False

**sampleRate**

Sample rate in hz. Sample rate is always 48000.

- **Type:** integer
- **Required:** False
- **Minimum:** 48000
- **Maximum:** 48000

**ltRtSurroundMixLevel**

Left total/Right total surround mix level. Only used for 3/2 coding mode. Valid values: -1.5 -3.0 -4.5 -6.0 -60

- **Type:** number
- **Required:** False
- **Format:** float
Properties

surroundExMode

Type: Eac3SurroundExMode (p. 371)
Required: False

dynamicRangeCompressionLine

Type: Eac3DynamicRangeCompressionLine (p. 367)
Required: False

lfeControl

Type: Eac3LfeControl (p. 368)
Required: False

codingMode

Type: Eac3CodingMode (p. 367)
Required: False

surroundMode

Type: Eac3SurroundMode (p. 372)
Required: False

attenuationControl

Type: Eac3AttenuationControl (p. 367)
Required: False

lfeFilter

Type: Eac3LfeFilter (p. 368)
Required: False

phaseControl

Type: Eac3PhaseControl (p. 368)
Required: False

ltRtCenterMixLevel

Left total/Right total center mix level. Only used for 3/2 coding mode. Valid values: 3.0, 1.5, 0.0, -1.5
-3.0 -4.5 -6.0 -60

Type: number
Required: False
Format: float
Minimum: -60.0
Maximum: 3.0

dcFilter

  Type: Eac3DcFilter (p. 367)
  Required: False

stereoDownmix

  Type: Eac3StereoDownmix (p. 371)
  Required: False

bitstreamMode

  Type: Eac3BitstreamMode (p. 367)
  Required: False

loRoSurroundMixLevel

Left only/Right only surround mix level. Only used for 3/2 coding mode. Valid values: -1.5 -3.0 -4.5 -6.0 -60

  Type: number
  Required: False
  Format: float
  Minimum: -60.0
  Maximum: -1.5

loRoCenterMixLevel

Left only/Right only center mix level. Only used for 3/2 coding mode. Valid values: 3.0, 1.5, 0.0, -1.5 -3.0 -4.5 -6.0 -60

  Type: number
  Required: False
  Format: float
  Minimum: -60.0
  Maximum: 3.0

Eac3StereoDownmix (Enum)

Stereo downmix preference. Only used for 3/2 coding mode.

  NOT_INDICATED
  LO_RO
  LT_RT
  DPL2

Eac3SurroundExMode (Enum)

When encoding 3/2 audio, sets whether an extra center back surround channel is matrix encoded into the left and right surround channels.
Eac3SurroundMode (Enum)
When encoding 2/0 audio, sets whether Dolby Surround is matrix encoded into the two channels.

- NOT_INDICATED
- ENABLED
- DISABLED

EmbeddedConvert608To708 (Enum)
When set to UPCONVERT, 608 data is both passed through via the "608 compatibility bytes" fields of the 708 wrapper as well as translated into 708. 708 data present in the source content will be discarded.

- UPCONVERT
- DISABLED

EmbeddedSourceSettings
source608ChannelNumber
Specifies the 608/708 channel number within the video track from which to extract captions. Unused for passthrough.

- Type: integer
- Required: False
- Minimum: 1
- Maximum: 4

convert608To708

- Type: EmbeddedConvert608To708 (p. 372)
- Required: False

source608TrackNumber
Specifies the video track index used for extracting captions. The system only supports one input video track, so this should always be set to '1'.

- Type: integer
- Required: False
- Minimum: 1
- Maximum: 1

ExceptionBody
message

- Type: string
Required: False

F4vMoovPlacement (Enum)

If set to PROGRESSIVE_DOWNLOAD, the MOOV atom is relocated to the beginning of the archive as required for progressive downloading. Otherwise it is placed normally at the end.

   PROGRESSIVE_DOWNLOAD
   NORMAL

F4vSettings

moovPlacement

   Type: F4vMoovPlacement (p. 373)
   Required: False

FileGroupSettings

destination

Use Destination (Destination) to specify the S3 output location and the output filename base. Destination accepts format identifiers. If you do not specify the base filename in the URI, the service will use the filename of the input file. If your job has multiple inputs, the service uses the filename of the first input file.

   Type: string
   Required: True
   Pattern: ^s3:\/\/

FileSourceConvert608To708 (Enum)

If set to UPCONVERT, 608 caption data is both passed through via the "608 compatibility bytes" fields of the 708 wrapper as well as translated into 708. 708 data present in the source content will be discarded.

   UPCONVERT
   DISABLED

FileSourceSettings

timeDelta

Specifies a time delta in seconds to offset the captions from the source file.

   Type: integer
   Required: False
   Minimum: -2147483648
   Maximum: 2147483647

convert608To708

   Type: FileSourceConvert608To708 (p. 373)
   Required: False
sourceFile

External caption file used for loading captions. Accepted file extensions are 'scc', 'ttml', 'dfxp', 'stl', 'srt', and 'smi'. Auto-populated when Infer External Filename is checked.

Type: string
Required: True
Pattern: ^(s3://)(.*?)\.(scc|SCC|ttml|TTML|dfxp|DFXP|stl|STL|srt|SRT|smi|SMI)$

FrameCaptureSettings

framerateDenominator

Frame capture will encode the first frame of the output stream, then one frame every framerateDenominator/framerateNumerator seconds. For example, settings of framerateNumerator = 1 and framerateDenominator = 3 (a rate of 1/3 frame per second) will capture the first frame, then 1 frame every 3s. Files will be named as filename.n.jpg where n is the 0-based sequence number of each Capture.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

maxCaptures

Maximum number of captures (encoded jpg output files).

Type: integer
Required: False
Minimum: 1
Maximum: 10000000

framerateNumerator

Frame capture will encode the first frame of the output stream, then one frame every framerateDenominator/framerateNumerator seconds. For example, settings of framerateNumerator = 1 and framerateDenominator = 3 (a rate of 1/3 frame per second) will capture the first frame, then 1 frame every 3s. Files will be named as filename.NNNNNNN.jpg where N is the 0-based frame sequence number zero padded to 7 decimal places.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

quality

JPEG Quality - a higher value equals higher quality.

Type: integer
Required: False
Minimum: 1
Maximum: 100
H264AdaptiveQuantization (Enum)

Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

- OFF
- LOW
- MEDIUM
- HIGH
- HIGHER
- MAX

H264CodecLevel (Enum)

H.264 Level.

- AUTO
- LEVEL_1
- LEVEL_1_1
- LEVEL_1_2
- LEVEL_1_3
- LEVEL_2
- LEVEL_2_1
- LEVEL_2_2
- LEVEL_3
- LEVEL_3_1
- LEVEL_3_2
- LEVEL_4
- LEVEL_4_1
- LEVEL_4_2
- LEVEL_5
- LEVEL_5_1
- LEVEL_5_2

H264CodecProfile (Enum)

H.264 Profile. High 4:2:2 and 10-bit profiles are only available with the AVC-I License.

- BASELINE
- HIGH
- HIGH_10BIT
- HIGH_422
- HIGH_422_10BIT
- MAIN

H264EntropyEncoding (Enum)

Entropy encoding mode. Use CABAC (must be in Main or High profile) or CAVLC.

- CABAC
- CAVLC

H264FieldEncoding (Enum)

Choosing FORCE_FIELD disables PAFF encoding for interlaced outputs.

- PAFF
FORCE_FIELD

H264FlickerAdaptiveQuantization (Enum)
Adjust quantization within each frame to reduce flicker or 'pop' on I-frames.
   DISABLED
   ENABLED

H264FramerateControl (Enum)
Using the API, set FramerateControl to INITIALIZE_FROM_SOURCE if you want the service to use the framerate from the input. Using the console, do this by choosing INITIALIZE_FROM_SOURCE for Framerate.
   INITIALIZE_FROM_SOURCE
   SPECIFIED

H264FramerateConversionAlgorithm (Enum)
When set to INTERPOLATE, produces smoother motion during framerate conversion.
   DUPLICATE_DROP
   INTERPOLATE

H264GopBReference (Enum)
If enable, use reference B frames for GOP structures that have B frames > 1.
   DISABLED
   ENABLED

H264GopSizeUnits (Enum)
Indicates if the GOP Size in H264 is specified in frames or seconds. If seconds the system will convert the GOP Size into a frame count at run time.
   FRAMES
   SECONDS

H264InterlaceMode (Enum)
Use Interlace mode (InterlaceMode) to choose the scan line type for the output.
* Top Field First (TOP_FIELD) and Bottom Field First (BOTTOM_FIELD) produce interlaced output with the entire output having the same field polarity (top or bottom first).
* Follow, Default Top (FOLLOW_TOP_FIELD) and Follow, Default Bottom (FOLLOW_BOTTOM_FIELD) use the same field polarity as the source. Therefore, behavior depends on the input scan type. If the source is interlaced, the output will be interlaced with the same polarity as the source (it will follow the source). The output could therefore be a mix of "top field first" and "bottom field first". If the source is progressive, the output will be interlaced with "top field first" or "bottom field first" polarity, depending on which of the Follow options you chose.
   PROGRESSIVE
H264ParControl (Enum)
Using the API, enable ParFollowSource if you want the service to use the pixel aspect ratio from the input. Using the console, do this by choosing Follow source for Pixel aspect ratio.

INITIALIZE_FROM_SOURCE
SPECIFIED

H264QualityTuningLevel (Enum)
Use Quality tuning level (H264QualityTuningLevel) to specify whether to use fast single-pass, high-quality singlepass, or high-quality multipass video encoding.

SINGLE_PASS
SINGLE_PASS_HQ
MULTI_PASS_HQ

H264RateControlMode (Enum)
Rate control mode. CQ uses constant quantizer (qp), ABR (average bitrate) does not write HRD parameters.

VBR
CBR

H264RepeatPps (Enum)
Places a PPS header on each encoded picture, even if repeated.

DISABLED
ENABLED

H264SceneChangeDetect (Enum)
Scene change detection (inserts I-frames on scene changes).

DISABLED
ENABLED

H264Settings
slices
Number of slices per picture. Must be less than or equal to the number of macroblock rows for progressive pictures, and less than or equal to half the number of macroblock rows for interlaced pictures.

Type: integer
Required: False
Properties

Minimum: 1
Maximum: 32

minIInterval
Enforces separation between repeated (cadence) I-frames and I-frames inserted by Scene Change Detection. If a scene change I-frame is within I-interval frames of a cadence I-frame, the GOP is shrunk and/or stretched to the scene change I-frame. GOP stretch requires enabling lookahead as well as setting I-interval. The normal cadence resumes for the next GOP. This setting is only used when Scene Change Detect is enabled. Note: Maximum GOP stretch = GOP size + Min-I-interval - 1

Type: integer
Required: False
Minimum: 0
Maximum: 30

parNumerator
Pixel Aspect Ratio numerator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

flickerAdaptiveQuantization

Type: H264FlickerAdaptiveQuantization (p. 376)
Required: False

gopSizeUnits

Type: H264GopSizeUnits (p. 376)
Required: False

hrdBufferSize
Size of buffer (HRD buffer model). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

qualityTuningLevel

Type: H264QualityTuningLevel (p. 377)
Required: False

maxBitrate
Maximum bitrate in bits/second (for VBR mode only). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.
Type: integer  
Required: False  
Minimum: -2147483648  
Maximum: 2147483647

**bitrate**

Average bitrate in bits/second. Required for VBR, CBR, and ABR. Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.

Type: integer  
Required: False  
Minimum: 1000  
Maximum: 2147483647

**spatialAdaptiveQuantization**

Type: H264SpatialAdaptiveQuantization (p. 382)  
Required: False

**slowPal**

Type: H264SlowPal (p. 382)  
Required: False

**codecProfile**

Type: H264CodecProfile (p. 375)  
Required: False

**unregisteredSeiTmecode**

Type: H264UnregisteredSeiTmecode (p. 383)  
Required: False

**softness**

Softness. Selects quantizer matrix, larger values reduce high-frequency content in the encoded image.

Type: integer  
Required: False  
Minimum: 0  
Maximum: 128

**framerateControl**

Type: H264FramerateControl (p. 376)  
Required: False

**telecine**

Type: H264Telecine (p. 383)
Required: False

framerateConversionAlgorithm

Type: H264FramerateConversionAlgorithm (p. 376)
Required: False

codecLevel

Type: H264CodecLevel (p. 375)
Required: False

numberReferenceFrames

Number of reference frames to use. The encoder may use more than requested if using B-frames and/or interlaced encoding.

Type: integer
Required: False
Minimum: 1
Maximum: 6

temporalAdaptiveQuantization

Type: H264TemporalAdaptiveQuantization (p. 383)
Required: False

repeatPps

Type: H264RepeatPps (p. 377)
Required: False

hrdBufferInitialFillPercentage

Percentage of the buffer that should initially be filled (HRD buffer model).

Type: integer
Required: False
Minimum: 0
Maximum: 100

framerateNumerator

Framerate numerator - framerate is a fraction, e.g. 24000 / 1001 = 23.976 fps.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

numberBFramesBetweenReferenceFrames

Number of B-frames between reference frames.
Type: integer
Required: False
Minimum: 0
Maximum: 7

gopClosedCadence

Frequency of closed GOPs. In streaming applications, it is recommended that this be set to 1 so a decoder joining mid-stream will receive an IDR frame as quickly as possible. Setting this value to 0 will break output segmenting.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

fieldEncoding

Type: H264FieldEncoding (p. 375)
Required: False

entropyEncoding

Type: H264EntropyEncoding (p. 375)
Required: False

framerateDenominator

When you use the API for transcode jobs that use framerate conversion, specify the framerate as a fraction. For example, 24000 / 1001 = 23.976 fps. Use FramerateDenominator to specify the denominator of this fraction. In this example, use 1001 for the value of FramerateDenominator. When you use the console for transcode jobs that use framerate conversion, provide the value as a decimal number for Framerate. In this example, specify 23.976.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

adaptiveQuantization

Type: H264AdaptiveQuantization (p. 375)
Required: False

interlaceMode

Type: H264InterlaceMode (p. 376)
Required: False

gopSize

GOP Length (keyframe interval) in frames or seconds. Must be greater than zero.
Type: number  
Required: False  
Format: float  
Minimum: 0.0

gopBReference
Type: H264GopBReference (p. 376)  
Required: False

sceneChangeDetect
Type: H264SceneChangeDetect (p. 377)  
Required: False

parDenominator
Pixel Aspect Ratio denominator.
Type: integer  
Required: False  
Minimum: 1  
Maximum: 2147483647

parControl
Type: H264ParControl (p. 377)  
Required: False

syntax
Type: H264Syntax (p. 383)  
Required: False

rateControlMode
Type: H264RateControlMode (p. 377)  
Required: False

H264SlowPal (Enum)
Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.
DISABLED  
ENABLED

H264SpatialAdaptiveQuantization (Enum)
Adjust quantization within each frame based on spatial variation of content complexity.
DISABLED
**H264Syntax (Enum)**

Produces a bitstream compliant with SMPTE RP-2027.

* DEFAULT
  * RP2027

**H264Telecine (Enum)**

This field applies only if the Streams > Advanced > Framerate (framerate) field is set to 29.970. This field works with the Streams > Advanced > Preprocessors > Deinterlacer field (deinterlace_mode) and the Streams > Advanced > Interlaced Mode field (interlace_mode) to identify the scan type for the output:
- Progressive, Interlaced, Hard Telecine or Soft Telecine. - Hard: produces 29.97i output from 23.976 input. - Soft: produces 23.976; the player converts this output to 29.97i.

* NONE
  * SOFT
  * HARD

**H264TemporalAdaptiveQuantization (Enum)**

Adjust quantization within each frame based on temporal variation of content complexity.

* DISABLED
  * ENABLED

**H264UnregisteredSeiTimecode (Enum)**

Inserts timecode for each frame as 4 bytes of an unregistered SEI message.

* DISABLED
  * ENABLED

**H265AdaptiveQuantization (Enum)**

Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

* OFF
  * LOW
  * MEDIUM
  * HIGH
  * HIGHER
  * MAX

**H265AlternateTransferFunctionSei (Enum)**

Enables Alternate Transfer Function SEI message for outputs using Hybrid Log Gamma (HLG) Electro-Optical Transfer Function (EOTF).

* DISABLED
  * ENABLED
H265CodecLevel (Enum)

H.265 Level.

- AUTO
- LEVEL_1
- LEVEL_2
- LEVEL_2_1
- LEVEL_3
- LEVEL_3_1
- LEVEL_4
- LEVEL_4_1
- LEVEL_5
- LEVEL_5_1
- LEVEL_5_2
- LEVEL_6
- LEVEL_6_1
- LEVEL_6_2

H265CodecProfile (Enum)

Represents the Profile and Tier, per the HEVC (H.265) specification. Selections are grouped as [Profile] / [Tier], so "Main/High" represents Main Profile with High Tier. 4:2:2 profiles are only available with the HEVC 4:2:2 License.

- MAIN_MAIN
- MAIN_HIGH
- MAIN10_MAIN
- MAIN10_HIGH
- MAIN_422_8BIT_MAIN
- MAIN_422_8BIT_HIGH
- MAIN_422_10BIT_MAIN
- MAIN_422_10BIT_HIGH

H265FlickerAdaptiveQuantization (Enum)

Adjust quantization within each frame to reduce flicker or 'pop' on I-frames.

- DISABLED
- ENABLED

H265FramerateControl (Enum)

Using the API, set FramerateControl to INITIALIZE_FROM_SOURCE if you want the service to use the framerate from the input. Using the console, do this by choosing INITIALIZE_FROM_SOURCE for Framerate.

- INITIALIZE_FROM_SOURCE
- SPECIFIED

H265FramerateConversionAlgorithm (Enum)

When set to INTERPOLATE, produces smoother motion during framerate conversion.
DUPLICATE_DROP
INTERPOLATE

H265GopBReference (Enum)
If enable, use reference B frames for GOP structures that have B frames > 1.

DISABLED
ENABLED

H265GopSizeUnits (Enum)
Indicates if the GOP Size in H265 is specified in frames or seconds. If seconds the system will convert the
GOP Size into a frame count at run time.

FRAMES
SECONDS

H265InterlaceMode (Enum)
Use Interlace mode (InterlaceMode) to choose the scan line type for the output.

* Top Field First (TOP_FIELD) and Bottom Field First (BOTTOM_FIELD) produce interlaced output with the
entire output having the same field polarity (top or bottom first).

* Follow, Default Top (FOLLOW_TOP_FIELD) and Follow, Default Bottom (FOLLOW_BOTTOM_FIELD)
use the same field polarity as the source. Therefore, behavior depends on the input scan type. If the
source is interlaced, the output will be interlaced with the same polarity as the source (it will follow the
source). The output could therefore be a mix of "top field first" and "bottom field first". If the source is
progressive, the output will be interlaced with "top field first" or "bottom field first" polarity, depending
on which of the Follow options you chose.

PROGRESSIVE
TOP_FIELD
BOTTOM_FIELD
FOLLOW_TOP_FIELD
FOLLOW_BOTTOM_FIELD

H265ParControl (Enum)
Using the API, enable ParFollowSource if you want the service to use the pixel aspect ratio from the
input. Using the console, do this by choosing Follow source for Pixel aspect ratio.

INITIALIZE_FROM_SOURCE
SPECIFIED

H265QualityTuningLevel (Enum)
Use Quality tuning level (H265QualityTuningLevel) to specify whether to use fast single-pass, high-
quality singlepass, or high-quality multipass video encoding.

SINGLE_PASS
SINGLE_PASS_HQ
MULTI_PASS_HQ
H265RateControlMode (Enum)

Rate control mode. CQ uses constant quantizer (qp), ABR (average bitrate) does not write HRD parameters.

- VBR
- CBR

H265SampleAdaptiveOffsetFilterMode (Enum)

Specify Sample Adaptive Offset (SAO) filter strength. Adaptive mode dynamically selects best strength based on content.

- DEFAULT
- ADAPTIVE
- OFF

H265SceneChangeDetect (Enum)

Scene change detection (inserts I-frames on scene changes).

- DISABLED
- ENABLED

H265Settings

slices

Number of slices per picture. Must be less than or equal to the number of macroblock rows for progressive pictures, and less than or equal to half the number of macroblock rows for interlaced pictures.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 32

minInterval

Enforces separation between repeated (cadence) I-frames and I-frames inserted by Scene Change Detection. If a scene change I-frame is within I-interval frames of a cadence I-frame, the GOP is shrunk and/or stretched to the scene change I-frame. GOP stretch requires enabling lookahead as well as setting I-interval. The normal cadence resumes for the next GOP. This setting is only used when Scene Change Detect is enabled. Note: Maximum GOP stretch = GOP size + Min-I-interval - 1

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 30

parNumerator

Pixel Aspect Ratio numerator.

- **Type**: integer
**Properties**

- **flickerAdaptiveQuantization**
  - Type: `H265FlickerAdaptiveQuantization (p. 384)`
  - Required: False

- **gopSizeUnits**
  - Type: `H265GopSizeUnits (p. 385)`
  - Required: False

- **hrdBufferSize**
  - Size of buffer (HRD buffer model). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.
  - Type: integer
  - Required: False
  - Minimum: -2147483648
  - Maximum: 2147483647

- **qualityTuningLevel**
  - Type: `H265QualityTuningLevel (p. 385)`
  - Required: False

- **maxBitrate**
  - Maximum bitrate in bits/second (for VBR mode only). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.
  - Type: integer
  - Required: False
  - Minimum: -2147483648
  - Maximum: 2147483647

- **bitrate**
  - Average bitrate in bits/second. Required for VBR, CBR, and ABR. Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.
  - Type: integer
  - Required: False
  - Minimum: 1000
  - Maximum: 2147483647

- **spatialAdaptiveQuantization**
  - Type: `H265SpatialAdaptiveQuantization (p. 391)`
Properties

Required: False

sampleAdaptiveOffsetFilterMode

Type: H265SampleAdaptiveOffsetFilterMode (p. 386)
Required: False

temporalIds

Type: H265TemporalIds (p. 391)
Required: False

slowPal

Type: H265SlowPal (p. 391)
Required: False

tiles

Type: H265Tiles (p. 392)
Required: False

codecProfile

Type: H265CodecProfile (p. 384)
Required: False

alternateTransferFunctionSei

Type: H265AlternateTransferFunctionSei (p. 383)
Required: False

unregisteredSeiTimecode

Type: H265UnregisteredSeiTimecode (p. 392)
Required: False

framerateControl

Type: H265FramerateControl (p. 384)
Required: False

telecine

Type: H265Telecine (p. 391)
Required: False

framerateConversionAlgorithm

Type: H265FramerateConversionAlgorithm (p. 384)
Required: False
codecLevel
Type: H265CodecLevel (p. 384)  
Required: False

numberReferenceFrames
Number of reference frames to use. The encoder may use more than requested if using B-frames and/or interlaced encoding.
Type: integer  
Required: False  
Minimum: 1  
Maximum: 6

temporalAdaptiveQuantization
Type: H265TemporalAdaptiveQuantization (p. 391)  
Required: False

hrdBufferInitialFillPercentage
Percentage of the buffer that should initially be filled (HRD buffer model).
Type: integer  
Required: False  
Minimum: 0  
Maximum: 100

framerateNumerator
Framerate numerator - framerate is a fraction, e.g. 24000 / 1001 = 23.976 fps.
Type: integer  
Required: False  
Minimum: 1  
Maximum: 2147483647

numberBFramesBetweenReferenceFrames
Number of B-frames between reference frames.
Type: integer  
Required: False  
Minimum: 0  
Maximum: 7

gopClosedCadence
Frequency of closed GOPs. In streaming applications, it is recommended that this be set to 1 so a decoder joining mid-stream will receive an IDR frame as quickly as possible. Setting this value to 0 will break output segmenting.
Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

framerateDenominator

Framerate denominator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

adaptiveQuantization

Type: H265AdaptiveQuantization (p. 383)
Required: False

interlaceMode

Type: H265InterlaceMode (p. 385)
Required: False

gopSize

GOP Length (keyframe interval) in frames or seconds. Must be greater than zero.

Type: number
Required: False
Format: float
Minimum: 0.0

gopBReference

Type: H265GopBReference (p. 385)
Required: False

sceneChangeDetect

Type: H265SceneChangeDetect (p. 386)
Required: False

parDenominator

Pixel Aspect Ratio denominator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647
parControl

- **Type:** H265ParControl (p. 385)
- **Required:** False

rateControlMode

- **Type:** H265RateControlMode (p. 386)
- **Required:** False

H265SlowPal (Enum)

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

- DISABLED
- ENABLED

H265SpatialAdaptiveQuantization (Enum)

Adjust quantization within each frame based on spatial variation of content complexity.

- DISABLED
- ENABLED

H265Telecine (Enum)

This field applies only if the Streams > Advanced > Framerate (framerate) field is set to 29.970. This field works with the Streams > Advanced > Preprocessors > Deinterlacer field (deinterlace_mode) and the Streams > Advanced > Interlaced Mode field (interlace_mode) to identify the scan type for the output: Progressive, Interlaced, Hard Telecine or Soft Telecine. - Hard: produces 29.97i output from 23.976 input. - Soft: produces 23.976; the player converts this output to 29.97i.

- NONE
- SOFT
- HARD

H265TemporalAdaptiveQuantization (Enum)

Adjust quantization within each frame based on temporal variation of content complexity.

- DISABLED
- ENABLED

H265TemporalIds (Enum)

Enables temporal layer identifiers in the encoded bitstream. Up to 3 layers are supported depending on GOP structure: I- and P-frames form one layer, reference B-frames can form a second layer and non-reference b-frames can form a third layer. Decoders can optionally decode only the lower temporal layers to generate a lower frame rate output. For example, given a bitstream with temporal IDs and with b-frames = 1 (i.e. IbPbPb display order), a decoder could decode all the frames for full frame rate output or only the I and P frames (lowest temporal layer) for a half frame rate output.

- DISABLED
- ENABLED
H265Tiles (Enum)
Enable use of tiles, allowing horizontal as well as vertical subdivision of the encoded pictures.

- DISABLED
- ENABLED

H265UnregisteredSeiTimecode (Enum)
Inserts timecode for each frame as 4 bytes of an unregistered SEI message.

- DISABLED
- ENABLED

Hdr10Metadata

redPrimaryY
HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

- Type: integer
- Required: False
- Minimum: 0
- Maximum: 50000

greenPrimaryY
HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

- Type: integer
- Required: False
- Minimum: 0
- Maximum: 50000

whitePointX
HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

- Type: integer
- Required: False
- Minimum: 0
- Maximum: 50000

maxLuminance
Nominal maximum mastering display luminance in units of 0.0001 candelas per square meter.

- Type: integer
- Required: False
- Minimum: 0
- Maximum: 2147483647
greenPrimaryX

HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

Type: integer  
Required: False  
Minimum: 0  
Maximum: 50000

whitePointY

HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

Type: integer  
Required: False  
Minimum: 0  
Maximum: 50000

redPrimaryX

HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

Type: integer  
Required: False  
Minimum: 0  
Maximum: 50000

bluePrimaryX

HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

Type: integer  
Required: False  
Minimum: 0  
Maximum: 50000

maxFrameAverageLightLevel

Maximum average light level of any frame in the coded video sequence, in units of candelas per square meter.

Type: integer  
Required: True  
Minimum: 0  
Maximum: 65535

bluePrimaryY

HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.
Properties

**Type**: integer
**Required**: False
**Minimum**: 0
**Maximum**: 50000

**maxContentLightLevel**

Maximum light level among all samples in the coded video sequence, in units of candelas per square meter.

**Type**: integer
**Required**: True
**Minimum**: 0
**Maximum**: 65535

**minLuminance**

Nominal minimum mastering display luminance in units of 0.0001 candelas per square meter

**Type**: integer
**Required**: False
**Minimum**: 0
**Maximum**: 2147483647

**HlsAdMarkers (Enum)**

- ELEMENTAL
- ELEMENTAL_SCTE35

**HlsAudioTrackType (Enum)**

- AUDIO_ONLY_VARIANT_STREAM
- ALTERNATE_AUDIO_AUTO_SELECT_DEFAULT
- ALTERNATE_AUDIO_AUTO_SELECT
- ALTERNATE_AUDIO_NOT_AUTO_SELECT
- AUDIO_ONLY_VARIANT_STREAM

**HlsCaptionLanguageMapping**

**languageDescription**

Caption language description.

**Type**: string
**Required**: False
captionChannel
Caption channel.
  Type: integer
  Required: False
  Minimum: -2147483648
  Maximum: 2147483647

languageCode
  Type: LanguageCode (p. 410)
  Required: False

HlsCaptionLanguageSetting (Enum)
Applies only to 608 Embedded output captions. Insert: Include CLOSED-CAPTIONS lines in the manifest. Specify at least one language in the CC1 Language Code field. One CLOSED-CAPTION line is added for each Language Code you specify. Make sure to specify the languages in the order in which they appear in the original source (if the source is embedded format) or the order of the caption selectors (if the source is other than embedded). Otherwise, languages in the manifest will not match up properly with the output captions. None: Include CLOSED-CAPTIONS=NONE line in the manifest. Omit: Omit any CLOSED-CAPTIONS line from the manifest.
  INSERT
  OMIT
  NONE

HlsClientCache (Enum)
When set to ENABLED, sets #EXT-X-ALLOW-CACHE:no tag, which prevents client from saving media segments for later replay.
  DISABLED
  ENABLED

HlsCodecSpecification (Enum)
Specification to use (RFC-6381 or the default RFC-4281) during m3u8 playlist generation.
  RFC_6381
  RFC_4281

HlsDirectoryStructure (Enum)
Indicates whether segments should be placed in subdirectories.
  SINGLE_DIRECTORY
  SUBDIRECTORY_PER_STREAM

HlsEncryptionSettings
initializationVectorInManifest
  Type: HlsInitializationVectorInManifest (p. 399)
**constantInitializationVector**

This is a 128-bit, 16-byte hex value represented by a 32-character text string. If this parameter is not set then the Initialization Vector will follow the segment number by default.

**Type:** string  
**Required:** False  
**Pattern:** `^[0-9a-fA-F]{32}$`

**staticKeyProvider**

**Type:** StaticKeyProvider (p. 445)  
**Required:** False

**type**

**Type:** HlsKeyProviderType (p. 399)  
**Required:** True

**spekeKeyProvider**

**Type:** SpekeKeyProvider (p. 445)  
**Required:** False

**encryptionMethod**

**Type:** HlsEncryptionType (p. 396)  
**Required:** False

**HlsEncryptionType (Enum)**

Encrypts the segments with the given encryption scheme. Leave blank to disable. Selecting 'Disabled' in the web interface also disables encryption.

- AES128
- SAMPLE_AES

**HlsGroupSettings**

**segmentsPerSubdirectory**

Number of segments to write to a subdirectory before starting a new one. directoryStructure must be SINGLE_DIRECTORY for this setting to have an effect.

**Type:** integer  
**Required:** False  
**Minimum:** 1  
**Maximum:** 2147483647
**streamInfResolution**

Type: `HlsStreamInfResolution (p. 401)`
Required: False

**timestampDeltaMilliseconds**

Provides an extra millisecond delta offset to fine tune the timestamps.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

**outputSelection**

Type: `HlsOutputSelection (p. 400)`
Required: False

**captionLanguageMappings**

Type: Array of type `HlsCaptionLanguageMapping (p. 394)`
Required: False

**clientCache**

Type: `HlsClientCache (p. 395)`
Required: False

**codecSpecification**

Type: `HlsCodecSpecification (p. 395)`
Required: False

**destination**

Use Destination (Destination) to specify the S3 output location and the output filename base. Destination accepts format identifiers. If you do not specify the base filename in the URI, the service will use the filename of the input file. If your job has multiple inputs, the service uses the filename of the first input file.

Type: string
Required: True
Pattern: `s3://`

**segmentControl**

Type: `HlsSegmentControl (p. 400)`
Required: False

**timedMetadataId3Frame**

Type: `HlsTimedMetadataId3Frame (p. 401)`
**timedMetadataId3Period**
Timed Metadata interval in seconds.

**Type:** integer  
**Required:** False  
**Minimum:** -2147483648  
**Maximum:** 2147483647

**captionLanguageSetting**

**Type:** HlsCaptionLanguageSetting (p. 395)  
**Required:** False

**minSegmentLength**
When set, Minimum Segment Size is enforced by looking ahead and back within the specified range for a nearby avail and extending the segment size if needed.

**Type:** integer  
**Required:** True  
**Minimum:** 0  
**Maximum:** 2147483647

**directoryStructure**

**Type:** HlsDirectoryStructure (p. 395)  
**Required:** False

**programDateTime**

**Type:** HlsProgramDateTime (p. 400)  
**Required:** False

**baseUrl**
A partial URI prefix that will be prepended to each output in the media .m3u8 file. Can be used if base manifest is delivered from a different URL than the main .m3u8 file.

**Type:** string  
**Required:** False

**encryption**
DRM settings.

**Type:** HlsEncryptionSettings (p. 395)  
**Required:** False

**adMarkers**
Choose one or more ad marker types to pass SCTE35 signals through to this group of Apple HLS outputs.
Type: Array of type HlsAdMarkers (p. 394)
Required: False

programDateTimePeriod

Period of insertion of EXT-X-PROGRAM-DATE-TIME entry, in seconds.
Type: integer
Required: False
Minimum: 0
Maximum: 3600

manifestCompression

Type: HlsManifestCompression (p. 400)
Required: False

segmentLength

Length of MPEG-2 Transport Stream segments to create (in seconds). Note that segments will end on the next keyframe after this number of seconds, so actual segment length may be longer.
Type: integer
Required: True
Minimum: 1
Maximum: 2147483647

manifestDurationFormat

Type: HlsManifestDurationFormat (p. 400)
Required: False

HlsIFrameOnlyManifest (Enum)

When set to INCLUDE, writes I-Frame Only Manifest in addition to the HLS manifest

INCLUDE
EXCLUDE

HlsInitializationVectorInManifest (Enum)

The Initialization Vector is a 128-bit number used in conjunction with the key for encrypting blocks. If set to INCLUDE, Initialization Vector is listed in the manifest. Otherwise Initialization Vector is not in the manifest.

INCLUDE
EXCLUDE

HlsKeyProviderType (Enum)

Indicates which type of key provider is used for encryption.

SPEKE
STATIC_KEY

**HlsManifestCompression (Enum)**
When set to GZIP, compresses HLS playlist.

```
GZIP
NONE
```

**HlsManifestDurationFormat (Enum)**
Indicates whether the output manifest should use floating point values for segment duration.

```
FLOATING_POINT
INTEGER
```

**HlsOutputSelection (Enum)**
Indicates whether the .m3u8 manifest file should be generated for this HLS output group.

```
MANIFESTS_AND_SEGMENTS
SEGMENTS_ONLY
```

**HlsProgramDateTime (Enum)**
Includes or excludes EXT-X-PROGRAM-DATE-TIME tag in .m3u8 manifest files. The value is calculated as follows: either the program date and time are initialized using the input timecode source, or the time is initialized using the input timecode source and the date is initialized using the timestamp_offset.

```
INCLUDE
EXCLUDE
```

**HlsSegmentControl (Enum)**
When set to SINGLE_FILE, emits program as a single media resource (.ts) file, uses #EXT-X-BYTERANGE tags to index segment for playback.

```
SINGLE_FILE
SEGMENTED_FILES
```

**HlsSettings**

**iFrameOnlyManifest**

```
Type: HlsIFrameOnlyManifest (p. 399)
Required: False
```

**segmentModifier**
String concatenated to end of segment filenames. Accepts "Format Identifiers":#format_identifier_parameters.

```
Type: string
Required: False
```
audioTrackType

Type: HlsAudioTrackType (p. 394)
Required: False

audioRenditionSets

List all the audio groups that are used with the video output stream. Input all the audio GROUP-IDs that are associated to the video, separate by ",".

Type: string
Required: False

audioGroupId

Specifies the group to which the audio Rendition belongs.

Type: string
Required: False

HlsStreamInfResolution (Enum)

Include or exclude RESOLUTION attribute for video in EXT-X-STREAM-INF tag of variant manifest.

- INCLUDE
- EXCLUDE

HlsTimedMetadataId3Frame (Enum)

Indicates ID3 frame that has the timecode.

- NONE
- PRIV
- TDRL

Id3Insertion

id3

Use ID3 tag (Id3) to provide a tag value in base64-encode format.

Type: string
Required: True
Pattern: ^[A-Za-z0-9+/]{0,2}#

timecode

Provide a Timecode (TimeCode) in HH:MM:SS:FF or HH:MM:SS;FF format.

Type: string
Required: True
Format: timecode
Pattern: ^([01][0-9]|2[0-4]):[0-5][0-9]:[0-5][0-9]:[0-9]{2}#$
ImageInserter

insertableImages

Image to insert. Must be 32 bit windows BMP, PNG, or TGA file. Must not be larger than the output frames.

Type: Array of type InsertableImage (p. 405)
Required: True

Input

audioSelectors

Use Audio selectors (AudioSelectors) to specify a track or set of tracks from the input that you will use in your outputs. You can use multiple Audio selectors per input.

Type: object
Required: False

audioSelectorGroups

Specifies set of audio selectors within an input to combine. An input may have multiple audio selector groups. See "Audio Selector Group":#inputs-audio_selector_group for more information.

Type: object
Required: False

videoSelector

Type: VideoSelector (p. 453)
Required: False

denoiseFilter

Type: InputDenoiseFilter (p. 404)
Required: False

filterEnable

Type: InputFilterEnable (p. 404)
Required: False

deblockFilter

Type: InputDeblockFilter (p. 404)
Required: False

filterStrength

Use Filter strength (FilterStrength) to adjust the magnitude the input filter settings (Deblock and Denoise). The range is -5 to 5. Default is 0.
**Properties**

**Type**: integer  
**Required**: False  
**Minimum**: -5  
**Maximum**: 5

**programNumber**

Use Program (programNumber) to select a specific program from within a multi-program transport stream. Note that Quad 4K is not currently supported. Default is the first program within the transport stream. If the program you specify doesn’t exist, the transcoding service will use this default.

**Type**: integer  
**Required**: False  
**Minimum**: -2147483648  
**Maximum**: 2147483647

**timecodeSource**

**Type**: InputTimecodeSource (p. 405)  
**Required**: False

**captionSelectors**

Use Captions selectors (CaptionSelectors) to specify the captions data from the input that you will use in your outputs. You can use multiple captions selectors per input.

**Type**: object  
**Required**: False

**fileInput**

Use Input (fileInput) to define the source file used in the transcode job. There can be multiple inputs in a job. These inputs are concatenated, in the order they are specified in the job, to create the output.

**Type**: string  
**Required**: True  
**Pattern**: ^(s3://[^/]+/[^/]+)+([^/]+[mM]2[vV][mM][pP][eE][gG][aA][Vv][iI][mM][pP]4[EF][Ll][Vv][mM][pP][tT][mM][pP][gG][mM][4[vV]][tT][Rr][pP][EF][vV][mM][2[tT][sS][tT][sS][264[hH]264[mM][kK][Vv]][mM][Oo][Vv]][mM][tT][sS][mM][2[tT][wW][mM][Vv]][aA][sS][fF][vV][oO][bB][3[gG][pP]][mM][Xx][fF][dD][iI][Vv][xX][xX][vV][iI][dd][rR][aA][wW][dD][vV][gG][xX][fF][mM][lVv][3[gG]2[vV][mM][fF][mM][3[uU][8[lL][cC][hH]][gG][xX][fF][mM][pP][eE][gG]2[mM][xX][fF][mM][pP][eE][gG][2[mM][xX][fF][hh][dD][wW][aA][Vv][yY][4[mM]]])$

**inputClippings**

(InputClippings) contains sets of start and end times that together specify a portion of the input to be used in the outputs. If you provide only a start time, the clip will be the entire input from that point to the end. If you provide only an end time, it will be the entire input up to that point. When you specify more than one input clip, the transcoding service creates the job outputs by stringing the clips together in the order you specify them.

**Type**: Array of type InputClipping (p. 404)  
**Required**: False
psiControl

Type: InputPsiControl (p. 405)
Required: False

InputClipping

startTimecode

Set Start timecode (StartTimecode) to the beginning of the portion of the input you are clipping. The frame corresponding to the Start timecode value is included in the clip. Start timecode or End timecode may be left blank, but not both. When choosing this value, take into account your setting for Input timecode source. For example, if you have embedded timecodes that start at 01:00:00:00 and you want your clip to begin five minutes into the video, use 01:00:05:00.

Type: string
Required: False
Format: timecode
Pattern: ^([01][0-9]|2[0-4]):[0-5][0-9]:[0-5][0-9][;][0-9]{2}$

dendTimecode

Set End timecode (EndTimecode) to the end of the portion of the input you are clipping. The frame corresponding to the End timecode value is included in the clip. Start timecode or End timecode may be left blank, but not both. When choosing this value, take into account your setting for Input timecode source. For example, if you have embedded timecodes that start at 01:00:00:00 and you want your clip to begin five minutes into the video, use 01:00:05:00.

Type: string
Required: False
Format: timecode
Pattern: ^([01][0-9]|2[0-4]):[0-5][0-9]:[0-5][0-9][;][0-9]{2}$

InputDeblockFilter (Enum)

Enable Deblock (InputDeblockFilter) to produce smoother motion in the output. Default is disabled. Only manually controllable for MPEG2 and uncompressed video inputs.

ENABLED
DISABLED

InputDenoiseFilter (Enum)

Enable Denoise (InputDenoiseFilter) to filter noise from the input. Default is disabled. Only applicable to MPEG2, H.264, H.265, and uncompressed video inputs.

ENABLED
DISABLED

InputFilterEnable (Enum)

Use Filter enable (InputFilterEnable) to specify how the transcoding service applies the denoise and deblock filters. You must also enable the filters separately, with Denoise (InputDenoiseFilter) and Debloack (InputDeblockFilter).
* Auto - The transcoding service determines whether to apply filtering, depending on input type and quality.

* Disable - The input is not filtered. This is true even if you use the API to enable them in (InputDeblockFilter) and (InputDeblockFilter).

* Force - The input is filtered regardless of input type.

  AUTO
  DISABLE
  FORCE

**InputPsiControl (Enum)**

Set PSI control (InputPsiControl) for transport stream inputs to specify which data the demux process to scans.

* Ignore PSI - Scan all PIDs for audio and video.

* Use PSI - Scan only PSI data.

  IGNORE_PSI
  USE_PSI

**InputTimecodeSource (Enum)**

Use Timecode source (InputTimecodeSource) to specify how timecode information from your input is adjusted and encoded in all outputs for the job. Default is embedded. Set to Embedded (EMBEDDED) to use the timecode that is in the input video. If no embedded timecode is in the source, will set the timecode for the first frame to 00:00:00:00. Set to Start at 0 (ZEROBASED) to set the timecode of the initial frame to 00:00:00:00. Set to Specified start (SPECIFIEDSTART) to provide the initial timecode yourself the setting (Start).

  EMBEDDED
  ZEROBASED
  SPECIFIEDSTART

**InsertableImage**

**duration**

Use Duration (Duration) to set the time, in milliseconds, for the image to remain on the output video.

  Type: integer
  Required: False
  Minimum: -2147483648
  Maximum: 2147483647

**fadeOut**

Use Fade out (FadeOut) to set the length, in milliseconds, of the inserted image fade out. If you don't specify a value for Fade out, the image will disappear abruptly at the end of the inserted image duration.

  Type: integer
  Required: False
  Minimum: -2147483648
Maximum: 2147483647

imageY

Use Top (ImageY) to set the distance, in pixels, between the inserted image and the top edge of the video frame. Required for BMP, PNG and TGA input.

Type: integer
Required: True
Minimum: -2147483648
Maximum: 2147483647

fadeIn

Use Fade in (FadeIn) to set the length, in milliseconds, of the inserted image fade in. If you don't specify a value for Fade in, the image will appear abruptly at the Start time.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

imageX

Use Left (ImageX) to set the distance, in pixels, between the inserted image and the left edge of the frame. Required for BMP, PNG and TGA input.

Type: integer
Required: True
Minimum: -2147483648
Maximum: 2147483647

width

Specify the Width (Width) of the inserted image. Use a value that is less than or equal to the video resolution width. Leave this setting blank to use the native width of the image.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

startTime

Use Start time (StartTime) to specify the video timecode when the image is inserted in the output. This must be in timecode format (HH:MM:SS:FF)

Type: string
Required: False

opacity

Use Opacity (Opacity) to specify how much of the underlying video shows through the inserted image. 0 is transparent and 100 is fully opaque. Default is 50.
**Type**: integer  
**Required**: True  
**Minimum**: 0  
**Maximum**: 100

### layer

Use `Layer` to specify how overlapping inserted images appear. Images with higher values of `layer` appear on top of images with lower values of `layer`.

**Type**: integer  
**Required**: True  
**Minimum**: 0  
**Maximum**: 7

### height

Specify the `Height` of the inserted image. Use a value that is less than or equal to the video resolution height. Leave this setting blank to use the native height of the image.

**Type**: integer  
**Required**: False  
**Minimum**: -2147483648  
**Maximum**: 2147483647

### imageInserterInput

Use `Image location (imageInserterInput)` to specify the Amazon S3 location of the image to be inserted into the output. Use a 32 bit BMP, PNG, or TGA file that fits inside the video frame.

**Type**: string  
**Required**: True  
**Pattern**: ^s3:/\/(.*?).(bmp|BMP|png|PNG|tga|TGA)$

### Job

#### settings

**Type**: JobSettings (p. 409)  
**Required**: True

### outputGroupDetails

**Type**: Array of type OutputGroupDetail (p. 439)  
**Required**: False

### role

The IAM role you use for creating this job. For details about permissions, see the User Guide topic at the User Guide at http://docs.aws.amazon.com/mediaconvert/latest/ug/iam-role.html

**Type**: string
Required: True

**jobTemplate**

The job template that the job is created from, if it is created from a job template.

Type: string
Required: False

**timing**

Type: Timing (p. 449)
Required: False

**errorMessage**

Error message of job

Type: string
Required: False

**errorCode**

Type: integer
Required: False
Format: int32

**createdAt**

The time, in Unix epoch format in seconds, when the job got created.

Type: string
Required: False
Format: date-time

**id**

A portion of the job's ARN, unique within your AWS Elemental MediaConvert resources

Type: string
Required: False

**arn**

An identifier for this resource that is unique within all of AWS.

Type: string
Required: False

**userMetadata**

Type: object
**queue**

Optional. When you create a job, you can specify a queue to send it to. If you don’t specify, the job will go to the default queue. For more about queues, see the User Guide topic at http://docs.aws.amazon.com/mediaconvert/latest/ug/what-is.html

- **Type:** string
- **Required:** False

**status**

- **Type:** JobStatus (p. 410)
- **Required:** False

**JobSettings**

**timecodeConfig**

Contains settings used to acquire and adjust timecode information from inputs.

- **Type:** TimecodeConfig (p. 447)
- **Required:** False

**adAvailOffset**

When specified, this offset (in milliseconds) is added to the input Ad Avail PTS time.

- **Type:** integer
- **Required:** False
- **Minimum:** -1000
- **Maximum:** 1000

**nielsenConfiguration**

Nielsen configuration settings

- **Type:** NielsenConfiguration (p. 435)
- **Required:** False

**inputs**

Use Inputs (inputs) to define source file used in the transcode job. There can be multiple inputs added to a job. These inputs will be concatenated together to create the output.

- **Type:** Array of type Input (p. 402)
- **Required:** True

**outputGroups**

(OutputGroups) contains one group of settings for each set of outputs that share a common package type. All unpackaged files (MPEG-4, MPEG-2 TS, Quicktime, MXF, and no container) are
grouped in a single output group as well. Required in (OutputGroups) is a group of settings that apply to the whole group. This required object depends on the value you set for (Type) under (OutputGroups)>(OutputGroupSettings). Type, settings object pairs are as follows.

* FILE_GROUP_SETTINGS, FileGroupSettings
* HLS_GROUP_SETTINGS, HlsGroupSettings
* DASH_ISO_GROUP_SETTINGS, DashIsoGroupSettings
* MS_SMOOTH_GROUP_SETTINGS, MsSmoothGroupSettings

  Type: Array of type OutputGroup (p. 438)
  Required: True

**timedMetadataInsertion**

  Type: TimedMetadataInsertion (p. 449)
  Required: False

**availBlanking**

Settings for ad avail blanking. Video can be blanked or overlaid with an image, and audio muted during SCTE-35 triggered ad avails.

  Type: AvailBlanking (p. 348)
  Required: False

**JobStatus (Enum)**

A job's status can be SUBMITTED, PROGRESSING, COMPLETE, CANCELED, or ERROR.

  SUBMITTED
  PROGRESSING
  COMPLETE
  CANCELED
  ERROR

**LanguageCode (Enum)**

Selects a specific language code from within an audio source.

  ENG
  SPA
  FRA
  DEU
  GER
  ZHO
  ARA
  HIN
  JPN
  RUS
  POR
  ITA
  URD
| VIE | KOR | PAN | ABK | AAR | AFR | AKA | SQI | AMH | ARG | HYE | ASM | AVA | AVE | AYM | AZE | BAM | BAK | EUS | BEL | BEN | BIH | BIS | BOS | BRE | BUL | MYA | CAT | KHM | CHA | CHE | NYA | CHU | CHV | CHV | COR | COS | CRE | HRV | CES | DAN | DIV | NLD | DZO | ENM | EPO | EST | EWE | FAO | FIJ | FIN | FRM | FUL | GLA |
GLG
LUG
KAT
ELL
GRN
GUJ
HAT
HAU
HEB
HER
HMO
HUN
ISL
IDO
IBO
IND
INA
ILE
IKU
IPK
GLE
JAV
KAL
KAN
KAU
KAS
KAZ
KIK
KIN
KIR
KOM
KON
KUA
KUR
LAO
LAT
LAV
LIM
LIN
LIT
LUB
LTZ
MKD
MLG
MSA
MAL
MLT
GLV
MRI
MAR
MAH
MON
NAU
**ListJobsRequest**

**nextToken**

Use this string, provided with the response to a previous request, to request the next batch of jobs.

- **Type**: string
  - **Required**: False

**maxResults**

Optional. Number of jobs, up to twenty, that will be returned at one time.

- **Type**: integer
  - **Required**: False
  - **Format**: int32

**queue**

Provide a queue name to get back only jobs from that queue.

- **Type**: string
  - **Required**: False

**status**

- **Type**: JobStatus (p. 410)
  - **Required**: False

**order**

- **Type**: Order (p. 436)
  - **Required**: False
ListJobsResponse

nextToken

Use this string to request the next batch of jobs.

Type: string
Required: False

jobs

Type: Array of type Job (p. 407)
Required: False

M2tsAudioBufferModel (Enum)

Selects between the DVB and ATSC buffer models for Dolby Digital audio.

- DVB
- ATSC

M2tsBufferModel (Enum)

Controls what buffer model to use for accurate interleaving. If set to MULTIPLEX, use multiplex buffer model. If set to NONE, this can lead to lower latency, but low-memory devices may not be able to play back the stream without interruptions.

- MULTIPLEX
- NONE

M2tsEbpAudioInterval (Enum)

When set to VIDEO_AND_FIXED_INTERVALS, audio EBP markers will be added to partitions 3 and 4. The interval between these additional markers will be fixed, and will be slightly shorter than the video EBP marker interval. When set to VIDEO_INTERVAL, these additional markers will not be inserted. Only applicable when EBP segmentation markers are is selected (segmentationMarkers is EBP or EBP_LEGACY).

- VIDEO_AND_FIXED_INTERVALS
- VIDEO_INTERVAL

M2tsEbpPlacement (Enum)

Selects which PIDs to place EBP markers on. They can either be placed only on the video PID, or on both the video PID and all audio PIDs. Only applicable when EBP segmentation markers are is selected (segmentationMarkers is EBP or EBP_LEGACY).

- VIDEO_AND_AUDIO_PIDS
- VIDEO_PID

M2tsEsRateInPes (Enum)

Controls whether to include the ES Rate field in the PES header.
**M2tsPcrControl (Enum)**

When set to PCR_EVERY_PES_PACKET, a Program Clock Reference value is inserted for every Packetized Elementary Stream (PES) header. This is effective only when the PCR PID is the same as the video or audio elementary stream.

- PCR_EVERY_PES_PACKET
- CONFIGURED_PCR_PERIOD

**M2tsRateMode (Enum)**

When set to CBR, inserts null packets into transport stream to fill specified bitrate. When set to VBR, the bitrate setting acts as the maximum bitrate, but the output will not be padded up to that bitrate.

- VBR
- CBR

**M2tsScte35Source (Enum)**

Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output. This is only available for certain containers.

- PASSTHROUGH
- NONE

**M2tsSegmentationMarkers (Enum)**

Inserts segmentation markers at each segmentation_time period. rai_segstart sets the Random Access Indicator bit in the adaptation field. rai_adapt sets the RAI bit and adds the current timecode in the private data bytes. psi_segstart inserts PAT and PMT tables at the start of segments. ebp adds Encoder Boundary Point information to the adaptation field as per OpenCable specification OC-SP-EBP-I01-130118. ebp_legacy adds Encoder Boundary Point information to the adaptation field using a legacy proprietary format.

- NONE
- RAI_SEGSTART
- RAI_ADAPT
- PSI_SEGSTART
- EBP
- EBP_LEGACY

**M2tsSegmentationStyle (Enum)**

The segmentation style parameter controls how segmentation markers are inserted into the transport stream. With avails, it is possible that segments may be truncated, which can influence where future segmentation markers are inserted. When a segmentation style of "reset_cadence" is selected and a segment is truncated due to an avail, we will reset the segmentation cadence. This means the subsequent segment will have a duration of $segmentation_time seconds. When a segmentation style of "maintain_cadence" is selected and a segment is truncated due to an avail, we will not reset the segmentation cadence. This means the subsequent segment will likely be truncated as well. However, all segments after that will have a duration of $segmentation_time seconds. Note that EBP lookahead is a slight exception to this rule.
MAINTAIN_CADENCE
RESET_CADENCE

**M2tsSettings**

**dvbTeletextPid**
Packet Identifier (PID) for input source DVB Teletext data to this output. Can be entered as a decimal or hexadecimal value.

- **Type**: integer
- **Required**: False
- **Minimum**: 32
- **Maximum**: 8182

**bitrate**
The output bitrate of the transport stream in bits per second. Setting to 0 lets the muxer automatically determine the appropriate bitrate. Other common values are 3750000, 7500000, and 15000000.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 2147483647

**segmentationTime**
The length in seconds of each segment. Required unless markers is set to _none_.

- **Type**: number
- **Required**: False
- **Format**: float
- **Minimum**: 0.0

**audioPids**
Packet Identifier (PID) of the elementary audio stream(s) in the transport stream. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values.

- **Type**: Array of type integer
- **Required**: False

**rateMode**

- **Type**: M2tsRateMode (p. 416)
- **Required**: False

**ebpAudioInterval**

- **Type**: M2tsEbpAudioInterval (p. 415)
- **Required**: False
fragmentTime
The length in seconds of each fragment. Only used with EBP markers.

    Type: number
    Required: False
    Format: float
    Minimum: 0.0

audioFramesPerPes
The number of audio frames to insert for each PES packet.

    Type: integer
    Required: False
    Minimum: 0
    Maximum: 2147483647

maxPcrInterval
Maximum time in milliseconds between Program Clock References (PCRs) inserted into the transport stream.

    Type: integer
    Required: False
    Minimum: 0
    Maximum: 500

scte35Pid
Packet Identifier (PID) of the SCTE-35 stream in the transport stream. Can be entered as a decimal or hexadecimal value.

    Type: integer
    Required: False
    Minimum: 32
    Maximum: 8182

privateMetadataPid
Packet Identifier (PID) of the private metadata stream in the transport stream. Can be entered as a decimal or hexadecimal value.

    Type: integer
    Required: False
    Minimum: 32
    Maximum: 8182

pmtInterval
The number of milliseconds between instances of this table in the output transport stream.

    Type: integer
    Required: False
    Minimum: 0
    Maximum: 1000
**segmentationStyle**

Type: M2tsSegmentationStyle (p. 416)
Required: False

**audioBufferModel**

Type: M2tsAudioBufferModel (p. 415)
Required: False

**programNumber**

The value of the program number field in the Program Map Table.

Type: integer
Required: False
Minimum: 0
Maximum: 65535

**dvbNitSettings**

Type: DvbNitSettings (p. 361)
Required: False

**scte35Source**

Type: M2tsScte35Source (p. 416)
Required: False

**pmtPid**

Packet Identifier (PID) for the Program Map Table (PMT) in the transport stream. Can be entered as a decimal or hexadecimal value.

Type: integer
Required: False
Minimum: 32
Maximum: 8182

**bufferModel**

Type: M2tsBufferModel (p. 415)
Required: False

**ebpPlacement**

Type: M2tsEbpPlacement (p. 415)
Required: False

**dvbSdtSettings**

Type: DvbSdtSettings (p. 362)
nullPacketBitrate

Value in bits per second of extra null packets to insert into the transport stream. This can be used if a downstream encryption system requires periodic null packets.

  - **Type**: number
  - **Required**: False
  - **Format**: float
  - **Minimum**: 0.0

pcrPid

Packet Identifier (PID) of the Program Clock Reference (PCR) in the transport stream. When no value is given, the encoder will assign the same value as the Video PID. Can be entered as a decimal or hexadecimal value.

  - **Type**: integer
  - **Required**: False
  - **Minimum**: 32
  - **Maximum**: 8182

minEbpInterval

When set, enforces that Encoder Boundary Points do not come within the specified time interval of each other by looking ahead at input video. If another EBP is going to come in within the specified time interval, the current EBP is not emitted, and the segment is "stretched" to the next marker. The lookahead value does not add latency to the system. The Live Event must be configured elsewhere to create sufficient latency to make the lookahead accurate.

  - **Type**: integer
  - **Required**: False
  - **Minimum**: 0
  - **Maximum**: 10000

transportStreamId

The value of the transport stream ID field in the Program Map Table.

  - **Type**: integer
  - **Required**: False
  - **Minimum**: 0
  - **Maximum**: 65535

pcrControl

  - **Type**: M2tsPcrControl (p. 416)
  - **Required**: False

videoPid

Packet Identifier (PID) of the elementary video stream in the transport stream. Can be entered as a decimal or hexadecimal value.
**esRateInPes**

Type: M2tsEsRateInPes (p. 415)
Required: False

**segmentationMarkers**

Type: M2tsSegmentationMarkers (p. 416)
Required: False

**dvbTdtSettings**

Type: DvbTdtSettings (p. 367)
Required: False

**patInterval**

The number of milliseconds between instances of this table in the output transport stream.

Type: integer
Required: False
Minimum: 0
Maximum: 1000

**dvbSubPids**

Packet Identifier (PID) for input source DVB Subtitle data to this output. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values.

Type: Array of type integer
Required: False

**M3u8PcrControl (Enum)**

When set to PCR_EVERY_PES_PACKET a Program Clock Reference value is inserted for every Packetized Elementary Stream (PES) header. This parameter is effective only when the PCR PID is the same as the video or audio elementary stream.

- PCR_EVERY_PES_PACKET
- CONFIGURED_PCR_PERIOD

**M3u8Scte35Source (Enum)**

Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output. This is only available for certain containers.

- PASSTHROUGH
- NONE
M3u8Settings

pmtPid
Packet Identifier (PID) for the Program Map Table (PMT) in the transport stream. Can be entered as a decimal or hexadecimal value.

  Type: integer
  Required: False
  Minimum: 32
  Maximum: 8182

pcrPid
Packet Identifier (PID) of the Program Clock Reference (PCR) in the transport stream. When no value is given, the encoder will assign the same value as the Video PID. Can be entered as a decimal or hexadecimal value.

  Type: integer
  Required: False
  Minimum: 32
  Maximum: 8182

audioPids
Packet Identifier (PID) of the elementary audio stream(s) in the transport stream. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values.

  Type: Array of type integer
  Required: False

audioFramesPerPes
The number of audio frames to insert for each PES packet.

  Type: integer
  Required: False
  Minimum: 0
  Maximum: 2147483647

scte35Pid
Packet Identifier (PID) of the SCTE-35 stream in the transport stream. Can be entered as a decimal or hexadecimal value.

  Type: integer
  Required: False
  Minimum: 32
  Maximum: 8182

transportStreamId
The value of the transport stream ID field in the Program Map Table.
**videoPid**
Packet Identifier (PID) of the elementary video stream in the transport stream. Can be entered as a decimal or hexadecimal value.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 65535

**pcrControl**
- **Type:** `M3u8PcrControl` (p. 421)
- **Required:** False

**privateMetadataPid**
Packet Identifier (PID) of the private metadata stream in the transport stream. Can be entered as a decimal or hexadecimal value.

- **Type:** integer
- **Required:** False
- **Minimum:** 32
- **Maximum:** 8182

**pmtInterval**
The number of milliseconds between instances of this table in the output transport stream.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 1000

**patInterval**
The number of milliseconds between instances of this table in the output transport stream.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 1000

**programNumber**
The value of the program number field in the Program Map Table.

- **Type:** integer
- **Required:** False
Properties

Minimum: 0
Maximum: 65535

timedMetadataPid
Packet Identifier (PID) of the timed metadata stream in the transport stream. Can be entered as a
decimal or hexadecimal value.

Type: integer
Required: False
Minimum: 32
Maximum: 8182

timedMetadata

Type: TimedMetadata (p. 448)
Required: False

scte35Source

Type: M3u8Scte35Source (p. 421)
Required: False

MovClapAtom (Enum)

When enabled, include 'clap' atom if appropriate for the video output settings.

 INCLUDE
 EXCLUDE

MovCslgAtom (Enum)

When enabled, file composition times will start at zero, composition times in the 'ctts' (composition time
to sample) box for B-frames will be negative, and a 'cslg' (composition shift least greatest) box will be
included per 14496-1 amendment 1. This improves compatibility with Apple players and tools.

 INCLUDE
 EXCLUDE

MovMpeg2FourCCControl (Enum)

When set to XDCAM, writes MPEG2 video streams into the QuickTime file using XDCAM fourcc codes.
This increases compatibility with Apple editors and players, but may decrease compatibility with other
players. Only applicable when the video codec is MPEG2.

 XDCAM
 MPEG

MovPaddingControl (Enum)

If set to OMNEON, inserts Omneon-compatible padding

 OMNEON
 NONE
**MovReference (Enum)**

A value of 'external' creates separate media files and the wrapper file (.mov) contains references to these media files. A value of 'self_contained' creates only a wrapper (.mov) file and this file contains all of the media.

SELF_CONTAINED
EXTERNAL

**MovSettings**

**reference**

_Type_: MovReference (p. 425)  
_Required_: False

**paddingControl**

_Type_: MovPaddingControl (p. 424)  
_Required_: False

**mpeg2FourCCControl**

_Type_: MovMpeg2FourCCControl (p. 424)  
_Required_: False

**cslgAtom**

_Type_: MovCslgAtom (p. 424)  
_Required_: False

**clapAtom**

_Type_: MovClapAtom (p. 424)  
_Required_: False

**Mp2Settings**

**channels**

Set Channels to specify the number of channels in this output audio track. Choosing Mono in the console will give you 1 output channel; choosing Stereo will give you 2. In the API, valid values are 1 and 2.

_Type_: integer  
_Required_: False  
_Minimum_: 1  
_Maximum_: 2

**bitrate**

Average bitrate in bits/second.

_Type_: integer
Properties

**sampleRate**
Sample rate in hz.

- **Type:** integer
- **Required:** False
- **Minimum:** 32000
- **Maximum:** 48000

**Mp4CslgAtom (Enum)**
When enabled, file composition times will start at zero, composition times in the 'ctts' (composition time to sample) box for B-frames will be negative, and a 'cslg' (composition shift least greatest) box will be included per 14496-1 amendment 1. This improves compatibility with Apple players and tools.

- **INCLUDE**
- **EXCLUDE**

**Mp4FreeSpaceBox (Enum)**
Inserts a free-space box immediately after the moov box.

- **INCLUDE**
- **EXCLUDE**

**Mp4MoovPlacement (Enum)**
If set to PROGRESSIVE_DOWNLOAD, the MOOV atom is relocated to the beginning of the archive as required for progressive downloading. Otherwise it is placed normally at the end.

- **PROGRESSIVE_DOWNLOAD**
- **NORMAL**

**Mp4Settings**

**mp4MajorBrand**
Overrides the "Major Brand" field in the output file. Usually not necessary to specify.

- **Type:** string
- **Required:** False

**moovPlacement**

- **Type:** Mp4MoovPlacement (p. 426)
- **Required:** False

**cslgAtom**

- **Type:** Mp4CslgAtom (p. 426)
**Properties**

**Required:** False

**freeSpaceBox**

**Type:** Mp4FreeSpaceBox (p. 426)  
**Required:** False

**Mpeg2AdaptiveQuantization (Enum)**

Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

- OFF
- LOW
- MEDIUM
- HIGH

**Mpeg2CodecLevel (Enum)**

Use Level (Mpeg2CodecLevel) to set the MPEG-2 level for the video output.

- AUTO
- LOW
- MAIN
- HIGH
- HIGH1440

**Mpeg2CodecProfile (Enum)**

Use Profile (Mpeg2CodecProfile) to set the MPEG-2 profile for the video output.

- MAIN
- PROFILE_422

**Mpeg2FramerateControl (Enum)**

Using the API, set FramerateControl to INITIALIZE_FROM_SOURCE if you want the service to use the framerate from the input. Using the console, do this by choosing INITIALIZE_FROM_SOURCE for Framerate.

- INITIALIZE_FROM_SOURCE
- SPECIFIED

**Mpeg2FramerateConversionAlgorithm (Enum)**

When set to INTERPOLATE, produces smoother motion during framerate conversion.

- DUPLICATE_DROP
- INTERPOLATE

**Mpeg2GopSizeUnits (Enum)**

Indicates if the GOP Size in MPEG2 is specified in frames or seconds. If seconds the system will convert the GOP Size into a frame count at run time.
FRAMES
SECONDS

Mpeg2InterlaceMode (Enum)
Use Interlace mode (InterlaceMode) to choose the scan line type for the output.

* Top Field First (TOP_FIELD) and Bottom Field First (BOTTOM_FIELD) produce interlaced output with the entire output having the same field polarity (top or bottom first).

* Follow, Default Top (FOLLOW_TOP_FIELD) and Follow, Default Bottom (FOLLOW_BOTTOM_FIELD) use the same field polarity as the source. Therefore, behavior depends on the input scan type. If the source is interlaced, the output will be interlaced with the same polarity as the source (it will follow the source). The output could therefore be a mix of "top field first" and "bottom field first". If the source is progressive, the output will be interlaced with "top field first" or "bottom field first" polarity, depending on which of the Follow options you chose.

PROGRESSIVE
TOP_FIELD
BOTTOM_FIELD
FOLLOW_TOP_FIELD
FOLLOW_BOTTOM_FIELD

Mpeg2IntraDcPrecision (Enum)
Use Intra DC precision (Mpeg2IntraDcPrecision) to set quantization precision for intra-block DC coefficients. If you choose the value auto, the service will automatically select the precision based on the per-frame compression ratio.

AUTO
INTRA_DC_PRECISION_8
INTRA_DC_PRECISION_9
INTRA_DC_PRECISION_10
INTRA_DC_PRECISION_11

Mpeg2ParControl (Enum)
Using the API, enable ParFollowSource if you want the service to use the pixel aspect ratio from the input. Using the console, do this by choosing Follow source for Pixel aspect ratio.

INITIALIZE_FROM_SOURCE
SPECIFIED

Mpeg2QualityTuningLevel (Enum)
Use Quality tuning level (Mpeg2QualityTuningLevel) to specify whether to use single-pass or multipass video encoding.

SINGLE_PASS
MULTI_PASS

Mpeg2RateControlMode (Enum)
Use Rate control mode (Mpeg2RateControlMode) to specify whether the bitrate is variable (vbr) or constant (cbr).
VBR
CBR

**Mpeg2SceneChangeDetect (Enum)**

Scene change detection (inserts I-frames on scene changes).

- DISABLED
- ENABLED

**Mpeg2Settings**

**minIInterval**

Enforces separation between repeated (cadence) I-frames and I-frames inserted by Scene Change Detection. If a scene change I-frame is within I-interval frames of a cadence I-frame, the GOP is shrunk and/or stretched to the scene change I-frame. GOP stretch requires enabling lookahead as well as setting I-interval. The normal cadence resumes for the next GOP. This setting is only used when Scene Change Detect is enabled. Note: Maximum GOP stretch = GOP size + Min-I-interval - 1

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 30

**parNumerator**

Pixel Aspect Ratio numerator.

- **Type:** integer
- **Required:** False
- **Minimum:** 1
- **Maximum:** 2147483647

**gopSizeUnits**

- **Type:** Mpeg2GopSizeUnits (p. 427)
- **Required:** False

**hrdBufferSize**

Size of buffer (HRD buffer model). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.

- **Type:** integer
- **Required:** False
- **Minimum:** -2147483648
- **Maximum:** 2147483647

**qualityTuningLevel**

- **Type:** Mpeg2QualityTuningLevel (p. 428)
- **Required:** False
maxBitrate

Maximum bitrate in bits/second (for VBR mode only). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

bitrate

Average bitrate in bits/second. Required for VBR, CBR, and ABR. Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.

Type: integer
Required: False
Minimum: 1000
Maximum: 2147483647

spatialAdaptiveQuantization

Type: Mpeg2SpatialAdaptiveQuantization (p. 433)
Required: False

slowPal

Type: Mpeg2SlowPal (p. 433)
Required: False

codecProfile

Type: Mpeg2CodecProfile (p. 427)
Required: False

intraDcPrecision

Type: Mpeg2IntraDcPrecision (p. 428)
Required: False

softness

Softness. Selects quantizer matrix, larger values reduce high-frequency content in the encoded image.

Type: integer
Required: False
Minimum: 0
Maximum: 128

framerateControl

Type: Mpeg2FramerateControl (p. 427)
Required: False
telecine

- **Type:** Mpeg2Telecine (p. 433)
- **Required:** False

framerateConversionAlgorithm

- **Type:** Mpeg2FramerateConversionAlgorithm (p. 427)
- **Required:** False

codecLevel

- **Type:** Mpeg2CodecLevel (p. 427)
- **Required:** False

temporalAdaptiveQuantization

- **Type:** Mpeg2TemporalAdaptiveQuantization (p. 433)
- **Required:** False

hrdBufferInitialFillPercentage

Percentage of the buffer that should initially be filled (HRD buffer model).

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 100

framerateNumerator

Framerate numerator - framerate is a fraction, e.g. 24000 / 1001 = 23.976 fps.

- **Type:** integer
- **Required:** False
- **Minimum:** 1
- **Maximum:** 2147483647

numberBFramesBetweenReferenceFrames

Number of B-frames between reference frames.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 7

gopClosedCadence

Frequency of closed GOPs. In streaming applications, it is recommended that this be set to 1 so a decoder joining mid-stream will receive an IDR frame as quickly as possible. Setting this value to 0 will break output segmenting.
Type: integer  
Required: False  
Minimum: 0  
Maximum: 2147483647

framerateDenominator

Framerate denominator.

Type: integer  
Required: False  
Minimum: 1  
Maximum: 2147483647

adaptiveQuantization

Type: Mpeg2AdaptiveQuantization (p. 427)  
Required: False

interlaceMode

Type: Mpeg2InterlaceMode (p. 428)  
Required: False

gopSize

GOP Length (keyframe interval) in frames or seconds. Must be greater than zero.

Type: number  
Required: False  
Format: float  
Minimum: 0.0

sceneChangeDetect

Type: Mpeg2SceneChangeDetect (p. 429)  
Required: False

parDenominator

Pixel Aspect Ratio denominator.

Type: integer  
Required: False  
Minimum: 1  
Maximum: 2147483647

parControl

Type: Mpeg2ParControl (p. 428)  
Required: False
**syntax**

Type: Mpeg2Syntax (p. 433)
Required: False

**rateControlMode**

Type: Mpeg2RateControlMode (p. 428)
Required: False

**Mpeg2SlowPal (Enum)**

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

- DISABLED
- ENABLED

**Mpeg2SpatialAdaptiveQuantization (Enum)**

Adjust quantization within each frame based on spatial variation of content complexity.

- DISABLED
- ENABLED

**Mpeg2Syntax (Enum)**

Produces a Type D-10 compatible bitstream (SMPTE 356M-2001).

- DEFAULT
- D_10

**Mpeg2Telecine (Enum)**

Only use Telecine (Mpeg2Telecine) when you set Framerate (Framerate) to 29.970. Set Telecine (Mpeg2Telecine) to Hard (hard) to produce a 29.97i output from a 23.976 input. Set it to Soft (soft) to produce 23.976 output and leave conversion to the player.

- NONE
- SOFT
- HARD

**Mpeg2TemporalAdaptiveQuantization (Enum)**

Adjust quantization within each frame based on temporal variation of content complexity.

- DISABLED
- ENABLED

**MsSmoothAudioDeduplication (Enum)**

COMBINE_DUPLICATE_STREAMS combines identical audio encoding settings across a Microsoft Smooth output group into a single audio stream.

- COMBINE_DUPLICATE_STREAMS
NONE

MsSmoothEncryptionSettings

spekeKeyProvider

Type: SpekeKeyProvider (p. 445)
Required: True

MsSmoothGroupSettings

fragmentLength

Use Fragment length (FragmentLength) to specify the mp4 fragment sizes in seconds. Fragment length must be compatible with GOP size and framerate.

Type: integer
Required: True
Minimum: 1
Maximum: 2147483647

encryption

Type: MsSmoothEncryptionSettings (p. 434)
Required: False

audioDeduplication

Type: MsSmoothAudioDeduplication (p. 433)
Required: False

manifestEncoding

Type: MsSmoothManifestEncoding (p. 434)
Required: False

destination

Use Destination (Destination) to specify the S3 output location and the output filename base. Destination accepts format identifiers. If you do not specify the base filename in the URI, the service will use the filename of the input file. If your job has multiple inputs, the service uses the filename of the first input file.

Type: string
Required: True
Pattern: ^s3: // /

MsSmoothManifestEncoding (Enum)

Use Manifest encoding (MsSmoothManifestEncoding) to specify the encoding format for the server and client manifest. Valid options are utf8 and utf16.

UTF8
UTF16

**NielsenConfiguration**

**distributorId**

Use Distributor ID (DistributorID) to specify the distributor ID that is assigned to your organization by Nielsen.

- **Type:** string
- **Required:** False

**breakoutCode**

Use Nielsen Configuration (NielsenConfiguration) to set the Nielsen measurement system breakout code. Supported values are 0, 3, 7, and 9.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 9

**NoiseReducer**

**filter**

- **Type:** NoiseReducerFilter (p. 435)
- **Required:** True

**filterSettings**

- **Type:** NoiseReducerFilterSettings (p. 436)
- **Required:** False

**spatialFilterSettings**

- **Type:** NoiseReducerSpatialFilterSettings (p. 436)
- **Required:** False

**NoiseReducerFilter (Enum)**

Use Noise reducer filter (NoiseReducerFilter) to select one of the following spatial image filtering functions. To use this setting, you must also enable Noise reducer (NoiseReducer).

* Bilateral is an edge preserving noise reduction filter
* Mean (softest), Gaussian, Lanczos, and Sharpen (sharpest) are convolution filters
* Conserve is a min/max noise reduction filter
* Spatial is frequency-domain filter based on JND principles.

BILATERAL
MEAN
GAUSSIAN
LANCZOS
SHARPEN
CONSERVE
SPATIAL

**NoiseReducerFilterSettings**

**strength**
Relative strength of noise reducing filter. Higher values produce stronger filtering.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 3

**NoiseReducerSpatialFilterSettings**

**strength**
Relative strength of noise reducing filter. Higher values produce stronger filtering.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 16

**postFilterSharpenStrength**
Specify strength of post noise reduction sharpening filter, with 0 disabling the filter and 3 enabling it at maximum strength.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 3

**speed**
The speed of the filter, from -2 (lower speed) to 3 (higher speed), with 0 being the nominal value.

- **Type**: integer
- **Required**: False
- **Minimum**: -2
- **Maximum**: 3

**Order (Enum)**
When you request lists of resources, you can optionally specify whether they are sorted in ASCENDING or DESCENDING order. Default varies by resource.

- ASCENDING
- DESCENDING
Output

extension

Use Extension (Extension) to specify the file extension for outputs in File output groups. If you do not specify a value, the service will use default extensions by container type as follows:

* MPEG-2 transport stream, m2ts
* Quicktime, mov
* MXF container, mxf
* MPEG-4 container, mp4
* No Container, the service will use codec extensions (e.g. AAC, H265, H265, AC3)

  Type: string
  Required: False

videoDescription

(VideoDescription) contains a group of video encoding settings. The specific video settings depend on the video codec you choose when you specify a value for Video codec (codec). Include one instance of (VideoDescription) per output.

  Type: VideoDescription (p. 450)
  Required: False

audioDescriptions

(AudioDescriptions) contains groups of audio encoding settings organized by audio codec. Include one instance of (AudioDescriptions) per output. (AudioDescriptions) can contain multiple groups of encoding settings.

  Type: Array of type AudioDescription (p. 343)
  Required: False

containerSettings

  Type: ContainerSettings (p. 357)
  Required: False

preset

Use Preset (Preset) to specify a preset for your transcoding settings. Provide the system or custom preset name. You can specify either Preset (Preset) or Container settings (ContainerSettings), but not both.

  Type: string
  Required: False

outputSettings

  Type: OutputSettings (p. 440)
  Required: False
captionDescriptions

(CaptionDescriptions) contains groups of captions settings. For each output that has captions, include one instance of (CaptionDescriptions). (CaptionDescriptions) can contain multiple groups of captions settings.

  Type: Array of type CaptionDescription (p. 352)
  Required: False

nameModifier

Use Name modifier (NameModifier) to have the service add a string to the end of each output filename. You specify the base filename as part of your destination URI. When you create multiple outputs in the same output group, Name modifier is required. Name modifier also accepts format identifiers. For DASH ISO outputs, if you use the format identifiers $Number$ or $Time$ in one output, you must use them in the same way in all outputs of the output group.

  Type: string
  Required: False

OutputChannelMapping

inputChannels

  Type: Array of type integer
  Required: True

OutputDetail

durationInMs

  Type: integer
  Required: False

videoDetails

  Type: VideoDetail (p. 452)
  Required: False

OutputGroup

outputs

This object holds groups of encoding settings, one group of settings per output.

  Type: Array of type Output (p. 437)
  Required: True

outputGroupSettings

  Type: OutputGroupSettings (p. 439)
  Required: True
name

Type: string  
Required: False

customName

Use Custom Group Name (CustomName) to specify a name for the output group. This value is displayed
on the console and can make your job settings JSON more human-readable. It does not affect your
outputs. Use up to twelve characters that are either letters, numbers, spaces, or underscores.

Type: string  
Required: False

OutputGroupDetail

outputDetails

Type: Array of type OutputDetail (p. 438)  
Required: False

OutputGroupSettings

dashIsoGroupSettings

Type: DashIsoGroupSettings (p. 359)  
Required: False

fileGroupSettings

Type: FileGroupSettings (p. 373)  
Required: False

msSmoothGroupSettings

Type: MsSmoothGroupSettings (p. 434)  
Required: False

type

Type: OutputGroupType (p. 439)  
Required: True

hlsGroupSettings

Type: HlsGroupSettings (p. 396)  
Required: False

OutputGroupType (Enum)

HLS_GROUP_SETTINGS  
DASH_ISO_GROUP_SETTINGS
AWS Elemental MediaConvert API Reference
Properties

FILE_GROUP_SETTINGS
MS_SMOOTH_GROUP_SETTINGS

OutputSdt (Enum)
Selects method of inserting SDT information into output stream. "Follow input SDT" copies SDT information from input stream to output stream. "Follow input SDT if present" copies SDT information from input stream to output stream if SDT information is present in the input, otherwise it will fall back on the user-defined values. Enter "SDT Manually" means user will enter the SDT information. "No SDT" means output stream will not contain SDT information.

SDT_FOLLOW
SDT_FOLLOW_IF_PRESENT
SDT_MANUAL
SDT_NONE

OutputSettings

hlsSettings
Type: HlsSettings (p. 400)
Required: False

ProresCodecProfile (Enum)
Use Profile (ProResCodecProfile) to specify the type of Apple ProRes codec to use for this output.

APPLE_PRORES_422
APPLE_PRORES_422_HQ
APPLE_PRORES_422_LT
APPLE_PRORES_422_PROXY

ProresFramerateControl (Enum)
Using the API, set FramerateControl to INITIALIZE_FROM_SOURCE if you want the service to use the framerate from the input. Using the console, do this by choosing INITIALIZE_FROM_SOURCE for Framerate.

INITIALIZE_FROM_SOURCE
SPECIFIED

ProresFramerateConversionAlgorithm (Enum)
When set to INTERPOLATE, produces smoother motion during framerate conversion.

DUPLICATE_DROP
INTERPOLATE

ProresInterlaceMode (Enum)
Use Interlace mode (InterlaceMode) to choose the scan line type for the output.

* Top Field First (TOP_FIELD) and Bottom Field First (BOTTOM_FIELD) produce interlaced output with the entire output having the same field polarity (top or bottom first).
* Follow, Default Top (FOLLOW_TOP_FIELD) and Follow, Default Bottom (FOLLOW_BOTTOM_FIELD) use the same field polarity as the source. Therefore, behavior depends on the input scan type. If the source is interlaced, the output will be interlaced with the same polarity as the source (it will follow the source). The output could therefore be a mix of "top field first" and "bottom field first". If the source is progressive, the output will be interlaced with "top field first" or "bottom field first" polarity, depending on which of the Follow options you chose.

```
PROGRESSIVE
TOP_FIELD
BOTTOM_FIELD
FOLLOW_TOP_FIELD
FOLLOW_BOTTOM_FIELD
```

**ProresParControl (Enum)**

Use (ProresParControl) to specify how the service determines the pixel aspect ratio. Set to Follow source (INITIALIZE_FROM_SOURCE) to use the pixel aspect ratio from the input. To specify a different pixel aspect ratio: Using the console, choose it from the dropdown menu. Using the API, set ProresParControl to (SPECIFIED) and provide for (ParNumerator) and (ParDenominator).

```
INITIALIZE_FROM_SOURCE
SPECIFIED
```

**ProresSettings**

**slowPal**

* Type: **ProresSlowPal** (p. 442)
* Required: False

**framerateControl**

* Type: **ProresFramerateControl** (p. 440)
* Required: False

**telecine**

* Type: **ProresTelecine** (p. 443)
* Required: False

**framerateDenominator**

Framerate denominator.

* Type: integer
* Required: False
* Minimum: 1
* Maximum: 2147483647

**framerateConversionAlgorithm**

* Type: **ProresFramerateConversionAlgorithm** (p. 440)
Properties

interlaceMode

Type: ProresInterlaceMode (p. 440)
Required: False

codecProfile

Type: ProresCodecProfile (p. 440)
Required: False

parNumerator

Pixel Aspect Ratio numerator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

parControl

Type: ProresParControl (p. 441)
Required: False

parDenominator

Pixel Aspect Ratio denominator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

framerateNumerator

When you use the API for transcode jobs that use framerate conversion, specify the framerate as a fraction. For example, 24000 / 1001 = 23.976 fps. Use FramerateNumerator to specify the numerator of this fraction. In this example, use 24000 for the value of FramerateNumerator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

ProresSlowPal (Enum)

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

DISABLED
ENABLED
**ProresTelecine (Enum)**

Only use Telecine (ProresTelecine) when you set Framerate (Framerate) to 29.970. Set Telecine (ProresTelecine) to Hard (hard) to produce a 29.97i output from a 23.976 input. Set it to Soft (soft) to produce 23.976 output and leave conversion to the player.

- NONE
- HARD

**Rectangle**

**width**

Width of rectangle in pixels.

- **Type:** integer
- **Required:** True
- **Minimum:** -2147483648
- **Maximum:** 2147483647

**x**

The distance, in pixels, between the rectangle and the left edge of the video frame.

- **Type:** integer
- **Required:** True
- **Minimum:** -2147483648
- **Maximum:** 2147483647

**y**

The distance, in pixels, between the rectangle and the top edge of the video frame.

- **Type:** integer
- **Required:** True
- **Minimum:** -2147483648
- **Maximum:** 2147483647

**height**

Height of rectangle in pixels.

- **Type:** integer
- **Required:** True
- **Minimum:** -2147483648
- **Maximum:** 2147483647

**RemixSettings**

**channelsOut**

Specify the number of channels in this output after remixing. Valid values: 1, 2, 4, 6, 8

- **Type:** integer
- **Required:** True
Minimum: 1
Maximum: 8

channelMapping

Type: ChannelMapping (p. 355)
Required: True

channelsIn

Specify the number of audio channels from your input that you want to use in your output. With remixing, you might combine or split the data in these channels, so the number of channels in your final output might be different.

Type: integer
Required: True
Minimum: 1
Maximum: 16

RespondToAfd (Enum)

Use Respond to AFD (RespondToAfd) to specify how the service changes the video itself in response to AFD values in the input.

* Choose Respond to clip the input video frame according to the AFD value, input display aspect ratio, and output display aspect ratio.
* Choose Passthrough to include the input AFD values. Do not choose this when AfdSignaling is set to (NONE). A preferred implementation of this workflow is to set RespondToAfd to (NONE) and set AfdSignaling to (AUTO).
* Choose None to remove all input AFD values from this output.

NONE
RESPOND
PASSTHROUGH

ScalingBehavior (Enum)

Applies only if your input aspect ratio is different from your output aspect ratio. Enable Stretch to output (StretchToOutput) to have the service stretch your video image to fit. Leave this setting disabled to allow the service to letterbox your video instead. This setting overrides any positioning value you specify elsewhere in the job.

DEFAULT
STRETCH_TO_OUTPUT

SccDestinationFramerate (Enum)

Set Framerate (SccDestinationFramerate) to make sure that the captions and the video are synchronized in the output. Specify a framerate that matches the framerate of the associated video. If the video framerate is 29.97, choose 29.97 dropframe (FRAMERATE_29_97_DROPFRAME) only if the video has video_insertion=true and drop_frame_timecode=true; otherwise, choose 29.97 non-dropframe (FRAMERATE_29_97_NON_DROPFRAME).

FRAMERATE_23_97
Properties

FRAMERATE_24
FRAMERATE_29_97_DROPFRAME
FRAMERATE_29_97_NON_DROPFRAME

SccDestinationSettings

framerate

Type: SccDestinationFramerate (p. 444)
Required: False

SpekeKeyProvider

resourceId

The SPEKE-compliant server uses Resource ID (ResourceId) to identify content.

Type: string
Required: True
Pattern: ^[\w-]+$

systemIds

Relates to SPEKE implementation. DRM system identifiers. DASH output groups support a max of two system ids. Other group types support one system id.

Type: Array of type string
Required: True

url

Use URL (Url) to specify the SPEKE-compliant server that will provide keys for content.

Type: string
Required: True
Format: uri
Pattern: ^https:/

StaticKeyProvider

keyFormatVersions

Relates to DRM implementation. Either a single positive integer version value or a slash delimited list of version values (1/2/3).

Type: string
Required: False
Pattern: ^\d+(\d+)*$}

keyFormat

Relates to DRM implementation. Sets the value of the KEYFORMAT attribute. Must be 'identity' or a reverse DNS string. May be omitted to indicate an implicit value of 'identity'.
Type: string
Required: False
Pattern: ^\(identity\|[A-Za-z]{2,6}\(\.|[A-Za-z0-9-]{1,63}\)\{1,63\}$

**staticKeyValue**

Relates to DRM implementation. Use a 32-character hexadecimal string to specify Key Value (StaticKeyValue).

Type: string
Required: True
Pattern: ^[A-Za-z0-9]{32}$

**url**

Relates to DRM implementation. The location of the license server used for protecting content.

Type: string
Required: True
Format: uri

**TeletextDestinationSettings**

**pageNumber**

Set pageNumber to the Teletext page number for the destination captions for this output. This value must be a three-digit hexadecimal string; strings ending in -FF are invalid. If you are passing through the entire set of Teletext data, do not use this field.

Type: string
Required: False
Pattern: ^[1-8][0-9a-fA-F][0-9a-eA-E]$
equivalents. The supported range of characters is 0x20 through 0x7e. This includes letters, numbers, and all special characters represented on a standard English keyboard.

Type: string
Required: False
Pattern: ^[ -~]+$

**fontSize**

Use Font Size (FontSize) to set the font size of any burned-in timecode. Valid values are 10, 16, 32, 48.

Type: integer
Required: False
Minimum: 10
Maximum: 48

**position**

Type: TimecodeBurninPosition (p. 447)
Required: False

**TimecodeBurninPosition (Enum)**

Use Position (Position) under Timecode burn-in (TimecodeBurnIn) to specify the location the burned-in timecode on output video.

- TOP_CENTER
- TOP_LEFT
- TOP_RIGHT
- MIDDLE_LEFT
- MIDDLE_CENTER
- MIDDLE_RIGHT
- BOTTOM_LEFT
- BOTTOM_CENTER
- BOTTOM_RIGHT

**TimecodeConfig**

**timestampOffset**

Only applies to outputs that support program-date-time stamp. Use Time stamp offset (TimestampOffset) to overwrite the timecode date without affecting the time and frame number. To use this, you must also enable Insert program-date-time (InsertProgramDateTime) in the output settings.

Type: string
Required: False
Pattern: ^([0-9]{4})-(0[1-9]|1[0-2])-(0[1-9]|12)[0-9]|3[01])$

**anchor**

If you use an editing platform that relies on an anchor timecode, use Anchor Timecode (Anchor) to specify a timecode that will match the input video frame to the output video frame. Use 24-hour format
with frame number, (HH:MM:SS:FF) or (HH:MM:SS;FF). This setting ignores framerate conversion. System behavior for Anchor Timecode varies depending on your setting for Timecode source (TimecodeSource).

* If Timecode source (TimecodeSource) is set to Specified Start (specifiedstart), the first input frame is the specified value in Start Timecode (Start). Anchor Timecode (Anchor) and Start Timecode (Start) are used to calculate output timecode.

* If Timecode source (TimecodeSource) is set to Start at 0 (zerobased) the first frame is 00:00:00:00.

* If Timecode source (TimecodeSource) is set to Embedded (embedded), the first frame is the timecode value on the first input frame of the input.

    Type: string
    Required: False
    Format: timecode
    Pattern: ^([01][0-9]|2[0-4]):[0-5][0-9]:[0-5][0-9][:;][0-9]{2}$

start

Only use when you set Timecode Source (TimecodeSource) to Specified Start (SPECIFIEDSTART). Use Start timecode (Start) to specify the timecode for the initial frame. Use 24-hour format with frame number, (HH:MM:SS:FF) or (HH:MM:SS;FF).

    Type: string
    Required: False
    Format: timecode
    Pattern: ^([01][0-9]|2[0-4]):[0-5][0-9]:[0-5][0-9][:;][0-9]{2}$

source

    Type: TimecodeSource (p. 448)
    Required: False

TimecodeSource (Enum)

Use Timecode source (TimecodeSource) to set how timecodes are handled within this input. To make sure that your video, audio, captions, and markers are synchronized and that time-based features, such as image inserter, work correctly, choose the Timecode source option that matches your assets. All timecodes are in a 24-hour format with frame number (HH:MM:SS:FF).

* Embedded (EMBEDDED) - Use the timecode that is in the input video. If no embedded timecode is in the source, the service will use Start at 0 (ZEROBASED) instead.

* Start at 0 (ZEROBASED) - Set the timecode of the initial frame to 00:00:00:00.

* Specified Start (SPECIFIEDSTART) - Set the timecode of the initial frame to a value other than zero. You use Start timecode (Start) to provide this value.

    EMBEDDED
    ZEROBASED
    SPECIFIEDSTART

TimedMetadata (Enum)

If PASSTHROUGH, inserts ID3 timed metadata from the timed_metadata REST command into this output. Only available for certain containers.
PASSTHROUGH
NONE

TimedMetadataInsertion

id3Insertions
Id3Insertions contains the array of Id3Insertion instances.

  Type: Array of type Id3Insertion (p. 401)
  Required: True

Timing

finishTime
The time, in Unix epoch format, that the transcoding job finished

  Type: string
  Required: False
  Format: date-time

submitTime
The time, in Unix epoch format, that you submitted the job.

  Type: string
  Required: False
  Format: date-time

startTime
The time, in Unix epoch format, that transcoding for the job began.

  Type: string
  Required: False
  Format: date-time

TtmlDestinationSettings

stylePassthrough

  Type: TtmlStylePassthrough (p. 449)
  Required: False

TtmlStylePassthrough (Enum)
Pass through style and position information from a TTML-like input source (TTML, SMPTE-TT, CFF-TT) to the CFF-TT output or TTML output.

  ENABLED
  DISABLED
VideoCodec (Enum)
Type of video codec
- FRAME_CAPTURE
- H_264
- H_265
- MPEG2
- PRORES

VideoCodecSettings

h265Settings
Type: H265Settings (p. 386)
Required: False

codec
Type: VideoCodec (p. 450)
Required: True

proresSettings
Type: ProresSettings (p. 441)
Required: False

mpeg2Settings
Type: Mpeg2Settings (p. 429)
Required: False

h264Settings
Type: H264Settings (p. 377)
Required: False

frameCaptureSettings
Type: FrameCaptureSettings (p. 374)
Required: False

VideoDescription

fixedAfd
Applies only if you set AFD Signaling(AfdSignaling) to Fixed (FIXED). Use Fixed (FixedAfd) to specify a four-bit AFD value which the service will write on all frames of this video output.

Type: integer
Required: False
Minimum: 0
Maximum: 15

**scalingBehavior**

Type: ScalingBehavior (p. 444)  
Required: False

**respondToAfd**

Type: RespondToAfd (p. 444)  
Required: False

**codecSettings**

Type: VideoCodecSettings (p. 450)  
Required: True

**afdSignaling**

Type: AfdSignaling (p. 341)  
Required: False

**colorMetadata**

Type: ColorMetadata (p. 356)  
Required: False

**timecodeInsertion**

Type: VideoTimecodeInsertion (p. 454)  
Required: False

**width**

Use Width (Width) to define the video resolution width, in pixels, for this output. If you don't provide a value here, the service will use the input width.

Type: integer  
Required: False  
Minimum: 32  
Maximum: 4096

**sharpness**

Use Sharpness (Sharpness) setting to specify the strength of anti-aliasing. This setting changes the width of the anti-alias filter kernel used for scaling. Sharpness only applies if your output resolution is different from your input resolution, and if you set Anti-alias (AntiAlias) to ENABLED. 0 is the softest setting, 100 the sharpest, and 50 recommended for most content.

Type: integer  
Required: False  
Minimum: 0  
Maximum: 100
antiAlias

Type: AntiAlias (p. 342)
Required: False

videoPreprocessors

Find additional transcoding features under Preprocessors (VideoPreprocessors). Enable the features at each output individually. These features are disabled by default.

Type: VideoPreprocessor (p. 453)
Required: False

position

Use Position (Position) to point to a rectangle object to define your position. This setting overrides any other aspect ratio.

Type: Rectangle (p. 443)
Required: False

dropFrameTimecode

Type: DropFrameTimecode (p. 361)
Required: False

crop

Applies only if your input aspect ratio is different from your output aspect ratio. Use Input cropping rectangle (Crop) to specify the video area the service will include in the output. This will crop the input source, causing video pixels to be removed on encode. Do not use this setting if you have enabled Stretch to output (stretchToOutput) in your output settings.

Type: Rectangle (p. 443)
Required: False

height

Use the Height (Height) setting to define the video resolution height for this output. Specify in pixels. If you don’t provide a value here, the service will use the input height.

Type: integer
Required: False
Minimum: 32
Maximum: 2160

VideoDetail

heightInPx

Height in pixels for the output

Type: integer
**widthInPx**
Width in pixels for the output

- **Type:** integer
- **Required:** False

**VideoPreprocessor**

**timecodeBurnin**
Timecode burn-in (TimecodeBurnIn)--Burns the output timecode and specified prefix into the output.

- **Type:** TimecodeBurnin (p. 446)
- **Required:** False

**noiseReducer**
Enable the Noise reducer (NoiseReducer) feature to remove noise from your video output if necessary. Enable or disable this feature for each output individually. This setting is disabled by default.

- **Type:** NoiseReducer (p. 435)
- **Required:** False

**colorCorrector**
Enable the Color corrector (ColorCorrector) feature if necessary. Enable or disable this feature for each output individually. This setting is disabled by default.

- **Type:** ColorCorrector (p. 355)
- **Required:** False

**imageInserter**
Enable the Image inserter (ImageInserter) feature to include a graphic overlay on your video. Enable or disable this feature for each output individually. This setting is disabled by default.

- **Type:** ImageInserter (p. 402)
- **Required:** False

**deinterlacer**
Use Deinterlacer (Deinterlacer) to produce smoother motion and a clearer picture.

- **Type:** Deinterlacer (p. 361)
- **Required:** False

**VideoSelector**

**colorSpace**

- **Type:** ColorSpace (p. 356)
**Properties**

**Required**: False

**hdr10Metadata**

*Type*: Hdr10Metadata (p. 392)

**Required**: False

**programNumber**

Selects a specific program from within a multi-program transport stream. Note that Quad 4K is not currently supported.

*Type*: integer

**Required**: False

**Minimum**: -2147483648

**Maximum**: 2147483647

**pid**

Use PID (Pid) to select specific video data from an input file. Specify this value as an integer; the system automatically converts it to the hexadecimal value. For example, 257 selects PID 0x101. A PID, or packet identifier, is an identifier for a set of data in an MPEG-2 transport stream container.

*Type*: integer

**Required**: False

**Minimum**: 1

**Maximum**: 2147483647

**colorSpaceUsage**

*Type*: ColorSpaceUsage (p. 357)

**Required**: False

**VideoTimecodeInsertion (Enum)**

Enable Timecode insertion to include timecode information in this output. Do this in the API by setting (VideoTimecodeInsertion) to (PIC_TIMING_SEI). To get timecodes to appear correctly in your output, also set up the timecode configuration for your job in the input settings. Only enable Timecode insertion when the input framerate is identical to output framerate. Disable this setting to remove the timecode from the output. Default is disabled.

- **DISABLED**
- **PIC_TIMING_SEI**

**WavSettings**

**channels**

Set Channels to specify the number of channels in this output audio track. With WAV, valid values 1, 2, 4, and 8. In the console, these values are Mono, Stereo, 4-Channel, and 8-Channel, respectively.

*Type*: integer

**Required**: False

**Minimum**: 1
**Maximum**: 8

**bitDepth**

Specify Bit depth (BitDepth), in bits per sample, to choose the encoding quality for this audio track.

- **Type**: integer
- **Required**: False
- **Minimum**: 16
- **Maximum**: 24

**sampleRate**

Sample rate in Hz.

- **Type**: integer
- **Required**: False
- **Minimum**: 8000
- **Maximum**: 192000

---

## A Specific Job

### URI

/jobs/\(id\)

### HTTP Methods

**GET**

Operation ID: GetJob

Retrieve the JSON for a specific completed transcoding job.

### Path Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>String</td>
<td>True</td>
<td></td>
</tr>
</tbody>
</table>

### Responses

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>GetJobResponse (p. 458)</td>
<td>200: OkResponse</td>
</tr>
</tbody>
</table>
| 400         | ExceptionBody (p. 468) | 400: BadRequestException

The conditional request failed. The service can’t process your request because of a problem in the request. Please check your request form and syntax.
<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>500</td>
<td>ExceptionBody (p. 468)</td>
<td>500: InternalServiceException. The service encountered an unexpected condition and cannot fulfill your request.</td>
</tr>
<tr>
<td>403</td>
<td>ExceptionBody (p. 468)</td>
<td>403: AccessDeniedException. You don't have permissions for this action with the credentials you sent. Please check your authorization credentials. You should be sending credentials using the AWS Signature Version 4 signing process.</td>
</tr>
<tr>
<td>404</td>
<td>ExceptionBody (p. 468)</td>
<td>404: ResourceNotFoundException. The resource you requested does not exist.</td>
</tr>
<tr>
<td>429</td>
<td>ExceptionBody (p. 468)</td>
<td>429: LimitExceededException. Too many requests have been sent in too short of a time. The service limits the rate at which it will accept requests. For example, you may be hitting your account limits for preset creation or job submission.</td>
</tr>
<tr>
<td>409</td>
<td>ExceptionBody (p. 468)</td>
<td>409: ResourceInUseException. The service could not complete your request because there is a conflict with the current state of the resource. For example, you may be trying to delete a Queue that has jobs processing.</td>
</tr>
</tbody>
</table>

**DELETE**

Operation ID: CancelJob

Permanently remove a job from a queue. Once you have canceled a job, you can't start it again. You can't delete a running job.

**Path Parameters**

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>String</td>
<td>True</td>
<td></td>
</tr>
</tbody>
</table>
## Responses

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
</table>
| 400         | ExceptionBody (p. 468) | 400: BadRequestException  
The conditional request failed. The service can't process your request because of a problem in the request. Please check your request form and syntax. |
| 202         | CancelJobResponse (p. 468) | 202: AcceptedResponse  
Your request has been accepted. Processing has not yet begun. |
| 500         | ExceptionBody (p. 468) | 500: InternalServiceException  
The service encountered an unexpected condition and cannot fulfill your request. |
| 403         | ExceptionBody (p. 468) | 403: AccessDeniedException  
You don't have permissions for this action with the credentials you sent. Please check your authorization credentials. You should be sending credentials using the AWS Signature Version 4 signing process. |
| 404         | ExceptionBody (p. 468) | 404: ResourceNotFoundException  
The resource you requested does not exist. |
| 429         | ExceptionBody (p. 468) | 429: LimitExceededException  
Too many requests have been sent in too short of a time. The service limits the rate at which it will accept requests. For example, you may be hitting your account limits for preset creation or job submission. |
| 409         | ExceptionBody (p. 468) | 409: ResourceInUseException  
The service could not complete your request because there is a conflict with the current state of the resource. For example, you may be trying to delete a Queue that has jobs processing. |
Schemas

Request Bodies

Example GET

```
{
  "id (p. 505)": "string"
}
```

Example DELETE

```
{
  "id (p. 484)": "string"
}
```

Response Bodies

Example GetJobResponse

```
{
  "job (p. 505)": {
    "settings (p. 538)": {
      "timecodeConfig (p. 539)": {
        "timestampOffset (p. 577)": "string",
        "anchor (p. 577)": "string",
        "start (p. 578)": "string",
        "source (p. 578)": enum
      },
      "adAvailOffset (p. 540)": integer,
      "nielsenConfiguration (p. 540)": {
        "distributorId (p. 565)": "string",
        "breakoutCode (p. 565)": integer
      },
      "inputs (p. 540)": [
        {
          "audioSelectors (p. 532)": {
          },
          "audioSelectorGroups (p. 533)": {
          },
          "videoSelector (p. 533)": {
            "colorSpace (p. 583)": enum,
            "hdr10Metadata (p. 583)": {
              "redPrimaryY (p. 522)": integer,
              "greenPrimaryY (p. 523)": integer,
              "whitePointX (p. 523)": integer,
              "maxLuminance (p. 523)": integer,
              "greenPrimaryX (p. 523)": integer,
              "whitePointY (p. 523)": integer,
              "redPrimaryX (p. 523)": integer,
              "bluePrimaryX (p. 524)": integer,
              "maxFrameAverageLightLevel (p. 524)": integer,
              "bluePrimaryY (p. 524)": integer,
              "maxContentLightLevel (p. 524)": integer,
              "minLuminance (p. 524)": integer
            },
            "programNumber (p. 584)": integer,
            "pid (p. 584)": integer,
```
"colorSpaceUsage (p. 584)": enum,
"denoiseFilter (p. 533)": enum,
"filterEnable (p. 533)": enum,
"deblockFilter (p. 533)": enum,
"filterStrength (p. 533)": integer,
"programNumber (p. 533)": integer,
"timecodeSource (p. 533)": enum,
"captionSelectors (p. 534)": {
},
"fileInput (p. 534)": "string",
"inputClippings (p. 534)": [
    {
        "startTimecode (p. 534)": "string",
        "endTimecode (p. 535)": "string"
    }
],
"psiControl (p. 534)": enum
],
"outputGroups (p. 540)": [
    {
        "outputs (p. 568)": [
            {
                "extension (p. 566)": "string",
                "videoDescription (p. 567)": {
                    "fixedAfd (p. 580)": integer,
                    "scalingBehavior (p. 580)": enum,
                    "respondToAfd (p. 581)": enum,
                    "codecSettings (p. 581)": {
                        "h265Settings (p. 580)": {
                            "slices (p. 517)": integer,
                            "minIInterval (p. 517)": integer,
                            "parNumerator (p. 517)": integer,
                            "flickerAdaptiveQuantization (p. 517)": enum,
                            "gopSizeUnits (p. 517)": enum,
                            "hrdBufferSize (p. 517)": integer,
                            "qualityTuningLevel (p. 518)": enum,
                            "maxBitrate (p. 518)": integer,
                            "bitrate (p. 518)": integer,
                            "spatialAdaptiveQuantization (p. 518)": enum,
                            "sampleAdaptiveOffsetFilterMode (p. 518)": enum,
                            "temporalIds (p. 518)": enum,
                            "slowPal (p. 518)": enum,
                            "tiles (p. 518)": enum,
                            "codecProfile (p. 519)": enum,
                            "alternateTransferFunctionSei (p. 519)": enum,
                            "unregisteredSeiTimecode (p. 519)": enum,
                            "framerateControl (p. 519)": enum,
                            "telecine (p. 519)": enum,
                            "framerateConversionAlgorithm (p. 519)": enum,
                            "codecLevel (p. 519)": enum,
                            "numberReferenceFrames (p. 519)": integer,
                            "temporalAdaptiveQuantization (p. 519)": enum,
                            "hrdBufferInitialFillPercentage (p. 520)": integer,
                            "framerateNumerator (p. 520)": integer,
                            "numberBFramesBetweenReferenceFrames (p. 520)": integer,
                            "gopClosedCadence (p. 520)": integer,
                            "gopSize (p. 521)": number,
                            "gopBReference (p. 521)": enum,
                            "sceneChangeDetect (p. 521)": enum,
                            "parDenominator (p. 521)": integer,
                            "parControl (p. 521)": enum,
                        }
                    }
                }
            }
        ]
    }
]
"rateControlMode (p. 521)": enum
},
"codec (p. 580)": enum,
"proresSettings (p. 580)": {
  "slowPal (p. 571)": enum,
  "framerateControl (p. 571)": enum,
  "telecine (p. 571)": enum,
  "framerateDenominator (p. 571)": integer,
  "framerateConversionAlgorithm (p. 571)": enum,
  "interlaceMode (p. 571)": enum,
  "codecProfile (p. 572)": enum,
  "parNumerator (p. 572)": integer,
  "parControl (p. 572)": enum,
  "parDenominator (p. 572)": integer,
  "framerateNumerator (p. 572)": integer
},
"mpeg2Settings (p. 580)": {
  "minIInterval (p. 559)": integer,
  "parNumerator (p. 559)": integer,
  "gopSizeUnits (p. 559)": enum,
  "hrdBufferSize (p. 559)": integer,
  "qualityTuningLevel (p. 559)": enum,
  "maxBitrate (p. 559)": integer,
  "bitrate (p. 560)": integer,
  "spatialAdaptiveQuantization (p. 560)": enum,
  "slowPal (p. 560)": enum,
  "codecProfile (p. 560)": enum,
  "intraDcPrecision (p. 560)": enum,
  "softness (p. 560)": integer,
  "framerateControl (p. 560)": enum,
  "telecine (p. 560)": enum,
  "framerateConversionAlgorithm (p. 561)": enum,
  "framerateDenominator (p. 561)": integer,
  "framerateNumerator (p. 561)": integer,
  "numberBFramesBetweenReferenceFrames (p. 561)": integer,
  "gopClosedCadence (p. 561)": integer,
  "parNumerator (p. 562)": integer,
  "gopSize (p. 562)": number,
  "parDenominator (p. 562)": integer,
  "parControl (p. 562)": enum,
  "syntax (p. 562)": enum,
  "rateControlMode (p. 563)": enum
},
"h264Settings (p. 580)": {
  "slices (p. 508)": integer,
  "minIInterval (p. 508)": integer,
  "parNumerator (p. 508)": integer,
  "flickerAdaptiveQuantization (p. 509)": enum,
  "gopSizeUnits (p. 509)": enum,
  "hrdBufferSize (p. 509)": integer,
  "qualityTuningLevel (p. 509)": enum,
  "maxBitrate (p. 509)": integer,
  "bitrate (p. 509)": integer,
  "spatialAdaptiveQuantization (p. 509)": enum,
  "slowPal (p. 510)": integer,
  "codecProfile (p. 510)": enum,
  "unregisteredSeiTimecode (p. 510)": enum,
  "softness (p. 510)": integer,
  "framerateControl (p. 510)": enum,
  "telecine (p. 510)": enum,
  "framerateConversionAlgorithm (p. 510)": enum,
"codecLevel (p. 510)": enum,
"numberReferenceFrames (p. 510)": integer,
"temporalAdaptiveQuantization (p. 511)": enum,
"repeatPps (p. 511)": enum,
"hrdBufferInitialFillPercentage (p. 511)": integer,
"framerateNumerator (p. 511)": integer,
"numberOfFramesBetweenReferenceFrames (p. 511)": integer,
"gopClosedCadence (p. 511)": integer,
"fieldEncoding (p. 511)": enum,
"entropyEncoding (p. 512)": enum,
"framerateDenominator (p. 512)": integer,
"adaptiveQuantization (p. 512)": enum,
"interlaceMode (p. 512)": enum,
"gopSize (p. 512)": number,
"gopReference (p. 512)": enum,
"sceneChangeDetect (p. 512)": enum,
"parDenominator (p. 512)": integer,
"parControl (p. 513)": enum,
"syntax (p. 513)": enum,
"rateControlMode (p. 513)": enum
},
"frameCaptureSettings (p. 580)": {
  "framerateDenominator (p. 504)": integer,
  "maxCaptures (p. 504)": integer,
  "framerateNumerator (p. 505)": integer,
  "quality (p. 505)": integer
},
"afdSignaling (p. 581)": enum,
"colorMetadata (p. 581)": enum,
"timecodeInsertion (p. 581)": enum,
"width (p. 581)": integer,
"sharpness (p. 581)": integer,
"antiAlias (p. 581)": enum,
"videoPreprocessors (p. 582)": {
  "timecodeBurnin (p. 583)": {
    "prefix (p. 576)": "string",
    "fontSize (p. 577)": integer,
    "position (p. 577)": enum
  },
  "noiseReducer (p. 583)": {
    "filter (p. 565)": enum,
    "filterSettings (p. 565)": {
      "strength (p. 566)": integer
    },
    "spatialFilterSettings (p. 565)": {
      "strength (p. 566)": integer,
      "postFilterSharpenStrength (p. 566)": integer,
      "speed (p. 566)": integer
    }
  }
},
"colorCorrector (p. 583)": {
  "saturation (p. 487)": integer,
  "brightness (p. 487)": integer,
  "hdr10Metadata (p. 487)": {
    "redPrimaryY (p. 522)": integer,
    "greenPrimaryY (p. 523)": integer,
    "whitePointX (p. 523)": integer,
    "maxLuminance (p. 523)": integer,
    "greenPrimaryX (p. 523)": integer,
    "whitePointY (p. 523)": integer,
    "redPrimaryX (p. 523)": integer,
    "bluePrimaryX (p. 524)": integer,
    "maxFrameAverageLightLevel (p. 524)": integer,
    "bluePrimaryY (p. 524)": integer,
    "maxContentLightLevel (p. 524)": integer,
"minLuminance (p. 524)" : integer,
"contrast (p. 487)" : integer,
"hue (p. 487)" : integer,
"colorSpaceConversion (p. 487)" : enum
},
"imageInserter (p. 583)" : {
"insertableImages (p. 532)" : [
  "duration (p. 536)" : integer,
  "fadeOut (p. 536)" : integer,
  "imageY (p. 536)" : integer,
  "fadeIn (p. 536)" : integer,
  "imageX (p. 537)" : integer,
  "width (p. 537)" : integer,
  "startTime (p. 537)" : "string",
  "opacity (p. 537)" : integer,
  "layer (p. 537)" : integer,
  "height (p. 537)" : integer,
  "imageInserterInput (p. 538)" : "string"
]
},
"deinterlacer (p. 583)" : {
  "mode (p. 491)" : enum,
  "control (p. 491)" : enum,
  "algorithm (p. 491)" : enum
},
"position (p. 582)" : {
  "width (p. 573)" : integer,
  "x (p. 573)" : integer,
  "y (p. 573)" : integer,
  "height (p. 573)" : integer
},
"dropFrameTimecode (p. 582)" : enum,
"crop (p. 582)" : {
  "width (p. 573)" : integer,
  "x (p. 573)" : integer,
  "y (p. 573)" : integer,
  "height (p. 573)" : integer
},
"height (p. 582)" : integer
],
"audioDescriptions (p. 567)" : [
  {
    "languageCodeControl (p. 475)" : enum,
    "audioTypeControl (p. 475)" : enum,
    "remixSettings (p. 475)" : {
      "channelsOut (p. 573)" : integer,
      "channelMapping (p. 574)" : {
        "outputChannels (p. 486)" : [?
          "inputChannels (p. 568)" : [?
            integer
          ]
        ]
      }
    },
    "channelsIn (p. 574)" : integer
  },
  "audioType (p. 475)" : integer,
  "audioSourceName (p. 475)" : "string",
  "codecSettings (p. 476)" : {
    "codec (p. 474)" : enum,
    "wavSettings (p. 474)" : {?
"channels (p. 584)": integer,
"bitDepth (p. 584)": integer,
"sampleRate (p. 585)": integer
},
"ac3Settings (p. 474)": {
  "dynamicRangeCompressionProfile (p. 471)": enum,
  "dialnorm (p. 472)": integer,
  "codingMode (p. 472)": enum,
  "metadataControl (p. 472)": enum,
  "lifeFilter (p. 472)": enum,
  "bitrate (p. 472)": integer,
  "bitstreamMode (p. 472)": enum,
  "sampleRate (p. 472)": integer
},
"aacSettings (p. 474)": {
  "vbrQuality (p. 469)": enum,
  "codecProfile (p. 469)": enum,
  "codingMode (p. 469)": enum,
  "specification (p. 470)": enum,
  "bitrate (p. 470)": integer,
  "rawFormat (p. 470)": enum,
  "rateControlMode (p. 470)": enum,
  "sampleRate (p. 470)": integer,
  "audioDescriptionBroadcasterMix (p. 470)": enum
},
"aiffSettings (p. 474)": {
  "channels (p. 473)": integer,
  "bitDepth (p. 473)": integer,
  "sampleRate (p. 473)": integer
},
"eac3Settings (p. 474)": {
  "dialnorm (p. 499)": integer,
  "passThroughControl (p. 499)": enum,
  "metadataControl (p. 499)": enum,
  "bitrate (p. 499)": integer,
  "dynamicRangeCompressionRf (p. 500)": enum,
  "sampleRate (p. 500)": integer,
  "ltRtSurroundMixLevel (p. 500)": number,
  "surroundExMode (p. 500)": enum,
  "dynamicRangeCompressionLine (p. 500)": enum,
  "lifeControl (p. 500)": enum,
  "codingMode (p. 500)": enum,
  "surroundMode (p. 500)": enum,
  "attenuationControl (p. 501)": enum,
  "lifeFilter (p. 501)": enum,
  "phaseControl (p. 501)": enum,
  "ltRtCenterMixLevel (p. 501)": number,
  "dcfilter (p. 501)": enum,
  "stereoDownmix (p. 501)": enum,
  "bitstreamMode (p. 501)": enum,
  "loRoSurroundMixLevel (p. 501)": number,
  "loRoCenterMixLevel (p. 502)": number
},
"mp2Settings (p. 474)": {
  "channels (p. 555)": integer,
  "bitrate (p. 555)": integer,
  "sampleRate (p. 555)": integer
},
"languageCode (p. 476)": enum,
"streamName (p. 476)": "string",
"audioNormalizationSettings (p. 476)": {
  "targetLkfs (p. 477)": number,
  "algorithmControl (p. 477)": enum,
  "peakCalculation (p. 477)": enum,
  "loudnessLogging (p. 477)": enum,
  "targetLkfs (p. 477)": number,
  "algorithmControl (p. 477)": enum,
  "peakCalculation (p. 477)": enum,
  "loudnessLogging (p. 477)": enum,
"correctionGateLevel (p. 477)": integer,
"algorithm (p. 478)": enum
}
],
"containerSettings (p. 567)": {
"container (p. 488)": enum,
"mp4Settings (p. 488)": {
"mp4MajorBrand (p. 556)": "string",
"moovPlacement (p. 556)": enum,
"cslgAtom (p. 556)": enum,
"freeSpaceBox (p. 556)": enum
},
"m3u8Settings (p. 488)": {
"pmtPid (p. 551)": integer,
"pcrPid (p. 552)": integer,
"audioPids (p. 552)": [
  integer
],
"audioFramesPerPes (p. 552)": integer,
"scte35Pid (p. 552)": integer,
"transportStreamId (p. 552)": integer,
"videoPid (p. 552)": integer,
"pcrControl (p. 553)": enum,
"privateMetadataPid (p. 553)": integer,
"pmtInterval (p. 553)": integer,
"patInterval (p. 553)": integer,
"programNumber (p. 553)": integer,
"timedMetadataPid (p. 553)": integer,
"timedMetadata (p. 554)": enum,
"scte35Source (p. 554)": enum
},
"m2tsSettings (p. 489)": {
"dvbTeletextPid (p. 546)": integer,
"bitrate (p. 547)": integer,
"segmentationTime (p. 547)": number,
"audioPids (p. 547)": [
  integer
],
"rateMode (p. 547)": enum,
"ebpAudioInterval (p. 547)": enum,
"fragmentTime (p. 547)": number,
"audioFramesPerPes (p. 547)": integer,
"maxPcrInterval (p. 548)": integer,
"scte35Pid (p. 548)": integer,
"privateMetadataPid (p. 548)": integer,
"pmtInterval (p. 548)": integer,
"segmentationStyle (p. 548)": enum,
"audioBufferModel (p. 548)": enum,
"programNumber (p. 549)": integer,
"dvbNitSettings (p. 549)": {
  "networkName (p. 492)": "string",
  "networkId (p. 492)": integer,
  "nitInterval (p. 492)": integer
},
"scte35Source (p. 549)": enum,
"pmtPid (p. 549)": integer,
"bufferModel (p. 549)": enum,
"ebpPlacement (p. 549)": enum,
"dvbSdtSettings (p. 549)": {
  "sdtInterval (p. 493)": integer,
  "serviceName (p. 493)": "string",
  "serviceProviderName (p. 493)": "string",
  "outputSdt (p. 493)": enum
},
"nullPacketBitrate (p. 549)": number,
"pcrPid (p. 550)": integer,
"minEbpInterval (p. 550)": integer,
"transportStreamId (p. 550)": integer,
"pcrControl (p. 550)": enum,
"videoPid (p. 550)": integer,
"esRateInPes (p. 550)": enum,
"segmentationMarkers (p. 551)": enum,
"dvbTdtSettings (p. 551)": {
  "tdtInterval (p. 497)": integer
},
"patInterval (p. 551)": integer,
"dvbsubPids (p. 551)": [integer
],
"movSettings (p. 489)": {
  "reference (p. 555)": enum,
  "paddingControl (p. 555)": enum,
  "mpeg2FourCCControl (p. 555)": enum,
  "celgAtom (p. 555)": enum,
  "clapAtom (p. 555)": enum
},
"f4vSettings (p. 489)": {
  "moovPlacement (p. 503)": enum
},
"preset (p. 567)": "string",
"outputSettings (p. 567)": {
  "hlsSettings (p. 570)": {
    "iFrameOnlyManifest (p. 531)": enum,
    "segmentModifier (p. 531)": "string",
    "audioTrackType (p. 531)": enum,
    "audioRenditionSets (p. 531)": "string",
    "audioGroupId (p. 531)": "string"
  }
},
"captionDescriptions (p. 567)": [
  {
    "captionSelectorName (p. 484)": "string",
    "languageDescription (p. 484)": "string",
    "languageCode (p. 484)": enum,
    "destinationSettings (p. 484)": {
      "burninDestinationSettings (p. 484)": {
        "xPosition (p. 480)": integer,
        "backgroundColor (p. 480)": enum,
        "teletextSpacing (p. 480)": enum,
        "yPosition (p. 480)": integer,
        "backgroundColor (p. 481)": integer,
        "fontOpacity (p. 481)": integer,
        "shadowOpacity (p. 481)": integer,
        "fontResolution (p. 481)": integer,
        "shadowYOffset (p. 481)": integer,
        "outlineSize (p. 481)": integer,
        "outlineColor (p. 482)": enum,
        "fontSize (p. 482)": integer,
        "shadowXOffset (p. 482)": integer,
        "alignment (p. 482)": enum,
        "shadowColor (p. 482)": enum,
        "fontColor (p. 482)": enum
      },
      "teletextDestinationSettings (p. 485)": {
        "pageNumber (p. 576)": "string"
      }
    },
    "ttmlDestinationSettings (p. 485)": {
      "stylePassthrough (p. 579)": enum
    }
  }
}
"destinationType (p. 485)": enum,
"dvbSubDestinationSettings (p. 485)": {
  "xPosition (p. 493)": integer,
  "backgroundColor (p. 493)": enum,
  "teletextSpacing (p. 494)": enum,
  "yPosition (p. 494)": integer,
  "backgroundOpacity (p. 494)": integer,
  "fontOpacity (p. 494)": integer,
  "shadowOpacity (p. 494)": integer,
  "fontResolution (p. 494)": integer,
  "shadowYOffset (p. 495)": integer,
  "outlineSize (p. 495)": integer,
  "outlineColor (p. 495)": enum,
  "fontSize (p. 495)": integer,
  "shadowXOffset (p. 495)": integer,
  "alignment (p. 495)": enum,
  "shadowColor (p. 496)": enum,
  "fontColor (p. 496)": enum
},
"sccDestinationSettings (p. 485)": {
  "framerate (p. 575)": enum
}
"nameModifier (p. 568)": "string"
"outputGroupSettings (p. 568)": {
  "dashIsoGroupSettings (p. 569)": {
    "fragmentLength (p. 489)": integer,
    "baseUrl (p. 490)": "string",
    "minBufferTime (p. 490)": integer,
    "encryption (p. 490)": {
      "spekeKeyProvider (p. 489)": {
        "resourceId (p. 575)": "string",
        "systemIds (p. 575)": [ "string"
      },
      "url (p. 575)": "string"
    },
    "destination (p. 490)": "string",
    "segmentLength (p. 490)": integer,
    "segmentControl (p. 490)": enum,
    "hbbtvCompliance (p. 491)": enum
  },
  "fileGroupSettings (p. 569)": {
    "destination (p. 503)": "string"
  },
  "msSmoothGroupSettings (p. 569)": {
    "fragmentLength (p. 564)": integer,
    "encryption (p. 564)": {
      "spekeKeyProvider (p. 564)": {
        "resourceId (p. 575)": "string",
        "systemIds (p. 575)": [ "string"
      },
      "url (p. 575)": "string"
    },
    "audioDeduplication (p. 564)": enum,
    "manifestEncoding (p. 564)": enum,
    "destination (p. 564)": "string"
  },
  "type (p. 569)": enum,
"hlsGroupSettings (p. 569)": {
  "segmentsPerSubdirectory (p. 527)": integer,
  "streamInfResolution (p. 527)": enum,
  "timestampDeltaMilliseconds (p. 527)": integer,
  "outputSelection (p. 527)": enum,
  "captionLanguageMappings (p. 527)": [
    {
      "languageDescription (p. 525)": "string",
      "captionChannel (p. 525)": integer,
      "languageCode (p. 525)": enum
    }
  ],
  "clientCache (p. 528)": enum,
  "codecSpecification (p. 528)": enum,
  "destination (p. 528)": "string",
  "segmentControl (p. 528)": enum,
  "timedMetadataId3Frame (p. 528)": enum,
  "timedMetadataId3Period (p. 528)": integer,
  "captionLanguageSetting (p. 528)": enum,
  "minSegmentLength (p. 528)": integer,
  "directoryStructure (p. 529)": enum,
  "programDateTime (p. 529)": enum,
  "baseUrl (p. 529)": "string",
  "encryption (p. 529)": {
    "initializationVectorInManifest (p. 526)": enum,
    "constantInitializationVector (p. 526)": "string",
    "staticKeyProvider (p. 526)": {
      "keyFormatVersions (p. 575)": "string",
      "keyFormat (p. 575)": "string",
      "staticKeyValue (p. 576)": "string",
      "url (p. 576)": "string"
    },
    "type (p. 526)": enum,
    "spekeKeyProvider (p. 527)": {
      "resourceId (p. 575)": "string",
      "systemIds (p. 575)": ["string"
    ],
    "url (p. 575)": "string"
  },
  "encryptionMethod (p. 527)": enum
},
"adMarkers (p. 529)": [enum
],
"programDateTimePeriod (p. 529)": integer,
"manifestCompression (p. 529)": enum,
"segmentLength (p. 529)": integer,
"manifestDurationFormat (p. 530)": enum
},
"name (p. 568)": "string",
"customName (p. 569)": "string"
},
"timedMetadataInsertion (p. 540)": {
  "id3Insertions (p. 578)": [
    {
      "id3 (p. 532)": "string",
      "timecode (p. 532)": "string"
    }
  ],
  "availBlanking (p. 540)": {
    "availBlankingImage (p. 480)": "string"
Example CancelJobResponse

{
}

Example ExceptionBody

{
  "message (p. 503)" : "string"
}

Properties

AacAudioDescriptionBroadcasterMix (Enum)

Choose BROADCASTER_MIXED_AD when the input contains pre-mixed main audio + audio description (AD) as a stereo pair. The value for AudioType will be set to 3, which signals to downstream systems that this stream contains "broadcaster mixed AD". Note that the input received by the encoder must contain pre-mixed audio; the encoder does not perform the mixing. When you choose BROADCASTER_MIXED_AD, the encoder ignores any values you provide in AudioType and FollowInputAudioType. Choose NORMAL when the input does not contain pre-mixed audio + audio description (AD). In this case, the encoder will use any values you provide for AudioType and FollowInputAudioType.

BROADCASTER_MIXED_AD
NORMAL
AacCodecProfile (Enum)

AAC Profile.

LC
HEV1
HEV2

AacCodingMode (Enum)

Mono (Audio Description), Mono, Stereo, or 5.1 channel layout. Valid values depend on rate control mode and profile. "1.0 - Audio Description (Receiver Mix)" setting receives a stereo description plus control track and emits a mono AAC encode of the description track, with control data emitted in the PES header as per ETSI TS 101 154 Annex E.

AD_RECEIVER_MIX
CODING_MODE_1_0
CODING_MODE_1_1
CODING_MODE_2_0
CODING_MODE_5_1

AacRateControlMode (Enum)

Rate Control Mode.

CBR
VBR

AacRawFormat (Enum)

Enables LATM/LOAS AAC output. Note that if you use LATM/LOAS AAC in an output, you must choose "No container" for the output container.

LATM_LOAS
NONE

AacSettings

vbrQuality

Type: AacVbrQuality (p. 470)
Required: False

codecProfile

Type: AacCodecProfile (p. 469)
Required: False

codingMode

Type: AacCodingMode (p. 469)
Required: True
specification

Type: AacSpecification (p. 470)
Required: False

bitrate

Average bitrate in bits/second. Valid values depend on rate control mode and profile.

Type: integer
Required: False
Minimum: 6000
Maximum: 1024000

rawFormat

Type: AacRawFormat (p. 469)
Required: False

rateControlMode

Type: AacRateControlMode (p. 469)
Required: False

sampleRate

Sample rate in Hz. Valid values depend on rate control mode and profile.

Type: integer
Required: True
Minimum: 8000
Maximum: 96000

audioDescriptionBroadcasterMix

Type: AacAudioDescriptionBroadcasterMix (p. 468)
Required: False

AacSpecification (Enum)

Use MPEG-2 AAC instead of MPEG-4 AAC audio for raw or MPEG-2 Transport Stream containers.

MPEG2
MPEG4

AacVbrQuality (Enum)

VBR quality level. Only used if the rate control mode (AacRateControlMode) is VBR.

LOW
MEDIUM_LOW
MEDIUM_HIGH
HIGH
**Ac3BitstreamMode (Enum)**

Specifies the "Bitstream Mode" (bsmod) for the emitted AC-3 stream. See ATSC A/52-2012 for background on these values.

- COMPLETE_MAIN
- COMMENTARY
- DIALOGUE
- EMERGENCY
- HEARING_IMPAIRED
- MUSIC_AND_EFFECTS
- VISUALLY_IMPAIRED
- VOICE_OVER

**Ac3CodingMode (Enum)**

Dolby Digital coding mode. Determines number of channels.

- CODING_MODE_1_0
- CODING_MODE_1_1
- CODING_MODE_2_0
- CODING_MODE_3_2_LFE

**Ac3DynamicRangeCompressionProfile (Enum)**

If set to FILM_STANDARD, adds dynamic range compression signaling to the output bitstream as defined in the Dolby Digital specification.

- FILM_STANDARD
- NONE

**Ac3LfeFilter (Enum)**

Applies a 120Hz lowpass filter to the LFE channel prior to encoding. Only valid with 3_2_LFE coding mode.

- ENABLED
- DISABLED

**Ac3MetadataControl (Enum)**

When set to FOLLOW_INPUT, encoder metadata will be sourced from the DD, DD+, or DolbyE decoder that supplied this audio data. If audio was not supplied from one of these streams, then the static metadata settings will be used.

- FOLLOW_INPUT
- USE_CONFIGURED

**Ac3Settings**

dynamicRangeCompressionProfile

*Type:* Ac3DynamicRangeCompressionProfile (p. 471)  
*Required:* False
**dialnorm**

Sets the dialnorm for the output. If blank and input audio is Dolby Digital, dialnorm will be passed through.

- **Type:** integer
- **Required:** False
- **Minimum:** 1
- **Maximum:** 31

**codingMode**

- **Type:** Ac3CodingMode (p. 471)
- **Required:** False

**metadataControl**

- **Type:** Ac3MetadataControl (p. 471)
- **Required:** False

**lfeFilter**

- **Type:** Ac3LfeFilter (p. 471)
- **Required:** False

**bitrate**

Average bitrate in bits/second. Valid bitrates depend on the coding mode.

- **Type:** integer
- **Required:** False
- **Minimum:** 64000
- **Maximum:** 640000

**bitstreamMode**

- **Type:** Ac3BitstreamMode (p. 471)
- **Required:** False

**sampleRate**

Sample rate in hz. Sample rate is always 48000.

- **Type:** integer
- **Required:** False
- **Minimum:** 48000
- **Maximum:** 48000

**AfdSignaling (Enum)**

This setting only applies to H.264 and MPEG2 outputs. Use Insert AFD signaling (AfdSignaling) to whether there are AFD values in the output video data and what those values are.

* Choose None to remove all AFD values from this output.
* Choose Fixed to ignore input AFD values and instead encode the value specified in the job.
* Choose Auto to calculate output AFD values based on the input AFD scaler data.

```
NONE
AUTO
FIXED
```

**AiffSettings**

**channels**

Set Channels to specify the number of channels in this output audio track. Choosing Mono in the console will give you 1 output channel; choosing Stereo will give you 2. In the API, valid values are 1 and 2.

- **Type:** integer
- **Required:** False
- **Minimum:** 1
- **Maximum:** 2

**bitDepth**

Specify Bit depth (BitDepth), in bits per sample, to choose the encoding quality for this audio track.

- **Type:** integer
- **Required:** False
- **Minimum:** 16
- **Maximum:** 24

**sampleRate**

Sample rate in hz.

- **Type:** integer
- **Required:** False
- **Minimum:** 8000
- **Maximum:** 192000

**AncillarySourceSettings**

**sourceAncillaryChannelNumber**

Specifies the 608 channel number in the ancillary data track from which to extract captions. Unused for passthrough.

- **Type:** integer
- **Required:** False
- **Minimum:** 1
- **Maximum:** 4

**AntiAlias (Enum)**

Enable Anti-alias (AntiAlias) to enhance sharp edges in video output when your input resolution is much larger than your output resolution. Default is enabled.
**Properties**

- **DISABLED**
- **ENABLED**

**AudioCodec (Enum)**

Type of Audio codec.

- AAC
- MP2
- WAV
- AIFF
- AC3
- EAC3
- PASSTHROUGH

**AudioCodecSettings**

**codec**

- Type: AudioCodec (p. 474)
- Required: True

**wavSettings**

- Type: WavSettings (p. 584)
- Required: False

**ac3Settings**

- Type: Ac3Settings (p. 471)
- Required: False

**aacSettings**

- Type: AacSettings (p. 469)
- Required: False

**aiffSettings**

- Type: AiffSettings (p. 473)
- Required: False

**eac3Settings**

- Type: Eac3Settings (p. 499)
- Required: False

**mp2Settings**

- Type: Mp2Settings (p. 555)
Required: False

AudioDefaultSelection (Enum)

When an "Audio Description":#audio_description specifies an AudioSelector or AudioSelectorGroup for which no matching source is found in the input, then the audio selector marked as DEFAULT will be used. If none are marked as default, silence will be inserted for the duration of the input.

- DEFAULT
- NOT_DEFAULT

AudioDescription

languageCodeControl

Type: AudioLanguageCodeControl (p. 476)
Required: False

audioTypeControl

Type: AudioTypeControl (p. 479)
Required: False

remixSettings

Advanced audio remixing settings.

Type: RemixSettings (p. 573)
Required: False

audioType

Applies only if Follow Input Audio Type is unchecked (false). A number between 0 and 255. The following are defined in ISO-IEC 13818-1: 0 = Undefined, 1 = Clean Effects, 2 = Hearing Impaired, 3 = Visually Impaired Commentary, 4-255 = Reserved.

Type: integer
Required: False
Minimum: 0
Maximum: 255

audioSourceName

Specifies which audio data to use from each input. In the simplest case, specify an "Audio Selector":#inputs-audio_selector by name based on its order within each input. For example if you specify "Audio Selector 3", then the third audio selector will be used from each input. If an input does not have an "Audio Selector 3", then the audio selector marked as "default" in that input will be used. If there is no audio selector marked as "default", silence will be inserted for the duration of that input. Alternatively, an "Audio Selector Group":#inputs-audio_selector_group name may be specified, with similar default/silence behavior. If no audio_source_name is specified, then "Audio Selector 1" will be chosen automatically.

Type: string
Required: False
codecSettings

Type: AudioCodecSettings (p. 474)
Required: True

languageCode

Indicates the language of the audio output track. The ISO 639 language specified in the 'Language Code' drop down will be used when 'Follow Input Language Code' is not selected or when 'Follow Input Language Code' is selected but there is no ISO 639 language code specified by the input.

Type: LanguageCode (p. 541)
Required: False

streamName

Used for Microsoft Smooth Streaming and Apple HLS outputs. Indicates the name displayed by the player (eg. English, or Director Commentary). Alphanumeric characters, spaces, and underscore are legal.

Type: string
Required: False
Pattern: ^[\w\s]+$

audioNormalizationSettings

Advanced audio normalization settings.

Type: AudioNormalizationSettings (p. 477)
Required: False

AudioLanguageCodeControl (Enum)

Choosing FOLLOW_INPUT will cause the ISO 639 language code of the output to follow the ISO 639 language code of the input. The language specified for languageCode will be used when USE_CONFIGURED is selected or when FOLLOW_INPUT is selected but there is no ISO 639 language code specified by the input.

FOLLOW_INPUT
USE_CONFIGURED

AudioNormalizationAlgorithm (Enum)

Audio normalization algorithm to use. 1770-1 conforms to the CALM Act specification, 1770-2 conforms to the EBU R-128 specification.

ITU_BS_1770_1
ITU_BS_1770_2

AudioNormalizationAlgorithmControl (Enum)

When enabled the output audio is corrected using the chosen algorithm. If disabled, the audio will be measured but not adjusted.

CORRECT_AUDIO
**Properties**

**MEASURE_ONLY**

**AudioNormalizationLoudnessLogging (Enum)**
If set to LOG, log each output's audio track loudness to a CSV file.

- LOG
- DONT_LOG

**AudioNormalizationPeakCalculation (Enum)**
If set to TRUE_PEAK, calculate and log the TruePeak for each output's audio track loudness.

- TRUE_PEAK
- NONE

**AudioNormalizationSettings**

**targetLkfs**
Target LKFS (loudness) to adjust volume to. If no value is entered, a default value will be used according to the chosen algorithm. The CALM Act (1770-1) recommends a target of -24 LKFS. The EBU R-128 specification (1770-2) recommends a target of -23 LKFS.

- Type: number
- Required: False
- Format: float
- Minimum: -59.0
- Maximum: 0.0

**algorithmControl**

- Type: AudioNormalizationAlgorithmControl (p. 476)
- Required: False

**peakCalculation**

- Type: AudioNormalizationPeakCalculation (p. 477)
- Required: False

**loudnessLogging**

- Type: AudioNormalizationLoudnessLogging (p. 477)
- Required: False

**correctionGateLevel**
Content measuring above this level will be corrected to the target level. Content measuring below this level will not be corrected. Gating only applies when not using real_time_correction.

- Type: integer
- Required: False
- Minimum: -70
Maximum: 0

algorithm

Type: AudioNormalizationAlgorithm (p. 476)
Required: False

AudioSelector

remixSettings

Advanced audio remixing settings.

Type: RemixSettings (p. 573)
Required: False

programSelection

Applies only when input streams contain Dolby E. Enter the program ID (according to the metadata in the audio) of the Dolby E program to extract from the specified track. One program extracted per audio selector. To select multiple programs, create multiple selectors with the same Track and different Program numbers. "All channels" means to ignore the program IDs and include all the channels in this selector; useful if metadata is known to be incorrect.

Type: integer
Required: False
Minimum: 0
Maximum: 8

offset

Specifies a time delta in milliseconds to offset the audio from the input video.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

defaultSelection

Type: AudioDefaultSelection (p. 475)
Required: False

selectorType

Type: AudioSelectorType (p. 479)
Required: False

languageCode

Selects a specific language code from within an audio source.

Type: LanguageCode (p. 541)
Required: False
pids

Selects a specific PID from within an audio source (e.g. 257 selects PID 0x101).

Type: Array of type integer
Required: False

externalAudioFileInput

Specifies audio data from an external file source. Auto populated when Infer External Filename is checked

Type: string
Required: False

tracks

Identify the channel to include in this selector by entering the 1-based track index. To combine several tracks, enter a comma-separated list, e.g. "1,2,3" for tracks 1-3.

Type: Array of type integer
Required: False

AudioSelectorGroup

audioSelectorNames

Name of an "Audio Selector": #inputs-audio_selector within the same input to include in the group. Audio selector names are standardized, based on their order within the input (e.g. "Audio Selector 1"). The audio_selector_name parameter can be repeated to add any number of audio selectors to the group.

Type: Array of type string
Required: True

AudioSelectorType (Enum)

Specifies the type of the audio selector.

PID
TRACK
LANGUAGE_CODE

AudioTypeControl (Enum)

When set to FOLLOW_INPUT, if the input contains an ISO 639 audio_type, then that value is passed through to the output. If the input contains no ISO 639 audio_type, the value in Audio Type is included
in the output. Otherwise the value in Audio Type is included in the output. Note that this field and audioType are both ignored if audioDescriptionBroadcasterMix is set to BROADCASTER_MIXED_AD.

FOLLOW_INPUT
USE_CONFIGURED

**AvailBlanking**

*availBlankingImage*

Blanking image to be used. Leave empty for solid black. Only bmp and png images are supported.

- **Type**: string
- **Required**: False
- **Pattern**: `^s3:/(.*)\.(bmp|BMP|png|PNG)$`

**BurninDestinationSettings**

*xPosition*

Specifies the horizontal position of the caption relative to the left side of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the left of the output. If no explicit x_position is provided, the horizontal caption position will be determined by the alignment parameter. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 2147483647

*backgroundColor*

- **Type**: BurninSubtitleBackgroundColor (p. 483)
- **Required**: False

*teletextSpacing*

- **Type**: BurninSubtitleTeletextSpacing (p. 483)
- **Required**: False

*yPosition*

Specifies the vertical position of the caption relative to the top of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the top of the output. If no explicit y_position is provided, the caption will be positioned towards the bottom of the output. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 2147483647

480
backgroundOpacity

Specifies the opacity of the background rectangle. 255 is opaque; 0 is transparent. Leaving this parameter blank is equivalent to setting it to 0 (transparent). All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: 0
Maximum: 255

fontOpacity

Specifies the opacity of the burned-in captions. 255 is opaque; 0 is transparent. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: True
Minimum: 0
Maximum: 255

shadowOpacity

Specifies the opacity of the shadow. 255 is opaque; 0 is transparent. Leaving this parameter blank is equivalent to setting it to 0 (transparent). All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: 0
Maximum: 255

fontResolution

Font resolution in DPI (dots per inch); default is 96 dpi. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: 96
Maximum: 600

shadowYOffset

Specifies the vertical offset of the shadow relative to the captions in pixels. A value of -2 would result in a shadow offset 2 pixels above the text. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

outlineSize

Specifies font outline size in pixels. This option is not valid for source captions that are either 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.
Properties

Type: integer
Required: True
Minimum: 0
Maximum: 10

outlineColor

Type: BurninSubtitleOutlineColor (p. 483)
Required: True

tfontSize

A positive integer indicates the exact font size in points. Set to 0 for automatic font size selection. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: 0
Maximum: 96

shadowXOffset

Specifies the horizontal offset of the shadow relative to the captions in pixels. A value of -2 would result in a shadow offset 2 pixels to the left. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

alignment

Type: BurninSubtitleAlignment (p. 482)
Required: True

shadowColor

Type: BurninSubtitleShadowColor (p. 483)
Required: False

fontColor

Type: BurninSubtitleFontColor (p. 483)
Required: False

BurninSubtitleAlignment (Enum)

If no explicit x_position or y_position is provided, setting alignment to centered will place the captions at the bottom center of the output. Similarly, setting a left alignment will align captions to the bottom left of the output. If x and y positions are given in conjunction with the alignment parameter, the font will be justified (either left or centered) relative to those coordinates. This option is not valid for source captions.
that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

- CENTERED
- LEFT

**BurninSubtitleBackgroundColor (Enum)**

Specifies the color of the rectangle behind the captions. All burn-in and DVB-Sub font settings must match.

- NONE
- BLACK
- WHITE

**BurninSubtitleFontColor (Enum)**

Specifies the color of the burned-in captions. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

- WHITE
- BLACK
- YELLOW
- RED
- GREEN
- BLUE

**BurninSubtitleOutlineColor (Enum)**

Specifies font outline color. This option is not valid for source captions that are either 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

- BLACK
- WHITE
- YELLOW
- RED
- GREEN
- BLUE

**BurninSubtitleShadowColor (Enum)**

Specifies the color of the shadow cast by the captions. All burn-in and DVB-Sub font settings must match.

- NONE
- BLACK
- WHITE

**BurninSubtitleTeletextSpacing (Enum)**

Controls whether a fixed grid size or proportional font spacing will be used to generate the output subtitles bitmap. Only applicable for Teletext inputs and DVB-Sub/Burn-in outputs.
CancelJobRequest

id

- **Type**: string
- **Required**: True

CaptionDescription

captionSelectorName

Specifies which "Caption Selector":#inputs-caption_selector to use from each input when generating captions. The name should be of the format "Caption Selector <N>", which denotes that the Nth Caption Selector will be used from each input.

- **Type**: string
- **Required**: True

languageDescription

Human readable information to indicate captions available for players (eg, English, or Spanish). Alphanumeric characters, spaces, and underscore are legal.

- **Type**: string
- **Required**: False
- **Pattern**: `^[\w ]*$`

languageCode

Indicates the language of the caption output track.

- **Type**: LanguageCode (p. 541)
- **Required**: False

destinationSettings

- **Type**: CaptionDestinationSettings (p. 484)
- **Required**: False

CaptionDestinationSettings

burninDestinationSettings

- **Type**: BurninDestinationSettings (p. 480)
- **Required**: False
teletextDestinationSettings

  Type: TeletextDestinationSettings (p. 576)
  Required: False

ttmlDestinationSettings

  Type: TtmlDestinationSettings (p. 579)
  Required: False

destinationType

  Type: CaptionDestinationType (p. 485)
  Required: True

dvbSubDestinationSettings

  Type: DvbSubDestinationSettings (p. 493)
  Required: False

sccDestinationSettings

  Type: SccDestinationSettings (p. 575)
  Required: False

CaptionDestinationType (Enum)

  BURN_IN
  DVB_SUB
  EMBEDDED
  SCC
  SRT
  TELETEXT
  TTML
  WEBVTT

CaptionSelector

sourceSettings

  Type: CaptionSourceSettings (p. 486)
  Required: True

languageCode

  The specific language to extract from source. If input is SCTE-27, complete this field and/or PID to select the caption language to extract. If input is DVB-Sub and output is Burn-in or SMPTE-TT, complete this field and/or PID to select the caption language to extract. If input is DVB-Sub that is being passed through, omit this field (and PID field); there is no way to extract a specific language with pass-through captions.

  Type: LanguageCode (p. 541)
  Required: False
CaptionSourceSettings

fileSourceSettings
  Type: FileSourceSettings (p. 504)
  Required: False

ancillarySourceSettings
  Type: AncillarySourceSettings (p. 473)
  Required: False

embeddedSourceSettings
  Type: EmbeddedSourceSettings (p. 502)
  Required: False

sourceType
  Type: CaptionSourceType (p. 486)
  Required: True

dvbSubSourceSettings
  Type: DvbSubSourceSettings (p. 496)
  Required: False

teletextSourceSettings
  Type: TeletextSourceSettings (p. 576)
  Required: False

CaptionSourceType (Enum)

Use Source (SourceType) to identify the format of your input captions. The service cannot auto-detect caption format.

  ANCILLARY
  DVB_SUB
  EMBEDDED
  SCC
  TTML
  STL
  SRT
  TELETEXT
  NULL_SOURCE

ChannelMapping

outputChannels
  Type: Array of type OutputChannelMapping (p. 568)
Required: True

**ColorCorrector**

**saturation**
Saturation level.

Type: integer
Required: False
Minimum: 1
Maximum: 100

**brightness**
Brightness level.

Type: integer
Required: False
Minimum: 1
Maximum: 100

**hdr10Metadata**
Type: [Hdr10Metadata](p. 522)
Required: False

**contrast**
Contrast level.

Type: integer
Required: False
Minimum: 1
Maximum: 100

**hue**
Hue in degrees.

Type: integer
Required: False
Minimum: -180
Maximum: 180

**colorSpaceConversion**
Type: [ColorSpaceConversion](p. 488)
Required: False

**ColorMetadata (Enum)**
Enable Insert color metadata (ColorMetadata) to include color metadata in this output. This setting is enabled by default.
ColorSpace (Enum)

Specifies the colorspace of an input. This setting works in tandem with "Color Corrector":#color_corrector > color_space_conversion to determine if any conversion will be performed.

FOLLOW
REC_601
REC_709
HDR10
HLG_2020

ColorSpaceConversion (Enum)

Determines if colorspace conversion will be performed. If set to _None_, no conversion will be performed. If _Force 601_ or _Force 709_ are selected, conversion will be performed for inputs with differing colorspaces. An input's colorspace can be specified explicitly in the "Video Selector":#inputs-video_selector if necessary.

NONE
FORCE_601
FORCE_709
FORCE_HDR10
FORCE_HLG_2020

ColorSpaceUsage (Enum)

There are two sources for color metadata, the input file and the job configuration. This enum controls which takes precedence. FORCE: System will use color metadata supplied by user, if any. If the user does not supply color metadata the system will use data from the source. FALLBACK: System will use color metadata from the source. If source has no color metadata, the system will use user-supplied color metadata values if available.

FORCE
FALLBACK

ContainerSettings

container

Type: ContainerType (p. 489)
Required: True

mp4Settings

Type: Mp4Settings (p. 556)
Required: False

m3u8Settings

Type: M3u8Settings (p. 551)
Required: False

**m2tsSettings**

*Type: M2tsSettings (p. 546)*  
*Required: False*

**movSettings**

*Type: MovSettings (p. 555)*  
*Required: False*

**f4vSettings**

*Type: F4vSettings (p. 503)*  
*Required: False*

**ContainerType (Enum)**

Container for this output. Some containers require a container settings object. If not specified, the default object will be created.

F4V  
ISMV  
M2TS  
M3U8  
MOV  
MP4  
MPD  
MXF  
RAW

**DashIsoEncryptionSettings**

**spekeKeyProvider**

*Type: SpekeKeyProvider (p. 575)*  
*Required: True*

**DashIsoGroupSettings**

**fragmentLength**

Length of fragments to generate (in seconds). Fragment length must be compatible with GOP size and Framerate. Note that fragments will end on the next keyframe after this number of seconds, so actual fragment length may be longer. When Emit Single File is checked, the fragmentation is internal to a single output file and it does not cause the creation of many output files as in other output types.

*Type: integer*  
*Required: True*  
*Minimum: 1*
Maximum: 2147483647

baseUrl

A partial URI prefix that will be put in the manifest (.mpd) file at the top level BaseURL element. Can be used if streams are delivered from a different URL than the manifest file.

- **Type:** string
- **Required:** False

minBufferTime

Minimum time of initially buffered media that is needed to ensure smooth playout.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 2147483647

encryption

DRM settings.

- **Type:** DashIsoEncryptionSettings (p. 489)
- **Required:** False

destination

Use Destination (Destination) to specify the S3 output location and the output filename base. Destination accepts format identifiers. If you do not specify the base filename in the URI, the service will use the filename of the input file. If your job has multiple inputs, the service uses the filename of the first input file.

- **Type:** string
- **Required:** True
- **Pattern:** ^s3:/\/

segmentLength

Length of mpd segments to create (in seconds). Note that segments will end on the next keyframe after this number of seconds, so actual segment length may be longer. When Emit Single File is checked, the segmentation is internal to a single output file and it does not cause the creation of many output files as in other output types.

- **Type:** integer
- **Required:** True
- **Minimum:** 1
- **Maximum:** 2147483647

segmentControl

- **Type:** DashIsoSegmentControl (p. 491)
- **Required:** False
hbbtvCompliance

Type: DashIsoHbbtvCompliance (p. 491)
Required: False

DashIsoHbbtvCompliance (Enum)
Supports HbbTV specification as indicated

- HBBTV_1_5
- NONE

DashIsoSegmentControl (Enum)
When set to SINGLE_FILE, a single output file is generated, which is internally segmented using the Fragment Length and Segment Length. When set to SEGMENTED_FILES, separate segment files will be created.

- SINGLE_FILE
- SEGMENTED_FILES

DeinterlaceAlgorithm (Enum)
Only applies when you set Deinterlacer (DeinterlaceMode) to Deinterlace (DEINTERLACE) or Adaptive (ADAPTIVE). Motion adaptive interpolate (INTERPOLATE) produces sharper pictures, while blend (BLEND) produces smoother motion. Use (INTERPOLATE_TICKER) OR (BLEND_TICKER) if your source file includes a ticker, such as a scrolling headline at the bottom of the frame.

- INTERPOLATE
- INTERPOLATE_TICKER
- BLEND
- BLEND_TICKER

Deinterlacer

mode

Type: DeinterlacerMode (p. 492)
Required: False

control

Type: DeinterlacerControl (p. 491)
Required: False

algorithm

Type: DeinterlaceAlgorithm (p. 491)
Required: False

DeinterlacerControl (Enum)
- When set to NORMAL (default), the deinterlacer does not convert frames that are tagged in metadata as progressive. It will only convert those that are tagged as some other type. - When set to
FORCE_ALL_FRAMES, the deinterlacer converts every frame to progressive - even those that are already tagged as progressive. Turn Force mode on only if there is a good chance that the metadata has tagged frames as progressive when they are not progressive. Do not turn on otherwise; processing frames that are already progressive into progressive will probably result in lower quality video.

FORCE_ALL_FRAMES
NORMAL

**DeinterlacerMode (Enum)**

Use Deinterlacer (DeinterlaceMode) to choose how the service will do deinterlacing. Default is Deinterlace. - Deinterlace converts interlaced to progressive. - Inverse telecine converts Hard Telecine 29.97i to progressive 23.976p. - Adaptive auto-detects and converts to progressive.

DEINTERLACE
INVERSE_TELECINE
ADAPTIVE

**DropFrameTimecode (Enum)**

Applies only to 29.97 fps outputs. When this feature is enabled, the service will use drop-frame timecode on outputs. If it is not possible to use drop-frame timecode, the system will fall back to non-drop-frame. This setting is enabled by default when Timecode insertion (TimecodeInsertion) is enabled.

DISABLED
ENABLED

**DvbNitSettings**

**networkName**

The network name text placed in the network_name_descriptor inside the Network Information Table. Maximum length is 256 characters.

- **Type**: string
- **Required**: True

**networkId**

The numeric value placed in the Network Information Table (NIT).

- **Type**: integer
- **Required**: True
- **Minimum**: 0
- **Maximum**: 65535

**nitInterval**

The number of milliseconds between instances of this table in the output transport stream.

- **Type**: integer
- **Required**: True
- **Minimum**: 25
**Maximum**: 10000

**DvbSdtSettings**

**sdtInterval**

The number of milliseconds between instances of this table in the output transport stream.

- **Type**: integer
- **Required**: False
- **Minimum**: 25
- **Maximum**: 2000

**serviceName**

The service name placed in the service_descriptor in the Service Description Table. Maximum length is 256 characters.

- **Type**: string
- **Required**: False

**serviceProviderName**

The service provider name placed in the service_descriptor in the Service Description Table. Maximum length is 256 characters.

- **Type**: string
- **Required**: False

**outputSdt**

- **Type**: OutputSdt (p. 570)
- **Required**: False

**DvbSubDestinationSettings**

**xPosition**

Specifies the horizontal position of the caption relative to the left side of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the left of the output. If no explicit x_position is provided, the horizontal caption position will be determined by the alignment parameter. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 2147483647

**backgroundColor**

- **Type**: DvbSubtitleBackgroundColor (p. 496)
- **Required**: False
teletextSpacing

Type: DvbSubtitleTeletextSpacing (p. 497)
Required: False

yPosition

Specifies the vertical position of the caption relative to the top of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the top of the output. If no explicit y_position is provided, the caption will be positioned towards the bottom of the output. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

backgroundOpacity

Specifies the opacity of the background rectangle. 255 is opaque; 0 is transparent. Leaving this parameter blank is equivalent to setting it to 0 (transparent). All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: 0
Maximum: 255

fontOpacity

Specifies the opacity of the burned-in captions. 255 is opaque; 0 is transparent. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: True
Minimum: 0
Maximum: 255

shadowOpacity

Specifies the opacity of the shadow. 255 is opaque; 0 is transparent. Leaving this parameter blank is equivalent to setting it to 0 (transparent). All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: 0
Maximum: 255

fontResolution

Font resolution in DPI (dots per inch); default is 96 dpi. All burn-in and DVB-Sub font settings must match.

Type: integer
**shadowYOffset**

Specifies the vertical offset of the shadow relative to the captions in pixels. A value of -2 would result in a shadow offset 2 pixels above the text. All burn-in and DVB-Sub font settings must match.

**Type:** integer  
**Required:** False  
**Minimum:** -2147483648  
**Maximum:** 2147483647

**outlineSize**

Specifies font outline size in pixels. This option is not valid for source captions that are either 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

**Type:** integer  
**Required:** True  
**Minimum:** 0  
**Maximum:** 10

**outlineColor**

**Type:** DvbSubtitleOutlineColor (p. 497)  
**Required:** True

**fontSize**

A positive integer indicates the exact font size in points. Set to 0 for automatic font size selection. All burn-in and DVB-Sub font settings must match.

**Type:** integer  
**Required:** False  
**Minimum:** 0  
**Maximum:** 96

**shadowXOffset**

Specifies the horizontal offset of the shadow relative to the captions in pixels. A value of -2 would result in a shadow offset 2 pixels to the left. All burn-in and DVB-Sub font settings must match.

**Type:** integer  
**Required:** False  
**Minimum:** -2147483648  
**Maximum:** 2147483647

**alignment**

**Type:** DvbSubtitleAlignment (p. 496)  
**Required:** True
### shadowColor

**Type:** DvbSubtitleShadowColor (p. 497)  
**Required:** False

### fontColor

**Type:** DvbSubtitleFontColor (p. 496)  
**Required:** False

### DvbSubSourceSettings

#### pid

When using DVB-Sub with Burn-In or SMPTE-TT, use this PID for the source content. Unused for DVB-Sub passthrough. All DVB-Sub content is passed through, regardless of selectors.

**Type:** integer  
**Required:** False  
**Minimum:** 1  
**Maximum:** 2147483647

### DvbSubtitleAlignment (Enum)

If no explicit x_position or y_position is provided, setting alignment to centered will place the captions at the bottom center of the output. Similarly, setting a left alignment will align captions to the bottom left of the output. If x and y positions are given in conjunction with the alignment parameter, the font will be justified (either left or centered) relative to those coordinates. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

- CENTERED
- LEFT

### DvbSubtitleBackgroundColor (Enum)

Specifies the color of the rectangle behind the captions. All burn-in and DVB-Sub font settings must match.

- NONE
- BLACK
- WHITE

### DvbSubtitleFontColor (Enum)

Specifies the color of the burned-in captions. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

- WHITE
- BLACK
- YELLOW
- RED
- GREEN
- BLUE
DvbSubtitleOutlineColor (Enum)

Specifies font outline color. This option is not valid for source captions that are either 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

BLACK
WHITE
YELLOW
RED
GREEN
BLUE

DvbSubtitleShadowColor (Enum)

Specifies the color of the shadow cast by the captions. All burn-in and DVB-Sub font settings must match.

NONE
BLACK
WHITE

DvbSubtitleTeletextSpacing (Enum)

Controls whether a fixed grid size or proportional font spacing will be used to generate the output subtitles bitmap. Only applicable for Teletext inputs and DVB-Sub/Burn-in outputs.

FIXED_GRID
PROPORTIONAL

DvbTdtSettings

tdtInterval

The number of milliseconds between instances of this table in the output transport stream.

Type: integer
Required: True
Minimum: 1000
Maximum: 30000

Eac3AttenuationControl (Enum)

If set to ATTENUATE_3_DB, applies a 3 dB attenuation to the surround channels. Only used for 3/2 coding mode.

ATTENUATE_3_DB
NONE

Eac3BitstreamMode (Enum)

Specifies the "Bitstream Mode" (bsmod) for the emitted E-AC-3 stream. See ATSC A/52-2012 (Annex E) for background on these values.

COMPLETE_MAIN
Eac3CodingMode (Enum)
Dolby Digital Plus coding mode. Determines number of channels.

- CODING_MODE_1_0
- CODING_MODE_2_0
- CODING_MODE_3_2

Eac3DcFilter (Enum)
Activates a DC highpass filter for all input channels.

- ENABLED
- DISABLED

Eac3DynamicRangeCompressionLine (Enum)
Enables Dynamic Range Compression that restricts the absolute peak level for a signal.

- NONE
- FILM_STANDARD
- FILM_LIGHT
- MUSIC_STANDARD
- MUSIC_LIGHT
- SPEECH

Eac3DynamicRangeCompressionRf (Enum)
Enables Heavy Dynamic Range Compression, ensures that the instantaneous signal peaks do not exceed specified levels.

- NONE
- FILM_STANDARD
- FILM_LIGHT
- MUSIC_STANDARD
- MUSIC_LIGHT
- SPEECH

Eac3LfeControl (Enum)
When encoding 3/2 audio, controls whether the LFE channel is enabled

- LFE
- NO_LFE

Eac3LfeFilter (Enum)
Applies a 120Hz lowpass filter to the LFE channel prior to encoding. Only valid with 3_2_LFE coding mode.
ENABLED
DISABLED

**Eac3MetadataControl (Enum)**

When set to FOLLOW_INPUT, encoder metadata will be sourced from the DD, DD+, or DolbyE decoder that supplied this audio data. If audio was not supplied from one of these streams, then the static metadata settings will be used.

FOLLOW_INPUT
USE_CONFIGURED

**Eac3PassthroughControl (Enum)**

When set to WHEN_POSSIBLE, input DD+ audio will be passed through if it is present on the input. This detection is dynamic over the life of the transcode. Inputs that alternate between DD+ and non-DD+ content will have a consistent DD+ output as the system alternates between passthrough and encoding.

WHEN_POSSIBLE
NO_PASSTHROUGH

**Eac3PhaseControl (Enum)**

Controls the amount of phase-shift applied to the surround channels. Only used for 3/2 coding mode.

SHIFT_90_DEGREES
NO_SHIFT

**Eac3Settings**

dialnorm

Sets the dialnorm for the output. If blank and input audio is Dolby Digital Plus, dialnorm will be passed through.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 31

**passthroughControl**

- **Type**: Eac3PassthroughControl (p. 499)
- **Required**: False

**metadataControl**

- **Type**: Eac3MetadataControl (p. 499)
- **Required**: False

**bitrate**

Average bitrate in bits/second. Valid bitrates depend on the coding mode.
Properties

Type: integer
Required: False
Minimum: 64000
Maximum: 640000

dynamicRangeCompressionRf
Type: Eac3DynamicRangeCompressionRf (p. 498)
Required: False

sampleRate
Sample rate in hz. Sample rate is always 48000.
Type: integer
Required: False
Minimum: 48000
Maximum: 48000

ltRtSurroundMixLevel
Left total/Right total surround mix level. Only used for 3/2 coding mode. Valid values: -1.5 -3.0 -4.5 -6.0 -60
Type: number
Required: False
Format: float
Minimum: -60.0
Maximum: -1.5

surroundExMode
Type: Eac3SurroundExMode (p. 502)
Required: False

dynamicRangeCompressionLine
Type: Eac3DynamicRangeCompressionLine (p. 498)
Required: False

lfeControl
Type: Eac3LfeControl (p. 498)
Required: False

codingMode
Type: Eac3CodingMode (p. 498)
Required: False

surroundMode
Type: Eac3SurroundMode (p. 502)
Properties

Required: False

attenuationControl
Type: Eac3AttenuationControl (p. 497)
Required: False

lfeFilter
Type: Eac3LfeFilter (p. 498)
Required: False

phaseControl
Type: Eac3PhaseControl (p. 499)
Required: False

ltRtCenterMixLevel
Left total/Right total center mix level. Only used for 3/2 coding mode. Valid values: 3.0, 1.5, 0.0, -1.5
-3.0, -4.5, -6.0, -60
Type: number
Required: False
Format: float
Minimum: -60.0
Maximum: 3.0

dcFilter
Type: Eac3DcFilter (p. 498)
Required: False

stereoDownmix
Type: Eac3StereoDownmix (p. 502)
Required: False

bitstreamMode
Type: Eac3BitstreamMode (p. 497)
Required: False

loRoSurroundMixLevel
Left only/Right only surround mix level. Only used for 3/2 coding mode. Valid values: -1.5, -3.0, -4.5, -6.0
-60
Type: number
Required: False
Format: float
Minimum: -60.0
Maximum: -1.5
IoRoCenterMixLevel
Left only/Right only center mix level. Only used for 3/2 coding mode. Valid values: 3.0, 1.5, 0.0, -1.5 -3.0 -4.5 -6.0 -60

Type: number
Required: False
Format: float
Minimum: -60.0
Maximum: 3.0

Eac3StereoDownmix (Enum)
Stereo downmix preference. Only used for 3/2 coding mode.
NOT_INDICATED
LO_RO
LT_RT
DPL2

Eac3SurroundExMode (Enum)
When encoding 3/2 audio, sets whether an extra center back surround channel is matrix encoded into the left and right surround channels.
NOT_INDICATED
ENABLED
DISABLED

Eac3SurroundMode (Enum)
When encoding 2/0 audio, sets whether Dolby Surround is matrix encoded into the two channels.
NOT_INDICATED
ENABLED
DISABLED

EmbeddedConvert608To708 (Enum)
When set to UPCONVERT, 608 data is both passed through via the "608 compatibility bytes" fields of the 708 wrapper as well as translated into 708. 708 data present in the source content will be discarded.
UPCONVERT
DISABLED

EmbeddedSourceSettings

source608ChannelNumber
Specifies the 608/708 channel number within the video track from which to extract captions. Unused for passthrough.
Type: integer
Required: False
Minimum: 1
Maximum: 4
convert608To708

Type: EmbeddedConvert608To708 (p. 502)
Required: False

source608TrackNumber

Specifies the video track index used for extracting captions. The system only supports one input video track, so this should always be set to '1'.

Type: integer
Required: False
Minimum: 1
Maximum: 1

ExceptionBody

message

Type: string
Required: False

F4vMoovPlacement (Enum)

If set to PROGRESSIVE_DOWNLOAD, the MOOV atom is relocated to the beginning of the archive as required for progressive downloading. Otherwise it is placed normally at the end.

PROGRESSIVE_DOWNLOAD
NORMAL

F4vSettings

moovPlacement

Type: F4vMoovPlacement (p. 503)
Required: False

FileGroupSettings

destination

Use Destination (Destination) to specify the S3 output location and the output filename base. Destination accepts format identifiers. If you do not specify the base filename in the URI, the service will use the filename of the input file. If your job has multiple inputs, the service uses the filename of the first input file.

Type: string
Required: True
Pattern: ^s3:\/[A-Za-z0-9]+/

FileSourceConvert608To708 (Enum)

If set to UPCONVERT, 608 caption data is both passed through via the "608 compatibility bytes" fields of the 708 wrapper as well as translated into 708. 708 data present in the source content will be discarded.
UPCONVERT
DISABLED

**FileSourceSettings**

timeDelta

Specifies a time delta in seconds to offset the captions from the source file.

- **Type:** integer
- **Required:** False
- **Minimum:** -2147483648
- **Maximum:** 2147483647

convert608To708

- **Type:** FileSourceConvert608To708 (p. 503)
- **Required:** False

sourceFile

External caption file used for loading captions. Accepted file extensions are 'scc', 'ttml', 'dfxp', 'stl', 'srt', and 'smi'. Auto-populated when Infer External Filename is checked.

- **Type:** string
- **Required:** True
- **Pattern:** `^(s3://)(.*?)(\.|\?)(scc|SCC|ttml|TTML|dfxp|DFXP|stl|STL|srt|SRT|smi|SMI)$`

**FrameCaptureSettings**

framerateDenominator

Frame capture will encode the first frame of the output stream, then one frame every `framerateDenominator/framerateNumerator` seconds. For example, settings of `framerateNumerator = 1` and `framerateDenominator = 3` (a rate of 1/3 frame per second) will capture the first frame, then 1 frame every 3s. Files will be named as `filename.n.jpg` where `n` is the 0-based sequence number of each Capture.

- **Type:** integer
- **Required:** False
- **Minimum:** 1
- **Maximum:** 2147483647

maxCaptures

Maximum number of captures (encoded jpg output files).

- **Type:** integer
- **Required:** False
- **Minimum:** 1
- **Maximum:** 10000000
framerateNumerator

Frame capture will encode the first frame of the output stream, then one frame every framerateDenominator/framerateNumerator seconds. For example, settings of framerateNumerator = 1 and framerateDenominator = 3 (a rate of 1/3 frame per second) will capture the first frame, then 1 frame every 3s. Files will be named as filename.NNNNNNN.jpg where N is the 0-based frame sequence number zero padded to 7 decimal places.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

quality

JPEG Quality - a higher value equals higher quality.

Type: integer
Required: False
Minimum: 1
Maximum: 100

GetJobRequest

id

Type: string
Required: True

GetJobResponse

job

Type: Job (p. 538)
Required: False

H264AdaptiveQuantization (Enum)

Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

OFF
LOW
MEDIUM
HIGH
HIGHER
MAX

H264CodecLevel (Enum)

H.264 Level.

AUTO
LEVEL_1
LEVEL_1_1
H264CodecProfile (Enum)

H.264 Profile. High 4:2:2 and 10-bit profiles are only available with the AVC-I License.

- BASELINE
- HIGH
- HIGH_10BIT
- HIGH_422
- HIGH_422_10BIT
- MAIN

H264EntropyEncoding (Enum)

Entropy encoding mode. Use CABAC (must be in Main or High profile) or CAVLC.

- CABAC
- CAVLC

H264FieldEncoding (Enum)

Choosing FORCE_FIELD disables PAFF encoding for interlaced outputs.

- PAFF
- FORCE_FIELD

H264FlickerAdaptiveQuantization (Enum)

Adjust quantization within each frame to reduce flicker or 'pop' on I-frames.

- DISABLED
- ENABLED

H264FramerateControl (Enum)

Using the API, set FramerateControl to INITIALIZE_FROM_SOURCE if you want the service to use the framerate from the input. Using the console, do this by choosing INITIALIZE_FROM_SOURCE for Framerate.

- INITIALIZE_FROM_SOURCE
- SPECIFIED
H264FramerateConversionAlgorithm (Enum)
When set to INTERPOLATE, produces smoother motion during framerate conversion.

- DUPLICATE_DROP
- INTERPOLATE

H264GopBReference (Enum)
If enable, use reference B frames for GOP structures that have B frames > 1.

- DISABLED
- ENABLED

H264GopSizeUnits (Enum)
Indicates if the GOP Size in H264 is specified in frames or seconds. If seconds the system will convert the GOP Size into a frame count at runtime.

- FRAMES
- SECONDS

H264InterlaceMode (Enum)
Use Interlace mode (InterlaceMode) to choose the scan line type for the output.

- Top Field First (TOP_FIELD) and Bottom Field First (BOTTOM_FIELD) produce interlaced output with the entire output having the same field polarity (top or bottom first).
- Follow, Default Top (FOLLOW_TOP_FIELD) and Follow, Default Bottom (FOLLOW_BOTTOM_FIELD) use the same field polarity as the source. Therefore, behavior depends on the input scan type. If the source is interlaced, the output will be interlaced with the same polarity as the source (it will follow the source). The output could therefore be a mix of "top field first" and "bottom field first". If the source is progressive, the output will be interlaced with "top field first" or "bottom field first" polarity, depending on which of the Follow options you chose.

- PROGRESSIVE
- TOP_FIELD
- BOTTOM_FIELD
- FOLLOW_TOP_FIELD
- FOLLOW_BOTTOM_FIELD

H264ParControl (Enum)
Using the API, enable ParFollowSource if you want the service to use the pixel aspect ratio from the input. Using the console, do this by choosing Follow source for Pixel aspect ratio.

- INITIALIZE_FROM_SOURCE
- SPECIFIED

H264QualityTuningLevel (Enum)
Use Quality tuning level (H264QualityTuningLevel) to specify whether to use fast single-pass, high-quality singlepass, or high-quality multipass video encoding.

- SINGLE_PASS
- SINGLE_PASS_HQ
MULTI_PASS_HQ

**H264RateControlMode (Enum)**

Rate control mode. CQ uses constant quantizer (qp), ABR (average bitrate) does not write HRD parameters.

- VBR
- CBR

**H264RepeatPps (Enum)**

Places a PPS header on each encoded picture, even if repeated.

- DISABLED
- ENABLED

**H264SceneChangeDetect (Enum)**

Scene change detection (inserts I-frames on scene changes).

- DISABLED
- ENABLED

**H264Settings**

**slices**

Number of slices per picture. Must be less than or equal to the number of macroblock rows for progressive pictures, and less than or equal to half the number of macroblock rows for interlaced pictures.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 32

**minIInterval**

Enforces separation between repeated (cadence) I-frames and I-frames inserted by Scene Change Detection. If a scene change I-frame is within I-interval frames of a cadence I-frame, the GOP is shrunk and/or stretched to the scene change I-frame. GOP stretch requires enabling lookahead as well as setting I-interval. The normal cadence resumes for the next GOP. This setting is only used when Scene Change Detect is enabled. Note: Maximum GOP stretch = GOP size + Min-I-interval - 1

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 30

**parNumerator**

Pixel Aspect Ratio numerator.

- **Type**: integer
Properties

Required: False
Minimum: 1
Maximum: 2147483647

flickerAdaptiveQuantization

Type: H264FlickerAdaptiveQuantization (p. 506)
Required: False

gopSizeUnits

Type: H264GopSizeUnits (p. 507)
Required: False

hrdBufferSize

Size of buffer (HRD buffer model). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

qualityTuningLevel

Type: H264QualityTuningLevel (p. 507)
Required: False

maxBitrate

Maximum bitrate in bits/second (for VBR mode only). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

bitrate

Average bitrate in bits/second. Required for VBR, CBR, and ABR. Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.

Type: integer
Required: False
Minimum: 1000
Maximum: 2147483647

spatialAdaptiveQuantization

Type: H264SpatialAdaptiveQuantization (p. 513)
Required: False
slowPal

    Type: H264SlowPal (p. 513)
    Required: False

codecProfile

    Type: H264CodecProfile (p. 506)
    Required: False

unregisteredSeiTimecode

    Type: H264UnregisteredSeiTimecode (p. 514)
    Required: False

softness

Softness. Selects quantizer matrix, larger values reduce high-frequency content in the encoded image.

    Type: integer
    Required: False
    Minimum: 0
    Maximum: 128

framerateControl

    Type: H264FramerateControl (p. 506)
    Required: False

telecine

    Type: H264Telecine (p. 513)
    Required: False

framerateConversionAlgorithm

    Type: H264FramerateConversionAlgorithm (p. 507)
    Required: False

codecLevel

    Type: H264CodecLevel (p. 505)
    Required: False

numberReferenceFrames

Number of reference frames to use. The encoder may use more than requested if using B-frames and/or interlaced encoding.

    Type: integer
    Required: False
    Minimum: 1
Maximum: 6

temporalAdaptiveQuantization

Type: H264TemporalAdaptiveQuantization (p. 514)
Required: False

repeatPps

Type: H264RepeatPps (p. 508)
Required: False

hrdBufferInitialFillPercentage

Percentage of the buffer that should initially be filled (HRD buffer model).

Type: integer
Required: False
Minimum: 0
Maximum: 100

framerateNumerator

Framerate numerator - framerate is a fraction, e.g. 24000 / 1001 = 23.976 fps.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

numberBFramesBetweenReferenceFrames

Number of B-frames between reference frames.

Type: integer
Required: False
Minimum: 0
Maximum: 7

gopClosedCadence

Frequency of closed GOPs. In streaming applications, it is recommended that this be set to 1 so a
decoder joining mid-stream will receive an IDR frame as quickly as possible. Setting this value to 0 will
break output segmenting.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

fieldEncoding

Type: H264FieldEncoding (p. 506)
Required: False

**entropyEncoding**

Type: [H264EntropyEncoding](p. 506)
Required: False

**framerateDenominator**

When you use the API for transcode jobs that use framerate conversion, specify the framerate as a fraction. For example, 24000 / 1001 = 23.976 fps. Use FramerateDenominator to specify the denominator of this fraction. In this example, use 1001 for the value of FramerateDenominator. When you use the console for transcode jobs that use framerate conversion, provide the value as a decimal number for Framerate. In this example, specify 23.976.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

**adaptiveQuantization**

Type: [H264AdaptiveQuantization](p. 505)
Required: False

**interlaceMode**

Type: [H264InterlaceMode](p. 507)
Required: False

**gopSize**

GOP Length (keyframe interval) in frames or seconds. Must be greater than zero.

Type: number
Required: False
Format: float
Minimum: 0.0

**gopBReference**

Type: [H264GopBReference](p. 507)
Required: False

**sceneChangeDetect**

Type: [H264SceneChangeDetect](p. 508)
Required: False

**parDenominator**

Pixel Aspect Ratio denominator.
Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

parControl
Type: H264ParControl (p. 507)
Required: False

syntax
Type: H264Syntax (p. 513)
Required: False

rateControlMode
Type: H264RateControlMode (p. 508)
Required: False

H264SlowPal (Enum)
Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

DISABLED
ENABLED

H264SpatialAdaptiveQuantization (Enum)
Adjust quantization within each frame based on spatial variation of content complexity.

DISABLED
ENABLED

H264Syntax (Enum)
Produces a bitstream compliant with SMPTE RP-2027.

DEFAULT
RP2027

H264Telecine (Enum)
This field applies only if the Streams > Advanced > Framerate (framerate) field is set to 29.970. This field works with the Streams > Advanced > Preprocessors > Deinterlacer field (deinterlace_mode) and the Streams > Advanced > Interlaced Mode field (interlace_mode) to identify the scan type for the output: Progressive, Interlaced, Hard Telecine or Soft Telecine. - Hard: produces 29.97i output from 23.976 input. - Soft: produces 23.976; the player converts this output to 29.97i.

NONE
SOFT
HARD

**H264TemporalAdaptiveQuantization (Enum)**

Adjust quantization within each frame based on temporal variation of content complexity.

- DISABLED
- ENABLED

**H264UnregisteredSeiTimecode (Enum)**

Inserts timecode for each frame as 4 bytes of an unregistered SEI message.

- DISABLED
- ENABLED

**H265AdaptiveQuantization (Enum)**

Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

- OFF
- LOW
- MEDIUM
- HIGH
- HIGHER
- MAX

**H265AlternateTransferFunctionSei (Enum)**

Enables Alternate Transfer Function SEI message for outputs using Hybrid Log Gamma (HLG) Electro-Optical Transfer Function (EOTF).

- DISABLED
- ENABLED

**H265CodecLevel (Enum)**

H.265 Level.

- AUTO
- LEVEL_1
- LEVEL_2
- LEVEL_2_1
- LEVEL_3
- LEVEL_3_1
- LEVEL_4
- LEVEL_4_1
- LEVEL_5
- LEVEL_5_1
- LEVEL_5_2
- LEVEL_6
- LEVEL_6_1
- LEVEL_6_2
H265CodecProfile (Enum)

Represents the Profile and Tier, per the HEVC (H.265) specification. Selections are grouped as [Profile] / [Tier], so "Main/High" represents Main Profile with High Tier. 4:2:2 profiles are only available with the HEVC 4:2:2 License.

MAIN_MAIN
MAIN_HIGH
MAIN10.MAIN
MAIN10_HIGH
MAIN_422_8BIT_MAIN
MAIN_422_8BIT_HIGH
MAIN_422_10BIT_MAIN
MAIN_422_10BIT_HIGH

H265FlickerAdaptiveQuantization (Enum)

Adjust quantization within each frame to reduce flicker or 'pop' on I-frames.

DISABLED
ENABLED

H265FramerateControl (Enum)

Using the API, set FramerateControl to INITIALIZE_FROM_SOURCE if you want the service to use the framerate from the input. Using the console, do this by choosing INITIALIZE_FROM_SOURCE for Framerate.

INITIALIZE_FROM_SOURCE
SPECIFIED

H265FramerateConversionAlgorithm (Enum)

When set to INTERPOLATE, produces smoother motion during framerate conversion.

DUPLICATE_DROP
INTERPOLATE

H265GopBReference (Enum)

If enable, use reference B frames for GOP structures that have B frames > 1.

DISABLED
ENABLED

H265GopSizeUnits (Enum)

Indicates if the GOP Size in H265 is specified in frames or seconds. If seconds the system will convert the GOP Size into a frame count at run time.

FRAMES
SECONDS

H265InterlaceMode (Enum)

Use Interlace mode (InterlaceMode) to choose the scan line type for the output.
* Top Field First (TOP_FIELD) and Bottom Field First (BOTTOM_FIELD) produce interlaced output with the entire output having the same field polarity (top or bottom first).

* Follow, Default Top (FOLLOW_TOP_FIELD) and Follow, Default Bottom (FOLLOW_BOTTOM_FIELD) use the same field polarity as the source. Therefore, behavior depends on the input scan type. If the source is interlaced, the output will be interlaced with the same polarity as the source (it will follow the source). The output could therefore be a mix of "top field first" and "bottom field first". If the source is progressive, the output will be interlaced with "top field first" or "bottom field first" polarity, depending on which of the Follow options you chose.

```
PROGRESSIVE
TOP_FIELD
BOTTOM_FIELD
FOLLOW_TOP_FIELD
FOLLOW_BOTTOM_FIELD
```

H265ParControl (Enum)

Using the API, enable ParFollowSource if you want the service to use the pixel aspect ratio from the input. Using the console, do this by choosing Follow source for Pixel aspect ratio.

```
INITIALIZE_FROM_SOURCE
SPECIFIED
```

H265QualityTuningLevel (Enum)

Use Quality tuning level (H265QualityTuningLevel) to specify whether to use fast single-pass, high-quality singlepass, or high-quality multipass video encoding.

```
SINGLE_PASS
SINGLE_PASS_HQ
MULTI_PASS_HQ
```

H265RateControlMode (Enum)

Rate control mode. CQ uses constant quantizer (qp), ABR (average bitrate) does not write HRD parameters.

```
VBR
CBR
```

H265SampleAdaptiveOffsetFilterMode (Enum)

Specify Sample Adaptive Offset (SAO) filter strength. Adaptive mode dynamically selects best strength based on content

```
DEFAULT
ADAPTIVE
OFF
```

H265SceneChangeDetect (Enum)

Scene change detection (inserts I-frames on scene changes).

```
DISABLED
```
H265Settings

slices
Number of slices per picture. Must be less than or equal to the number of macroblock rows for progressive pictures, and less than or equal to half the number of macroblock rows for interlaced pictures.

Type: integer
Required: False
Minimum: 1
Maximum: 32

minIInterval
Enforces separation between repeated (cadence) I-frames and I-frames inserted by Scene Change Detection. If a scene change I-frame is within I-interval frames of a cadence I-frame, the GOP is shrunk and/or stretched to the scene change I-frame. GOP stretch requires enabling lookahead as well as setting I-interval. The normal cadence resumes for the next GOP. This setting is only used when Scene Change Detect is enabled. Note: Maximum GOP stretch = GOP size + Min-I-interval - 1

Type: integer
Required: False
Minimum: 0
Maximum: 30

parNumerator
Pixel Aspect Ratio numerator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

flickerAdaptiveQuantization

Type: H265FlickerAdaptiveQuantization (p. 515)
Required: False

gopSizeUnits

Type: H265GopSizeUnits (p. 515)
Required: False

hrdBufferSize
Size of buffer (HRD buffer model). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.

Type: integer
**qualityTuningLevel**

Type: H265QualityTuningLevel (p. 516)
Required: False

**maxBitrate**

Maximum bitrate in bits/second (for VBR mode only). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

**bitrate**

Average bitrate in bits/second. Required for VBR, CBR, and ABR. Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.

Type: integer
Required: False
Minimum: 1000
Maximum: 2147483647

**spatialAdaptiveQuantization**

Type: H265SpatialAdaptiveQuantization (p. 521)
Required: False

**sampleAdaptiveOffsetFilterMode**

Type: H265SampleAdaptiveOffsetFilterMode (p. 516)
Required: False

**temporalIds**

Type: H265TemporalIds (p. 522)
Required: False

**slowPal**

Type: H265SlowPal (p. 521)
Required: False

**tiles**

Type: H265Tiles (p. 522)
Properties

**Required**: False

**codecProfile**

*Type*: H265CodecProfile (p. 515)

**Required**: False

**alternateTransferFunctionSei**

*Type*: H265AlternateTransferFunctionSei (p. 514)

**Required**: False

**unregisteredSeiTimecode**

*Type*: H265UnregisteredSeiTimecode (p. 522)

**Required**: False

**framerateControl**

*Type*: H265FramerateControl (p. 515)

**Required**: False

**telecine**

*Type*: H265Telecine (p. 522)

**Required**: False

**framerateConversionAlgorithm**

*Type*: H265FramerateConversionAlgorithm (p. 515)

**Required**: False

**codecLevel**

*Type*: H265CodecLevel (p. 514)

**Required**: False

**numberReferenceFrames**

Number of reference frames to use. The encoder may use more than requested if using B-frames and/or interlaced encoding.

*Type*: integer

**Required**: False

**Minimum**: 1

**Maximum**: 6

**temporalAdaptiveQuantization**

*Type*: H265TemporalAdaptiveQuantization (p. 522)

**Required**: False
hrdBufferInitialFillPercentage

Percentage of the buffer that should initially be filled (HRD buffer model).

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 100

framerateNumerator

Framerate numerator - framerate is a fraction, e.g. 24000 / 1001 = 23.976 fps.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 2147483647

numberOfFramesBetweenReferenceFrames

Number of B-frames between reference frames.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 7

gopClosedCadence

Frequency of closed GOPs. In streaming applications, it is recommended that this be set to 1 so a decoder joining mid-stream will receive an IDR frame as quickly as possible. Setting this value to 0 will break output segmenting.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 2147483647

framerateDenominator

Framerate denominator.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 2147483647

adaptiveQuantization

- **Type**: H265AdaptiveQuantization (p. 514)
- **Required**: False

interlaceMode

- **Type**: H265InterlaceMode (p. 515)
Required: False

gopSize

GOP Length (keyframe interval) in frames or seconds. Must be greater than zero.

Type: number
Required: False
Format: float
Minimum: 0.0

gopBReference

Type: H265GopBReference (p. 515)
Required: False

sceneChangeDetect

Type: H265SceneChangeDetect (p. 516)
Required: False

parDenominator

Pixel Aspect Ratio denominator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

parControl

Type: H265ParControl (p. 516)
Required: False

rateControlMode

Type: H265RateControlMode (p. 516)
Required: False

H265SlowPal (Enum)

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

DISABLED
ENABLED

H265SpatialAdaptiveQuantization (Enum)

Adjust quantization within each frame based on spatial variation of content complexity.

DISABLED
ENABLED
**H265Telecine (Enum)**

This field applies only if the Streams > Advanced > Framerate (framerate) field is set to 29.970. This field works with the Streams > Advanced > Preprocessors > Deinterlacer field (deinterlace_mode) and the Streams > Advanced > Interlaced Mode field (interlace_mode) to identify the scan type for the output: Progressive, Interlaced, Hard Telecine or Soft Telecine. - Hard: produces 29.97i output from 23.976 input. - Soft: produces 23.976; the player converts this output to 29.97i.

NONE
SOFT
HARD

**H265TemporalAdaptiveQuantization (Enum)**

Adjust quantization within each frame based on temporal variation of content complexity.

DISABLED
ENABLED

**H265TemporalIds (Enum)**

Enables temporal layer identifiers in the encoded bitstream. Up to 3 layers are supported depending on GOP structure: I- and P-frames form one layer, reference B-frames can form a second layer and non-reference b-frames can form a third layer. Decoders can optionally decode only the lower temporal layers to generate a lower frame rate output. For example, given a bitstream with temporal IDs and with b-frames = 1 (i.e. IbPbPb display order), a decoder could decode all the frames for full frame rate output or only the I and P frames (lowest temporal layer) for a half frame rate output.

DISABLED
ENABLED

**H265Tiles (Enum)**

Enable use of tiles, allowing horizontal as well as vertical subdivision of the encoded pictures.

DISABLED
ENABLED

**H265UnregisteredSeiTimecode (Enum)**

Inserts timecode for each frame as 4 bytes of an unregistered SEI message.

DISABLED
ENABLED

**Hdr10Metadata**

**redPrimaryY**

HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

Type: integer
Required: False
Minimum: 0
Maximum: 50000

greenPrimaryY

HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

Type: integer
Required: False
Minimum: 0
Maximum: 50000

whitePointX

HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

Type: integer
Required: False
Minimum: 0
Maximum: 50000

maxLuminance

Nominal maximum mastering display luminance in units of 0.0001 candelas per square meter.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

greenPrimaryX

HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

Type: integer
Required: False
Minimum: 0
Maximum: 50000

whitePointY

HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

Type: integer
Required: False
Minimum: 0
Maximum: 50000

redPrimaryX

HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.
Properties

**Type**: integer  
**Required**: False  
**Minimum**: 0  
**Maximum**: 50000

**bluePrimaryX**

HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

**Type**: integer  
**Required**: False  
**Minimum**: 0  
**Maximum**: 50000

**maxFrameAverageLightLevel**

Maximum average light level of any frame in the coded video sequence, in units of candelas per square meter.

**Type**: integer  
**Required**: True  
**Minimum**: 0  
**Maximum**: 65535

**bluePrimaryY**

HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

**Type**: integer  
**Required**: False  
**Minimum**: 0  
**Maximum**: 50000

**maxContentLightLevel**

Maximum light level among all samples in the coded video sequence, in units of candelas per square meter.

**Type**: integer  
**Required**: True  
**Minimum**: 0  
**Maximum**: 65535

**minLuminance**

Nominal minimum mastering display luminance in units of 0.0001 candelas per square meter

**Type**: integer  
**Required**: False  
**Minimum**: 0
Maximum: 2147483647

HlsAdMarkers (Enum)

- ELEMENTAL
- ELEMENTAL_SCTE35

HlsAudioTrackType (Enum)

Four types of audio-only tracks are supported: Audio-Only Variant Stream The client can play back this audio-only stream instead of video in low-bandwidth scenarios. Represented as an EXT-X-STREAM-INF in the HLS manifest. Alternate Audio, Auto Select, Default Alternate rendition that the client should try to play back by default. Represented as an EXT-X-MEDIA in the HLS manifest with DEFAULT=YES, AUTOSELECT=YES Alternate Audio, Auto Select, Not Default Alternate rendition that the client may try to play back by default. Represented as an EXT-X-MEDIA in the HLS manifest with DEFAULT=NO, AUTOSELECT=YES Alternate Audio, not Auto Select Alternate rendition that the client will not try to play back by default. Represented as an EXT-X-MEDIA in the HLS manifest with DEFAULT=NO, AUTOSELECT=NO

- ALTERNATE_AUDIO_AUTO_SELECT_DEFAULT
- ALTERNATE_AUDIO_AUTO_SELECT
- ALTERNATE_AUDIO_NOT_AUTO_SELECT
- AUDIO_ONLY_VARIANT_STREAM

HlsCaptionLanguageMapping

languageDescription

Caption language description.

- Type: string
- Required: False

captionChannel

Caption channel.

- Type: integer
- Required: False
- Minimum: -2147483648
- Maximum: 2147483647

languageCode

- Type: LanguageCode (p. 541)
- Required: False

HlsCaptionLanguageSetting (Enum)

Applies only to 608 Embedded output captions. Insert: Include CLOSED-CAPTIONS lines in the manifest. Specify at least one language in the CC1 Language Code field. One CLOSED-CAPTION line is added for each Language Code you specify. Make sure to specify the languages in the order in which they appear in the original source (if the source is embedded format) or the order of the caption selectors (if the source is other than embedded). Otherwise, languages in the manifest will not match up properly with the
output captions. None: Include CLOSED-CAPTIONS=NONE line in the manifest. Omit: Omit any CLOSED-CAPTIONS line from the manifest.

Insert
Omit
None

HlsClientCache (Enum)
When set to ENABLED, sets #EXT-X-ALLOW-CACHE:no tag, which prevents client from saving media segments for later replay.

Disabled
Enabled

HlsCodecSpecification (Enum)
Specification to use (RFC-6381 or the default RFC-4281) during m3u8 playlist generation.

RFC_6381
RFC_4281

HlsDirectoryStructure (Enum)
Indicates whether segments should be placed in subdirectories.

Single_Directory
Subdirectory_Per_Stream

HlsEncryptionSettings

initializationVectorInManifest
Type: HlsInitializationVectorInManifest (p. 530)
Required: False

constantInitializationVector
This is a 128-bit, 16-byte hex value represented by a 32-character text string. If this parameter is not set then the Initialization Vector will follow the segment number by default.

Type: string
Required: False
Pattern: ^[0-9a-fA-F]{32}$

staticKeyProvider
Type: StaticKeyProvider (p. 575)
Required: False

type
Type: HlsKeyProviderType (p. 530)
Required: True
spekeKeyProvider

Type: SpekeKeyProvider (p. 575)
Required: False

encryptionMethod

Type: HlsEncryptionType (p. 527)
Required: False

HlsEncryptionType (Enum)

Encrypts the segments with the given encryption scheme. Leave blank to disable. Selecting 'Disabled' in the web interface also disables encryption.

AES128
SAMPLE_AES

HlsGroupSettings

segmentsPerSubdirectory

Number of segments to write to a subdirectory before starting a new one. directoryStructure must be SINGLE_DIRECTORY for this setting to have an effect.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

streamInfResolution

Type: HlsStreamInfResolution (p. 532)
Required: False

timestampDeltaMilliseconds

Provides an extra millisecond delta offset to fine tune the timestamps.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

outputSelection

Type: HlsOutputSelection (p. 530)
Required: False

captionLanguageMappings

Type: Array of type HlsCaptionLanguageMapping (p. 525)
Required: False
clientCache

Type: HlsClientCache (p. 526)
Required: False

codecSpecification

Type: HlsCodecSpecification (p. 526)
Required: False

destination

Use Destination (Destination) to specify the S3 output location and the output filename base. Destination accepts format identifiers. If you do not specify the base filename in the URI, the service will use the filename of the input file. If your job has multiple inputs, the service uses the filename of the first input file.

Type: string
Required: True
Pattern: ^s3://\/

segmentControl

Type: HlsSegmentControl (p. 531)
Required: False

timedMetadataId3Frame

Type: HlsTimedMetadataId3Frame (p. 532)
Required: False

timedMetadataId3Period

Timed Metadata interval in seconds.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

captionLanguageSetting

Type: HlsCaptionLanguageSetting (p. 525)
Required: False

minSegmentLength

When set, Minimum Segment Size is enforced by looking ahead and back within the specified range for a nearby avail and extending the segment size if needed.

Type: integer
Required: True
Minimum: 0
Maximum: 2147483647

directoryStructure
  
  Type: HlsDirectoryStructure (p. 526)
  Required: False

programDateTime
  
  Type: HlsProgramDateTime (p. 531)
  Required: False

baseUrl
  
  A partial URI prefix that will be prepended to each output in the media .m3u8 file. Can be used if base manifest is delivered from a different URL than the main .m3u8 file.
    
    Type: string
    Required: False

encryption
  
  DRM settings.
    
    Type: HlsEncryptionSettings (p. 526)
    Required: False

adMarkers
  
  Choose one or more ad marker types to pass SCTE35 signals through to this group of Apple HLS outputs.
    
    Type: Array of type HlsAdMarkers (p. 525)
    Required: False

programDateTimePeriod
  
  Period of insertion of EXT-X-PROGRAM-DATE-TIME entry, in seconds.
    
    Type: integer
    Required: False
    Minimum: 0
    Maximum: 3600

manifestCompression
  
  Type: HlsManifestCompression (p. 530)
  Required: False

segmentLength
  
  Length of MPEG-2 Transport Stream segments to create (in seconds). Note that segments will end on the next keyframe after this number of seconds, so actual segment length may be longer.
Type: integer  
**Required: True**  
**Minimum: 1**  
**Maximum: 2147483647**

**manifestDurationFormat**

**Type:** HlsManifestDurationFormat (p. 530)  
**Required:** False

**HlsIframeOnlyManifest (Enum)**

When set to INCLUDE, writes I-Frame Only Manifest in addition to the HLS manifest

- INCLUDE  
- EXCLUDE

**HlsInitializationVectorInManifest (Enum)**

The Initialization Vector is a 128-bit number used in conjunction with the key for encrypting blocks. If set to INCLUDE, Initialization Vector is listed in the manifest. Otherwise Initialization Vector is not in the manifest.

- INCLUDE  
- EXCLUDE

**HlsKeyProviderType (Enum)**

Indicates which type of key provider is used for encryption.

- SPEKE  
- STATIC_KEY

**HlsManifestCompression (Enum)**

When set to GZIP, compresses HLS playlist.

- GZIP  
- NONE

**HlsManifestDurationFormat (Enum)**

Indicates whether the output manifest should use floating point values for segment duration.

- FLOATING_POINT  
- INTEGER

**HlsOutputSelection (Enum)**

Indicates whether the .m3u8 manifest file should be generated for this HLS output group.

- MANIFESTS_AND_SEGMENTS
**SEGMENTSONLY**

**HlsProgramDateTime (Enum)**

Includes or excludes EXT-X-PROGRAM-DATE-TIME tag in .m3u8 manifest files. The value is calculated as follows: either the program date and time are initialized using the input timecode source, or the time is initialized using the input timecode source and the date is initialized using the timestamp_offset.

- INCLUDE
- EXCLUDE

**HlsSegmentControl (Enum)**

When set to SINGLE_FILE, emits program as a single media resource (.ts) file, uses #EXT-X-BYTERANGE tags to index segment for playback.

- SINGLE_FILE
- SEGMENTED_FILES

**HlsSettings**

**iFrameOnlyManifest**

- Type: HlsIFrameOnlyManifest (p. 530)
- Required: False

**segmentModifier**

String concatenated to end of segment filenames. Accepts "Format Identifiers":#format_identifier_parameters.

- Type: string
- Required: False

**audioTrackType**

- Type: HlsAudioTrackType (p. 525)
- Required: False

**audioRenditionSets**

List all the audio groups that are used with the video output stream. Input all the audio GROUP-IDs that are associated to the video, separate by '.

- Type: string
- Required: False

**audioGroupId**

Specifies the group to which the audio Rendition belongs.

- Type: string
- Required: False
**HlsStreamInfResolution (Enum)**

Include or exclude RESOLUTION attribute for video in EXT-X-STREAM-INF tag of variant manifest.

- INCLUDE
- EXCLUDE

**HlsTimedMetadataId3Frame (Enum)**

Indicates ID3 frame that has the timecode.

- NONE
- PRIV
- TDRL

**Id3Insertion**

**id3**

Use ID3 tag (id3) to provide a tag value in base64-encode format.

- **Type**: string
- **Required**: True
- **Pattern**: `^[A-Za-z0-9+/=]{0,2}$`

**timecode**

Provide a Timecode (TimeCode) in HH:MM:SS:FF or HH:MM:SS;FF format.

- **Type**: string
- **Required**: True
- **Format**: timecode
- **Pattern**: `^([01][0-9]|2[0-4]):[0-5][0-9]:[0-9][0-9]:[0-9]{2}$`

**ImageInserter**

**insertableImages**

Image to insert. Must be 32 bit windows BMP, PNG, or TGA file. Must not be larger than the output frames.

- **Type**: Array of type `InsertableImage (p. 536)`
- **Required**: True

**Input**

**audioSelectors**

Use Audio selectors (AudioSelectors) to specify a track or set of tracks from the input that you will use in your outputs. You can use mutiple Audio selectors per input.

- **Type**: object
- **Required**: False
audioSelectorGroups

Specifies set of audio selectors within an input to combine. An input may have multiple audio selector groups. See "Audio Selector Group" in the input documentation for more information.

Type: object
Required: False

videoSelector

Type: VideoSelector (p. 583)
Required: False

denoiseFilter

Type: InputDenoiseFilter (p. 535)
Required: False

filterEnable

Type: InputFilterEnable (p. 535)
Required: False

deblockFilter

Type: InputDeblockFilter (p. 535)
Required: False

filterStrength

Use Filter strength (FilterStrength) to adjust the magnitude the input filter settings (Deblock and Denoise). The range is -5 to 5. Default is 0.

Type: integer
Required: False
Minimum: -5
Maximum: 5

programNumber

Use Program (programNumber) to select a specific program from within a multi-program transport stream. Note that Quad 4K is not currently supported. Default is the first program within the transport stream. If the program you specify doesn't exist, the transcoding service will use this default.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

timecodeSource

Type: InputTimecodeSource (p. 536)
Required: False
captionSelectors

Use Captions selectors (CaptionSelectors) to specify the captions data from the input that you will use in your outputs. You can use multiple captions selectors per input.

Type: object
Required: False

fileInput

Use Input (fileInput) to define the source file used in the transcoding job. There can be multiple inputs in a job. These inputs are concatenated, in order they are specified in the job, to create the output.

Type: string
Required: True

Format: timecode
Pattern: ^([01][0-9]|2[0-4]):([0-5][0-9]):([0-5][0-9]):([0-5][0-9])$
endTimecode

Set End timecode (EndTimecode) to the end of the portion of the input you are clipping. The frame corresponding to the End timecode value is included in the clip. Start timecode or End timecode may be left blank, but not both. When choosing this value, take into account your setting for Input timecode source. For example, if you have embedded timecodes that start at 01:00:00:00 and you want your clip to begin five minutes into the video, use 01:00:05:00.

Type: string
Required: False
Format: timecode
Pattern: \^([01][0-9]|2[0-4]):[0-5][0-9]:[0-5][0-9]::[0-9]{2}$

InputDeblockFilter (Enum)

Enable Deblock (InputDeblockFilter) to produce smoother motion in the output. Default is disabled. Only manually controllable for MPEG2 and uncompressed video inputs.

ENABLED
DISABLED

InputDenoiseFilter (Enum)

Enable Denoise (InputDenoiseFilter) to filter noise from the input. Default is disabled. Only applicable to MPEG2, H.264, H.265, and uncompressed video inputs.

ENABLED
DISABLED

InputFilterEnable (Enum)

Use Filter enable (InputFilterEnable) to specify how the transcoding service applies the denoise and deblock filters. You must also enable the filters separately, with Denoise (InputDenoiseFilter) and Deblock (InputDeblockFilter).

* Auto - The transcoding service determines whether to apply filtering, depending on input type and quality.

* Disable - The input is not filtered. This is true even if you use the API to enable them in (InputDeblockFilter) and (InputDeblockFilter).

* Force - The input is filtered regardless of input type.

AUTO
DISABLE
FORCE

InputPsiControl (Enum)

Set PSI control (InputPsiControl) for transport stream inputs to specify which data the demux process to scans.

* Ignore PSI - Scan all PIDs for audio and video.

* Use PSI - Scan only PSI data.

IGNORE_PSI
USE_PSI

InputTimecodeSource (Enum)

Use Timecode source (InputTimecodeSource) to specify how timecode information from your input is adjusted and encoded in all outputs for the job. Default is embedded. Set to Embedded (EMBEDDED) to use the timecode that is in the input video. If no embedded timecode is in the source, will set the timecode for the first frame to 00:00:00:00. Set to Start at 0 (ZEROBASED) to set the timecode of the initial frame to 00:00:00:00. Set to Specified start (SPECIFIEDSTART) to provide the initial timecode yourself the setting (Start).

EMBEDDED
ZEROBASED
SPECIFIEDSTART

InsertableImage

duration

Use Duration (Duration) to set the time, in milliseconds, for the image to remain on the output video.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

fadeOut

Use Fade out (FadeOut) to set the length, in milliseconds, of the inserted image fade out. If you don't specify a value for Fade out, the image will disappear abruptly at the end of the inserted image duration.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

imageY

Use Top (ImageY) to set the distance, in pixels, between the inserted image and the top edge of the video frame. Required for BMP, PNG and TGA input.

Type: integer
Required: True
Minimum: -2147483648
Maximum: 2147483647

fadeIn

Use Fade in (FadeIn) to set the length, in milliseconds, of the inserted image fade in. If you don't specify a value for Fade in, the image will appear abruptly at the Start time.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

**imageX**

Use Left (ImageX) to set the distance, in pixels, between the inserted image and the left edge of the frame. Required for BMP, PNG and TGA input.

- **Type:** integer
- **Required:** True
- **Minimum:** -2147483648
- **Maximum:** 2147483647

**width**

Specify the Width (Width) of the inserted image. Use a value that is less than or equal to the video resolution width. Leave this setting blank to use the native width of the image.

- **Type:** integer
- **Required:** False
- **Minimum:** -2147483648
- **Maximum:** 2147483647

**startTime**

Use Start time (StartTime) to specify the video timecode when the image is inserted in the output. This must be in timecode format (HH:MM:SS:FF)

- **Type:** string
- **Required:** False

**opacity**

Use Opacity (Opacity) to specify how much of the underlying video shows through the inserted image. 0 is transparent and 100 is fully opaque. Default is 50.

- **Type:** integer
- **Required:** True
- **Minimum:** 0
- **Maximum:** 100

**layer**

Use Layer (Layer) to specify how overlapping inserted images appear. Images with higher values of layer appear on top of images with lower values of layer.

- **Type:** integer
- **Required:** True
- **Minimum:** 0
- **Maximum:** 7

**height**

Specify the Height (Height) of the inserted image. Use a value that is less than or equal to the video resolution height. Leave this setting blank to use the native height of the image.
Properties

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

imageInserterInput

Use Image location (imageInserterInput) to specify the Amazon S3 location of the image to be inserted into the output. Use a 32 bit BMP, PNG, or TGA file that fits inside the video frame.

Type: string
Required: True
Pattern: ^s3://(.*?)\.(bmp|BMP|png|PNG|tga|TGA)$

Job

settings

Type: JobSettings (p. 539)
Required: True

outputGroupDetails

Type: Array of type OutputGroupDetail (p. 569)
Required: False

role

The IAM role you use for creating this job. For details about permissions, see the User Guide topic at the User Guide at http://docs.aws.amazon.com/mediaconvert/latest/ug/iam-role.html

Type: string
Required: True

jobTemplate

The job template that the job is created from, if it is created from a job template.

Type: string
Required: False

timing

Type: Timing (p. 579)
Required: False

errorMessage

Error message of job

Type: string
Required: False
errorCode

Type: integer  
Required: False  
Format: int32

createdAt

The time, in Unix epoch format in seconds, when the job got created.

Type: string  
Required: False  
Format: date-time

id

A portion of the job's ARN, unique within your AWS Elemental MediaConvert resources

Type: string  
Required: False

arn

An identifier for this resource that is unique within all of AWS.

Type: string  
Required: False

userMetadata

Type: object  
Required: False

queue

Optional. When you create a job, you can specify a queue to send it to. If you don’t specify, the job will go to the default queue. For more about queues, see the User Guide topic at http://docs.aws.amazon.com/mediaconvert/latest/ug/what-is.html

Type: string  
Required: False

status

Type: JobStatus (p. 541)  
Required: False

JobSettings

timecodeConfig

Contains settings used to acquire and adjust timecode information from inputs.

Type: TimecodeConfig (p. 577)
AWS Elemental MediaConvert API Reference
Properties

Required: False

adAvailOffset
When specified, this offset (in milliseconds) is added to the input Ad Avail PTS time.

Type: integer
  Required: False
  Minimum: -1000
  Maximum: 1000

nielsenConfiguration
Nielsen configuration settings

Type: NielsenConfiguration (p. 565)
  Required: False

inputs
Use Inputs (inputs) to define source file used in the transcode job. There can be multiple inputs added to a job. These inputs will be concatenated together to create the output.

Type: Array of type Input (p. 532)
  Required: True

outputGroups
(OutputGroups) contains one group of settings for each set of outputs that share a common package type. All unpackaged files (MPEG-4, MPEG-2 TS, Quicktime, MXF, and no container) are grouped in a single output group as well. Required in (OutputGroups) is a group of settings that apply to the whole group. This required object depends on the value you set for (Type) under (OutputGroups)>(OutputGroupSettings). Type, settings object pairs are as follows.

* FILE_GROUP_SETTINGS, FileGroupSettings
* HLS_GROUP_SETTINGS, HlsGroupSettings
* DASH_ISO_GROUP_SETTINGS, DashIsoGroupSettings
* MS_SMOOTH_GROUP_SETTINGS, MsSmoothGroupSettings

Type: Array of type OutputGroup (p. 568)
  Required: True

timedMetadataInsertion

Type: TimedMetadataInsertion (p. 578)
  Required: False

availBlanking
Settings for ad avail blanking. Video can be blanked or overlaid with an image, and audio muted during SCTE-35 triggered ad avails.
**Type:** AvailBlanking (p. 480)  
**Required:** False

**JobStatus (Enum)**

A job's status can be SUBMITTED, PROGRESSING, COMPLETE, CANCELED, or ERROR.

- SUBMITTED
- PROGRESSING
- COMPLETE
- CANCELED
- ERROR

**LanguageCode (Enum)**

Selects a specific language code from within an audio source.

- ENG
- SPA
- FRA
- DEU
- GER
- ZHO
- ARA
- HIN
- JPN
- RUS
- POR
- ITA
- URD
- VIE
- KOR
- PAN
- ABK
- AAR
- AFR
- AKA
- SQI
- AMH
- ARG
- HYE
- ASM
-AVA
- AVE
- AYM
- AZE
- BAM
- BAK
- EUS
- BEL
- BEN
- BIH
<table>
<thead>
<tr>
<th>Code</th>
<th>Language</th>
</tr>
</thead>
<tbody>
<tr>
<td>GLE</td>
<td></td>
</tr>
<tr>
<td>JAV</td>
<td></td>
</tr>
<tr>
<td>KAL</td>
<td></td>
</tr>
<tr>
<td>KAN</td>
<td></td>
</tr>
<tr>
<td>KAU</td>
<td></td>
</tr>
<tr>
<td>KAS</td>
<td></td>
</tr>
<tr>
<td>KAZ</td>
<td></td>
</tr>
<tr>
<td>KIK</td>
<td></td>
</tr>
<tr>
<td>KIN</td>
<td></td>
</tr>
<tr>
<td>KIR</td>
<td></td>
</tr>
<tr>
<td>KOM</td>
<td></td>
</tr>
<tr>
<td>KON</td>
<td></td>
</tr>
<tr>
<td>KUA</td>
<td></td>
</tr>
<tr>
<td>KUR</td>
<td></td>
</tr>
<tr>
<td>LAO</td>
<td></td>
</tr>
<tr>
<td>LAT</td>
<td></td>
</tr>
<tr>
<td>LAV</td>
<td></td>
</tr>
<tr>
<td>LIM</td>
<td></td>
</tr>
<tr>
<td>LIN</td>
<td></td>
</tr>
<tr>
<td>LIT</td>
<td></td>
</tr>
<tr>
<td>LUB</td>
<td></td>
</tr>
<tr>
<td>LTZ</td>
<td></td>
</tr>
<tr>
<td>MKD</td>
<td></td>
</tr>
<tr>
<td>MLG</td>
<td></td>
</tr>
<tr>
<td>MSA</td>
<td></td>
</tr>
<tr>
<td>MAL</td>
<td></td>
</tr>
<tr>
<td>MLT</td>
<td></td>
</tr>
<tr>
<td>GLV</td>
<td></td>
</tr>
<tr>
<td>MRI</td>
<td></td>
</tr>
<tr>
<td>MAR</td>
<td></td>
</tr>
<tr>
<td>MAH</td>
<td></td>
</tr>
<tr>
<td>MON</td>
<td></td>
</tr>
<tr>
<td>NAU</td>
<td></td>
</tr>
<tr>
<td>NAV</td>
<td></td>
</tr>
<tr>
<td>NDE</td>
<td></td>
</tr>
<tr>
<td>NBL</td>
<td></td>
</tr>
<tr>
<td>NDO</td>
<td></td>
</tr>
<tr>
<td>NEP</td>
<td></td>
</tr>
<tr>
<td>SME</td>
<td></td>
</tr>
<tr>
<td>NOR</td>
<td></td>
</tr>
<tr>
<td>NOB</td>
<td></td>
</tr>
<tr>
<td>NNO</td>
<td></td>
</tr>
<tr>
<td>OCI</td>
<td></td>
</tr>
<tr>
<td>OJI</td>
<td></td>
</tr>
<tr>
<td>ORI</td>
<td></td>
</tr>
<tr>
<td>ORM</td>
<td></td>
</tr>
<tr>
<td>OSS</td>
<td></td>
</tr>
<tr>
<td>PLI</td>
<td></td>
</tr>
<tr>
<td>FAS</td>
<td></td>
</tr>
<tr>
<td>POL</td>
<td></td>
</tr>
<tr>
<td>PUS</td>
<td></td>
</tr>
<tr>
<td>Code</td>
<td>Language</td>
</tr>
<tr>
<td>------</td>
<td>----------</td>
</tr>
<tr>
<td>QUE</td>
<td>Quechua</td>
</tr>
<tr>
<td>QAA</td>
<td>Qaraqosh</td>
</tr>
<tr>
<td>RON</td>
<td>Romanian</td>
</tr>
<tr>
<td>ROH</td>
<td>Romanian</td>
</tr>
<tr>
<td>RUN</td>
<td>Russian</td>
</tr>
<tr>
<td>SMO</td>
<td>Sango</td>
</tr>
<tr>
<td>SAG</td>
<td>Sango</td>
</tr>
<tr>
<td>SAN</td>
<td>Sande</td>
</tr>
<tr>
<td>SRD</td>
<td>Serbo-Croatian Dalmatian Dalmatian</td>
</tr>
<tr>
<td>SRB</td>
<td>Serbian</td>
</tr>
<tr>
<td>SNA</td>
<td>Sranak</td>
</tr>
</tbody>
</table>
Properties

QPC
TNG

**M2tsAudioBufferModel (Enum)**

Selects between the DVB and ATSC buffer models for Dolby Digital audio.

- DVB
- ATSC

**M2tsBufferModel (Enum)**

Controls what buffer model to use for accurate interleaving. If set to MULTIPLEX, use multiplex buffer model. If set to NONE, this can lead to lower latency, but low-memory devices may not be able to play back the stream without interruptions.

- MULTIPLEX
- NONE

**M2tsEbpAudioInterval (Enum)**

When set to VIDEO_AND_FIXED_INTERVALS, audio EBP markers will be added to partitions 3 and 4. The interval between these additional markers will be fixed, and will be slightly shorter than the video EBP marker interval. When set to VIDEO_INTERVAL, these additional markers will not be inserted. Only applicable when EBP segmentation markers are is selected (segmentationMarkers is EBP or EBP_LEGACY).

- VIDEO_AND_FIXED_INTERVALS
- VIDEO_INTERVAL

**M2tsEbpPlacement (Enum)**

Selects which PIDs to place EBP markers on. They can either be placed only on the video PID, or on both the video PID and all audio PIDs. Only applicable when EBP segmentation markers are is selected (segmentationMarkers is EBP or EBP_LEGACY).

- VIDEO_AND_AUDIO_PIDS
- VIDEO_PID

**M2tsEsRateInPes (Enum)**

Controls whether to include the ES Rate field in the PES header.

- INCLUDE
- EXCLUDE

**M2tsPcrControl (Enum)**

When set to PCR_EVERY_PES_PACKET, a Program Clock Reference value is inserted for every Packetized Elementary Stream (PES) header. This is effective only when the PCR PID is the same as the video or audio elementary stream.

- PCR_EVERY_PES_PACKET
- CONFIGURED_PCR_PERIOD
**M2tsRateMode (Enum)**

When set to CBR, inserts null packets into transport stream to fill specified bitrate. When set to VBR, the bitrate setting acts as the maximum bitrate, but the output will not be padded up to that bitrate.

- VBR
- CBR

**M2tsScte35Source (Enum)**

Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output. This is only available for certain containers.

- PASSTHROUGH
- NONE

**M2tsSegmentationMarkers (Enum)**

Inserts segmentation markers at each segmentation_time period. rai_segstart sets the Random Access Indicator bit in the adaptation field. rai_adapt sets the RAI bit and adds the current timecode in the private data bytes. psi_segstart inserts PAT and PMT tables at the start of segments. ebp adds Encoder Boundary Point information to the adaptation field as per OpenCable specification OC-SP-EBP-I01-130118. ebp_legacy adds Encoder Boundary Point information to the adaptation field using a legacy proprietary format.

- NONE
- RAI_SEGSTART
- RAI_ADAPT
- PSI_SEGSTART
- EBP
- EBP_LEGACY

**M2tsSegmentationStyle (Enum)**

The segmentation style parameter controls how segmentation markers are inserted into the transport stream. With avails, it is possible that segments may be truncated, which can influence where future segmentation markers are inserted. When a segmentation style of "reset_cadence" is selected and a segment is truncated due to an avail, we will reset the segmentation cadence. This means the subsequent segment will have a duration of $segmentation_time seconds. When a segmentation style of "maintain_cadence" is selected and a segment is truncated due to an avail, we will not reset the segmentation cadence. This means the subsequent segment will likely be truncated as well. However, all segments after that will have a duration of $segmentation_time seconds. Note that EBP lookahead is a slight exception to this rule.

- MAINTAIN_CADENCE
- RESET_CADENCE

**M2tsSettings**

**dvbTeletextPid**

Packet Identifier (PID) for input source DVB Teletext data to this output. Can be entered as a decimal or hexadecimal value.

- **Type**: integer
Required: False
Minimum: 32
Maximum: 8182

**bitrate**

The output bitrate of the transport stream in bits per second. Setting to 0 lets the muxer automatically determine the appropriate bitrate. Other common values are 3750000, 7500000, and 15000000.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

**segmentationTime**

The length in seconds of each segment. Required unless markers is set to _none_.

Type: number
Required: False
Format: float
Minimum: 0.0

**audioPids**

Packet Identifier (PID) of the elementary audio stream(s) in the transport stream. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values.

Type: Array of type integer
Required: False

**rateMode**

Type: M2tsRateMode (p. 546)
Required: False

**ebpAudioInterval**

Type: M2tsEbpAudioInterval (p. 545)
Required: False

**fragmentTime**

The length in seconds of each fragment. Only used with EBP markers.

Type: number
Required: False
Format: float
Minimum: 0.0

**audioFramesPerPes**

The number of audio frames to insert for each PES packet.
**Properties**

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Required</th>
<th>Minimum</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>integer</td>
<td>False</td>
<td>0</td>
<td>2147483647</td>
</tr>
<tr>
<td>maxPCRInterval</td>
<td>integer</td>
<td>False</td>
<td>0</td>
<td>500</td>
</tr>
<tr>
<td>scte35Pid</td>
<td>integer</td>
<td>False</td>
<td>32</td>
<td>8182</td>
</tr>
<tr>
<td>privateMetadataPid</td>
<td>integer</td>
<td>False</td>
<td>32</td>
<td>8182</td>
</tr>
<tr>
<td>pmtInterval</td>
<td>integer</td>
<td>False</td>
<td>0</td>
<td>1000</td>
</tr>
<tr>
<td>segmentationStyle</td>
<td>M2tsSegmentationStyle</td>
<td>False</td>
<td></td>
<td></td>
</tr>
<tr>
<td>audioBufferModel</td>
<td>M2tsAudioBufferModel</td>
<td>False</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
programNumber

The value of the program number field in the Program Map Table.

  Type: integer
  Required: False
  Minimum: 0
  Maximum: 65535

dvbNitSettings

  Type: DvbNitSettings (p. 492)
  Required: False

scte35Source

  Type: M2tsScte35Source (p. 546)
  Required: False

pmtPid

Packet Identifier (PID) for the Program Map Table (PMT) in the transport stream. Can be entered as a decimal or hexadecimal value.

  Type: integer
  Required: False
  Minimum: 32
  Maximum: 8182

bufferModel

  Type: M2tsBufferModel (p. 545)
  Required: False

ebpPlacement

  Type: M2tsEbpPlacement (p. 545)
  Required: False

dvbSdtSettings

  Type: DvbSdtSettings (p. 493)
  Required: False

nullPacketBitrate

Value in bits per second of extra null packets to insert into the transport stream. This can be used if a downstream encryption system requires periodic null packets.

  Type: number
  Required: False
  Format: float
Minimum: 0.0

**pcrPid**

Packet Identifier (PID) of the Program Clock Reference (PCR) in the transport stream. When no value is given, the encoder will assign the same value as the Video PID. Can be entered as a decimal or hexadecimal value.

- **Type:** integer
- **Required:** False
- **Minimum:** 32
- **Maximum:** 8182

**minEbpInterval**

When set, enforces that Encoder Boundary Points do not come within the specified time interval of each other by looking ahead at input video. If another EBP is going to come in within the specified time interval, the current EBP is not emitted, and the segment is "stretched" to the next marker. The lookahead value does not add latency to the system. The Live Event must be configured elsewhere to create sufficient latency to make the lookahead accurate.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 10000

**transportStreamId**

The value of the transport stream ID field in the Program Map Table.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 65535

**pcrControl**

- **Type:** [M2tsPcrControl](p. 545)
- **Required:** False

**videoPid**

Packet Identifier (PID) of the elementary video stream in the transport stream. Can be entered as a decimal or hexadecimal value.

- **Type:** integer
- **Required:** False
- **Minimum:** 32
- **Maximum:** 8182

**esRateInPes**

- **Type:** [M2tsEsRateInPes](p. 545)
- **Required:** False
segmentationMarkers
Type: M2tsSegmentationMarkers (p. 546)
Required: False

dvbTdtSettings
Type: DvbTdtSettings (p. 497)
Required: False

patInterval
The number of milliseconds between instances of this table in the output transport stream.
Type: integer
Required: False
Minimum: 0
Maximum: 1000

dvbSubPids
Packet Identifier (PID) for input source DVB Subtitle data to this output. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values.
Type: Array of type integer
Required: False

M3u8PcrControl (Enum)
When set to PCR_EVERY_PES_PACKET a Program Clock Reference value is inserted for every Packetized Elementary Stream (PES) header. This parameter is effective only when the PCR PID is the same as the video or audio elementary stream.

PCR_EVERY_PES_PACKET
CONFIGURED_PCR_PERIOD

M3u8Scte35Source (Enum)
Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output. This is only available for certain containers.

PASSTHROUGH
NONE

M3u8Settings
pmtPid
Packet Identifier (PID) for the Program Map Table (PMT) in the transport stream. Can be entered as a decimal or hexadecimal value.
Type: integer
Required: False
Minimum: 32  
Maximum: 8182

**pcrPid**
Packet Identifier (PID) of the Program Clock Reference (PCR) in the transport stream. When no value is given, the encoder will assign the same value as the Video PID. Can be entered as a decimal or hexadecimal value.

Type: integer  
Required: False  
Minimum: 32  
Maximum: 8182

**audioPids**
Packet Identifier (PID) of the elementary audio stream(s) in the transport stream. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values.

Type: Array of type integer  
Required: False

**audioFramesPerPes**
The number of audio frames to insert for each PES packet.

Type: integer  
Required: False  
Minimum: 0  
Maximum: 2147483647

**scte35Pid**
Packet Identifier (PID) of the SCTE-35 stream in the transport stream. Can be entered as a decimal or hexadecimal value.

Type: integer  
Required: False  
Minimum: 32  
Maximum: 8182

**transportStreamId**
The value of the transport stream ID field in the Program Map Table.

Type: integer  
Required: False  
Minimum: 0  
Maximum: 65535

**videoPid**
Packet Identifier (PID) of the elementary video stream in the transport stream. Can be entered as a decimal or hexadecimal value.
**Properties**

- **Type**: integer
  - **Required**: False
  - **Minimum**: 32
  - **Maximum**: 8182

**pcrControl**

- **Type**: M3u8PcrControl (p. 551)
- **Required**: False

**privateMetadataPid**

Packet Identifier (PID) of the private metadata stream in the transport stream. Can be entered as a decimal or hexadecimal value.

- **Type**: integer
- **Required**: False
- **Minimum**: 32
- **Maximum**: 8182

**pmtInterval**

The number of milliseconds between instances of this table in the output transport stream.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 1000

**patInterval**

The number of milliseconds between instances of this table in the output transport stream.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 1000

**programNumber**

The value of the program number field in the Program Map Table.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 65535

**timedMetadataPid**

Packet Identifier (PID) of the timed metadata stream in the transport stream. Can be entered as a decimal or hexadecimal value.

- **Type**: integer
**Properties**

**Required**: False
**Minimum**: 32
**Maximum**: 8182

**timedMetadata**

*Type*: TimedMetadata (p. 578)
*Required*: False

**scte35Source**

*Type*: M3u8Scte35Source (p. 551)
*Required*: False

**MovClapAtom (Enum)**

When enabled, include 'clap' atom if appropriate for the video output settings.

- **INCLUDE**
- **EXCLUDE**

**MovCslgAtom (Enum)**

When enabled, file composition times will start at zero, composition times in the 'ctts' (composition time to sample) box for B-frames will be negative, and a 'cslg' (composition shift least greatest) box will be included per 14496-1 amendment 1. This improves compatibility with Apple players and tools.

- **INCLUDE**
- **EXCLUDE**

**MovMpeg2FourCCControl (Enum)**

When set to XDCAM, writes MPEG2 video streams into the QuickTime file using XDCAM fourcc codes. This increases compatibility with Apple editors and players, but may decrease compatibility with other players. Only applicable when the video codec is MPEG2.

- **XDCAM**
- **MPEG**

**MovPaddingControl (Enum)**

If set to OMNEON, inserts Omneon-compatible padding

- **OMNEON**
- **NONE**

**MovReference (Enum)**

A value of 'external' creates separate media files and the wrapper file (.mov) contains references to these media files. A value of 'self_contained' creates only a wrapper (.mov) file and this file contains all of the media.

- **SELF_CONTAINED**
- **EXTERNAL**
MovSettings

reference
- Type: MovReference (p. 554)
- Required: False

paddingControl
- Type: MovPaddingControl (p. 554)
- Required: False

mpeg2FourCCControl
- Type: MovMpeg2FourCCControl (p. 554)
- Required: False

cslgAtom
- Type: MovCslgAtom (p. 554)
- Required: False

clapAtom
- Type: MovClapAtom (p. 554)
- Required: False

Mp2Settings

channels
Set Channels to specify the number of channels in this output audio track. Choosing Mono in the console will give you 1 output channel; choosing Stereo will give you 2. In the API, valid values are 1 and 2.

- Type: integer
- Required: False
- Minimum: 1
- Maximum: 2

bitrate
Average bitrate in bits/second.

- Type: integer
- Required: False
- Minimum: 32000
- Maximum: 384000

sampleRate
Sample rate in hz.
Properties

**Type**
Type: integer  
Required: False  
Minimum: 32000  
Maximum: 48000

**Mp4CslgAtom (Enum)**
When enabled, file composition times will start at zero, composition times in the 'ctts' (composition time to sample) box for B-frames will be negative, and a 'cslg' (composition shift least greatest) box will be included per 14496-1 amendment 1. This improves compatibility with Apple players and tools.

INCLUDE  
EXCLUDE

**Mp4FreeSpaceBox (Enum)**
Inserts a free-space box immediately after the moov box.

INCLUDE  
EXCLUDE

**Mp4MoovPlacement (Enum)**
If set to PROGRESSIVE_DOWNLOAD, the MOOV atom is relocated to the beginning of the archive as required for progressive downloading. Otherwise it is placed normally at the end.

PROGRESSIVE_DOWNLOAD  
NORMAL

**Mp4Settings**

**mp4MajorBrand**
Overrides the "Major Brand" field in the output file. Usually not necessary to specify.

Type: string  
Required: False

**moovPlacement**

Type: Mp4MoovPlacement (p. 556)  
Required: False

**cslgAtom**

Type: Mp4CslgAtom (p. 556)  
Required: False

**freeSpaceBox**

Type: Mp4FreeSpaceBox (p. 556)  
Required: False
Mpeg2AdaptiveQuantization (Enum)
Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

- OFF
- LOW
- MEDIUM
- HIGH

Mpeg2CodecLevel (Enum)
Use Level (Mpeg2CodecLevel) to set the MPEG-2 level for the video output.

- AUTO
- LOW
- MAIN
- HIGH1440
- HIGH

Mpeg2CodecProfile (Enum)
Use Profile (Mpeg2CodecProfile) to set the MPEG-2 profile for the video output.

- MAIN
- PROFILE_422

Mpeg2FramerateControl (Enum)
Using the API, set FramerateControl to INITIALIZE_FROM_SOURCE if you want the service to use the framerate from the input. Using the console, do this by choosing INITIALIZE_FROM_SOURCE for Framerate.

- INITIALIZE_FROM_SOURCE
- SPECIFIED

Mpeg2FramerateConversionAlgorithm (Enum)
When set to INTERPOLATE, produces smoother motion during framerate conversion.

- DUPLICATE_DROP
- INTERPOLATE

Mpeg2GopSizeUnits (Enum)
Indicates if the GOP Size in MPEG2 is specified in frames or seconds. If seconds the system will convert the GOP Size into a frame count at run time.

- FRAMES
- SECONDS

Mpeg2InterlaceMode (Enum)
Use Interlace mode (InterlaceMode) to choose the scan line type for the output.
* Top Field First (TOP_FIELD) and Bottom Field First (BOTTOM_FIELD) produce interlaced output with the entire output having the same field polarity (top or bottom first).

* Follow, Default Top (FOLLOW_TOP_FIELD) and Follow, Default Bottom (FOLLOW_BOTTOM_FIELD) use the same field polarity as the source. Therefore, behavior depends on the input scan type. If the source is interlaced, the output will be interlaced with the same polarity as the source (it will follow the source). The output could therefore be a mix of "top field first" and "bottom field first". If the source is progressive, the output will be interlaced with "top field first" or "bottom field first" polarity, depending on which of the Follow options you chose.

PROGRESSIVE
  TOP_FIELD
  BOTTOM_FIELD
  FOLLOW_TOP_FIELD
  FOLLOW_BOTTOM_FIELD

**Mpeg2IntraDcPrecision (Enum)**

Use Intra DC precision (Mpeg2IntraDcPrecision) to set quantization precision for intra-block DC coefficients. If you choose the value auto, the service will automatically select the precision based on the per-frame compression ratio.

  AUTO
  INTRA_DC_PRECISION_8
  INTRA_DC_PRECISION_9
  INTRA_DC_PRECISION_10
  INTRA_DC_PRECISION_11

**Mpeg2ParControl (Enum)**

Using the API, enable ParFollowSource if you want the service to use the pixel aspect ratio from the input. Using the console, do this by choosing Follow source for Pixel aspect ratio.

  INITIALIZE_FROM_SOURCE
  SPECIFIED

**Mpeg2QualityTuningLevel (Enum)**

Use Quality tuning level (Mpeg2QualityTuningLevel) to specify whether to use single-pass or multipass video encoding.

  SINGLE_PASS
  MULTI_PASS

**Mpeg2RateControlMode (Enum)**

Use Rate control mode (Mpeg2RateControlMode) to specify whether the bitrate is variable (vbr) or constant (cbr).

  VBR
  CBR

**Mpeg2SceneChangeDetect (Enum)**

Scene change detection (inserts I-frames on scene changes).
DISABLED
ENABLED

**Mpeg2Settings**

**minInterval**

Enforces separation between repeated (cadence) I-frames and I-frames inserted by Scene Change Detection. If a scene change I-frame is within I-interval frames of a cadence I-frame, the GOP is shrunk and/or stretched to the scene change I-frame. GOP stretch requires enabling lookahead as well as setting I-interval. The normal cadence resumes for the next GOP. This setting is only used when Scene Change Detect is enabled. Note: Maximum GOP stretch = GOP size + Min-I-interval - 1

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 30

**parNumerator**

Pixel Aspect Ratio numerator.

- **Type:** integer
- **Required:** False
- **Minimum:** 1
- **Maximum:** 2147483647

**gopSizeUnits**

- **Type:** Mpeg2GopSizeUnits (p. 557)
- **Required:** False

**hrdBufferSize**

Size of buffer (HRD buffer model). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.

- **Type:** integer
- **Required:** False
- **Minimum:** -2147483648
- **Maximum:** 2147483647

**qualityTuningLevel**

- **Type:** Mpeg2QualityTuningLevel (p. 558)
- **Required:** False

**maxBitrate**

Maximum bitrate in bits/second (for VBR mode only). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.

- **Type:** integer
**Properties**

**Required**: False

**Minimum**: -2147483648

**Maximum**: 2147483647

**bitrate**

Average bitrate in bits/second. Required for VBR, CBR, and ABR. Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.

**Type**: integer

**Required**: False

**Minimum**: 1000

**Maximum**: 2147483647

**spatialAdaptiveQuantization**

**Type**: Mpeg2SpatialAdaptiveQuantization (p. 563)

**Required**: False

**slowPal**

**Type**: Mpeg2SlowPal (p. 563)

**Required**: False

**codecProfile**

**Type**: Mpeg2CodecProfile (p. 557)

**Required**: False

**intraDcPrecision**

**Type**: Mpeg2IntraDcPrecision (p. 558)

**Required**: False

**softness**

Softness. Selects quantizer matrix, larger values reduce high-frequency content in the encoded image.

**Type**: integer

**Required**: False

**Minimum**: 0

**Maximum**: 128

**framerateControl**

**Type**: Mpeg2FramerateControl (p. 557)

**Required**: False

**telecine**

**Type**: Mpeg2Telecine (p. 563)
Required: False

framerateConversionAlgorithm

Type: Mpeg2FramerateConversionAlgorithm (p. 557)
Required: False

codecLevel

Type: Mpeg2CodecLevel (p. 557)
Required: False

temporalAdaptiveQuantization

Type: Mpeg2TemporalAdaptiveQuantization (p. 563)
Required: False

hrdBufferInitialFillPercentage

Percentage of the buffer that should initially be filled (HRD buffer model).

Type: integer
Required: False
Minimum: 0
Maximum: 100

framerateNumerator

Framerate numerator - framerate is a fraction, e.g. 24000 / 1001 = 23.976 fps.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

numberBFramesBetweenReferenceFrames

Number of B-frames between reference frames.

Type: integer
Required: False
Minimum: 0
Maximum: 7

gopClosedCadence

Frequency of closed GOPs. In streaming applications, it is recommended that this be set to 1 so a decoder joining mid-stream will receive an IDR frame as quickly as possible. Setting this value to 0 will break output segmenting.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

framerateDenominator
Framerate denominator.
Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

adaptiveQuantization
Type: Mpeg2AdaptiveQuantization (p. 557)
Required: False

interlaceMode
Type: Mpeg2InterlaceMode (p. 557)
Required: False

gopSize
GOP Length (keyframe interval) in frames or seconds. Must be greater than zero.
Type: number
Required: False
Format: float
Minimum: 0.0

sceneChangeDetect
Type: Mpeg2SceneChangeDetect (p. 558)
Required: False

parDenominator
Pixel Aspect Ratio denominator.
Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

parControl
Type: Mpeg2ParControl (p. 558)
Required: False

syntax
Type: Mpeg2Syntax (p. 563)
**Required**: False

**rateControlMode**

**Type**: Mpeg2RateControlMode (p. 558)

**Required**: False

**Mpeg2SlowPal (Enum)**

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

- DISABLED
- ENABLED

**Mpeg2SpatialAdaptiveQuantization (Enum)**

Adjust quantization within each frame based on spatial variation of content complexity.

- DISABLED
- ENABLED

**Mpeg2Syntax (Enum)**

Produces a Type D-10 compatible bitstream (SMPTE 356M-2001).

- DEFAULT
- D_10

**Mpeg2Telecine (Enum)**

Only use Telecine (Mpeg2Telecine) when you set Framerate (Framerate) to 29.970. Set Telecine (Mpeg2Telecine) to Hard (hard) to produce a 29.97i output from a 23.976 input. Set it to Soft (soft) to produce 23.976 output and leave conversion to the player.

- NONE
- SOFT
- HARD

**Mpeg2TemporalAdaptiveQuantization (Enum)**

Adjust quantization within each frame based on temporal variation of content complexity.

- DISABLED
- ENABLED

**MsSmoothAudioDeduplication (Enum)**

COMBINE_DUPLICATE_STREAMS combines identical audio encoding settings across a Microsoft Smooth output group into a single audio stream.

- COMBINE_DUPLICATE_STREAMS
- NONE
MsSmoothEncryptionSettings

spekeKeyProvider

  Type: SpekeKeyProvider (p. 575)
  Required: True

MsSmoothGroupSettings

fragmentLength

Use Fragment length (FragmentLength) to specify the mp4 fragment sizes in seconds. Fragment length must be compatible with GOP size and framerate.

  Type: integer
  Required: True
  Minimum: 1
  Maximum: 2147483647

encryption

  Type: MsSmoothEncryptionSettings (p. 564)
  Required: False

audioDeduplication

  Type: MsSmoothAudioDeduplication (p. 563)
  Required: False

manifestEncoding

  Type: MsSmoothManifestEncoding (p. 564)
  Required: False

destination

Use Destination (Destination) to specify the S3 output location and the output filename base. Destination accepts format identifiers. If you do not specify the base filename in the URI, the service will use the filename of the input file. If your job has multiple inputs, the service uses the filename of the first input file.

  Type: string
  Required: True
  Pattern: ^s3: /\/

MsSmoothManifestEncoding (Enum)

Use Manifest encoding (MsSmoothManifestEncoding) to specify the encoding format for the server and client manifest. Valid options are utf8 and utf16.

  UTF8
  UTF16
NielsenConfiguration

distributorID

Use Distributor ID (DistributorID) to specify the distributor ID that is assigned to your organization by Nielsen.

Type: string
Required: False

breakoutCode

Use Nielsen Configuration (NielsenConfiguration) to set the Nielsen measurement system breakout code. Supported values are 0, 3, 7, and 9.

Type: integer
Required: False
Minimum: 0
Maximum: 9

NoiseReducer

filter

Type: NoiseReducerFilter (p. 565)
Required: True

filterSettings

Type: NoiseReducerFilterSettings (p. 566)
Required: False

spatialFilterSettings

Type: NoiseReducerSpatialFilterSettings (p. 566)
Required: False

NoiseReducerFilter (Enum)

Use Noise reducer filter (NoiseReducerFilter) to select one of the following spatial image filtering functions. To use this setting, you must also enable Noise reducer (NoiseReducer).

* Bilateral is an edge preserving noise reduction filter
* Mean (softest), Gaussian, Lanczos, and Sharpen (sharpest) are convolution filters
* Conserve is a min/max noise reduction filter
* Spatial is frequency-domain filter based on JND principles.

BILATERAL
MEAN
GAUSSIAN
LANCZOS
SHARPEN
CONSERVE
SPATIAL

**NoiseReducerFilterSettings**

**strength**

Relative strength of noise reducing filter. Higher values produce stronger filtering.

  **Type:** integer  
  **Required:** False  
  **Minimum:** 0  
  **Maximum:** 3

**NoiseReducerSpatialFilterSettings**

**strength**

Relative strength of noise reducing filter. Higher values produce stronger filtering.

  **Type:** integer  
  **Required:** False  
  **Minimum:** 0  
  **Maximum:** 16

**postFilterSharpenStrength**

Specify strength of post noise reduction sharpening filter, with 0 disabling the filter and 3 enabling it at maximum strength.

  **Type:** integer  
  **Required:** False  
  **Minimum:** 0  
  **Maximum:** 3

**speed**

The speed of the filter, from -2 (lower speed) to 3 (higher speed), with 0 being the nominal value.

  **Type:** integer  
  **Required:** False  
  **Minimum:** -2  
  **Maximum:** 3

**Output**

**extension**

Use Extension (Extension) to specify the file extension for outputs in File output groups. If you do not specify a value, the service will use default extensions by container type as follows:
* MPEG-2 transport stream, m2ts
* Quicktime, mov
* MXF container, mxf
* MPEG-4 container, mp4
* No Container, the service will use codec extensions (e.g. AAC, H265, H265, AC3)

  Type: string
  Required: False

**videoDescription**

(VideoDescription) contains a group of video encoding settings. The specific video settings depend on the video codec you choose when you specify a value for Video codec (codec). Include one instance of (VideoDescription) per output.

  Type: VideoDescription (p. 580)
  Required: False

**audioDescriptions**

(AudioDescriptions) contains groups of audio encoding settings organized by audio codec. Include one instance of (AudioDescriptions) per output. (AudioDescriptions) can contain multiple groups of encoding settings.

  Type: Array of type AudioDescription (p. 475)
  Required: False

**containerSettings**

  Type: ContainerSettings (p. 488)
  Required: False

**preset**

Use Preset (Preset) to specify a preset for your transcoding settings. Provide the system or custom preset name. You can specify either Preset (Preset) or Container settings (ContainerSettings), but not both.

  Type: string
  Required: False

**outputSettings**

  Type: OutputSettings (p. 570)
  Required: False

**captionDescriptions**

(CaptionDescriptions) contains groups of captions settings. For each output that has captions, include one instance of (CaptionDescriptions). (CaptionDescriptions) can contain multiple groups of captions settings.
Properties

**Type**
- **Array of type** CaptionDescription (p. 484)
- **Required**: False

**nameModifier**

Use Name modifier (NameModifier) to have the service add a string to the end of each output filename. You specify the base filename as part of your destination URI. When you create multiple outputs in the same output group, Name modifier is required. Name modifier also accepts format identifiers. For DASH ISO outputs, if you use the format identifiers $Number$ or $Time$ in one output, you must use them in the same way in all outputs of the output group.

- **Type**: string
- **Required**: False

**OutputChannelMapping**

**inputChannels**

- **Type**: Array of type integer
- **Required**: True

**OutputDetail**

**durationInMs**

- **Type**: integer
- **Required**: False

**videoDetails**

- **Type**: VideoDetail (p. 582)
- **Required**: False

**OutputGroup**

**outputs**

This object holds groups of encoding settings, one group of settings per output.

- **Type**: Array of type Output (p. 566)
- **Required**: True

**outputGroupSettings**

- **Type**: OutputGroupSettings (p. 569)
- **Required**: True

**name**

- **Type**: string
**Properties**

**Required**: False

**customName**

Use Custom Group Name (CustomName) to specify a name for the output group. This value is displayed on the console and can make your job settings JSON more human-readable. It does not affect your outputs. Use up to twelve characters that are either letters, numbers, spaces, or underscores.

*Type*: string  
*Required*: False

**OutputGroupDetail**

**outputDetails**

*Type*: Array of type `OutputDetail (p. 568)`  
*Required*: False

**OutputGroupSettings**

**dashIsoGroupSettings**

*Type*: `DashIsoGroupSettings (p. 489)`  
*Required*: False

**fileGroupSettings**

*Type*: `FileGroupSettings (p. 503)`  
*Required*: False

**msSmoothGroupSettings**

*Type*: `MsSmoothGroupSettings (p. 564)`  
*Required*: False

**type**

*Type*: `OutputGroupType (p. 569)`  
*Required*: True

**hlsGroupSettings**

*Type*: `HlsGroupSettings (p. 527)`  
*Required*: False

**OutputGroupType (Enum)**

- HLS_GROUP_SETTINGS
- DASH_ISO_GROUP_SETTINGS
- FILE_GROUP_SETTINGS
- MS_SMOOTH_GROUP_SETTINGS
OutputSdt (Enum)

Selects method of inserting SDT information into output stream. "Follow input SDT" copies SDT information from input stream to output stream. "Follow input SDT if present" copies SDT information from input stream to output stream if SDT information is present in the input, otherwise it will fall back on the user-defined values. Enter "SDT Manually" means user will enter the SDT information. "No SDT" means output stream will not contain SDT information.

- SDT_FOLLOW
- SDT_FOLLOW_IF_PRESENT
- SDT_MANUAL
- SDT_NONE

OutputSettings

hlsSettings

- Type: HlsSettings (p. 531)
- Required: False

ProresCodecProfile (Enum)

Use Profile (ProResCodecProfile) to specify the type of Apple ProRes codec to use for this output.

- APPLE_PRORES_422
- APPLE_PRORES_422_HQ
- APPLE_PRORES_422_LT
- APPLE_PRORES_422_PROXY

ProresFramerateControl (Enum)

Using the API, set FramerateControl to INITIALIZE_FROM_SOURCE if you want the service to use the framerate from the input. Using the console, do this by choosing INITIALIZE_FROM_SOURCE for Framerate.

- INITIALIZE_FROM_SOURCE
- SPECIFIED

ProresFramerateConversionAlgorithm (Enum)

When set to INTERPOLATE, produces smoother motion during framerate conversion.

- DUPLICATE_DROP
- INTERPOLATE

ProresInterlaceMode (Enum)

Use Interlace mode (InterlaceMode) to choose the scan line type for the output.

* Top Field First (TOP_FIELD) and Bottom Field First (BOTTOM_FIELD) produce interlaced output with the entire output having the same field polarity (top or bottom first).

* Follow, Default Top (FOLLOW_TOP_FIELD) and Follow, Default Bottom (FOLLOW_BOTTOM_FIELD) use the same field polarity as the source. Therefore, behavior depends on the input scan type. If the source is interlaced, the output will be interlaced with the same polarity as the source (it will follow the source). The output could therefore be a mix of "top field first" and "bottom field first". If the source is
progressive, the output will be interlaced with "top field first" or "bottom field first" polarity, depending on which of the Follow options you chose.

```plaintext
PROGRESSIVE
TOP_FIELD
BOTTOM_FIELD
FOLLOW_TOP_FIELD
FOLLOW_BOTTOM_FIELD
```

**ProresParControl (Enum)**

Use (ProresParControl) to specify how the service determines the pixel aspect ratio. Set to Follow source (INITIALIZE FROM SOURCE) to use the pixel aspect ratio from the input. To specify a different pixel aspect ratio: Using the console, choose it from the dropdown menu. Using the API, set ProresParControl to (SPECIFIED) and provide for (ParNumerator) and (ParDenominator).

```plaintext
INITIALIZE_FROM_SOURCE
SPECIFIED
```

**ProresSettings**

**SlowPal**

- **Type**: ProresSlowPal (p. 572)
- **Required**: False

**FramerateControl**

- **Type**: ProresFramerateControl (p. 570)
- **Required**: False

**Telecine**

- **Type**: ProresTelecine (p. 572)
- **Required**: False

**FramerateDenominator**

Framerate denominator.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 2147483647

**FramerateConversionAlgorithm**

- **Type**: ProresFramerateConversionAlgorithm (p. 570)
- **Required**: False

**InterlaceMode**

- **Type**: ProresInterlaceMode (p. 570)
Properties

 codecProfile

 Required: False

 Type: ProresCodecProfile (p. 570)

 parNumerator

 Pixel Aspect Ratio numerator.

 Type: integer
 Required: False
 Minimum: 1
 Maximum: 2147483647

 parControl

 Type: ProresParControl (p. 571)
 Required: False

 parDenominator

 Pixel Aspect Ratio denominator.

 Type: integer
 Required: False
 Minimum: 1
 Maximum: 2147483647

 framerateNumerator

 When you use the API for transcode jobs that use framerate conversion, specify the framerate as a fraction. For example, 24000 / 1001 = 23.976 fps. Use FramerateNumerator to specify the numerator of this fraction. In this example, use 24000 for the value of FramerateNumerator.

 Type: integer
 Required: False
 Minimum: 1
 Maximum: 2147483647

 ProresSlowPal (Enum)

 Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

 DISABLED
 ENABLED

 ProresTelecine (Enum)

 Only use Telecine (ProresTelecine) when you set Framerate (Framerate) to 29.970. Set Telecine (ProresTelecine) to Hard (hard) to produce a 29.97i output from a 23.976 input. Set it to Soft (soft) to produce 23.976 output and leave conversion to the player.
NONE
HARD

**Rectangle**

**width**

Width of rectangle in pixels.

*Type: integer*

*Required: True*

*Minimum: -2147483648*

*Maximum: 2147483647*

**x**

The distance, in pixels, between the rectangle and the left edge of the video frame.

*Type: integer*

*Required: True*

*Minimum: -2147483648*

*Maximum: 2147483647*

**y**

The distance, in pixels, between the rectangle and the top edge of the video frame.

*Type: integer*

*Required: True*

*Minimum: -2147483648*

*Maximum: 2147483647*

**height**

Height of rectangle in pixels.

*Type: integer*

*Required: True*

*Minimum: -2147483648*

*Maximum: 2147483647*

**RemixSettings**

**channelsOut**

Specify the number of channels in this output after remixing. Valid values: 1, 2, 4, 6, 8

*Type: integer*

*Required: True*

*Minimum: 1*

*Maximum: 8*
channelMapping

Type: ChannelMapping (p. 486)
Required: True

channelsIn

Specify the number of audio channels from your input that you want to use in your output. With remixing, you might combine or split the data in these channels, so the number of channels in your final output might be different.

Type: integer
Required: True
Minimum: 1
Maximum: 16

RespondToAfd (Enum)

Use Respond to AFD (RespondToAfd) to specify how the service changes the video itself in response to AFD values in the input.

* Choose Respond to clip the input video frame according to the AFD value, input display aspect ratio, and output display aspect ratio.

* Choose Passthrough to include the input AFD values. Do not choose this when AfdSignaling is set to (NONE). A preferred implementation of this workflow is to set RespondToAfd to (NONE) and set AfdSignaling to (AUTO).

* Choose None to remove all input AFD values from this output.

NONE
RESPOND
PASSTHROUGH

ScalingBehavior (Enum)

Applies only if your input aspect ratio is different from your output aspect ratio. Enable Stretch to output (StretchToOutput) to have the service stretch your video image to fit. Leave this setting disabled to allow the service to letterbox your video instead. This setting overrides any positioning value you specify elsewhere in the job.

DEFAULT
STRETCH_TO_OUTPUT

SccDestinationFramerate (Enum)

Set Framerate (SccDestinationFramerate) to make sure that the captions and the video are synchronized in the output. Specify a framerate that matches the framerate of the associated video. If the video framerate is 29.97, choose 29.97 dropframe (FRAMERATE_29_97_DROPFRAME) only if the video has video_insertion=true and drop_frame_timecode=true; otherwise, choose 29.97 non-dropframe (FRAMERATE_29_97_NON_DROPFRAME).

FRAMERATE_23_97
FRAMERATE_24
FRAMERATE_29_97_DROPFRAME
FRAMERATE_29_97_NON_DROPFRAME
**SccDestinationSettings**

**framerate**
- **Type**: SccDestinationFramerate (p. 574)
- **Required**: False

**SpekeKeyProvider**

**resourceld**
The SPEKE-compliant server uses Resource ID (Resourceld) to identify content.
- **Type**: string
- **Required**: True
- **Pattern**: ^\[\w-]+$

**systemIds**
Relates to SPEKE implementation. DRM system identifiers. DASH output groups support a max of two system ids. Other group types support one system id.
- **Type**: Array of type string
- **Required**: True

**url**
Use URL (Url) to specify the SPEKE-compliant server that will provide keys for content.
- **Type**: string
- **Required**: True
- **Format**: uri
- **Pattern**: ^https:\/\/

**StaticKeyProvider**

**keyFormatVersions**
Relates to DRM implementation. Either a single positive integer version value or a slash delimited list of version values (1/2/3).
- **Type**: string
- **Required**: False
- **Pattern**: ^\d+(\d+\d+)*$

**keyFormat**
Relates to DRM implementation. Sets the value of the KEYFORMAT attribute. Must be 'identity' or a reverse DNS string. May be omitted to indicate an implicit value of 'identity'.
- **Type**: string
- **Required**: False
- **Pattern**: ^(identity|[A-Za-z]{2,6}(\.[A-Za-z0-9-]{1,63}){1,63})+$
staticKeyValue

Relates to DRM implementation. Use a 32-character hexadecimal string to specify Key Value
(StaticKeyValue).

  Type: string
  Required: True
  Pattern: ^[A-Za-z0-9]{32}$

url

Relates to DRM implementation. The location of the license server used for protecting content.

  Type: string
  Required: True
  Format: uri

TeletextDestinationSettings

pageNumber

Set pageNumber to the Teletext page number for the destination captions for this output. This value
must be a three-digit hexadecimal string; strings ending in -FF are invalid. If you are passing through the
entire set of Teletext data, do not use this field.

  Type: string
  Required: False
  Pattern: ^[1-8][0-9a-fA-F][0-9a-eA-E]$}

TeletextSourceSettings

pageNumber

Use Page Number (PageNumber) to specify the three-digit hexadecimal page number that will be used
for Teletext captions. Do not use this setting if you are passing through teletext from the input source to
output.

  Type: string
  Required: False
  Pattern: ^[1-8][0-9a-fA-F][0-9a-eA-E]$}

TimecodeBurnin

prefix

Use Prefix (Prefix) to place ASCII characters before any burned-in timecode. For example, a prefix of "EZ-"
will result in the timecode "EZ-00:00:00:00". Provide either the characters themselves or the ASCII code
equivalents. The supported range of characters is 0x20 through 0x7e. This includes letters, numbers, and
all special characters represented on a standard English keyboard.

  Type: string
  Required: False
  Pattern: ^[ -]+$
fontSize

Use Font Size (FontSize) to set the font size of any burned-in timecode. Valid values are 10, 16, 32, 48.

  Type: integer
  Required: False
  Minimum: 10
  Maximum: 48

position

  Type: TimecodeBurninPosition (p. 577)
  Required: False

TimecodeBurninPosition (Enum)

Use Position (Position) under Timecode burn-in (TimecodeBurnIn) to specify the location the burned-in timecode on output video.

  TOP_CENTER
  TOP_LEFT
  TOP_RIGHT
  MIDDLE_LEFT
  MIDDLE_CENTER
  MIDDLE_RIGHT
  BOTTOM_LEFT
  BOTTOM_CENTER
  BOTTOM_RIGHT

TimecodeConfig

timestampOffset

Only applies to outputs that support program-date-time stamp. Use Time stamp offset (TimestampOffset) to overwrite the timecode date without affecting the time and frame number. To use this, you must also enable Insert program-date-time (InsertProgramDateTime) in the output settings.

  Type: string
  Required: False
  Pattern: ^([0-9]{4})-(0[1-9]|1[0-2])-(0[1-9]|1[2][0-9]|3[01])$

anchor

If you use an editing platform that relies on an anchor timecode, use Anchor Timecode (Anchor) to specify a timecode that will match the input video frame to the output video frame. Use 24-hour format with frame number, (HH:MM:SS:FF) or (HH:MM:SS;FF). This setting ignores framerate conversion. System behavior for Anchor Timecode varies depending on your setting for Timecode source (TimecodeSource).

* If Timecode source (TimecodeSource) is set to Specified Start (specifiedstart), the first input frame is the specified value in Start Timecode (Start). Anchor Timecode (Anchor) and Start Timecode (Start) are used calculate output timecode.

* If Timecode source (TimecodeSource) is set to Start at 0 (zerobased) the first frame is 00:00:00:00.

* If Timecode source (TimecodeSource) is set to Embedded (embedded), the first frame is the timecode value on the first input frame of the input.
Properties

Type: string
Required: False
Format: timecode
Pattern: ^([01][0-9]|2[0-4]):[0-5][0-9]:[0-5][0-9]:;[0-9]{2}$

start

Only use when you set Timecode Source (TimecodeSource) to Specified Start (SPECIFIEDSTART). Use Start timecode (Start) to specify the timecode for the initial frame. Use 24-hour format with frame number, (HH:MM:SS:FF) or (HH:MM:SS;FF).

Type: string
Required: False
Format: timecode
Pattern: ^([01][0-9]|2[0-4]):[0-5][0-9]:[0-5][0-9]:;[0-9]{2}$

source

Type: TimecodeSource (p. 578)
Required: False

TimecodeSource (Enum)

Use Timecode source (TimecodeSource) to set how timecodes are handled within this input. To make sure that your video, audio, captions, and markers are synchronized and that time-based features, such as image inserter, work correctly, choose the Timecode source option that matches your assets. All timecodes are in a 24-hour format with frame number (HH:MM:SS:FF).

* Embedded (EMBEDDED) - Use the timecode that is in the input video. If no embedded timecode is in the source, the service will use Start at 0 (ZEROBASED) instead.

* Start at 0 (ZEROBASED) - Set the timecode of the initial frame to 00:00:00:00.

* Specified Start (SPECIFIEDSTART) - Set the timecode of the initial frame to a value other than zero. You use Start timecode (Start) to provide this value.

EMBEDDED
ZEROBASED
SPECIFIEDSTART

TimedMetadata (Enum)

If PASSTHROUGH, inserts ID3 timed metadata from the timed_metadata REST command into this output. Only available for certain containers.

PASSTHROUGH
NONE

TimedMetadataInsertion

id3Insertions

Id3Insertions contains the array of Id3Insertion instances.
Properties

Type: Array of type Id3Insertion (p. 532)
Required: True

Timing

finishTime
The time, in Unix epoch format, that the transcoding job finished

Type: string
Required: False
Format: date-time

submitTime
The time, in Unix epoch format, that you submitted the job.

Type: string
Required: False
Format: date-time

startTime
The time, in Unix epoch format, that transcoding for the job began.

Type: string
Required: False
Format: date-time

TtmlDestinationSettings

stylePassthrough

Type: TtmlStylePassthrough (p. 579)
Required: False

TtmlStylePassthrough (Enum)
Pass through style and position information from a TTML-like input source (TTML, SMPTE-TT, CFF-TT) to the CFF-TT output or TTML output.

ENABLED
DISABLED

VideoCodec (Enum)

Type of video codec

FRAME_CAPTURE
H_264
H_265
Properties

MPEG2
PRORES

VideoCodecSettings

h265Settings
  Type: H265Settings
  Required: False

codec
  Type: VideoCodec
  Required: True

proresSettings
  Type: ProresSettings
  Required: False

mpeg2Settings
  Type: Mpeg2Settings
  Required: False

h264Settings
  Type: H264Settings
  Required: False

frameCaptureSettings
  Type: FrameCaptureSettings
  Required: False

VideoDescription

fixedAfd
Applies only if you set AFD Signaling(AfdSignaling) to Fixed (FIXED). Use Fixed (FixedAfd) to specify a four-bit AFD value which the service will write on all frames of this video output.

  Type: integer
  Required: False
  Minimum: 0
  Maximum: 15

scalingBehavior
  Type: ScalingBehavior

Required: False

**respondToAfd**

Type: **RespondToAfd (p. 574)**  
Required: False

**codecSettings**

Type: **VideoCodecSettings (p. 580)**  
Required: True

**afdSignaling**

Type: **AfdSignaling (p. 472)**  
Required: False

**colorMetadata**

Type: **ColorMetadata (p. 487)**  
Required: False

**timecodeInsertion**

Type: **VideoTimecodeInsertion (p. 584)**  
Required: False

**width**

Use **Width (Width)** to define the video resolution width, in pixels, for this output. If you don't provide a value here, the service will use the input width.

Type: integer  
Required: False  
Minimum: 32  
Maximum: 4096

**sharpness**

Use **Sharpness (Sharpness)** setting to specify the strength of anti-aliasing. This setting changes the width of the anti-alias filter kernel used for scaling. Sharpness only applies if your output resolution is different from your input resolution, and if you set **Anti-alias (AntiAlias)** to ENABLED. 0 is the softest setting, 100 the sharpest, and 50 recommended for most content.

Type: integer  
Required: False  
Minimum: 0  
Maximum: 100

**antiAlias**

Type: **AntiAlias (p. 473)**  
Required: False
**videoPreprocessors**

Find additional transcoding features under Preprocessors (VideoPreprocessors). Enable the features at each output individually. These features are disabled by default.

*Type: VideoPreprocessor (p. 583)*
*Required: False*

**position**

Use Position (Position) to point to a rectangle object to define your position. This setting overrides any other aspect ratio.

*Type: Rectangle (p. 573)*
*Required: False*

**dropFrameTimecode**

*Type: DropFrameTimecode (p. 492)*
*Required: False*

**crop**

Applies only if your input aspect ratio is different from your output aspect ratio. Use Input cropping rectangle (Crop) to specify the video area the service will include in the output. This will crop the input source, causing video pixels to be removed on encode. Do not use this setting if you have enabled Stretch to output (stretchToOutput) in your output settings.

*Type: Rectangle (p. 573)*
*Required: False*

**height**

Use the Height (Height) setting to define the video resolution height for this output. Specify in pixels. If you don’t provide a value here, the service will use the input height.

*Type: integer*
*Required: False*
*Minimum: 32*
*Maximum: 2160*

**VideoDetail**

**heightInPx**

Height in pixels for the output

*Type: integer*
*Required: False*

**widthInPx**

Width in pixels for the output

*Type: integer*
VideoPreprocessor

timecodeBurnin
Timecode burn-in (TimecodeBurnIn)--Burns the output timecode and specified prefix into the output.

Type: TimecodeBurnin (p. 576)
Required: False

noiseReducer
Enable the Noise reducer (NoiseReducer) feature to remove noise from your video output if necessary. Enable or disable this feature for each output individually. This setting is disabled by default.

Type: NoiseReducer (p. 565)
Required: False

colorCorrector
Enable the Color corrector (ColorCorrector) feature if necessary. Enable or disable this feature for each output individually. This setting is disabled by default.

Type: ColorCorrector (p. 487)
Required: False

imageInserter
Enable the Image inserter (ImageInserter) feature to include a graphic overlay on your video. Enable or disable this feature for each output individually. This setting is disabled by default.

Type: ImageInserter (p. 532)
Required: False

deinterlacer
Use Deinterlacer (Deinterlacer) to produce smoother motion and a clearer picture.

Type: Deinterlacer (p. 491)
Required: False

VideoSelector

colorSpace

Type: ColorSpace (p. 488)
Required: False

hdr10Metadata

Type: Hdr10Metadata (p. 522)
Required: False
programNumber

Selects a specific program from within a multi-program transport stream. Note that Quad 4K is not currently supported.

- **Type**: integer
- **Required**: False
- **Minimum**: -2147483648
- **Maximum**: 2147483647

pid

Use PID (Pid) to select specific video data from an input file. Specify this value as an integer; the system automatically converts it to the hexadecimal value. For example, 257 selects PID 0x101. A PID, or packet identifier, is an identifier for a set of data in an MPEG-2 transport stream container.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 2147483647

colorSpaceUsage

- **Type**: ColorSpaceUsage (p. 488)
- **Required**: False

VideoTimecodeInsertion (Enum)

Enable Timecode insertion to include timecode information in this output. Do this in the API by setting (VideoTimecodeInsertion) to (PIC_TIMING_SEI). To get timecodes to appear correctly in your output, also set up the timecode configuration for your job in the input settings. Only enable Timecode insertion when the input framerate is identical to output framerate. Disable this setting to remove the timecode from the output. Default is disabled.

- DISABLED
- PIC_TIMING_SEI

WavSettings

channels

Set Channels to specify the number of channels in this output audio track. With WAV, valid values 1, 2, 4, and 8. In the console, these values are Mono, Stereo, 4-Channel, and 8-Channel, respectively.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 8

bitDepth

Specify Bit depth (BitDepth), in bits per sample, to choose the encoding quality for this audio track.

- **Type**: integer
- **Required**: False
Presets

sampleRate

Sample rate in Hz.

Type: integer
Required: False
Minimum: 8000
Maximum: 192000

Presets

URI

/presets

HTTP Methods

GET

Operation ID: ListPresets

Retrieve a JSON array of up to twenty of your presets. This will return the presets themselves, not just a list of them. To retrieve the next twenty presets, use the nextToken string returned with the array.

Query Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>listBy</td>
<td>String</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>nextToken</td>
<td>String</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>maxResults</td>
<td>String</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>order</td>
<td>String</td>
<td>False</td>
<td></td>
</tr>
</tbody>
</table>

Responses

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>ListPresetsResponse (p. 59)</td>
<td>200: OkResponse</td>
</tr>
<tr>
<td>400</td>
<td>ExceptionBody (p. 609)</td>
<td>400: BadRequestException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The conditional request failed. The service can't process your request because of a problem in the request. Please check your request form and syntax.</td>
</tr>
<tr>
<td>500</td>
<td>ExceptionBody (p. 609)</td>
<td>500: InternalServiceException</td>
</tr>
</tbody>
</table>
HTTP Methods

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>403</td>
<td>ExceptionBody (p. 609)</td>
<td>The service encountered an unexpected condition and cannot fulfill your request.</td>
</tr>
<tr>
<td>403</td>
<td>ExceptionBody (p. 609)</td>
<td>403: AccessDeniedException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>You don't have permissions for this action with the credentials you sent. Please check your authorization credentials. You should be sending credentials using the AWS Signature Version 4 signing process.</td>
</tr>
<tr>
<td>404</td>
<td>ExceptionBody (p. 609)</td>
<td>404: ResourceNotFoundException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The resource you requested does not exist.</td>
</tr>
<tr>
<td>429</td>
<td>ExceptionBody (p. 609)</td>
<td>429: LimitExceededException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Too many requests have been sent in too short of a time. The service limits the rate at which it will accept requests. For example, you may be hitting your account limits for preset creation or job submission.</td>
</tr>
<tr>
<td>409</td>
<td>ExceptionBody (p. 609)</td>
<td>409: ResourceInUseException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The service could not complete your request because there is a conflict with the current state of the resource. For example, you may be trying to delete a Queue that has jobs processing.</td>
</tr>
</tbody>
</table>

POST

Operation ID: CreatePreset

Create a new preset. For information about job templates see the User Guide at http://docs.aws.amazon.com/mediaconvert/latest/ug/what-is.html

Responses

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>201</td>
<td>CreatePresetResponse (p. 602)</td>
<td>201: CreatedResponse</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Your resource has been successfully created.</td>
</tr>
<tr>
<td>Status Code</td>
<td>Response Model</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>----------------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| 400         | ExceptionBody (p. 609) | 400: BadRequestException  
The conditional request failed.  
The service can't process your request because of a problem in the request. Please check your request form and syntax. |
| 500         | ExceptionBody (p. 609) | 500: InternalServiceException  
The service encountered an unexpected condition and cannot fulfill your request. |
| 403         | ExceptionBody (p. 609) | 403: AccessDeniedException  
You don't have permissions for this action with the credentials you sent. Please check your authorization credentials. You should be sending credentials using the [AWS Signature Version 4](https://docs.aws.amazon.com/general/latest/gr/sigv4-signed-header-howto.html) signing process. |
| 404         | ExceptionBody (p. 609) | 404: ResourceNotFoundException  
The resource you requested does not exist. |
| 429         | ExceptionBody (p. 609) | 429: LimitExceededException  
Too many requests have been sent in too short of a time.  
The service limits the rate at which it will accept requests.  
For example, you may be hitting your account limits for preset creation or job submission. |
| 409         | ExceptionBody (p. 609) | 409: ResourceInUseException  
The service could not complete your request because there is a conflict with the current state of the resource. For example, you may be trying to delete a Queue that has jobs processing. |
Schemas

Request Bodies

Example GET

```json
{
  "nextToken (p. 664)": "string",
  "maxResults (p. 664)": integer,
  "category (p. 665)": "string",
  "listBy (p. 665)": enum,
  "order (p. 665)": enum
}
```

Example POST

```json
{
  "settings (p. 626)": {
    "videoDescription (p. 687)": {
      "fixedAfd (p. 695)": integer,
      "scalingBehavior (p. 695)": enum,
      "respondToAfd (p. 695)": enum,
      "codecSettings (p. 695)": {
        "h265Settings (p. 694)": {
          "slices (p. 650)": integer,
          "minIInterval (p. 651)": integer,
          "parNumerator (p. 651)": integer,
          "flickerAdaptiveQuantization (p. 651)": enum,
          "gopSizeUnits (p. 651)": enum,
          "hrdBufferSize (p. 651)": integer,
          "qualityTuningLevel (p. 651)": enum,
          "maxBitrate (p. 651)": integer,
          "bitrate (p. 652)": integer,
          "spatialAdaptiveQuantization (p. 652)": enum,
          "sampleAdaptiveOffsetFilterMode (p. 652)": enum,
          "temporalIds (p. 652)": enum,
          "slowPal (p. 652)": enum,
          "tiles (p. 652)": enum,
          "codecProfile (p. 652)": enum,
          "alternateTransferFunctionSei (p. 652)": enum,
          "unregisteredSeiTimecode (p. 652)": enum,
          "framerateControl (p. 653)": enum,
          "telecine (p. 653)": enum,
          "framerateConversionAlgorithm (p. 653)": enum,
          "codecLevel (p. 653)": enum,
          "numberReferenceFrames (p. 653)": integer,
          "temporalAdaptiveQuantization (p. 653)": enum,
          "hrdBufferInitialFillPercentage (p. 653)": integer,
          "framerateDenominator (p. 653)": enum,
          "numberBFramesBetweenReferenceFrames (p. 654)": integer,
          "gopClosedCadence (p. 654)": integer,
          "framerateDenominator (p. 654)": integer,
          "adaptiveQuantization (p. 654)": enum,
          "interlaceMode (p. 654)": enum,
          "gopSize (p. 654)": number,
          "gopBReference (p. 654)": enum,
          "sceneChangeDetect (p. 655)": enum,
          "parDenominator (p. 655)": integer,
          "parControl (p. 655)": enum,
          "rateControlMode (p. 655)": enum
        }
      }
    }
  }
}
```
"codec (p. 694)": enum,
"proresSettings (p. 694)": {
  "slowPal (p. 689)": enum,
  "telecine (p. 689)": enum,
  "frameRateControl (p. 689)": enum,
  "frameRateDenominator (p. 689)": integer,
  "frameRateConversionAlgorithm (p. 689)": enum,
  "interlaceMode (p. 689)": enum,
  "codecProfile (p. 689)": enum,
  "parNumerator (p. 689)": integer,
  "parControl (p. 690)": enum,
  "parDenominator (p. 690)": integer,
  "frameRateNumerator (p. 690)": integer
},
"mpeg2Settings (p. 694)": {
  "minIInterval (p. 679)": integer,
  "parNumerator (p. 679)": integer,
  "gopSizeUnits (p. 680)": enum,
  "hrdBufferSize (p. 680)": integer,
  "qualityTuningLevel (p. 680)": enum,
  "maxBitrate (p. 680)": integer,
  "bitrate (p. 680)": integer,
  "spatialAdaptiveQuantization (p. 680)": enum,
  "slowPal (p. 680)": enum,
  "codecProfile (p. 681)": enum,
  "intraDCPrecision (p. 681)": enum,
  "softness (p. 681)": integer,
  "frameRateControl (p. 681)": enum,
  "telecine (p. 681)": enum,
  "frameRateConversionAlgorithm (p. 681)": enum,
  "codecLevel (p. 681)": enum,
  "temporalAdaptiveQuantization (p. 681)": enum,
  "hrdBufferInitialFillPercentage (p. 681)": integer,
  "frameRateNumerator (p. 681)": integer,
  "numberBFramesBetweenReferenceFrames (p. 682)": integer,
  "gopClosedCadence (p. 682)": integer,
  "frameRateDenominator (p. 682)": integer,
  "adaptiveQuantization (p. 682)": integer,
  "interlaceMode (p. 682)": enum,
  "gopSize (p. 682)": number,
  "sceneChangeDetect (p. 683)": enum,
  "parDenominator (p. 683)": integer,
  "parControl (p. 683)": enum,
  "syntax (p. 683)": enum,
  "rateControlMode (p. 683)": enum
},
"h264Settings (p. 695)": {
  "slices (p. 642)": integer,
  "minIInterval (p. 642)": integer,
  "parNumerator (p. 642)": integer,
  "flickerAdaptiveQuantization (p. 642)": enum,
  "gopSizeUnits (p. 642)": enum,
  "hrdBufferSize (p. 643)": integer,
  "qualityTuningLevel (p. 643)": enum,
  "maxBitrate (p. 643)": integer,
  "bitrate (p. 643)": integer,
  "spatialAdaptiveQuantization (p. 643)": enum,
  "slowPal (p. 643)": enum,
  "codecProfile (p. 643)": enum,
  "unregisteredSelTimecode (p. 644)": enum,
  "softness (p. 644)": integer,
  "frameRateControl (p. 644)": enum,
  "telecine (p. 644)": enum,
  "frameRateConversionAlgorithm (p. 644)": enum,
  "codecLevel (p. 644)": enum,
  "numberReferenceFrames (p. 644)": integer,
"temporalAdaptiveQuantization (p. 644)": enum,
"repeatPps (p. 644)": enum,
"hrdBufferInitialFillPercentage (p. 645)": integer,
"framerateNumerator (p. 645)": integer,
"numberBFramesBetweenReferenceFrames (p. 645)": integer,
"gopClosedCadence (p. 645)": integer,
"fieldEncoding (p. 645)": enum,
"entropyEncoding (p. 645)": enum,
"framerateDenominator (p. 645)": integer,
"adaptiveQuantization (p. 646)": enum,
"interlaceMode (p. 646)": enum,
"gopSize (p. 646)": number,
"gopBReference (p. 646)": enum,
"sceneChangeDetect (p. 646)": enum,
"parDenominator (p. 646)": integer,
"parControl (p. 646)": enum,
"syntax (p. 646)": enum,
"rateControlMode (p. 647)": enum
},
"frameCaptureSettings (p. 695)": {
  "framerateDenominator (p. 638)": integer,
  "maxCaptures (p. 638)": integer,
  "framerateNumerator (p. 639)": integer,
  "quality (p. 639)": integer
},
"afdSignaling (p. 695)": enum,
"colorMetadata (p. 695)": enum,
"timecodeInsertion (p. 695)": enum,
"width (p. 696)": integer,
"sharpness (p. 696)": integer,
"antiAlias (p. 696)": enum,
"videoPreprocessors (p. 696)": {
  "timecodeBurnin (p. 697)": {
    "prefix (p. 693)": "string",
    "fontSize (p. 693)": integer,
    "position (p. 693)": enum
  },
  "noiseReducer (p. 697)": {
    "filter (p. 684)": enum,
    "filterSettings (p. 684)": {
      "strength (p. 685)": integer
    },
    "spatialFilterSettings (p. 684)": {
      "strength (p. 685)": integer,
      "postFilterSharpenStrength (p. 685)": integer,
      "speed (p. 685)": integer
    }
  },
  "colorCorrector (p. 697)": {
    "saturation (p. 624)": integer,
    "brightness (p. 624)": integer,
    "hdr10Metadata (p. 624)": {
      "redPrimaryY (p. 656)": integer,
      "greenPrimaryY (p. 656)": integer,
      "whitePointX (p. 656)": integer,
      "maxLuminance (p. 657)": integer,
      "greenPrimaryX (p. 657)": integer,
      "whitePointY (p. 657)": integer,
      "redPrimaryX (p. 657)": integer,
      "bluePrimaryX (p. 657)": integer,
      "maxFrameAverageLightLevel (p. 658)": integer,
      "bluePrimaryY (p. 658)": integer,
      "maxContentLightLevel (p. 658)": integer,
      "minLuminance (p. 658)": integer
    }
  }
},
"afdSignaling (p. 695)": enum,
"contrast (p. 624)": integer,
"hue (p. 624)": integer,
"colorSpaceConversion (p. 624)": enum
},
"imageInserter (p. 697)": {
  "insertableImages (p. 658)": [
    {"duration (p. 658)": integer,
     "fadeOut (p. 659)": integer,
     "imageY (p. 659)": integer,
     "fadeIn (p. 659)": integer,
     "imageX (p. 659)": integer,
     "width (p. 659)": integer,
     "startTime (p. 660)": "string",
     "opacity (p. 660)": integer,
     "layer (p. 660)": integer,
     "height (p. 660)": integer,
     "imageInserterInput (p. 660)": "string"
    }
  ]
},
"deinterlacer (p. 697)": {
  "mode (p. 627)": enum,
  "control (p. 627)": enum,
  "algorithm (p. 627)": enum
},
"position (p. 696)": {
  "width (p. 690)": integer,
  "x (p. 691)": integer,
  "y (p. 691)": integer,
  "height (p. 691)": integer
},
"dropFrameTimecode (p. 696)": enum,
"crop (p. 696)": {
  "width (p. 690)": integer,
  "x (p. 691)": integer,
  "y (p. 691)": integer,
  "height (p. 691)": integer
},
"height (p. 697)": integer
],
"audioDescriptions (p. 687)": [
  {
    "languageCodeControl (p. 615)": enum,
    "audioTypeControl (p. 615)": enum,
    "remixSettings (p. 616)": {
      "channelsOut (p. 691)": integer,
      "channelMapping (p. 691)": {
        "outputChannels (p. 623)": [
          "inputChannels (p. 685)": [integer]
        ]
      },
      "channelsIn (p. 691)": integer
    },
    "audioType (p. 616)": integer,
    "audioSourceName (p. 616)": "string",
    "codecSettings (p. 616)": {
      "codec (p. 615)": enum,
      "wavSettings (p. 615)": {
        "channels (p. 698)": integer,
        "bitDepth (p. 698)": integer,
"sampleRate (p. 698)": integer
},
"ac3Settings (p. 615)": {
  "dynamicRangeCompressionProfile (p. 612)": enum,
  "dialnorm (p. 613)": integer,
  "codingMode (p. 613)": enum,
  "metadataControl (p. 613)": enum,
  "lfeFilter (p. 613)": enum,
  "bitrate (p. 613)": integer,
  "bitstreamMode (p. 613)": enum,
  "sampleRate (p. 613)": integer
},
"aacSettings (p. 615)": {
  "vbrQuality (p. 610)": enum,
  "codecProfile (p. 610)": enum,
  "codingMode (p. 610)": enum,
  "specification (p. 611)": enum,
  "bitrate (p. 611)": integer,
  "rawFormat (p. 611)": enum,
  "rateControlMode (p. 611)": enum,
  "sampleRate (p. 611)": integer,
  "audioDescriptionBroadcasterMix (p. 611)": enum
},
"aiffSettings (p. 615)": {
  "channels (p. 614)": integer,
  "bitDepth (p. 614)": integer,
  "sampleRate (p. 614)": integer
},
"aac3Settings (p. 615)": {
  "dialnorm (p. 635)": integer,
  "passthroughControl (p. 635)": enum,
  "metadataControl (p. 635)": enum,
  "bitrate (p. 635)": integer,
  "dynamicRangeCompressionRf (p. 635)": enum,
  "sampleRate (p. 635)": integer,
  "ltRtSurroundMixLevel (p. 635)": number,
  "surroundExMode (p. 636)": enum,
  "dynamicRangeCompressionLine (p. 636)": enum,
  "lfeControl (p. 636)": enum,
  "codingMode (p. 636)": enum,
  "surroundMode (p. 636)": enum,
  "attenuationControl (p. 636)": enum,
  "lfeFilter (p. 636)": enum,
  "phaseControl (p. 636)": enum,
  "ltRtCenterMixLevel (p. 636)": number,
  "dCFilter (p. 637)": enum,
  "stereoDownmix (p. 637)": enum,
  "bitstreamMode (p. 637)": enum,
  "loRoSurroundMixLevel (p. 637)": number,
  "loRoCenterMixLevel (p. 637)": number
},
"mp2Settings (p. 615)": {
  "channels (p. 676)": integer,
  "bitrate (p. 676)": integer,
  "sampleRate (p. 676)": integer
},
"languageCode (p. 616)": enum,
"streamName (p. 616)": "string",
"audioNormalizationSettings (p. 617)": {
  "targetLkfs (p. 617)": number,
  "algorithmControl (p. 618)": enum,
  "peakCalculation (p. 618)": enum,
  "loudnessLogging (p. 618)": enum,
  "correctionGateLevel (p. 618)": integer,
  "algorithm (p. 618)": enum
"containerSettings (p. 687)": {
  "container (p. 625)": enum,
  "mp4Settings (p. 625)": {
    "mp4MajorBrand (p. 677)": "string",
    "moovPlacement (p. 677)": enum,
    "cs1gAtom (p. 677)": enum,
    "freeSpaceBox (p. 677)": enum
  },
  "m3u8Settings (p. 625)": {
    "pmtPid (p. 672)": integer,
    "pcrPid (p. 672)": integer,
    "audioPids (p. 672)": [
      integer
    ],
    "audioFramesPerPes (p. 673)": integer,
    "scte35Pid (p. 673)": integer,
    "transportStreamId (p. 673)": integer,
    "videoPid (p. 673)": integer,
    "pcrControl (p. 673)": enum,
    "privateMetadataPid (p. 673)": integer,
    "pmtInterval (p. 674)": integer,
    "patInterval (p. 674)": integer,
    "programNumber (p. 674)": integer,
    "timedMetadataPid (p. 674)": integer,
    "timedMetadata (p. 674)": enum,
    "scte35Source (p. 674)": enum
  },
  "m2tsSettings (p. 625)": {
    "dvbTeletextPid (p. 667)": integer,
    "bitrate (p. 667)": integer,
    "segmentationTime (p. 667)": number,
    "audioPids (p. 668)": [
      integer
    ],
    "rateMode (p. 668)": enum,
    "ebpAudioInterval (p. 668)": enum,
    "fragmentTime (p. 668)": number,
    "audioFramesPerPes (p. 668)": integer,
    "maxPcrInterval (p. 668)": integer,
    "scte35Pid (p. 668)": integer,
    "privateMetadataPid (p. 669)": integer,
    "pmtInterval (p. 669)": integer,
    "segmentationStyle (p. 669)": enum,
    "audioBufferModel (p. 669)": enum,
    "ebpPlacement (p. 669)": enum,
    "dvbSdtSettings (p. 669)": {
      "sdtInterval (p. 628)": integer,
      "serviceName (p. 628)": "string",
      "serviceProviderName (p. 628)": "string",
      "outputSdt (p. 629)": enum
    },
    "scte35Source (p. 669)": enum,
    "pmtPid (p. 670)": integer,
    "bufferModel (p. 670)": enum,
    "ebpPlacement (p. 670)": enum,
    "dvbSdtSettings (p. 670)": {
      "adtInterval (p. 628)": integer,
      "serviceName (p. 628)": "string",
      "serviceProviderName (p. 628)": "string",
      "outputSdt (p. 629)": enum
    },
    "nullPacketBitrate (p. 670)": number,
    "pcrPid (p. 670)": integer,
    "minEbpInterval (p. 670)": integer,
"transportStreamId (p. 671)" : integer,
"pcrControl (p. 671)" : enum,
"videoPid (p. 671)" : integer,
"esRateInPes (p. 671)" : enum,
"segmentationMarkers (p. 671)" : enum,
"dvbTdtSettings (p. 671)" : {
    "tdtInterval (p. 633)" : integer
},
"patInterval (p. 671)" : integer,
"dvbSubPids (p. 672)" : [
    integer
],
"movSettings (p. 625)" : {
    "reference (p. 675)" : enum,
    "paddingControl (p. 675)" : enum,
    "mpeg2FourCCControl (p. 675)" : enum,
    "csigAtom (p. 676)" : enum,
    "clapAtom (p. 676)" : enum
},
"f4vSettings (p. 625)" : {
    "moovPlacement (p. 638)" : enum
},
"captionDescriptions (p. 687)" : [
    {
        "languageDescription (p. 622)" : "string",
        "languageCode (p. 622)" : enum,
        "destinationSettings (p. 622)" : {
            "burninDestinationSettings (p. 623)" : {
                "xPosition (p. 618)" : integer,
                "backgroundColor (p. 619)" : enum,
                "teletextSpacing (p. 619)" : enum,
                "yPosition (p. 619)" : integer,
                "backgroundColorOpacity (p. 619)" : integer,
                "fontOpacity (p. 619)" : integer,
                "shadowOpacity (p. 620)" : integer,
                "fontResolution (p. 620)" : integer,
                "shadowYOffset (p. 620)" : integer,
                "outlineSize (p. 620)" : integer,
                "outlineColor (p. 620)" : enum,
                "fontSize (p. 620)" : integer,
                "shadowXOffset (p. 621)" : integer,
                "alignment (p. 621)" : enum,
                "shadowColor (p. 621)" : enum,
                "fontColor (p. 621)" : enum
            },
            "teletextDestinationSettings (p. 623)" : {
                "pageNumber (p. 692)" : "string"
            },
            "ttmlDestinationSettings (p. 623)" : {
                "stylePassthrough (p. 694)" : enum
            },
            "destinationType (p. 623)" : enum,
            "dvbSubDestinationSettings (p. 623)" : {
                "xPosition (p. 629)" : integer,
                "backgroundColor (p. 629)" : enum,
                "teletextSpacing (p. 629)" : enum,
                "yPosition (p. 629)" : integer,
                "backgroundColorOpacity (p. 629)" : integer,
                "fontOpacity (p. 630)" : integer,
                "shadowOpacity (p. 630)" : integer,
                "fontResolution (p. 630)" : integer,
                "shadowYOffset (p. 630)" : integer,
                "outlineSize (p. 630)" : integer,
                "outlineColor (p. 631)" : enum,
                "drawableRegion (p. 631)" : enum
            }}}}
Response Bodies

Example ListPresetsResponse

```json
{
  "presets (p. 665)": [
    {
      "settings (p. 686)": {
        "videoDescription (p. 687)": {
          "fixedAfd (p. 695)": integer,
          "scalingBehavior (p. 695)": enum,
          "respondToAfd (p. 695)": enum,
          "codecSettings (p. 695)": {
            "h265Settings (p. 694)": {
              "slices (p. 650)": integer,
              "minInterval (p. 651)": integer,
              "parNumerator (p. 651)": integer,
              "flickerAdaptiveQuantization (p. 651)": enum,
              "gopSizeUnits (p. 651)": enum,
              "hrdBufferSize (p. 651)": integer,
              "qualityTuningLevel (p. 651)": enum,
              "maxBitrate (p. 651)": integer,
              "bitrate (p. 652)": integer,
              "spatialAdaptiveQuantization (p. 652)": enum,
              "sampleAdaptiveOffsetFilterMode (p. 652)": enum,
              "temporalIds (p. 652)": enum,
              "slowPal (p. 652)": enum,
              "tiles (p. 652)": enum,
              "codecProfile (p. 652)": enum,
              "alternateTransferFunctionSei (p. 652)": enum,
              "unregisteredSeiTimecode (p. 652)": enum,
              "framerateControl (p. 653)": enum,
              "telecine (p. 653)": enum,
              "framerateConversionAlgorithm (p. 653)": enum,
              "codecLevel (p. 653)": enum,
              "numberReferenceFrames (p. 653)": integer,
              "temporalAdaptiveQuantization (p. 653)": enum,
              "hrdBufferInitialFillPercentage (p. 653)": integer,
              "framerateNumerator (p. 653)": integer,
              "framerateDenominator (p. 653)": integer,
              "adaptiveQuantization (p. 654)": enum,
              "interlaceMode (p. 654)": enum,
              "gopSize (p. 654)": number,
```
"gopBReference (p. 654)": enum,
"sceneChangeDetect (p. 655)": enum,
"parDenominator (p. 655)": integer,
"parControl (p. 655)": enum,
"rateControlMode (p. 655)": enum
},
"codec (p. 694)": enum,
"proresSettings (p. 694)": {
  "slowPal (p. 689)": enum,
  "framerateControl (p. 689)": enum,
  "telecine (p. 689)": enum,
  "framerateDenominator (p. 689)": integer,
  "framerateConversionAlgorithm (p. 689)": enum,
  "interlaceMode (p. 689)": enum,
  "codecProfile (p. 689)": enum,
  "parNumerator (p. 689)": integer,
  "parControl (p. 689)": enum,
  "parDenominator (p. 690)": integer,
  "framerateNumerator (p. 690)": integer
},
"mpeg2Settings (p. 694)": {
  "minIInterval (p. 679)": integer,
  "parNumerator (p. 679)": integer,
  "gopSizeUnits (p. 680)": enum,
  "hrdBufferSize (p. 680)": integer,
  "qualityTuningLevel (p. 680)": enum,
  "maxBitrate (p. 680)": integer,
  "bitrate (p. 680)": integer,
  "spatialAdaptiveQuantization (p. 680)": enum,
  "slowPal (p. 680)": enum,
  "intraDcPrecision (p. 681)": enum,
  "softness (p. 681)": integer,
  "framerateControl (p. 681)": enum,
  "telecine (p. 681)": enum,
  "framerateConversionAlgorithm (p. 681)": enum,
  "codecProfile (p. 681)": enum,
  "intraDcPrecision (p. 681)": enum,
  "softness (p. 681)": integer,
  "framerateControl (p. 681)": enum,
  "telecine (p. 682)": enum,
  "framerateConversionAlgorithm (p. 682)": enum,
  "framerateNumerator (p. 682)": integer,
  "numberBFramesBetweenReferenceFrames (p. 682)": integer,
  "gopClosedCadence (p. 682)": integer,
  "framerateDenominator (p. 682)": integer,
  "adaptiveQuantization (p. 682)": integer,
  "interlaceMode (p. 682)": enum,
  "gopSize (p. 682)": number,
  "sceneChangeDetect (p. 683)": enum,
  "parDenominator (p. 683)": integer,
  "parControl (p. 683)": enum,
  "syntax (p. 683)": enum,
  "rateControlMode (p. 683)": enum
},
"h264Settings (p. 695)": {
  "slices (p. 642)": integer,
  "minIInterval (p. 642)": integer,
  "parNumerator (p. 642)": integer,
  "flickerAdaptiveQuantization (p. 642)": enum,
  "gopSizeUnits (p. 642)": enum,
  "hrdBufferSize (p. 643)": integer,
  "qualityTuningLevel (p. 643)": enum,
  "maxBitrate (p. 643)": integer,
  "bitrate (p. 643)": integer,
  "spatialAdaptiveQuantization (p. 643)": enum,
  "slowPal (p. 643)": enum,
  "codecProfile (p. 643)": enum,
  "unregisteredSeiTimecode (p. 644)": enum,
"softness (p. 644)": integer,
"framerateControl (p. 644)": enum,
"telecine (p. 644)": enum,
"framerateConversionAlgorithm (p. 644)": enum,
"codecLevel (p. 644)": enum,
"numberReferenceFrames (p. 644)": integer,
"temporalAdaptiveQuantization (p. 644)": enum,
"repeatFps (p. 644)": enum,
"hrdBufferInitialFillPercentage (p. 645)": integer,
"framerateNumerator (p. 645)": integer,
"numberBFramesBetweenReferenceFrames (p. 645)": integer,
"gopClosedCadence (p. 645)": integer,
"fieldEncoding (p. 645)": enum,
"entropyEncoding (p. 645)": enum,
"framerateDenominator (p. 645)": integer,
"adaptiveQuantization (p. 646)": enum,
"interlaceMode (p. 646)": enum,
"gopSize (p. 646)": number,
"gopBReference (p. 646)": enum,
"sceneChangeDetect (p. 646)": enum,
"parDenominator (p. 646)": integer,
"parControl (p. 646)": enum,
"syntax (p. 646)": enum,
"rateControlMode (p. 647)": enum
},
"frameCaptureSettings (p. 695)": {
"framerateDenominator (p. 638)": integer,
"maxCaptures (p. 638)": integer,
"framerateNumerator (p. 639)": integer,
"quality (p. 639)": integer
}
"afdSignaling (p. 695)": enum,
"colorMetadata (p. 695)": enum,
"timecodeInsertion (p. 695)": enum,
"width (p. 696)": integer,
"sharpness (p. 696)": integer,
"antiAlias (p. 696)": enum,
"videoPreprocessors (p. 696)": {
"timecodeBurnin (p. 697)": {
"prefix (p. 693)": "string",
"fontSize (p. 693)": integer,
"position (p. 693)": enum
},
"noiseReducer (p. 697)": {
"filter (p. 684)": enum,
"filterSettings (p. 684)": {
"strength (p. 685)": integer
},
"spatialFilterSettings (p. 684)": {
"strength (p. 685)": integer,
"postFilterSharpenStrength (p. 685)": integer,
"speed (p. 685)": integer
}
},
"colorCorrector (p. 697)": {
"saturation (p. 624)": integer,
"brightness (p. 624)": integer,
"hdr10Metadata (p. 624)": {
"redPrimaryY (p. 656)": integer,
"greenPrimaryY (p. 656)": integer,
"whitePointX (p. 656)": integer,
"maxLuminance (p. 657)": integer,
"greenPrimaryX (p. 657)": integer,
"whitePointY (p. 657)": integer,
"redPrimaryX (p. 657)": integer,
"bluePrimaryX (p. 657)": integer,
"maxFrameAverageLightLevel (p. 658)": integer,
"bluePrimaryY (p. 658)": integer,
"maxContentLightLevel (p. 658)": integer,
"minLuminance (p. 658)": integer
},
"contrast (p. 624)": integer,
"hue (p. 624)": integer,
"colorSpaceConversion (p. 624)": enum
},
"imageInserter (p. 697)": {
"insertableImages (p. 658)": [ {
"duration (p. 658)": integer,
"fadeOut (p. 659)": integer,
"imageY (p. 659)": integer,
"fadeIn (p. 659)": integer,
"imageX (p. 659)": integer,
"width (p. 659)": integer,
"startTime (p. 660)": "string",
"opacity (p. 660)": integer,
"layer (p. 660)": integer,
"height (p. 660)": integer,
"imageInserterInput (p. 660)": "string"
} ]
},
"deinterlacer (p. 697)": {
"mode (p. 627)": enum,
"control (p. 627)": enum,
"algorithm (p. 627)": enum
}
",
"position (p. 696)": {
"width (p. 690)": integer,
"x (p. 691)": integer,
"y (p. 691)": integer,
"height (p. 691)": integer
},
"dropFrameTimecode (p. 696)": enum,
"crop (p. 696)": {
"width (p. 690)": integer,
"x (p. 691)": integer,
"y (p. 691)": integer,
"height (p. 691)": integer
},
"height (p. 697)": integer
},
"audioDescriptions (p. 687)": [ {
"languageCodeControl (p. 615)": enum,
"audioTypeControl (p. 615)": enum,
"remixSettings (p. 616)": {
"channelsOut (p. 691)": integer,
"channelMapping (p. 691)": {
"outputChannels (p. 623)": [ {
"inputChannels (p. 685)": [ integer
] ]
},
"channelsIn (p. 691)": integer
},
"audioType (p. 616)": integer,
"audioSourceName (p. 616)": "string",
"codecSettings (p. 616)": {
  "codec (p. 615)": enum,
  "wavSettings (p. 615)": {
    "channels (p. 698)": integer,
    "bitDepth (p. 698)": integer,
    "sampleRate (p. 698)": integer
  },
  "ac3Settings (p. 615)": {
    "dynamicRangeCompressionProfile (p. 612)": enum,
    "dialnorm (p. 613)": integer,
    "codingMode (p. 613)": enum,
    "metadataControl (p. 613)": enum,
    "lfeFilter (p. 613)": enum,
    "bitrate (p. 613)": integer,
    "bitstreamMode (p. 613)": enum,
    "sampleRate (p. 613)": integer
  },
  "aacSettings (p. 615)": {
    "vbrQuality (p. 610)": enum,
    "codecProfile (p. 610)": enum,
    "codingMode (p. 610)": enum,
    "specification (p. 611)": enum,
    "bitrate (p. 611)": integer,
    "rawFormat (p. 611)": enum,
    "rateControlMode (p. 611)": enum,
    "sampleRate (p. 611)": integer,
    "audioDescriptionBroadcasterMix (p. 611)": enum
  },
  "wiffSettings (p. 615)": {
    "channels (p. 614)": integer,
    "bitDepth (p. 614)": integer,
    "sampleRate (p. 614)": integer
  },
  "eac3Settings (p. 615)": {
    "dialnorm (p. 635)": integer,
    "passthroughControl (p. 635)": enum,
    "metadataControl (p. 635)": enum,
    "dynamicRangeCompressionRf (p. 635)": enum,
    "sampleRate (p. 635)": integer,
    "leRtsSurroundMixLevel (p. 635)": number,
    "surroundExMode (p. 636)": enum,
    "dynamicRangeCompressionLine (p. 636)": enum,
    "lfeControl (p. 636)": enum,
    "codingMode (p. 636)": enum,
    "surroundMode (p. 636)": enum,
    "attenuationControl (p. 636)": enum,
    "lfeFilter (p. 636)": enum,
    "phaseControl (p. 636)": enum,
    "leRtCenterMixLevel (p. 636)": number,
    "dcFilter (p. 637)": enum,
    "stereoDownmix (p. 637)": enum,
    "bitstreamMode (p. 637)": enum,
    "loRoSurroundMixLevel (p. 637)": number,
    "loRoCenterMixLevel (p. 637)": number
  },
  "mp2Settings (p. 615)": {
    "channels (p. 676)": integer,
    "bitrate (p. 676)": integer,
    "sampleRate (p. 676)": integer
  }
},
"languageCode (p. 616)": enum,
"streamName (p. 616)": "string",
"audioNormalizationSettings (p. 617)": {
}
"targetLkfs (p. 617)": number,
"algorithmControl (p. 618)": enum,
"peakCalculation (p. 618)": enum,
"loudnessLogging (p. 618)": enum,
"correctionGateLevel (p. 618)": integer,
"algorithm (p. 618)": enum
},
"containerSettings (p. 687)": {
  "container (p. 625)": enum,
  "mp4Settings (p. 625)": {
    "mp4MajorBrand (p. 677)": "string",
    "moovPlacement (p. 677)": enum,
    "cslgAtom (p. 677)": enum,
    "freeSpaceBox (p. 677)": enum
  },
  "m3u8Settings (p. 625)": {
    "pmtPid (p. 672)": integer,
    "pcrPid (p. 672)": integer,
    "audioPids (p. 672)": [ integer ]
  },
  "audioFramesPerPes (p. 673)": integer,
  "scte35Pid (p. 673)": integer,
  "transportStreamId (p. 673)": integer,
  "videoPid (p. 673)": integer,
  "pcrControl (p. 673)": enum,
  "privateMetadataPid (p. 673)": integer,
  "pmtInterval (p. 674)": integer,
  "patInterval (p. 674)": integer,
  "programNumber (p. 674)": integer,
  "timedMetadataPid (p. 674)": integer,
  "timedMetadata (p. 674)": enum,
  "scte35Source (p. 674)": enum
},
"m2tsSettings (p. 625)": {
  "dvbTeletextPid (p. 667)": integer,
  "bitrate (p. 667)": integer,
  "segmentationTime (p. 667)": number,
  "audioPids (p. 668)": [ integer ]
},
"rateMode (p. 668)": enum,
"ebpAudioInterval (p. 668)": enum,
"fragmentTime (p. 668)": number,
"audioFramesPerPes (p. 668)": integer,
"maxPcrInterval (p. 668)": integer,
"scte35Pid (p. 668)": integer,
"privateMetadataPid (p. 669)": integer,
"pmtInterval (p. 669)": integer,
"segmentationStyle (p. 669)": enum,
"audioBufferModel (p. 669)": enum,
"programNumber (p. 669)": integer,
"dvbNitSettings (p. 669)": {
  "networkName (p. 628)": "string",
  "networkId (p. 628)": integer,
  "nitInterval (p. 628)": integer
},
"scte35Source (p. 669)": enum,
"pmtPid (p. 670)": integer,
"bufferModel (p. 670)": enum,
"ebpPlacement (p. 670)": enum,
"dvbSdtSettings (p. 670)": {
  "sdtInterval (p. 628)": integer,
  "serviceName (p. 628)": "string",}
"serviceProviderName (p. 628)": "string",
"outputSdt (p. 629)": enum
},
"nullPacketBitrate (p. 670)": number,
"pcrPid (p. 670)": integer,
"minEbpInterval (p. 670)": integer,
"transportStreamId (p. 671)": integer,
"pcrControl (p. 671)": enum,
"videoPid (p. 671)": integer,
"esRateInPes (p. 671)": enum,
"segmentationMarkers (p. 671)": enum,
"dvbTdtSettings (p. 671)": {
  "tdtInterval (p. 633)": integer
},
"patInterval (p. 671)": integer,
"dvbSubPids (p. 672)": [
  integer
],
"movSettings (p. 625)": {
  "reference (p. 675)": enum,
  "paddingControl (p. 675)": enum,
  "mpeg2FourCCControl (p. 675)": enum,
  "cslgAtom (p. 676)": enum,
  "clapAtom (p. 676)": enum
},
"f4vSettings (p. 625)": {
  "moovPlacement (p. 638)": enum
},
"captionDescriptions (p. 687)": [
  {
    "languageDescription (p. 622)": "string",
    "languageCode (p. 622)": enum,
    "destinationSettings (p. 622)": enum,
    "burninDestinationSettings (p. 623)": {
      "xPosition (p. 618)": integer,
      "backgroundColor (p. 619)": enum,
      "teletextSpacing (p. 619)": enum,
      "yPosition (p. 619)": integer,
      "backgroundOpacity (p. 619)": integer,
      "fontOpacity (p. 619)": integer,
      "shadowOpacity (p. 620)": integer,
      "fontResolution (p. 620)": integer,
      "shadowYOffset (p. 620)": integer,
      "outlineSize (p. 620)": integer,
      "shadowColor (p. 620)": enum,
      "alignment (p. 621)": enum,
      "shadowColor (p. 621)": enum,
      "fontColor (p. 621)": enum
    },
    "teletextDestinationSettings (p. 623)": {
      "pageNumber (p. 692)": "string"
    },
    "ttmlDestinationSettings (p. 623)": {
      "stylePassthrough (p. 694)": enum
    },
    "destinationType (p. 623)": enum,
    "dvbSubDestinationSettings (p. 623)": {
      "xPosition (p. 629)": integer,
      "backgroundColor (p. 629)": enum,
      "teletextSpacing (p. 629)": enum,
      "yPosition (p. 629)": integer,
      "backgroundOpacity (p. 629)": integer,
Example CreatePresetResponse

```json
{
  "preset (p. 626)": {
    "settings (p. 686)": {
      "videoDescription (p. 687)": {
        "fixedAfd (p. 695)": integer,
        "scalingBehavior (p. 695)": enum,
        "respondToAfd (p. 695)": enum,
        "codecSettings (p. 695)": {
          "h265Settings (p. 694)": {
            "slices (p. 650)": integer,
            "minIInterval (p. 651)": integer,
            "parNumerator (p. 651)": integer,
            "flickerAdaptiveQuantization (p. 651)": enum,
            "gopSizeUnits (p. 651)": enum,
            "hrdBufferSize (p. 651)": integer,
            "qualityTuningLevel (p. 651)": enum,
            "maxBitrate (p. 651)": integer,
            "bitrate (p. 652)": integer,
            "spatialAdaptiveQuantization (p. 652)": enum,
            "sampleAdaptiveOffsetFilterMode (p. 652)": enum,
            "temporalIds (p. 652)": enum,
            "slowPal (p. 652)": enum,
            "tiles (p. 652)": enum,
            "codecProfile (p. 652)": enum,
            "alternateTransferFunctionSei (p. 652)": enum,
            "unregisteredSeiTimecode (p. 652)": enum,
            "framerateControl (p. 653)": enum,
            "telecine (p. 653)": enum,
            "framerateConversionAlgorithm (p. 653)": enum,
            "codecLevel (p. 653)": enum,
            "numberReferenceFrames (p. 653)": integer,
          }
        }
      }
    }
  }
}
```
"temporalAdaptiveQuantization (p. 653)" : enum,
"hrdBufferInitialFillPercentage (p. 653)" : integer,
"frameRateNumerator (p. 653)" : integer,
"numberBFramesBetweenReferenceFrames (p. 654)" : integer,
"gopClosedCadence (p. 654)" : integer,
"frameRateDenominator (p. 654)" : integer,
"adaptiveQuantization (p. 654)" : enum,
"interlaceMode (p. 654)" : enum,
"gopSize (p. 654)" : number,
"gopBReference (p. 654)" : enum,
"sceneChangeDetect (p. 655)" : enum,
"parDenominator (p. 655)" : integer,
"parControl (p. 655)" : enum,
"rateControlMode (p. 655)" : enum
},
"codec (p. 694)" : enum,
"proresSettings (p. 694)" : {
  "slowPal (p. 689)" : enum,
  "frameRateControl (p. 689)" : enum,
  "telecine (p. 689)" : enum,
  "frameRateDenominator (p. 689)" : integer,
  "frameRateConversionAlgorithm (p. 689)" : enum,
  "interlaceMode (p. 689)" : enum,
  "codecProfile (p. 689)" : enum,
  "parNumerator (p. 689)" : integer,
  "parControl (p. 690)" : enum,
  "parDenominator (p. 690)" : integer,
  "frameRateNumerator (p. 690)" : integer
},
"mpeg2Settings (p. 694)" : {
  "minIInterval (p. 679)" : integer,
  "parNumerator (p. 679)" : integer,
  "gopSizeUnits (p. 680)" : enum,
  "hrdBufferSize (p. 680)" : integer,
  "qualityTuningLevel (p. 680)" : enum,
  "maxBitrate (p. 680)" : integer,
  "bitrate (p. 680)" : integer,
  "spatialAdaptiveQuantization (p. 680)" : enum,
  "slowPal (p. 680)" : enum,
  "codecProfile (p. 681)" : enum,
  "intraDcPrecision (p. 681)" : enum,
  "softness (p. 681)" : integer,
  "frameRateControl (p. 681)" : enum,
  "telecine (p. 681)" : enum,
  "frameRateConversionAlgorithm (p. 681)" : enum,
  "codecLevel (p. 681)" : enum,
  "temporalAdaptiveQuantization (p. 681)" : enum,
  "hrdBufferInitialFillPercentage (p. 681)" : integer,
  "frameRateNumerator (p. 681)" : integer,
  "numberBFramesBetweenReferenceFrames (p. 682)" : integer,
  "gopClosedCadence (p. 682)" : integer,
  "frameRateDenominator (p. 682)" : integer,
  "adaptiveQuantization (p. 682)" : enum,
  "interlaceMode (p. 682)" : enum,
  "gopSize (p. 682)" : number,
  "sceneChangeDetect (p. 683)" : enum,
  "parDenominator (p. 683)" : integer,
  "parControl (p. 683)" : enum,
  "rateControlMode (p. 683)" : enum
},
"h264Settings (p. 695)" : {
  "slices (p. 642)" : integer,
  "minIInterval (p. 642)" : integer,
  "parNumerator (p. 642)" : integer,
  "flickerAdaptiveQuantization (p. 642)" : enum,
"gopSizeUnits (p. 642)": enum,
"hrdBufferSize (p. 643)": integer,
"qualityTuningLevel (p. 643)": enum,
"maxBitrate (p. 643)": integer,
"bitrate (p. 643)": integer,
"spatialAdaptiveQuantization (p. 643)": enum,
"slowPal (p. 643)": enum,
"codecProfile (p. 643)": enum,
"unregisteredSelTimecode (p. 644)": enum,
"softness (p. 644)": integer,
"framerateControl (p. 644)": enum,
"telecine (p. 644)": enum,
"framerateConversionAlgorithm (p. 644)": enum,
"codecLevel (p. 644)": enum,
"numberReferenceFrames (p. 644)": integer,
"temporalAdaptiveQuantization (p. 644)": enum,
"repeatPps (p. 644)": enum,
"hrdBufferInitialFillPercentage (p. 645)": integer,
"framerateNumerator (p. 645)": integer,
"hrdBufferInitialFillPercentage (p. 645)": integer,
"framerateDenominator (p. 645)": integer,
"numberBFramesBetweenReferenceFrames (p. 645)": integer,
"gopClosedCadence (p. 645)": integer,
"fieldEncoding (p. 645)": enum,
"entropyEncoding (p. 645)": enum,
"frameCaptureSettings (p. 695)": {
  "framerateDenominator (p. 638)": integer,
  "maxCaptures (p. 638)": integer,
  "framerateNumerator (p. 639)": integer,
  "quality (p. 639)": integer
}.
"afdSignaling (p. 695)": enum,
"colorMetadata (p. 695)": enum,
"timecodeInsertion (p. 695)": enum,
"width (p. 696)": integer,
"sharpness (p. 696)": integer,
"antiAlias (p. 696)": enum,
"videoPreprocessors (p. 696)": {
  "timecodeBurnin (p. 697)": {
    "prefix (p. 693)": "string",
    "fontSize (p. 693)": integer,
    "position (p. 693)": enum
  },
  "noiseReducer (p. 697)": {
    "filter (p. 684)": enum,
    "filterSettings (p. 684)": {
      "strength (p. 685)": integer
    },
    "spatialFilterSettings (p. 684)": {
      "strength (p. 685)": integer,
      "postFilterSharpenStrength (p. 685)": integer,
      "speed (p. 685)": integer
    }
  },
  "colorCorrector (p. 697)": {
    "saturation (p. 624)": integer,
"brightness (p. 624)" : integer,
"hdr10Metadata (p. 624)" : {
  "redPrimaryY (p. 656)" : integer,
  "greenPrimaryY (p. 656)" : integer,
  "whitePointY (p. 656)" : integer,
  "maxLuminance (p. 657)" : integer,
  "greenPrimaryX (p. 657)" : integer,
  "whitePointX (p. 657)" : integer,
  "redPrimaryX (p. 657)" : integer,
  "bluePrimaryX (p. 657)" : integer,
  "maxFrameAverageLightLevel (p. 658)" : integer,
  "bluePrimaryY (p. 658)" : integer,
  "maxContentLightLevel (p. 658)" : integer,
  "minLuminance (p. 658)" : integer
},
"contrast (p. 624)" : integer,
"hue (p. 624)" : integer,
"colorSpaceConversion (p. 624)" : enum
},
"imageInserter (p. 697)" : {
  "insertableImages (p. 658)" : [ 
    { 
      "duration (p. 658)" : integer,
      "fadeOut (p. 659)" : integer,
      "imageY (p. 659)" : integer,
      "fadeIn (p. 659)" : integer,
      "imageX (p. 659)" : integer,
      "width (p. 659)" : integer,
      "startTime (p. 660)" : "string",
      "opacity (p. 660)" : integer,
      "layer (p. 660)" : integer,
      "height (p. 660)" : integer,
      "imageInserterInput (p. 660)" : "string"
    }
  ],
  "deinterlacer (p. 697)" : { 
    "mode (p. 627)" : enum,
    "control (p. 627)" : enum,
    "algorithm (p. 627)" : enum
  },
  "position (p. 696)" : { 
    "width (p. 690)" : integer,
    "x (p. 691)" : integer,
    "y (p. 691)" : integer,
    "height (p. 691)" : integer
  },
  "dropFrameTimecode (p. 696)" : enum,
  "crop (p. 696)" : { 
    "width (p. 690)" : integer,
    "x (p. 691)" : integer,
    "y (p. 691)" : integer,
    "height (p. 691)" : integer
  },
  "height (p. 697)" : integer
},
"audioDescriptions (p. 687)" : [ 
  { 
    "languageCodeControl (p. 615)" : enum,
    "audioTypeControl (p. 615)" : enum,
    "remixSettings (p. 616)" : { 
      "channelsOut (p. 691)" : integer,
      "channelMapping (p. 691)" : { 
        "outputChannels (p. 623)" : [ 
          
```
"inputChannels (p. 685)" : [
  integer
]
}

"channelsIn (p. 691)" : integer

"audioType (p. 616)" : integer,
"audioSourceName (p. 616)" : "string",
"codecSettings (p. 616)" : {
  "codec (p. 615)" : enum,
  "wavSettings (p. 615)" : {
    "channels (p. 698)" : integer,
    "bitDepth (p. 698)" : integer,
    "sampleRate (p. 698)" : integer
  },
  "ac3Settings (p. 615)" : {
    "dynamicRangeCompressionProfile (p. 612)" : enum,
    "dialnorm (p. 613)" : integer,
    "codingMode (p. 613)" : enum,
    "metadataControl (p. 613)" : enum,
    "lfeFilter (p. 613)" : integer,
    "bitrate (p. 613)" : integer,
    "bitstreamMode (p. 613)" : enum,
    "sampleRate (p. 613)" : integer
  },
  "aiffSettings (p. 615)" : {
    "channels (p. 614)" : integer,
    "bitDepth (p. 614)" : integer,
    "sampleRate (p. 614)" : integer
  },
  "eac3Settings (p. 615)" : {
    "dialnorm (p. 635)" : integer,
    "passsthroughControl (p. 635)" : enum,
    "metadataControl (p. 635)" : enum,
    "bitrate (p. 635)" : integer,
    "dynamicRangeCompressionRf (p. 635)" : enum,
    "sampleRate (p. 635)" : integer,
    "ltRtSurroundMixLevel (p. 635)" : number,
    "surroundExMode (p. 636)" : enum,
    "dynamicRangeCompressionLine (p. 636)" : enum,
    "lfeControl (p. 636)" : enum,
    "codingMode (p. 636)" : enum,
    "surroundMode (p. 636)" : enum,
    "attenuationControl (p. 636)" : enum,
    "lfeFilter (p. 636)" : enum,
    "phaseControl (p. 636)" : enum,
    "ltRtCenterMixLevel (p. 636)" : number,
    "dcFilter (p. 637)" : enum,
    "stereoDownmix (p. 637)" : enum,
    "bitstreamMode (p. 637)" : enum,
    "loRoSurroundMixLevel (p. 637)" : number,
    "loRoCenterMixLevel (p. 637)" : number
  }
},

"audioDescriptionBroadcasterMix (p. 611)" : enum

"siffSettings (p. 615)" : {
  "channels (p. 614)" : integer,
  "bitDepth (p. 614)" : integer,
  "sampleRate (p. 614)" : integer
}

"avxSettings (p. 615)" : {
  "vbrQuality (p. 610)" : enum,
  "codecProfile (p. 610)" : enum,
  "codingMode (p. 610)" : enum,
  "specification (p. 611)" : enum,
  "bitrate (p. 611)" : integer,
  "rawFormat (p. 611)" : enum,
  "rateControlMode (p. 611)" : enum,
  "sampleRate (p. 611)" : integer,
  "audioDescriptionBroadcasterMix (p. 611)" : enum
}
"mp2Settings (p. 615)" : {
  "channels (p. 676)" : integer,
  "bitrate (p. 676)" : integer,
  "sampleRate (p. 676)" : integer
},
"languageCode (p. 616)" : enum,
"streamName (p. 616)" : "string",
"audioNormalizationSettings (p. 617)" : {
  "targetLkfs (p. 617)" : number,
  "algorithmControl (p. 618)" : enum,
  "peakCalculation (p. 618)" : enum,
  "loudnessLogging (p. 618)" : enum,
  "correctionGateLevel (p. 618)" : integer,
  "algorithm (p. 618)" : enum
}
},
"containerSettings (p. 687)" : {
  "container (p. 625)" : enum,
  "mp4Settings (p. 625)" : {
    "mp4MajorBrand (p. 677)" : "string",
    "moovPlacement (p. 677)" : enum,
    "cslgAtom (p. 677)" : enum,
    "freeSpaceBox (p. 677)" : enum
  },
  "m3u8Settings (p. 625)" : {
    "pmtPid (p. 672)" : integer,
    "pcrPid (p. 672)" : integer,
    "audioPids (p. 672)" : [ integer
    ],
    "audioFramesPerPes (p. 673)" : integer,
    "scte35Pid (p. 673)" : integer,
    "transportStreamId (p. 673)" : integer,
    "videoPid (p. 673)" : integer,
    "pcrControl (p. 673)" : enum,
    "privateMetadataPid (p. 673)" : integer,
    "patInterval (p. 674)" : integer,
    "programNumber (p. 674)" : integer,
    "scte35Source (p. 674)" : enum
  },
  "m2tsSettings (p. 625)" : {
    "dvbTeletextPid (p. 667)" : integer,
    "bitrate (p. 667)" : integer,
    "segmentationTime (p. 667)" : number,
    "audioPids (p. 668)" : [ integer
    ],
    "rateMode (p. 668)" : enum,
    "ebpAudioInterval (p. 668)" : enum,
    "fragmentTime (p. 668)" : number,
    "audioFramesPerPes (p. 668)" : integer,
    "maxPcrInterval (p. 668)" : integer,
    "privateMetadataPid (p. 669)" : integer,
    "scte35Pid (p. 669)" : integer,
    "pmtInterval (p. 669)" : integer,
    "segmentationStyle (p. 669)" : enum,
    "audioBufferModel (p. 669)" : enum,
    "programNumber (p. 669)" : integer,
    "dvbNitSettings (p. 669)" : {
      "networkName (p. 628)" : "string",
      "networkId (p. 628)" : integer,
    }
  }
}
"nitInterval (p. 628)": integer,
"scte35Source (p. 669)": enum,
"pmtPid (p. 670)": integer,
"bufferModel (p. 670)": enum,
"ebpPlacement (p. 670)": enum,
"dvbSdtSettings (p. 670)": {
  "sdtInterval (p. 628)": integer,
  "serviceName (p. 628)": "string",
  "serviceProviderName (p. 628)": "string",
  "outputSdt (p. 629)": enum
},
"nullPacketBitrate (p. 670)": number,
"pcrPid (p. 670)": integer,
"minEbpInterval (p. 670)": integer,
"transportStreamId (p. 671)": integer,
"pcrControl (p. 671)": enum,
"videoPid (p. 671)": integer,
"esRateInPes (p. 671)": enum,
"segmentationMarkers (p. 671)": enum,
"dvbTdtSettings (p. 671)": {
  "tdtInterval (p. 633)": integer
},
"patInterval (p. 671)": integer,
"dvbSubPids (p. 672)": [ integer ]
},
"movSettings (p. 625)": {
  "reference (p. 675)": enum,
  "paddingControl (p. 675)": enum,
  "mpeg2FourCCControl (p. 675)": enum,
  "cslgAtom (p. 676)": enum,
  "clapAtom (p. 676)": enum
},
"f4vSettings (p. 625)": {
  "moovPlacement (p. 638)": enum
},
"captionDescriptions (p. 687)": [
  { "languageDescription (p. 622)": "string",
    "languageCode (p. 622)": enum,
    "destinationSettings (p. 622)": {
      "burninDestinationSettings (p. 623)": {
        "xPosition (p. 618)": integer,
        "backgroundColor (p. 619)": enum,
        "teletextSpacing (p. 619)": enum,
        "yPosition (p. 619)": integer,
        "backgroundOpacity (p. 619)": integer,
        "fontOpacity (p. 619)": integer,
        "shadowOpacity (p. 620)": integer,
        "fontResolution (p. 620)": integer,
        "shadowYOffset (p. 620)": integer,
        "outlineSize (p. 620)": integer,
        "outlineColor (p. 620)": enum,
        "fontSize (p. 620)": integer,
        "shadowXOffset (p. 621)": integer,
        "alignment (p. 621)": enum,
        "shadowColor (p. 621)": enum,
        "fontColor (p. 621)": enum
      },
      "teletextDestinationSettings (p. 623)": {
        "pageNumber (p. 692)": "string"
      },
      "ttmlDestinationSettings (p. 623)": {

"stylePassthrough (p. 694)": enum,
"destinationType (p. 623)": enum,
"dvbSubDestinationSettings (p. 623)": {
  "xPosition (p. 629)": integer,
  "backgroundColor (p. 629)": enum,
  "teletextSpacing (p. 629)": enum,
  "yPosition (p. 629)": integer,
  "backgroundOpacity (p. 629)": integer,
  "fontOpacity (p. 630)": integer,
  "shadowOpacity (p. 630)": integer,
  "fontResolution (p. 630)": integer,
  "shadowYOffset (p. 630)": integer,
  "outlineSize (p. 630)": integer,
  "outlineColor (p. 631)": enum,
  "fontSize (p. 631)": integer,
  "shadowXOffset (p. 631)": integer,
  "alignment (p. 631)": enum,
  "shadowColor (p. 631)": enum,
  "fontColor (p. 631)": enum
},
"sccDestinationSettings (p. 623)": {
  "framerate (p. 692)": enum
}
",
"lastUpdated (p. 686)": "string",
"createdAt (p. 686)": "string",
"name (p. 686)": "string",
"description (p. 686)": "string",
"category (p. 686)": "string",
"type (p. 687)": enum,
"arn (p. 687)": "string"
}

Example ExceptionBody

{
  "message (p. 638)": "string"
}

Properties

AacAudioDescriptionBroadcasterMix (Enum)

Choose BROADCASTER_MIXED_AD when the input contains pre-mixed main audio + audio description (AD) as a stereo pair. The value for AudioType will be set to 3, which signals to downstream systems that this stream contains "broadcaster mixed AD". Note that the input received by the encoder must contain pre-mixed audio; the encoder does not perform the mixing. When you choose BROADCASTER_MIXED_AD, the encoder ignores any values you provide in AudioType and FollowInputAudioType. Choose NORMAL when the input does not contain pre-mixed audio + audio description (AD). In this case, the encoder will use any values you provide for AudioType and FollowInputAudioType.

BROADCASTER_MIXED_AD
NORMAL
AacCodecProfile (Enum)

AAC Profile.

LC
HEV1
HEV2

AacCodingMode (Enum)

 Mono (Audio Description), Mono, Stereo, or 5.1 channel layout. Valid values depend on rate control mode and profile. “1.0 - Audio Description (Receiver Mix)” setting receives a stereo description plus control track and emits a mono AAC encode of the description track, with control data emitted in the PES header as per ETSI TS 101 154 Annex E.

AD_RECEIVER_MIX
CODING_MODE_1_0
CODING_MODE_1_1
CODING_MODE_2_0
CODING_MODE_5_1

AacRateControlMode (Enum)

 Rate Control Mode.

 CBR
VBR

AacRawFormat (Enum)

 Enables LATM/LOAS AAC output. Note that if you use LATM/LOAS AAC in an output, you must choose "No container" for the output container.

 LATM_LOAS
NONE

AacSettings

vbrQuality

 Type: AacVbrQuality (p. 611)
 Required: False

codecProfile

 Type: AacCodecProfile (p. 610)
 Required: False

codingMode

 Type: AacCodingMode (p. 610)
 Required: True
**specification**

*Type:* AacSpecification (p. 611)  
*Required:* False

**bitrate**

Average bitrate in bits/second. Valid values depend on rate control mode and profile.

*Type:* integer  
*Required:* False  
*Minimum:* 6000  
*Maximum:* 1024000

**rawFormat**

*Type:* AacRawFormat (p. 610)  
*Required:* False

**rateControlMode**

*Type:* AacRateControlMode (p. 610)  
*Required:* False

**sampleRate**

Sample rate in Hz. Valid values depend on rate control mode and profile.

*Type:* integer  
*Required:* True  
*Minimum:* 8000  
*Maximum:* 96000

**audioDescriptionBroadcasterMix**

*Type:* AacAudioDescriptionBroadcasterMix (p. 609)  
*Required:* False

### AacSpecification (Enum)

Use MPEG-2 AAC instead of MPEG-4 AAC audio for raw or MPEG-2 Transport Stream containers.

- MPEG2
- MPEG4

### AacVbrQuality (Enum)

VBR quality level. Only used if the rate control mode (AacRateControlMode) is VBR.

- LOW
- MEDIUM_LOW
- MEDIUM_HIGH
- HIGH
**Ac3BitstreamMode (Enum)**

Specifies the "Bitstream Mode" (bsmod) for the emitted AC-3 stream. See ATSC A/52-2012 for background on these values.

- COMPLETE_MAIN
- COMMENTARY
- DIALOGUE
- EMERGENCY
- HEARING_IMPAIRED
- MUSIC_AND_EFFECTS
- VISUALLY_IMPAIRED
- VOICE_OVER

**Ac3CodingMode (Enum)**

Dolby Digital coding mode. Determines number of channels.

- CODING_MODE_1_0
- CODING_MODE_1_1
- CODING_MODE_2_0
- CODING_MODE_3_2_LFE

**Ac3DynamicRangeCompressionProfile (Enum)**

If set to FILM_STANDARD, adds dynamic range compression signaling to the output bitstream as defined in the Dolby Digital specification.

- FILM_STANDARD
- NONE

**Ac3LfeFilter (Enum)**

Applies a 120Hz lowpass filter to the LFE channel prior to encoding. Only valid with 3_2_LFE coding mode.

- ENABLED
- DISABLED

**Ac3MetadataControl (Enum)**

When set to FOLLOW_INPUT, encoder metadata will be sourced from the DD, DD+, or DolbyE decoder that supplied this audio data. If audio was not supplied from one of these streams, then the static metadata settings will be used.

- FOLLOW_INPUT
- USE_CONFIGURED

**Ac3Settings**

`dynamicRangeCompressionProfile`  

Type: Ac3DynamicRangeCompressionProfile (p. 612)  
Required: False
dialnorm
Sets the dialnorm for the output. If blank and input audio is Dolby Digital, dialnorm will be passed through.

  Type: integer
  Required: False
  Minimum: 1
  Maximum: 31

codingMode

  Type: Ac3CodingMode (p. 612)
  Required: False

metadataControl

  Type: Ac3MetadataControl (p. 612)
  Required: False

lfeFilter

  Type: Ac3LfeFilter (p. 612)
  Required: False

bitrate
Average bitrate in bits/second. Valid bitrates depend on the coding mode.

  Type: integer
  Required: False
  Minimum: 64000
  Maximum: 640000

bitstreamMode

  Type: Ac3BitstreamMode (p. 612)
  Required: False

sampleRate
Sample rate in hz. Sample rate is always 48000.

  Type: integer
  Required: False
  Minimum: 48000
  Maximum: 48000

AfdSignaling (Enum)
This setting only applies to H.264 and MPEG2 outputs. Use Insert AFD signaling (AfdSignaling) to whether there are AFD values in the output video data and what those values are.

* Choose None to remove all AFD values from this output.
* Choose Fixed to ignore input AFD values and instead encode the value specified in the job.

* Choose Auto to calculate output AFD values based on the input AFD scaler data.

NONE
AUTO
FIXED

**AiffSettings**

**channels**

Set Channels to specify the number of channels in this output audio track. Choosing Mono in the console will give you 1 output channel; choosing Stereo will give you 2. In the API, valid values are 1 and 2.

Type: integer
Required: False
Minimum: 1
Maximum: 2

**bitDepth**

Specify Bit depth (BitDepth), in bits per sample, to choose the encoding quality for this audio track.

Type: integer
Required: False
Minimum: 16
Maximum: 24

**sampleRate**

Sample rate in hz.

Type: integer
Required: False
Minimum: 8000
Maximum: 192000

**AntiAlias (Enum)**

Enable Anti-alias (AntiAlias) to enhance sharp edges in video output when your input resolution is much larger than your output resolution. Default is enabled.

DISABLED
ENABLED

**AudioCodec (Enum)**

Type of Audio codec.

AAC
MP2
WAV
AIFF
AC3
EAC3
PASSTHROUGH

AudioCodecSettings

codec

Type: AudioCodec (p. 614)
Required: True

wavSettings

Type: WavSettings (p. 698)
Required: False

ac3Settings

Type: Ac3Settings (p. 612)
Required: False

aacSettings

Type: AacSettings (p. 610)
Required: False

aiFFSettings

Type: AiffSettings (p. 614)
Required: False

eac3Settings

Type: Eac3Settings (p. 635)
Required: False

mp2Settings

Type: Mp2Settings (p. 676)
Required: False

AudioDescription

languageCodeControl

Type: AudioLanguageCodeControl (p. 617)
Required: False

audioTypeControl

Type: AudioTypeControl (p. 618)
Required: False
remixSettings

Advanced audio remixing settings.

Type: RemixSettings (p. 691)
Required: False

audioType

Applies only if Follow Input Audio Type is unchecked (false). A number between 0 and 255. The following are defined in ISO-IEC 13818-1: 0 = Undefined, 1 = Clean Effects, 2 = Hearing Impaired, 3 = Visually Impaired Commentary, 4-255 = Reserved.

Type: integer
Required: False
Minimum: 0
Maximum: 255

audioSourceName

Specifies which audio data to use from each input. In the simplest case, specify an "Audio Selector":#inputs-audio_selector by name based on its order within each input. For example if you specify 'Audio Selector 3", then the third audio selector will be used from each input. If an input does not have an "Audio Selector 3", then the audio selector marked as "default" in that input will be used. If there is no audio selector marked as "default", silence will be inserted for the duration of that input. Alternatively, an "Audio Selector Group":#inputs-audio_selector_group name may be specified, with similar default/silence behavior. If no audio_source_name is specified, then "Audio Selector 1" will be chosen automatically.

Type: string
Required: False

codecSettings

Type: AudioCodecSettings (p. 615)
Required: True

languageCode

Indicates the language of the audio output track. The ISO 639 language specified in the 'Language Code' drop down will be used when 'Follow Input Language Code' is not selected or when 'Follow Input Language Code' is selected but there is no ISO 639 language code specified by the input.

Type: LanguageCode (p. 660)
Required: False

streamName

Used for Microsoft Smooth Streaming and Apple HLS outputs. Indicates the name displayed by the player (eg. English, or Director Commentary). Alphanumeric characters, spaces, and underscore are legal.

Type: string
Required: False
Pattern: ^[\w\s]*$
audioNormalizationSettings

Advanced audio normalization settings.

Type: AudioNormalizationSettings (p. 617)
Required: False

AudioLanguageCodeControl (Enum)

Choosing FOLLOW_INPUT will cause the ISO 639 language code of the output to follow the ISO 639 language code of the input. The language specified for languageCode' will be used when USE_CONFIGURED is selected or when FOLLOW_INPUT is selected but there is no ISO 639 language code specified by the input.

FOLLOW_INPUT
USE_CONFIGURED

AudioNormalizationAlgorithm (Enum)

Audio normalization algorithm to use. 1770-1 conforms to the CALM Act specification, 1770-2 conforms to the EBU R-128 specification.

ITU_BS_1770_1
ITU_BS_1770_2

AudioNormalizationAlgorithmControl (Enum)

When enabled the output audio is corrected using the chosen algorithm. If disabled, the audio will be measured but not adjusted.

CORRECT_AUDIO
MEASURE_ONLY

AudioNormalizationLoudnessLogging (Enum)

If set to LOG, log each output's audio track loudness to a CSV file.

LOG
DONT_LOG

AudioNormalizationPeakCalculation (Enum)

If set to TRUE_PEAK, calculate and log the TruePeak for each output's audio track loudness.

TRUE_PEAK
NONE

AudioNormalizationSettings

targetLkfs

Target LKFS(loudness) to adjust volume to. If no value is entered, a default value will be used according to the chosen algorithm. The CALM Act (1770-1) recommends a target of -24 LKFS. The EBU R-128 specification (1770-2) recommends a target of -23 LKFS.
**algorithmControl**

Type: `AudioNormalizationAlgorithmControl (p. 617)`
Required: False

**peakCalculation**

Type: `AudioNormalizationPeakCalculation (p. 617)`
Required: False

**loudnessLogging**

Type: `AudioNormalizationLoudnessLogging (p. 617)`
Required: False

**correctionGateLevel**

Content measuring above this level will be corrected to the target level. Content measuring below this level will not be corrected. Gating only applies when not using `real_time_correction`.

Type: integer
Required: False
Minimum: -70
Maximum: 0

**algorithm**

Type: `AudioNormalizationAlgorithm (p. 617)`
Required: False

**AudioTypeControl (Enum)**

When set to `FOLLOW_INPUT`, if the input contains an ISO 639 `audio_type`, then that value is passed through to the output. If the input contains no ISO 639 `audio_type`, the value in `Audio Type` is included in the output. Otherwise the value in `Audio Type` is included in the output. Note that this field and `audioType` are both ignored if `audioDescriptionBroadcasterMix` is set to `BROADCASTER_MIXED_AD`.

- FOLLOW_INPUT
- USE_CONFIGURED

**BurninDestinationSettings**

**xPosition**

Specifies the horizontal position of the caption relative to the left side of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the left of the output. If no explicit `x_position` is provided, the horizontal caption position will be determined by the alignment parameter. This option is
not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

**Type:** integer  
**Required:** False  
**Minimum:** 0  
**Maximum:** 2147483647

**backgroundColor**

**Type:** BurninSubtitleBackgroundColor (p. 621)  
**Required:** False

**teletextSpacing**

**Type:** BurninSubtitleTeletextSpacing (p. 622)  
**Required:** False

**yPosition**

Specifies the vertical position of the caption relative to the top of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the top of the output. If no explicit y_position is provided, the caption will be positioned towards the bottom of the output. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

**Type:** integer  
**Required:** False  
**Minimum:** 0  
**Maximum:** 2147483647

**backgroundOpacity**

Specifies the opacity of the background rectangle. 255 is opaque; 0 is transparent. Leaving this parameter blank is equivalent to setting it to 0 (transparent). All burn-in and DVB-Sub font settings must match.

**Type:** integer  
**Required:** False  
**Minimum:** 0  
**Maximum:** 255

**fontOpacity**

Specifies the opacity of the burned-in captions. 255 is opaque; 0 is transparent. All burn-in and DVB-Sub font settings must match.

**Type:** integer  
**Required:** True  
**Minimum:** 0  
**Maximum:** 255
shadowOpacity

Specifies the opacity of the shadow. 255 is opaque; 0 is transparent. Leaving this parameter blank is equivalent to setting it to 0 (transparent). All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: 0
Maximum: 255

fontResolution

Font resolution in DPI (dots per inch); default is 96 dpi. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: 96
Maximum: 600

shadowYOffset

Specifies the vertical offset of the shadow relative to the captions in pixels. A value of -2 would result in a shadow offset 2 pixels above the text. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

outlineSize

Specifies font outline size in pixels. This option is not valid for source captions that are either 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: True
Minimum: 0
Maximum: 10

outlineColor

Type: BurninSubtitleOutlineColor (p. 622)
Required: True

fontSize

A positive integer indicates the exact font size in points. Set to 0 for automatic font size selection. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: 0
Maximum: 96
shadowXOffset

Specifies the horizontal offset of the shadow relative to the captions in pixels. A value of -2 would result in a shadow offset 2 pixels to the left. All burn-in and DVB-Sub font settings must match.

- **Type**: integer
- **Required**: False
- **Minimum**: -2147483648
- **Maximum**: 2147483647

alignment

- **Type**: BurninSubtitleAlignment (p. 621)
- **Required**: True

shadowColor

- **Type**: BurninSubtitleShadowColor (p. 622)
- **Required**: False

fontColor

- **Type**: BurninSubtitleFontColor (p. 621)
- **Required**: False

**BurninSubtitleAlignment (Enum)**

If no explicit x_position or y_position is provided, setting alignment to centered will place the captions at the bottom center of the output. Similarly, setting a left alignment will align captions to the bottom left of the output. If x and y positions are given in conjunction with the alignment parameter, the font will be justified (either left or centered) relative to those coordinates. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

- CENTERED
- LEFT

**BurninSubtitleBackgroundColor (Enum)**

Specifies the color of the rectangle behind the captions. All burn-in and DVB-Sub font settings must match.

- NONE
- BLACK
- WHITE

**BurninSubtitleFontColor (Enum)**

Specifies the color of the burned-in captions. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

- WHITE
- BLACK
- YELLOW
RED
GREEN
BLUE

**BurninSubtitleOutlineColor (Enum)**
Specifies font outline color. This option is not valid for source captions that are either 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

BLACK
WHITE
YELLOW
RED
GREEN
BLUE

**BurninSubtitleShadowColor (Enum)**
Specifies the color of the shadow cast by the captions. All burn-in and DVB-Sub font settings must match.

NONE
BLACK
WHITE

**BurninSubtitleTeletextSpacing (Enum)**
Controls whether a fixed grid size or proportional font spacing will be used to generate the output subtitles bitmap. Only applicable for Teletext inputs and DVB-Sub/Burn-in outputs.

FIXED_GRID
PROPORTIONAL

**CaptionDescriptionPreset**

**languageDescription**
Human readable information to indicate captions available for players (e.g., English, or Spanish). Alphanumeric characters, spaces, and underscore are legal.

*Type:* string
*Required:* False
*Pattern:* ^[^\w ]*$

**languageCode**
Indicates the language of the caption output track.

*Type:* LanguageCode (p. 660)
*Required:* False

**destinationSettings**
*Type:* CaptionDestinationSettings (p. 623)
CaptionDestinationSettings

burninDestinationSettings
   Type: BurninDestinationSettings (p. 618)
   Required: False

teletextDestinationSettings
   Type: TeletextDestinationSettings (p. 692)
   Required: False

ttmlDestinationSettings
   Type: TtmlDestinationSettings (p. 694)
   Required: False

destinationType
   Type: CaptionDestinationType (p. 623)
   Required: True

dvbSubDestinationSettings
   Type: DvbSubDestinationSettings (p. 629)
   Required: False

sccDestinationSettings
   Type: SccDestinationSettings (p. 692)
   Required: False

CaptionDestinationType (Enum)

BURN_IN
DVB_SUB
EMBEDDED
SCC
SRT
TELETEXT
TTML
WEBVTT

ChannelMapping

outputChannels
   Type: Array of type OutputChannelMapping (p. 685)
   Required: True
**ColorCorrector**

**saturation**

Saturation level.

Type: integer  
Required: False  
Minimum: 1  
Maximum: 100

**brightness**

Brightness level.

Type: integer  
Required: False  
Minimum: 1  
Maximum: 100

**hdr10Metadata**

Type: Hdr10Metadata (p. 656)  
Required: False

**contrast**

Contrast level.

Type: integer  
Required: False  
Minimum: 1  
Maximum: 100

**hue**

Hue in degrees.

Type: integer  
Required: False  
Minimum: -180  
Maximum: 180

**colorSpaceConversion**

Type: ColorSpaceConversion (p. 625)  
Required: False

**ColorMetadata (Enum)**

Enable Insert color metadata (ColorMetadata) to include color metadata in this output. This setting is enabled by default.

IGNORE
ColorSpaceConversion (Enum)

Determines if colorspace conversion will be performed. If set to _None_, no conversion will be performed. If _Force 601_ or _Force 709_ are selected, conversion will be performed for inputs with differing colorspaces. An input's colorspace can be specified explicitly in the "Video Selector".#inputs-video_selector if necessary.

- NONE
- FORCE_601
- FORCE_709
- FORCE_HDR10
- FORCE_HLG_2020

ContainerSettings

container

- Type: ContainerType (p. 625)
- Required: True

mp4Settings

- Type: Mp4Settings (p. 677)
- Required: False

m3u8Settings

- Type: M3u8Settings (p. 672)
- Required: False

m2tsSettings

- Type: M2tsSettings (p. 667)
- Required: False

movSettings

- Type: MovSettings (p. 675)
- Required: False

f4vSettings

- Type: F4vSettings (p. 638)
- Required: False

ContainerType (Enum)

Container for this output. Some containers require a container settings object. If not specified, the default object will be created.
Properties

F4V
ISMV
M2TS
M3U8
MOV
MP4
MPD
MXF
RAW

CreatePresetRequest

settings

Type: PresetSettings (p. 687)
Required: True

ame

The name of the preset you are creating.

Type: string
Required: True

description

Optional. A description of the preset you are creating.

Type: string
Required: False
category

Optional. A category for the preset you are creating.

Type: string
Required: False

CreatePresetResponse

preset

Type: Preset (p. 686)
Required: False

DeinterlaceAlgorithm (Enum)

Only applies when you set Deinterlacer (DeinterlaceMode) to Deinterlace (DEINTERLACE) or Adaptive (ADAPTIVE). Motion adaptive interpolate (INTERPOLATE) produces sharper pictures, while blend (BLEND) produces smoother motion. Use (INTERPOLATE_TICKER) OR (BLEND_TICKER) if your source file includes a ticker, such as a scrolling headline at the bottom of the frame.
Deinterlacer

**mode**

- **Type**: DeinterlacerMode (p. 627)
- **Required**: False

**control**

- **Type**: DeinterlacerControl (p. 627)
- **Required**: False

**algorithm**

- **Type**: DeinterlaceAlgorithm (p. 626)
- **Required**: False

**DeinterlacerControl (Enum)**

- When set to NORMAL (default), the deinterlacer does not convert frames that are tagged in metadata as progressive. It will only convert those that are tagged as some other type. - When set to FORCE_ALL_FRAMES, the deinterlacer converts every frame to progressive - even those that are already tagged as progressive. Turn Force mode on only if there is a good chance that the metadata has tagged frames as progressive when they are not progressive. Do not turn on otherwise; processing frames that are already progressive into progressive will probably result in lower quality video.

  - FORCE_ALL_FRAMES
  - NORMAL

**DeinterlacerMode (Enum)**

Use Deinterlace (DeinterlaceMode) to choose how the service will do deinterlacing. Default is Deinterlace. - Deinterlace converts interlaced to progressive. - Inverse telecine converts Hard Telecine 29.97i to progressive 23.976p. - Adaptive auto-detects and converts to progressive.

  - DEINTERLACE
  - INVERSE_TELECINE
  - ADAPTIVE

**DropFrameTimecode (Enum)**

Applies only to 29.97 fps outputs. When this feature is enabled, the service will use drop-frame timecode on outputs. If it is not possible to use drop-frame timecode, the system will fall back to non-drop-frame. This setting is enabled by default when Timecode insertion (TimecodeInsertion) is enabled.

  - DISABLED
  - ENABLED
DvbNitSettings

**networkName**

The network name text placed in the network_name_descriptor inside the Network Information Table. Maximum length is 256 characters.

- **Type:** string
- **Required:** True

**networkId**

The numeric value placed in the Network Information Table (NIT).

- **Type:** integer
- **Required:** True
- **Minimum:** 0
- **Maximum:** 65535

**nitInterval**

The number of milliseconds between instances of this table in the output transport stream.

- **Type:** integer
- **Required:** True
- **Minimum:** 25
- **Maximum:** 10000

DvbSdtSettings

**sdtInterval**

The number of milliseconds between instances of this table in the output transport stream.

- **Type:** integer
- **Required:** False
- **Minimum:** 25
- **Maximum:** 2000

**serviceName**

The service name placed in the service_descriptor in the Service Description Table. Maximum length is 256 characters.

- **Type:** string
- **Required:** False

**serviceProviderName**

The service provider name placed in the service_descriptor in the Service Description Table. Maximum length is 256 characters.

- **Type:** string
Required: False

**outputSdt**

Type: OutputSdt (p. 686)
Required: False

**DvbSubDestinationSettings**

**xPosition**

Specifies the horizontal position of the caption relative to the left side of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the left of the output. If no explicit x_position is provided, the horizontal caption position will be determined by the alignment parameter. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

**backgroundColor**

Type: DvbSubtitleBackgroundColor (p. 632)
Required: False

**teletextSpacing**

Type: DvbSubtitleTeletextSpacing (p. 632)
Required: False

**yPosition**

Specifies the vertical position of the caption relative to the top of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the top of the output. If no explicit y_position is provided, the caption will be positioned towards the bottom of the output. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

**backgroundOpacity**

Specifies the opacity of the background rectangle. 255 is opaque; 0 is transparent. Leaving this parameter blank is equivalent to setting it to 0 (transparent). All burn-in and DVB-Sub font settings must match.

Type: integer
fontOpacity

Specifies the opacity of the burned-in captions. 255 is opaque; 0 is transparent. All burn-in and DVB-Sub font settings must match.

- **Type**: integer
- **Required**: True
- **Minimum**: 0
- **Maximum**: 255

shadowOpacity

Specifies the opacity of the shadow. 255 is opaque; 0 is transparent. Leaving this parameter blank is equivalent to setting it to 0 (transparent). All burn-in and DVB-Sub font settings must match.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 255

fontResolution

Font resolution in DPI (dots per inch); default is 96 dpi. All burn-in and DVB-Sub font settings must match.

- **Type**: integer
- **Required**: False
- **Minimum**: 96
- **Maximum**: 600

shadowYOffset

Specifies the vertical offset of the shadow relative to the captions in pixels. A value of -2 would result in a shadow offset 2 pixels above the text. All burn-in and DVB-Sub font settings must match.

- **Type**: integer
- **Required**: False
- **Minimum**: -2147483648
- **Maximum**: 2147483647

outlineSize

Specifies font outline size in pixels. This option is not valid for source captions that are either 608/ embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

- **Type**: integer
- **Required**: True
- **Minimum**: 0
Maximum: 10

outlineColor

Type: DvbSubtitleOutlineColor (p. 632)
Required: True

fontSize

A positive integer indicates the exact font size in points. Set to 0 for automatic font size selection. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: 0
Maximum: 96

shadowXOffset

Specifies the horizontal offset of the shadow relative to the captions in pixels. A value of -2 would result in a shadow offset 2 pixels to the left. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

alignment

Type: DvbSubtitleAlignment (p. 631)
Required: True

shadowColor

Type: DvbSubtitleShadowColor (p. 632)
Required: False

fontColor

Type: DvbSubtitleFontColor (p. 632)
Required: False

DvbSubtitleAlignment (Enum)

If no explicit x_position or y_position is provided, setting alignment to centered will place the captions at the bottom center of the output. Similarly, setting a left alignment will align captions to the bottom left of the output. If x and y positions are given in conjunction with the alignment parameter, the font will be justified (either left or centered) relative to those coordinates. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

CENTERED
LEFT

**DvbSubtitleBackgroundColor (Enum)**

Specifies the color of the rectangle behind the captions. All burn-in and DVB-Sub font settings must match.

- NONE
- BLACK
- WHITE

**DvbSubtitleFontColor (Enum)**

Specifies the color of the burned-in captions. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

- WHITE
- BLACK
- YELLOW
- RED
- GREEN
- BLUE

**DvbSubtitleOutlineColor (Enum)**

Specifies font outline color. This option is not valid for source captions that are either 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

- BLACK
- WHITE
- YELLOW
- RED
- GREEN
- BLUE

**DvbSubtitleShadowColor (Enum)**

Specifies the color of the shadow cast by the captions. All burn-in and DVB-Sub font settings must match.

- NONE
- BLACK
- WHITE

**DvbSubtitleTeletextSpacing (Enum)**

Controls whether a fixed grid size or proportional font spacing will be used to generate the output subtitles bitmap. Only applicable for Teletext inputs and DVB-Sub/Burn-in outputs.

- FIXED_GRID
- PROPORTIONAL
**DvbTdtSettings**

**tdtInterval**

The number of milliseconds between instances of this table in the output transport stream.

- **Type:** integer
- **Required:** True
- **Minimum:** 1000
- **Maximum:** 30000

**Eac3AttenuationControl (Enum)**

If set to ATTENUATE_3_DB, applies a 3 dB attenuation to the surround channels. Only used for 3/2 coding mode.

- ATTENUATE_3_DB
- NONE

**Eac3BitstreamMode (Enum)**

Specifies the "Bitstream Mode" (bsmod) for the emitted E-AC-3 stream. See ATSC A/52-2012 (Annex E) for background on these values.

- COMPLETE_MAIN
- COMMENTARY
- EMERGENCY
- HEARING_IMPAIRED
- VISUALLY_IMPAIRED

**Eac3CodingMode (Enum)**

Dolby Digital Plus coding mode. Determines number of channels.

- CODING_MODE_1_0
- CODING_MODE_2_0
- CODING_MODE_3_2

**Eac3DcFilter (Enum)**

Activates a DC highpass filter for all input channels.

- ENABLED
- DISABLED

**Eac3DynamicRangeCompressionLine (Enum)**

Enables Dynamic Range Compression that restricts the absolute peak level for a signal.

- NONE
- FILM_STANDARD
- FILM_LIGHT
- MUSIC_STANDARD
MUSIC_LIGHT
SPEECH

Eac3DynamicRangeCompressionRf (Enum)
Enables Heavy Dynamic Range Compression, ensures that the instantaneous signal peaks do not exceed specified levels.

NONE
FILM_STANDARD
FILM_LIGHT
MUSIC_STANDARD
MUSIC_LIGHT
SPEECH

Eac3LfeControl (Enum)
When encoding 3/2 audio, controls whether the LFE channel is enabled

LFE
NO_LFE

Eac3LfeFilter (Enum)
Applies a 120Hz lowpass filter to the LFE channel prior to encoding. Only valid with 3_2_LFE coding mode.

ENABLED
DISABLED

Eac3MetadataControl (Enum)
When set to FOLLOW_INPUT, encoder metadata will be sourced from the DD, DD+, or DolbyE decoder that supplied this audio data. If audio was not supplied from one of these streams, then the static metadata settings will be used.

FOLLOW_INPUT
USE_CONFIGURED

Eac3PassthroughControl (Enum)
When set to WHEN_POSSIBLE, input DD+ audio will be passed through if it is present on the input. This detection is dynamic over the life of the transcode. Inputs that alternate between DD+ and non-DD+ content will have a consistent DD+ output as the system alternates between passthrough and encoding.

WHEN_POSSIBLE
NO_PASSTHROUGH

Eac3PhaseControl (Enum)
Controls the amount of phase-shift applied to the surround channels. Only used for 3/2 coding mode.

SHIFT_90_DEGREES
NO_SHIFT
**Eac3Settings**

**dialnorm**
Sets the dialnorm for the output. If blank and input audio is Dolby Digital Plus, dialnorm will be passed through.

*Type:* integer  
*Required:* False  
*Minimum:* 1  
*Maximum:* 31

**passthroughControl**

*Type:* Eac3PassthroughControl (p. 634)  
*Required:* False

**metadataControl**

*Type:* Eac3MetadataControl (p. 634)  
*Required:* False

**bitrate**
Average bitrate in bits/second. Valid bitrates depend on the coding mode.

*Type:* integer  
*Required:* False  
*Minimum:* 64000  
*Maximum:* 640000

**dynamicRangeCompressionRf**

*Type:* Eac3DynamicRangeCompressionRf (p. 634)  
*Required:* False

**sampleRate**
Sample rate in hz. Sample rate is always 48000.

*Type:* integer  
*Required:* False  
*Minimum:* 48000  
*Maximum:* 48000

**ltRtSurroundMixLevel**
Left total/Right total surround mix level. Only used for 3/2 coding mode. Valid values: -1.5 -3.0 -4.5 -6.0 -60

*Type:* number  
*Required:* False  
*Format:* float  
*Minimum:* -60.0
Maximum: -1.5

**surroundExMode**

Type: `Eac3SurroundExMode (p. 637)`  
Required: False

**dynamicRangeCompressionLine**

Type: `Eac3DynamicRangeCompressionLine (p. 633)`  
Required: False

**lfeControl**

Type: `Eac3LfeControl (p. 634)`  
Required: False

**codingMode**

Type: `Eac3CodingMode (p. 633)`  
Required: False

**surroundMode**

Type: `Eac3SurroundMode (p. 638)`  
Required: False

**attenuationControl**

Type: `Eac3AttenuationControl (p. 633)`  
Required: False

**lfeFilter**

Type: `Eac3LfeFilter (p. 634)`  
Required: False

**phaseControl**

Type: `Eac3PhaseControl (p. 634)`  
Required: False

**ltRtCenterMixLevel**

Left total/Right total center mix level. Only used for 3/2 coding mode. Valid values: 3.0, 1.5, 0.0, -1.5, -3.0, -4.5, -6.0, -60

Type: number  
Required: False  
Format: float  
Minimum: -60.0  
Maximum: 3.0
dcFilter

Type: Eac3DcFilter (p. 633)
Required: False

stereoDownmix

Type: Eac3StereoDownmix (p. 637)
Required: False

bitstreamMode

Type: Eac3BitstreamMode (p. 633)
Required: False

loRoSurroundMixLevel

Left only/Right only surround mix level. Only used for 3/2 coding mode. Valid values: -1.5 -3.0 -4.5 -6.0 -60

Type: number
Required: False
Format: float
Minimum: -60.0
Maximum: -1.5

loRoCenterMixLevel

Left only/Right only center mix level. Only used for 3/2 coding mode. Valid values: 3.0, 1.5, 0.0, -1.5 -3.0 -4.5 -6.0 -60

Type: number
Required: False
Format: float
Minimum: -60.0
Maximum: 3.0

Eac3StereoDownmix (Enum)

Stereo downmix preference. Only used for 3/2 coding mode.

NOT_INDICATED
LO_RO
LT_RT
DPL2

Eac3SurroundExMode (Enum)

When encoding 3/2 audio, sets whether an extra center back surround channel is matrix encoded into the left and right surround channels.

NOT_INDICATED
ENABLED
DISABLED
Eac3SurroundMode (Enum)
When encoding 2/0 audio, sets whether Dolby Surround is matrix encoded into the two channels.

- NOT_INDICATED
- ENABLED
- DISABLED

ExceptionBody

message

Type: string
Required: False

F4vMoovPlacement (Enum)
If set to PROGRESSIVE_DOWNLOAD, the MOOV atom is relocated to the beginning of the archive as required for progressive downloading. Otherwise it is placed normally at the end.

- PROGRESSIVE_DOWNLOAD
- NORMAL

F4vSettings

moovPlacement

Type: F4vMoovPlacement (p. 638)
Required: False

FrameCaptureSettings

framerateDenominator

Frame capture will encode the first frame of the output stream, then one frame every framerateDenominator/framerateNumerator seconds. For example, settings of framerateNumerator = 1 and framerateDenominator = 3 (a rate of 1/3 frame per second) will capture the first frame, then 1 frame every 3s. Files will be named as filename.n.jpg where n is the 0-based sequence number of each Capture.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

maxCaptures

Maximum number of captures (encoded jpg output files).

Type: integer
Required: False
Minimum: 1
Maximum: 10000000
framerateNumerator

Frame capture will encode the first frame of the output stream, then one frame every framerateDenominator/framerateNumerator seconds. For example, settings of framerateNumerator = 1 and framerateDenominator = 3 (a rate of 1/3 frame per second) will capture the first frame, then 1 frame every 3s. Files will be named as filename.NNNNNNN.jpg where N is the 0-based frame sequence number zero padded to 7 decimal places.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

quality

JPEG Quality - a higher value equals higher quality.

Type: integer
Required: False
Minimum: 1
Maximum: 100

H264AdaptiveQuantization (Enum)

Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

OFF
LOW
MEDIUM
HIGH
HIGHER
MAX

H264CodecLevel (Enum)

H.264 Level.

AUTO
LEVEL_1
LEVEL_1_1
LEVEL_1_2
LEVEL_1_3
LEVEL_2
LEVEL_2_1
LEVEL_2_2
LEVEL_3
LEVEL_3_1
LEVEL_3_2
LEVEL_4
LEVEL_4_1
LEVEL_4_2
LEVEL_5
LEVEL_5_1
LEVEL_5_2
**H264CodecProfile (Enum)**

H.264 Profile. High 4:2:2 and 10-bit profiles are only available with the AVC-I License.

- BASELINE
- HIGH
- HIGH_10BIT
- HIGH_422
- HIGH_422_10BIT
- MAIN

**H264EntropyEncoding (Enum)**

Entropy encoding mode. Use CABAC (must be in Main or High profile) or CAVLC.

- CABAC
- CAVLC

**H264FieldEncoding (Enum)**

Choosing FORCE_FIELD disables PAFF encoding for interlaced outputs.

- PAFF
- FORCE_FIELD

**H264FlickerAdaptiveQuantization (Enum)**

Adjust quantization within each frame to reduce flicker or 'pop' on I-frames.

- DISABLED
- ENABLED

**H264FramerateControl (Enum)**

Using the API, set FramerateControl to INITIALIZE_FROM_SOURCE if you want the service to use the framerate from the input. Using the console, do this by choosing INITIALIZE_FROM_SOURCE for Framerate.

- INITIALIZE_FROM_SOURCE
- SPECIFIED

**H264FramerateConversionAlgorithm (Enum)**

When set to INTERPOLATE, produces smoother motion during framerate conversion.

- DUPLICATE_DROP
- INTERPOLATE

**H264GopBReference (Enum)**

If enable, use reference B frames for GOP structures that have B frames > 1.

- DISABLED
- ENABLED
H264GopSizeUnits (Enum)

Indicates if the GOP Size in H264 is specified in frames or seconds. If seconds the system will convert the GOP Size into a frame count at run time.

   FRAMES
   SECONDS

H264InterlaceMode (Enum)

Use Interlace mode (InterlaceMode) to choose the scan line type for the output.

* Top Field First (TOP_FIELD) and Bottom Field First (BOTTOM_FIELD) produce interlaced output with the entire output having the same field polarity (top or bottom first).

* Follow, Default Top (FOLLOW_TOP_FIELD) and Follow, Default Bottom (FOLLOW_BOTTOM_FIELD) use the same field polarity as the source. Therefore, behavior depends on the input scan type. If the source is interlaced, the output will be interlaced with the same polarity as the source (it will follow the source). The output could therefore be a mix of "top field first" and "bottom field first". If the source is progressive, the output will be interlaced with "top field first" or "bottom field first" polarity, depending on which of the Follow options you chose.

   PROGRESSIVE
   TOP_FIELD
   BOTTOM_FIELD
   FOLLOW_TOP_FIELD
   FOLLOW_BOTTOM_FIELD

H264ParControl (Enum)

Using the API, enable ParFollowSource if you want the service to use the pixel aspect ratio from the input. Using the console, do this by choosing Follow source for Pixel aspect ratio.

   INITIALIZE_FROM_SOURCE
   SPECIFIED

H264QualityTuningLevel (Enum)

Use Quality tuning level (H264QualityTuningLevel) to specify whether to use fast single-pass, high-quality singlepass, or high-quality multipass video encoding.

   SINGLE_PASS
   SINGLE_PASS_HQ
   MULTI_PASS_HQ

H264RateControlMode (Enum)

Rate control mode. CQ uses constant quantizer (qp), ABR (average bitrate) does not write HRD parameters.

   VBR
   CBR

H264RepeatPps (Enum)

Places a PPS header on each encoded picture, even if repeated.
**DISABLED**
**ENABLED**

**H264SceneChangeDetect (Enum)**

Scene change detection (inserts I-frames on scene changes).

**DISABLED**
**ENABLED**

**H264Settings**

**slices**

Number of slices per picture. Must be less than or equal to the number of macroblock rows for progressive pictures, and less than or equal to half the number of macroblock rows for interlaced pictures.

*Type:* integer  
*Required:* False  
*Minimum:* 1  
*Maximum:* 32

**minInterval**

Enforces separation between repeated (cadence) I-frames and I-frames inserted by Scene Change Detection. If a scene change I-frame is within I-interval frames of a cadence I-frame, the GOP is shrunk and/or stretched to the scene change I-frame. GOP stretch requires enabling lookahead as well as setting I-interval. The normal cadence resumes for the next GOP. This setting is only used when Scene Change Detect is enabled. Note: Maximum GOP stretch = GOP size + Min-I-interval - 1

*Type:* integer  
*Required:* False  
*Minimum:* 0  
*Maximum:* 30

**parNumerator**

Pixel Aspect Ratio numerator.

*Type:* integer  
*Required:* False  
*Minimum:* 1  
*Maximum:* 2147483647

**flickerAdaptiveQuantization**

*Type:* H264FlickerAdaptiveQuantization (p. 640)  
*Required:* False

**gopSizeUnits**

*Type:* H264GopSizeUnits (p. 641)
Properties

Required: False

**hrdBufferSize**

Size of buffer (HRD buffer model). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.

*Type*: integer
*Required*: False
*Minimum*: -2147483648
*Maximum*: 2147483647

**qualityTuningLevel**

*Type*: H264QualityTuningLevel (p. 641)
*Required*: False

**maxBitrate**

Maximum bitrate in bits/second (for VBR mode only). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.

*Type*: integer
*Required*: False
*Minimum*: -2147483648
*Maximum*: 2147483647

**bitrate**

Average bitrate in bits/second. Required for VBR, CBR, and ABR. Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.

*Type*: integer
*Required*: False
*Minimum*: 1000
*Maximum*: 2147483647

**spatialAdaptiveQuantization**

*Type*: H264SpatialAdaptiveQuantization (p. 647)
*Required*: False

**slowPal**

*Type*: H264SlowPal (p. 647)
*Required*: False

**codecProfile**

*Type*: H264CodecProfile (p. 640)
*Required*: False
unregisteredSeiTimecode

Type: H264UnregisteredSeiTimecode (p. 647)
Required: False

softness

Softness. Selects quantizer matrix, larger values reduce high-frequency content in the encoded image.

Type: integer
Required: False
Minimum: 0
Maximum: 128

framerateControl

Type: H264FramerateControl (p. 640)
Required: False

telecine

Type: H264Telecine (p. 647)
Required: False

framerateConversionAlgorithm

Type: H264FramerateConversionAlgorithm (p. 640)
Required: False

codecLevel

Type: H264CodecLevel (p. 639)
Required: False

numberReferenceFrames

Number of reference frames to use. The encoder may use more than requested if using B-frames and/or interlaced encoding.

Type: integer
Required: False
Minimum: 1
Maximum: 6

temporalAdaptiveQuantization

Type: H264TemporalAdaptiveQuantization (p. 647)
Required: False

repeatPps

Type: H264RepeatPps (p. 641)
Required: False
**hrdBufferInitialFillPercentage**
Percentage of the buffer that should initially be filled (HRD buffer model).

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 100

**framerateNumerator**
Framerate numerator - framerate is a fraction, e.g. 24000 / 1001 = 23.976 fps.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 2147483647

**numberBFramesBetweenReferenceFrames**
Number of B-frames between reference frames.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 7

**gopClosedCadence**
Frequency of closed GOPs. In streaming applications, it is recommended that this be set to 1 so a decoder joining mid-stream will receive an IDR frame as quickly as possible. Setting this value to 0 will break output segmenting.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 2147483647

**fieldEncoding**

- **Type**: H264FieldEncoding (p. 640)
- **Required**: False

**entropyEncoding**

- **Type**: H264EntropyEncoding (p. 640)
- **Required**: False

**framerateDenominator**
When you use the API for transcode jobs that use framerate conversion, specify the framerate as a fraction. For example, 24000 / 1001 = 23.976 fps. Use FramerateDenominator to specify the denominator of this fraction. In this example, use 1001 for the value of FramerateDenominator. When you use the console for transcode jobs that use framerate conversion, provide the value as a decimal number for Framerate. In this example, specify 23.976.
**Type:** integer  
**Required:** False  
**Minimum:** 1  
**Maximum:** 2147483647

**adaptiveQuantization**

**Type:** H264AdaptiveQuantization (p. 639)  
**Required:** False

**interlaceMode**

**Type:** H264InterlaceMode (p. 641)  
**Required:** False

**gopSize**

GOP Length (keyframe interval) in frames or seconds. Must be greater than zero.

**Type:** number  
**Required:** False  
**Format:** float  
**Minimum:** 0.0

**gopBReference**

**Type:** H264GopBReference (p. 640)  
**Required:** False

**sceneChangeDetect**

**Type:** H264SceneChangeDetect (p. 642)  
**Required:** False

**parDenominator**

Pixel Aspect Ratio denominator.

**Type:** integer  
**Required:** False  
**Minimum:** 1  
**Maximum:** 2147483647

**parControl**

**Type:** H264ParControl (p. 641)  
**Required:** False

**syntax**

**Type:** H264Syntax (p. 647)
**Required**: False

**rateControlMode**

**Type**: H264RateControlMode (p. 641)

**Required**: False

**H264SlowPal (Enum)**

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

- DISABLED
- ENABLED

**H264SpatialAdaptiveQuantization (Enum)**

Adjust quantization within each frame based on spatial variation of content complexity.

- DISABLED
- ENABLED

**H264Syntax (Enum)**

Produces a bitstream compliant with SMPTE RP-2027.

- DEFAULT
- RP2027

**H264Telecine (Enum)**

This field applies only if the Streams > Advanced > Framerate (framerate) field is set to 29.970. This field works with the Streams > Advanced > Preprocessors > Deinterlacer field (deinterlace_mode) and the Streams > Advanced > Interlaced Mode field (interlace_mode) to identify the scan type for the output: Progressive, Interlaced, Hard Telecine or Soft Telecine. - Hard: produces 29.97i output from 23.976 input.

- NONE
- SOFT
- HARD

**H264TemporalAdaptiveQuantization (Enum)**

Adjust quantization within each frame based on temporal variation of content complexity.

- DISABLED
- ENABLED

**H264UnregisteredSeiTimecode (Enum)**

Inserts timecode for each frame as 4 bytes of an unregistered SEI message.

- DISABLED
ENABLED

**H265AdaptiveQuantization (Enum)**

Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

- OFF
- LOW
- MEDIUM
- HIGH
- HIGHER
- MAX

**H265AlternateTransferFunctionSei (Enum)**

Enables Alternate Transfer Function SEI message for outputs using Hybrid Log Gamma (HLG) Electro-Optical Transfer Function (EOTF).

- DISABLED
- ENABLED

**H265CodecLevel (Enum)**

H.265 Level.

- AUTO
- LEVEL_1
- LEVEL_2
- LEVEL_2_1
- LEVEL_3
- LEVEL_3_1
- LEVEL_4
- LEVEL_4_1
- LEVEL_5
- LEVEL_5_1
- LEVEL_5_2
- LEVEL_6
- LEVEL_6_1
- LEVEL_6_2

**H265CodecProfile (Enum)**

Represents the Profile and Tier, per the HEVC (H.265) specification. Selections are grouped as [Profile] / [Tier], so "Main/High" represents Main Profile with High Tier. 4:2:2 profiles are only available with the HEVC 4:2:2 License.

- MAIN_MAIN
- MAIN_HIGH
- MAIN10_MAIN
- MAIN10_HIGH
- MAIN_422_8BIT_MAIN
- MAIN_422_8BIT_HIGH
- MAIN_422_10BIT_MAIN
- MAIN_422_10BIT_HIGH
**H265FlickerAdaptiveQuantization (Enum)**
Adjust quantization within each frame to reduce flicker or 'pop' on I-frames.
- DISABLED
- ENABLED

**H265FramerateControl (Enum)**
Using the API, set FramerateControl to INITIALIZE_FROM_SOURCE if you want the service to use the framerate from the input. Using the console, do this by choosing INITIALIZE_FROM_SOURCE for Framerate.
- INITIALIZE_FROM_SOURCE
- SPECIFIED

**H265FramerateConversionAlgorithm (Enum)**
When set to INTERPOLATE, produces smoother motion during framerate conversion.
- DUPLICATE_DROP
- INTERPOLATE

**H265GopBReference (Enum)**
If enable, use reference B frames for GOP structures that have B frames > 1.
- DISABLED
- ENABLED

**H265GopSizeUnits (Enum)**
Indicates if the GOP Size in H265 is specified in frames or seconds. If seconds the system will convert the GOP Size into a frame count at run time.
- FRAMES
- SECONDS

**H265InterlaceMode (Enum)**
Use Interlace mode (InterlaceMode) to choose the scan line type for the output.
- * Top Field First (TOP_FIELD) and Bottom Field First (BOTTOM_FIELD) produce interlaced output with the entire output having the same field polarity (top or bottom first).

- * Follow, Default Top (FOLLOW_TOP_FIELD) and Follow, Default Bottom (FOLLOW_BOTTOM_FIELD) use the same field polarity as the source. Therefore, behavior depends on the input scan type. If the source is interlaced, the output will be interlaced with the same polarity as the source (it will follow the source). The output could therefore be a mix of "top field first" and "bottom field first". If the source is progressive, the output will be interlaced with "top field first" or "bottom field first" polarity, depending on which of the Follow options you chose.
- PROGRESSIVE
- TOP_FIELD
- BOTTOM_FIELD
- FOLLOW_TOP_FIELD
H265ParControl (Enum)

Using the API, enable ParFollowSource if you want the service to use the pixel aspect ratio from the input. Using the console, do this by choosing Follow source for Pixel aspect ratio.

INITIALIZE_FROM_SOURCE
SPECIFIED

H265QualityTuningLevel (Enum)

Use Quality tuning level (H265QualityTuningLevel) to specify whether to use fast single-pass, high-quality singlepass, or high-quality multipass video encoding.

SINGLE_PASS
SINGLE_PASS_HQ
MULTI_PASS_HQ

H265RateControlMode (Enum)

Rate control mode. CQ uses constant quantizer (qp), ABR (average bitrate) does not write HRD parameters.

VBR
CBR

H265SampleAdaptiveOffsetFilterMode (Enum)

Specify Sample Adaptive Offset (SAO) filter strength. Adaptive mode dynamically selects best strength based on content

DEFAULT
ADAPTIVE
OFF

H265SceneChangeDetect (Enum)

Scene change detection (inserts I-frames on scene changes).

DISABLED
ENABLED

H265Settings

slices

Number of slices per picture. Must be less than or equal to the number of macroblock rows for progressive pictures, and less than or equal to half the number of macroblock rows for interlaced pictures.

Type: integer
Required: False
Minimum: 1
Maximum: 32
**minIInterval**

Enforces separation between repeated (cadence) I-frames and I-frames inserted by Scene Change Detection. If a scene change I-frame is within I-interval frames of a cadence I-frame, the GOP is shrunk and/or stretched to the scene change I-frame. GOP stretch requires enabling lookahead as well as setting I-interval. The normal cadence resumes for the next GOP. This setting is only used when Scene Change Detect is enabled. Note: Maximum GOP stretch = GOP size + Min-I-interval - 1

**Type:** integer  
**Required:** False  
**Minimum:** 0  
**Maximum:** 30

**parNumerator**

Pixel Aspect Ratio numerator.

**Type:** integer  
**Required:** False  
**Minimum:** 1  
**Maximum:** 2147483647

**flickerAdaptiveQuantization**

**Type:** H265FlickerAdaptiveQuantization (p. 649)  
**Required:** False

**gopSizeUnits**

**Type:** H265GopSizeUnits (p. 649)  
**Required:** False

**hrdBufferSize**

Size of buffer (HRD buffer model). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.

**Type:** integer  
**Required:** False  
**Minimum:** -2147483648  
**Maximum:** 2147483647

**qualityTuningLevel**

**Type:** H265QualityTuningLevel (p. 650)  
**Required:** False

**maxBitrate**

Maximum bitrate in bits/second (for VBR mode only). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.

**Type:** integer  
**Required:** False
Properties

Minimum: -2147483648
Maximum: 2147483647

**bitrate**

Average bitrate in bits/second. Required for VBR, CBR, and ABR. Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.

  - **Type:** integer
  - **Required:** False
  - **Minimum:** 1000
  - **Maximum:** 2147483647

**spatialAdaptiveQuantization**

  - **Type:** H265SpatialAdaptiveQuantization (p. 655)
  - **Required:** False

**sampleAdaptiveOffsetFilterMode**

  - **Type:** H265SampleAdaptiveOffsetFilterMode (p. 650)
  - **Required:** False

**temporalIds**

  - **Type:** H265TemporalIds (p. 656)
  - **Required:** False

**slowPal**

  - **Type:** H265SlowPal (p. 655)
  - **Required:** False

**tiles**

  - **Type:** H265Tiles (p. 656)
  - **Required:** False

**codecProfile**

  - **Type:** H265CodecProfile (p. 648)
  - **Required:** False

**alternateTransferFunctionSei**

  - **Type:** H265AlternateTransferFunctionSei (p. 648)
  - **Required:** False

**unregisteredSeiTimecode**

  - **Type:** H265UnregisteredSeiTimecode (p. 656)
Properties

Required: False

framerateControl

Type: H265FramerateControl (p. 649)
Required: False

television

Type: H265Television (p. 655)
Required: False

framerateConversionAlgorithm

Type: H265FramerateConversionAlgorithm (p. 649)
Required: False

codecLevel

Type: H265CodecLevel (p. 648)
Required: False

numberReferenceFrames

Number of reference frames to use. The encoder may use more than requested if using B-frames and/or
interlaced encoding.

Type: integer
Required: False
Minimum: 1
Maximum: 6

temporalAdaptiveQuantization

Type: H265TemporalAdaptiveQuantization (p. 655)
Required: False

hrdBufferInitialFillPercentage

Percentage of the buffer that should initially be filled (HRD buffer model).

Type: integer
Required: False
Minimum: 0
Maximum: 100

framerateNumerator

Framerate numerator - framerate is a fraction, e.g. 24000 / 1001 = 23.976 fps.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

numberBFramesBetweenReferenceFrames
Number of B-frames between reference frames.

Type: integer
Required: False
Minimum: 0
Maximum: 7

gopClosedCadence
Frequency of closed GOPs. In streaming applications, it is recommended that this be set to 1 so a decoder joining mid-stream will receive an IDR frame as quickly as possible. Setting this value to 0 will break output segmenting.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

framerateDenominator
Framerate denominator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

adaptiveQuantization

Type: H265AdaptiveQuantization (p. 648)
Required: False

interlaceMode

Type: H265InterlaceMode (p. 649)
Required: False

gopSize
GOP Length (keyframe interval) in frames or seconds. Must be greater than zero.

Type: number
Required: False
Format: float
Minimum: 0.0

gopBReference

Type: H265GopBReference (p. 649)
Required: False
sceneChangeDetect

Type: H265SceneChangeDetect (p. 650)
Required: False

parDenominator

Pixel Aspect Ratio denominator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

parControl

Type: H265ParControl (p. 650)
Required: False

rateControlMode

Type: H265RateControlMode (p. 650)
Required: False

H265SlowPal (Enum)

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

DISABLED
ENABLED

H265SpatialAdaptiveQuantization (Enum)

Adjust quantization within each frame based on spatial variation of content complexity.

DISABLED
ENABLED

H265Telecine (Enum)

This field applies only if the Streams > Advanced > Framerate (framerate) field is set to 29.970. This field works with the Streams > Advanced > Preprocessors > Deinterlacer field (deinterlace_mode) and the Streams > Advanced > Interlaced Mode field (interlace_mode) to identify the scan type for the output: Progressive, Interlaced, Hard Telecine or Soft Telecine. - Hard: produces 29.97i output from 23.976 input. - Soft: produces 23.976; the player converts this output to 29.97i.

NONE
SOFT
HARD

H265TemporalAdaptiveQuantization (Enum)

Adjust quantization within each frame based on temporal variation of content complexity.
DISABLED
ENABLED

**H265TemporalIds (Enum)**
Enables temporal layer identifiers in the encoded bitstream. Up to 3 layers are supported depending on GOP structure: I- and P-frames form one layer, reference B-frames can form a second layer and non-reference B-frames can form a third layer. Decoders can optionally decode only the lower temporal layers to generate a lower frame rate output. For example, given a bitstream with temporal IDs and with b-frames = 1 (i.e. IbPbPb display order), a decoder could decode all the frames for full frame rate output or only the I and P frames (lowest temporal layer) for a half frame rate output.

DISABLED
ENABLED

**H265Tiles (Enum)**
Enable use of tiles, allowing horizontal as well as vertical subdivision of the encoded pictures.

DISABLED
ENABLED

**H265UnregisteredSeiTimecode (Enum)**
Inserts timecode for each frame as 4 bytes of an unregistered SEI message.

DISABLED
ENABLED

**Hdr10Metadata**

**redPrimaryY**
HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 50000

**greenPrimaryY**
HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 50000

**whitePointX**
HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.
Properties

maxLuminance
Nominal maximum mastering display luminance in units of 0.0001 candelas per square meter.

Type: integer
Required: False
Minimum: 0
Maximum: 50000

greenPrimaryX
HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

Type: integer
Required: False
Minimum: 0
Maximum: 50000

whitePointY
HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

Type: integer
Required: False
Minimum: 0
Maximum: 50000

redPrimaryX
HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

Type: integer
Required: False
Minimum: 0
Maximum: 50000

bluePrimaryX
HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

Type: integer
Required: False
Minimum: 0
Maximum: 50000
maxFrameAverageLightLevel
Maximum average light level of any frame in the coded video sequence, in units of candelas per square meter.

Type: integer
Required: True
Minimum: 0
Maximum: 65535

bluePrimaryY
HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

Type: integer
Required: False
Minimum: 0
Maximum: 50000

maxContentLightLevel
Maximum light level among all samples in the coded video sequence, in units of candelas per square meter.

Type: integer
Required: True
Minimum: 0
Maximum: 65535

minLuminance
Nominal minimum mastering display luminance in units of 0.0001 candelas per square meter

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

ImageInserter
insertableImages
Image to insert. Must be 32 bit windows BMP, PNG, or TGA file. Must not be larger than the output frames.

Type: Array of type InsertableImage (p. 658)
Required: True

InsertableImage
duration
Use Duration (Duration) to set the time, in milliseconds, for the image to remain on the output video.
**Properties**

- **Type**: integer
  - **Required**: False
  - **Minimum**: -2147483648
  - **Maximum**: 2147483647

  **fadeOut**

  Use Fade out (FadeOut) to set the length, in milliseconds, of the inserted image fade out. If you don’t specify a value for Fade out, the image will disappear abruptly at the end of the inserted image duration.

  - **Type**: integer
    - **Required**: False
    - **Minimum**: -2147483648
    - **Maximum**: 2147483647

- **imageY**

  Use Top (ImageY) to set the distance, in pixels, between the inserted image and the top edge of the video frame. Required for BMP, PNG and TGA input.

  - **Type**: integer
    - **Required**: True
    - **Minimum**: -2147483648
    - **Maximum**: 2147483647

  **fadeIn**

  Use Fade in (FadeIn) to set the length, in milliseconds, of the inserted image fade in. If you don’t specify a value for Fade in, the image will appear abruptly at the Start time.

  - **Type**: integer
    - **Required**: False
    - **Minimum**: -2147483648
    - **Maximum**: 2147483647

- **imageX**

  Use Left (ImageX) to set the distance, in pixels, between the inserted image and the left edge of the frame. Required for BMP, PNG and TGA input.

  - **Type**: integer
    - **Required**: True
    - **Minimum**: -2147483648
    - **Maximum**: 2147483647

  **width**

  Specify the Width (Width) of the inserted image. Use a value that is less than or equal to the video resolution width. Leave this setting blank to use the native width of the image.

  - **Type**: integer
    - **Required**: False
    - **Minimum**: -2147483648

659
Maximum: 2147483647

startTime
Use Start time (StartTime) to specify the video timecode when the image is inserted in the output. This must be in timecode format (HH:MM:SS:FF)

Type: string
Required: False

opacity
Use Opacity (Opacity) to specify how much of the underlying video shows through the inserted image. 0 is transparent and 100 is fully opaque. Default is 50.

Type: integer
Required: True
Minimum: 0
Maximum: 100

layer
Use Layer (Layer) to specify how overlapping inserted images appear. Images with higher values of layer appear on top of images with lower values of layer.

Type: integer
Required: True
Minimum: 0
Maximum: 7

height
Specify the Height (Height) of the inserted image. Use a value that is less than or equal to the video resolution height. Leave this setting blank to use the native height of the image.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

imageInserterInput
Use Image location (imageInserterInput) to specify the Amazon S3 location of the image to be inserted into the output. Use a 32 bit BMP, PNG, or TGA file that fits inside the video frame.

Type: string
Required: True
Pattern: ^s3://(.*?)\.(bmp|BMP|png|PNG|tga|TGA)$

LanguageCode (Enum)
Indicates the language of the audio output track. The ISO 639 language specified in the 'Language Code' drop down will be used when 'Follow Input Language Code' is not selected or when 'Follow Input Language Code' is selected but there is no ISO 639 language code specified by the input.
<table>
<thead>
<tr>
<th>Language Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENG</td>
</tr>
<tr>
<td>SPA</td>
</tr>
<tr>
<td>FRA</td>
</tr>
<tr>
<td>DEU</td>
</tr>
<tr>
<td>GER</td>
</tr>
<tr>
<td>ZHO</td>
</tr>
<tr>
<td>ARA</td>
</tr>
<tr>
<td>HIN</td>
</tr>
<tr>
<td>JPN</td>
</tr>
<tr>
<td>RUS</td>
</tr>
<tr>
<td>POR</td>
</tr>
<tr>
<td>ITA</td>
</tr>
<tr>
<td>URD</td>
</tr>
<tr>
<td>VIE</td>
</tr>
<tr>
<td>KOR</td>
</tr>
<tr>
<td>PAN</td>
</tr>
<tr>
<td>ABK</td>
</tr>
<tr>
<td>AAR</td>
</tr>
<tr>
<td>AFR</td>
</tr>
<tr>
<td>AKA</td>
</tr>
<tr>
<td>SQI</td>
</tr>
<tr>
<td>AMH</td>
</tr>
<tr>
<td>ARG</td>
</tr>
<tr>
<td>HYE</td>
</tr>
<tr>
<td>ASM</td>
</tr>
<tr>
<td>AVA</td>
</tr>
<tr>
<td>AVE</td>
</tr>
<tr>
<td>AYM</td>
</tr>
<tr>
<td>AZE</td>
</tr>
<tr>
<td>BAM</td>
</tr>
<tr>
<td>BAK</td>
</tr>
<tr>
<td>EUS</td>
</tr>
<tr>
<td>BEL</td>
</tr>
<tr>
<td>BEN</td>
</tr>
<tr>
<td>BIH</td>
</tr>
<tr>
<td>BIS</td>
</tr>
<tr>
<td>BOS</td>
</tr>
<tr>
<td>BRE</td>
</tr>
<tr>
<td>BUL</td>
</tr>
<tr>
<td>MYA</td>
</tr>
<tr>
<td>CAT</td>
</tr>
<tr>
<td>KHM</td>
</tr>
<tr>
<td>CHA</td>
</tr>
<tr>
<td>CHE</td>
</tr>
<tr>
<td>NYA</td>
</tr>
<tr>
<td>CHU</td>
</tr>
<tr>
<td>CHV</td>
</tr>
<tr>
<td>COR</td>
</tr>
<tr>
<td>COS</td>
</tr>
<tr>
<td>CRE</td>
</tr>
<tr>
<td>HRV</td>
</tr>
<tr>
<td>Code</td>
</tr>
<tr>
<td>------</td>
</tr>
<tr>
<td>LIM</td>
</tr>
<tr>
<td>LIN</td>
</tr>
<tr>
<td>LIT</td>
</tr>
<tr>
<td>LUB</td>
</tr>
<tr>
<td>LTZ</td>
</tr>
<tr>
<td>MKD</td>
</tr>
<tr>
<td>MLG</td>
</tr>
<tr>
<td>MSA</td>
</tr>
<tr>
<td>MAL</td>
</tr>
<tr>
<td>MLT</td>
</tr>
<tr>
<td>GLV</td>
</tr>
<tr>
<td>MRI</td>
</tr>
<tr>
<td>MAR</td>
</tr>
<tr>
<td>MAH</td>
</tr>
<tr>
<td>MON</td>
</tr>
<tr>
<td>NAU</td>
</tr>
<tr>
<td>NAV</td>
</tr>
<tr>
<td>NDE</td>
</tr>
<tr>
<td>NBL</td>
</tr>
<tr>
<td>NDO</td>
</tr>
<tr>
<td>NEP</td>
</tr>
<tr>
<td>SME</td>
</tr>
<tr>
<td>NOR</td>
</tr>
<tr>
<td>NOB</td>
</tr>
<tr>
<td>NNO</td>
</tr>
<tr>
<td>OCI</td>
</tr>
<tr>
<td>OJI</td>
</tr>
<tr>
<td>ORI</td>
</tr>
<tr>
<td>ORM</td>
</tr>
<tr>
<td>OSS</td>
</tr>
<tr>
<td>PLI</td>
</tr>
<tr>
<td>FAS</td>
</tr>
<tr>
<td>POL</td>
</tr>
<tr>
<td>PUS</td>
</tr>
<tr>
<td>QUE</td>
</tr>
<tr>
<td>QAA</td>
</tr>
<tr>
<td>RON</td>
</tr>
<tr>
<td>ROH</td>
</tr>
<tr>
<td>RUN</td>
</tr>
<tr>
<td>SMO</td>
</tr>
<tr>
<td>SAG</td>
</tr>
<tr>
<td>SAN</td>
</tr>
<tr>
<td>SRD</td>
</tr>
<tr>
<td>SRB</td>
</tr>
<tr>
<td>SNA</td>
</tr>
<tr>
<td>III</td>
</tr>
<tr>
<td>SND</td>
</tr>
<tr>
<td>SIN</td>
</tr>
<tr>
<td>SLK</td>
</tr>
<tr>
<td>SLV</td>
</tr>
<tr>
<td>SOM</td>
</tr>
<tr>
<td>SOT</td>
</tr>
</tbody>
</table>
ListPresetsRequest

nextToken

Use this string, provided with the response to a previous request, to request the next batch of presets.

Type: string
Required: False

maxResults

Optional. Number of presets, up to twenty, that will be returned at one time

Type: integer
Required: False
Format: int32

category
Optionally, specify a preset category to limit responses to only presets from that category.

Type: string
Required: False

listBy

Type: PresetListBy (p. 687)
Required: False

order

Type: Order (p. 685)
Required: False

ListPresetsResponse

presets

Type: Array of type Preset (p. 686)
Required: False

nextToken

Use this string to request the next batch of presets.

Type: string
Required: False

M2tsAudioBufferModel (Enum)
Selects between the DVB and ATSC buffer models for Dolby Digital audio.

DVB
ATSC

M2tsBufferModel (Enum)
Controls what buffer model to use for accurate interleaving. If set to MULTIPLEX, use multiplex buffer model. If set to NONE, this can lead to lower latency, but low-memory devices may not be able to play back the stream without interruptions.

MULTIPLEX
NONE

M2tsEbpAudioInterval (Enum)
When set to VIDEO_AND_FIXED_INTERVALS, audio EBP markers will be added to partitions 3 and 4. The interval between these additional markers will be fixed, and will be slightly shorter than the video
EBP marker interval. When set to VIDEO_INTERVAL, these additional markers will not be inserted. Only applicable when EBP segmentation markers are is selected (segmentationMarkers is EBP or EBP_LEGACY).

- VIDEO_AND_FIXED_INTERVALS
- VIDEO_INTERVAL

**M2tsEbpPlacement (Enum)**

Selects which PIDs to place EBP markers on. They can either be placed only on the video PID, or on both the video PID and all audio PIDs. Only applicable when EBP segmentation markers are is selected (segmentationMarkers is EBP or EBP_LEGACY).

- VIDEO_AND_AUDIO_PIDS
- VIDEO_PID

**M2tsEsRateInPes (Enum)**

Controls whether to include the ES Rate field in the PES header.

- INCLUDE
- EXCLUDE

**M2tsPcrControl (Enum)**

When set to PCR_EVERY_PES_PACKET, a Program Clock Reference value is inserted for every Packetized Elementary Stream (PES) header. This is effective only when the PCR PID is the same as the video or audio elementary stream.

- PCR_EVERY_PES_PACKET
- CONFIGURED_PCR_PERIOD

**M2tsRateMode (Enum)**

When set to CBR, inserts null packets into transport stream to fill specified bitrate. When set to VBR, the bitrate setting acts as the maximum bitrate, but the output will not be padded up to that bitrate.

- VBR
- CBR

**M2tsScte35Source (Enum)**

Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output. This is only available for certain containers.

- PASSTHROUGH
- NONE

**M2tsSegmentationMarkers (Enum)**

Inserts segmentation markers at each segmentation_time period. rai_segstart sets the Random Access Indicator bit in the adaptation field. rai_adapt sets the RAI bit and adds the current timecode in the private data bytes. psi_segstart inserts PAT and PMT tables at the start of segments. ebp adds Encoder Boundary Point information to the adaptation field as per OpenCable specification OC-SP-EBP-
ebp_legacy adds Encoder Boundary Point information to the adaptation field using a legacy proprietary format.

NONE
RAI_SEGSTART
RAI_ADAPT
PSI_SEGSTART
EBP
EBP_LEGACY

**M2tsSegmentationStyle (Enum)**

The segmentation style parameter controls how segmentation markers are inserted into the transport stream. With avails, it is possible that segments may be truncated, which can influence where future segmentation markers are inserted. When a segmentation style of "reset_cadence" is selected and a segment is truncated due to an avail, we will reset the segmentation cadence. This means the subsequent segment will have a duration of $segmentation_time$ seconds. When a segmentation style of "maintain_cadence" is selected and a segment is truncated due to an avail, we will not reset the segmentation cadence. This means the subsequent segment will likely be truncated as well. However, all segments after that will have a duration of $segmentation_time$ seconds. Note that EBP lookahead is a slight exception to this rule.

MAINTAIN_CADENCE
RESET_CADENCE

**M2tsSettings**

**dvbTeletextPid**

Packet Identifier (PID) for input source DVB Teletext data to this output. Can be entered as a decimal or hexadecimal value.

*Type: integer*

*Required: False*

*Minimum: 32*

*Maximum: 8182*

**bitrate**

The output bitrate of the transport stream in bits per second. Setting to 0 lets the muxer automatically determine the appropriate bitrate. Other common values are 3750000, 7500000, and 15000000.

*Type: integer*

*Required: False*

*Minimum: 0*

*Maximum: 2147483647*

**segmentationTime**

The length in seconds of each segment. Required unless markers is set to _none_.

*Type: number*

*Required: False*

*Format: float*
Minimum: 0.0

**audioPids**
Packet Identifier (PID) of the elementary audio stream(s) in the transport stream. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values.

  Type: Array of type integer  
  Required: False

**rateMode**

  Type: M2tsRateMode (p. 666)  
  Required: False

**ebpAudioInterval**

  Type: M2tsEbpAudioInterval (p. 665)  
  Required: False

**fragmentTime**
The length in seconds of each fragment. Only used with EBP markers.

  Type: number  
  Required: False  
  Format: float  
  Minimum: 0.0

**audioFramesPerPes**
The number of audio frames to insert for each PES packet.

  Type: integer  
  Required: False  
  Minimum: 0  
  Maximum: 2147483647

**maxPcrInterval**
Maximum time in milliseconds between Program Clock References (PCRs) inserted into the transport stream.

  Type: integer  
  Required: False  
  Minimum: 0  
  Maximum: 500

**scte35Pid**
Packet Identifier (PID) of the SCTE-35 stream in the transport stream. Can be entered as a decimal or hexadecimal value.
properties

privateMetadataPid
Packet Identifier (PID) of the private metadata stream in the transport stream. Can be entered as a
decimal or hexadecimal value.

Type: integer
    Required: False
    Minimum: 32
    Maximum: 8182

pmtInterval
The number of milliseconds between instances of this table in the output transport stream.

Type: integer
    Required: False
    Minimum: 0
    Maximum: 1000

segmentationStyle
    Type: M2tsSegmentationStyle (p. 667)
    Required: False

audioBufferModel
    Type: M2tsAudioBufferModel (p. 665)
    Required: False

programNumber
The value of the program number field in the Program Map Table.

    Type: integer
    Required: False
    Minimum: 0
    Maximum: 65535

dvbNitSettings
    Type: DvbNitSettings (p. 628)
    Required: False

scte35Source
    Type: M2tsScte35Source (p. 666)
    Required: False
pmtPid

Packet Identifier (PID) for the Program Map Table (PMT) in the transport stream. Can be entered as a decimal or hexadecimal value.

Type: integer
Required: False
Minimum: 32
Maximum: 8182

bufferModel

Type: M2tsBufferModel (p. 665)
Required: False

ebpPlacement

Type: M2tsEbpPlacement (p. 666)
Required: False

dvbSdtSettings

Type: DvbSdtSettings (p. 628)
Required: False

nullPacketBitrate

Value in bits per second of extra null packets to insert into the transport stream. This can be used if a downstream encryption system requires periodic null packets.

Type: number
Required: False
Format: float
Minimum: 0.0

pcrPid

Packet Identifier (PID) of the Program Clock Reference (PCR) in the transport stream. When no value is given, the encoder will assign the same value as the Video PID. Can be entered as a decimal or hexadecimal value.

Type: integer
Required: False
Minimum: 32
Maximum: 8182

minEbpInterval

When set, enforces that Encoder Boundary Points do not come within the specified time interval of each other by looking ahead at input video. If another EBP is going to come in within the specified time interval, the current EBP is not emitted, and the segment is "stretched" to the next marker. The lookahead value does not add latency to the system. The Live Event must be configured elsewhere to create sufficient latency to make the lookahead accurate.
**Properties**

**Type**: integer  
**Required**: False  
**Minimum**: 0  
**Maximum**: 10000

**transportStreamId**

The value of the transport stream ID field in the Program Map Table.

**Type**: integer  
**Required**: False  
**Minimum**: 0  
**Maximum**: 65535

**pcrControl**

**Type**: M2tsPcrControl (p. 666)  
**Required**: False

**videoPid**

Packet Identifier (PID) of the elementary video stream in the transport stream. Can be entered as a decimal or hexadecimal value.

**Type**: integer  
**Required**: False  
**Minimum**: 32  
**Maximum**: 8182

**esRateInPes**

**Type**: M2tsEsRateInPes (p. 666)  
**Required**: False

**segmentationMarkers**

**Type**: M2tsSegmentationMarkers (p. 666)  
**Required**: False

**dvbTdtSettings**

**Type**: DvbTdtSettings (p. 633)  
**Required**: False

**patInterval**

The number of milliseconds between instances of this table in the output transport stream.

**Type**: integer  
**Required**: False  
**Minimum**: 0  
**Maximum**: 1000
**Properties**

**dvbSubPids**
Packet Identifier (PID) for input source DVB Subtitle data to this output. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values.

- **Type:** Array of type integer
- **Required:** False

**M3u8PcrControl (Enum)**
When set to PCR_EVERY_PES_PACKET a Program Clock Reference value is inserted for every Packetized Elementary Stream (PES) header. This parameter is effective only when the PCR PID is the same as the video or audio elementary stream.

- **PCR_EVERY_PES_PACKET**
- **CONFIGURED_PCR_PERIOD**

**M3u8Scte35Source (Enum)**
Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output. This is only available for certain containers.

- **PASSTHROUGH**
- **NONE**

**M3u8Settings**

**pmtPid**
Packet Identifier (PID) for the Program Map Table (PMT) in the transport stream. Can be entered as a decimal or hexadecimal value.

- **Type:** integer
- **Required:** False
- **Minimum:** 32
- **Maximum:** 8182

**pcrPid**
Packet Identifier (PID) of the Program Clock Reference (PCR) in the transport stream. When no value is given, the encoder will assign the same value as the Video PID. Can be entered as a decimal or hexadecimal value.

- **Type:** integer
- **Required:** False
- **Minimum:** 32
- **Maximum:** 8182

**audioPids**
Packet Identifier (PID) of the elementary audio stream(s) in the transport stream. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values.
Properties

**Type**: Array of type integer

**Required**: False

### audioFramesPerPes

The number of audio frames to insert for each PES packet.

**Type**: integer

**Required**: False

**Minimum**: 0

**Maximum**: 2147483647

### scte35Pid

Packet Identifier (PID) of the SCTE-35 stream in the transport stream. Can be entered as a decimal or hexadecimal value.

**Type**: integer

**Required**: False

**Minimum**: 32

**Maximum**: 8182

### transportStreamId

The value of the transport stream ID field in the Program Map Table.

**Type**: integer

**Required**: False

**Minimum**: 0

**Maximum**: 65535

### videoPid

Packet Identifier (PID) of the elementary video stream in the transport stream. Can be entered as a decimal or hexadecimal value.

**Type**: integer

**Required**: False

**Minimum**: 32

**Maximum**: 8182

### pcrControl

**Type**: M3u8PcrControl (p. 672)

**Required**: False

### privateMetadataPid

Packet Identifier (PID) of the private metadata stream in the transport stream. Can be entered as a decimal or hexadecimal value.

**Type**: integer

**Required**: False
Minimum: 32
Maximum: 8182

**pmtInterval**

The number of milliseconds between instances of this table in the output transport stream.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 1000

**patInterval**

The number of milliseconds between instances of this table in the output transport stream.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 1000

**programNumber**

The value of the program number field in the Program Map Table.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 65535

**timedMetadataPid**

Packet Identifier (PID) of the timed metadata stream in the transport stream. Can be entered as a decimal or hexadecimal value.

- **Type**: integer
- **Required**: False
- **Minimum**: 32
- **Maximum**: 8182

**timedMetadata**

- **Type**: [TimedMetadata](p. 693)
- **Required**: False

**scte35Source**

- **Type**: [M3u8Scte35Source](p. 672)
- **Required**: False

**MovClapAtom (Enum)**

When enabled, include 'clap' atom if appropriate for the video output settings.
**MovCslgAtom (Enum)**

When enabled, file composition times will start at zero, composition times in the 'ctts' (composition time to sample) box for B-frames will be negative, and a 'cslg' (composition shift least greatest) box will be included per 14496-1 amendment 1. This improves compatibility with Apple players and tools.

INCLUDE
EXCLUDE

**MovMpeg2FourCCControl (Enum)**

When set to XDCAM, writes MPEG2 video streams into the QuickTime file using XDCAM fourcc codes. This increases compatibility with Apple editors and players, but may decrease compatibility with other players. Only applicable when the video codec is MPEG2.

XDCAM
MPEG

**MovPaddingControl (Enum)**

If set to OMNEON, inserts Omneon-compatible padding

OMNEON
NONE

**MovReference (Enum)**

A value of 'external' creates separate media files and the wrapper file (.mov) contains references to these media files. A value of 'self_contained' creates only a wrapper (.mov) file and this file contains all of the media.

SELF_CONTAINED
EXTERNAL

**MovSettings**

**reference**

*Type: MovReference (p. 675)*  
*Required: False*

**paddingControl**

*Type: MovPaddingControl (p. 675)*  
*Required: False*

**mpeg2FourCCControl**

*Type: MovMpeg2FourCCControl (p. 675)*  
*Required: False*
**cslgAtom**

*Type:* MovCslgAtom (p. 675)

*Required:* False

**clapAtom**

*Type:* MovClapAtom (p. 674)

*Required:* False

**Mp2Settings**

**channels**

Set Channels to specify the number of channels in this output audio track. Choosing Mono in the console will give you 1 output channel; choosing Stereo will give you 2. In the API, valid values are 1 and 2.

*Type:* integer  
*Required:* False  
*Minimum:* 1  
*Maximum:* 2

**bitrate**

Average bitrate in bits/second.

*Type:* integer  
*Required:* False  
*Minimum:* 32000  
*Maximum:* 384000

**sampleRate**

Sample rate in hz.

*Type:* integer  
*Required:* False  
*Minimum:* 32000  
*Maximum:* 48000

**Mp4CslgAtom (Enum)**

When enabled, file composition times will start at zero, composition times in the 'ctts' (composition time to sample) box for B-frames will be negative, and a 'cslg' (composition shift least greatest) box will be included per 14496-1 amendment 1. This improves compatibility with Apple players and tools.

*INCLUDE*  
*EXCLUDE*

**Mp4FreeSpaceBox (Enum)**

Inserts a free-space box immediately after the moov box.

*INCLUDE*  
*EXCLUDE*
**Mp4MoovPlacement (Enum)**

If set to PROGRESSIVE_DOWNLOAD, the MOOV atom is relocated to the beginning of the archive as required for progressive downloading. Otherwise it is placed normally at the end.

- PROGRESSIVE_DOWNLOAD
- NORMAL

**Mp4Settings**

**mp4MajorBrand**

Overrides the "Major Brand" field in the output file. Usually not necessary to specify.

- **Type**: string
- **Required**: False

**moovPlacement**

- **Type**: Mp4MoovPlacement (p. 677)
- **Required**: False

**cslgAtom**

- **Type**: Mp4CslgAtom (p. 676)
- **Required**: False

**freeSpaceBox**

- **Type**: Mp4FreeSpaceBox (p. 676)
- **Required**: False

**Mpeg2AdaptiveQuantization (Enum)**

Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

- OFF
- LOW
- MEDIUM
- HIGH

**Mpeg2CodecLevel (Enum)**

Use Level (Mpeg2CodecLevel) to set the MPEG-2 level for the video output.

- AUTO
- LOW
- MAIN
- HIGH1440
- HIGH

**Mpeg2CodecProfile (Enum)**

Use Profile (Mpeg2CodecProfile) to set the MPEG-2 profile for the video output.
MAIN
PROFILE_422

**Mpeg2FramerateControl (Enum)**

Using the API, set FramerateControl to INITIALIZE_FROM_SOURCE if you want the service to use the framerate from the input. Using the console, do this by choosing INITIALIZE_FROM_SOURCE for Framerate.

INITIALIZE_FROM_SOURCE
SPECIFIED

**Mpeg2FramerateConversionAlgorithm (Enum)**

When set to INTERPOLATE, produces smoother motion during framerate conversion.

DUPLICATE_DROP
INTERPOLATE

**Mpeg2GopSizeUnits (Enum)**

Indicates if the GOP Size in MPEG2 is specified in frames or seconds. If seconds the system will convert the GOP Size into a frame count at run time.

FRAMES
SECONDS

**Mpeg2InterlaceMode (Enum)**

Use Interlace mode (InterlaceMode) to choose the scan line type for the output.

* Top Field First (TOP_FIELD) and Bottom Field First (BOTTOM_FIELD) produce interlaced output with the entire output having the same field polarity (top or bottom first).

* Follow, Default Top (FOLLOW_TOP_FIELD) and Follow, Default Bottom (FOLLOW_BOTTOM_FIELD) use the same field polarity as the source. Therefore, behavior depends on the input scan type. If the source is interlaced, the output will be interlaced with the same polarity as the source (it will follow the source). The output could therefore be a mix of "top field first" and "bottom field first". If the source is progressive, the output will be interlaced with "top field first" or "bottom field first" polarity, depending on which of the Follow options you chose.

PROGRESSIVE
TOP_FIELD
BOTTOM_FIELD
FOLLOW_TOP_FIELD
FOLLOW_BOTTOM_FIELD

**Mpeg2IntraDcPrecision (Enum)**

Use Intra DC precision (Mpeg2IntraDcPrecision) to set quantization precision for intra-block DC coefficients. If you choose the value auto, the service will automatically select the precision based on the per-frame compression ratio.

AUTO
INTRA_DC_PRECISION_8
INTRA_DC_PRECISION_9
INTRA_DC_PRECISION_10
INTRA_DC_PRECISION_11

Mpeg2ParControl (Enum)
Using the API, enable ParFollowSource if you want the service to use the pixel aspect ratio from the input. Using the console, do this by choosing Follow source for Pixel aspect ratio.

INITIALIZE_FROM_SOURCE
SPECIFIED

Mpeg2QualityTuningLevel (Enum)
Use Quality tuning level (Mpeg2QualityTuningLevel) to specify whether to use single-pass or multipass video encoding.

SINGLE_PASS
MULTI_PASS

Mpeg2RateControlMode (Enum)
Use Rate control mode (Mpeg2RateControlMode) to specify whether the bitrate is variable (vbr) or constant (cbr).

VBR
CBR

Mpeg2SceneChangeDetect (Enum)
Scene change detection (inserts I-frames on scene changes).

DISABLED
ENABLED

Mpeg2Settings

minIInterval
Enforces separation between repeated (cadence) I-frames and I-frames inserted by Scene Change Detection. If a scene change I-frame is within I-interval frames of a cadence I-frame, the GOP is shrunk and/or stretched to the scene change I-frame. GOP stretch requires enabling lookahead as well as setting I-interval. The normal cadence resumes for the next GOP. This setting is only used when Scene Change Detect is enabled. Note: Maximum GOP stretch = GOP size + Min-I-interval - 1

Type: integer
Required: False
Minimum: 0
Maximum: 30

parNumerator
Pixel Aspect Ratio numerator.

Type: integer
AWS Elemental MediaConvert API Reference
Properties

**Required**: False
**Minimum**: 1
**Maximum**: 2147483647

**gopSizeUnits**

**Type**: Mpeg2GopSizeUnits (p. 678)
**Required**: False

**hrdBufferSize**

Size of buffer (HRD buffer model). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.

**Type**: integer
**Required**: False
**Minimum**: -2147483648
**Maximum**: 2147483647

**qualityTuningLevel**

**Type**: Mpeg2QualityTuningLevel (p. 679)
**Required**: False

**maxBitrate**

Maximum bitrate in bits/second (for VBR mode only). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.

**Type**: integer
**Required**: False
**Minimum**: -2147483648
**Maximum**: 2147483647

**bitrate**

Average bitrate in bits/second. Required for VBR, CBR, and ABR. Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.

**Type**: integer
**Required**: False
**Minimum**: 1000
**Maximum**: 2147483647

**spatialAdaptiveQuantization**

**Type**: Mpeg2SpatialAdaptiveQuantization (p. 683)
**Required**: False

**slowPal**

**Type**: Mpeg2SlowPal (p. 683)
**Required**: False
**Properties**

**codecProfile**
Type: Mpeg2CodecProfile (p. 677)  
Required: False

**intraDcPrecision**
Type: Mpeg2IntraDcPrecision (p. 678)  
Required: False

**softness**
Softness. Selects quantizer matrix, larger values reduce high-frequency content in the encoded image.
Type: integer  
Required: False  
Minimum: 0  
Maximum: 128

**framerateControl**
Type: Mpeg2FramerateControl (p. 678)  
Required: False

**telecine**
Type: Mpeg2Telecine (p. 684)  
Required: False

**framerateConversionAlgorithm**
Type: Mpeg2FramerateConversionAlgorithm (p. 678)  
Required: False

**codecLevel**
Type: Mpeg2CodecLevel (p. 677)  
Required: False

**temporalAdaptiveQuantization**
Type: Mpeg2TemporalAdaptiveQuantization (p. 684)  
Required: False

**hrdBufferInitialFillPercentage**
Percentage of the buffer that should initially be filled (HRD buffer model).
Type: integer  
Required: False  
Minimum: 0  
Maximum: 100
framerateNumerator

Framerate numerator - framerate is a fraction, e.g. 24000 / 1001 = 23.976 fps.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

numberBFramesBetweenReferenceFrames

Number of B-frames between reference frames.

Type: integer
Required: False
Minimum: 0
Maximum: 7

gopClosedCadence

Frequency of closed GOPs. In streaming applications, it is recommended that this be set to 1 so a decoder joining mid-stream will receive an IDR frame as quickly as possible. Setting this value to 0 will break output segmenting.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

framerateDenominator

Framerate denominator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

adaptiveQuantization

Type: Mpeg2AdaptiveQuantization (p. 677)
Required: False

interlaceMode

Type: Mpeg2InterlaceMode (p. 678)
Required: False

gopSize

GOP Length (keyframe interval) in frames or seconds. Must be greater than zero.

Type: number
Required: False
**Format**: float  
**Minimum**: 0.0

**sceneChangeDetect**

**Type**: Mpeg2SceneChangeDetect (p. 679)  
**Required**: False

**parDenominator**

Pixel Aspect Ratio denominator.

**Type**: integer  
**Required**: False  
**Minimum**: 1  
**Maximum**: 2147483647

**parControl**

**Type**: Mpeg2ParControl (p. 679)  
**Required**: False

**syntax**

**Type**: Mpeg2Syntax (p. 683)  
**Required**: False

**rateControlMode**

**Type**: Mpeg2RateControlMode (p. 679)  
**Required**: False

**Mpeg2SlowPal (Enum)**

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

- DISABLED
- ENABLED

**Mpeg2SpatialAdaptiveQuantization (Enum)**

Adjust quantization within each frame based on spatial variation of content complexity.

- DISABLED
- ENABLED

**Mpeg2Syntax (Enum)**

Produces a Type D-10 compatible bitstream (SMPTE 356M-2001).

- DEFAULT
- D_10
Mpeg2Telecine (Enum)

Only use Telecine (Mpeg2Telecine) when you set Framerate (Framerate) to 29.970. Set Telecine (Mpeg2Telecine) to Hard (hard) to produce a 29.97i output from a 23.976 input. Set it to Soft (soft) to produce 23.976 output and leave conversion to the player.

NONE
SOFT
HARD

Mpeg2TemporalAdaptiveQuantization (Enum)

Adjust quantization within each frame based on temporal variation of content complexity.

DISABLED
ENABLED

NoiseReducer

filter

Type: NoiseReducerFilter (p. 684)
Required: True

filterSettings

Type: NoiseReducerFilterSettings (p. 685)
Required: False

spatialFilterSettings

Type: NoiseReducerSpatialFilterSettings (p. 685)
Required: False

NoiseReducerFilter (Enum)

Use Noise reducer filter (NoiseReducerFilter) to select one of the following spatial image filtering functions. To use this setting, you must also enable Noise reducer (NoiseReducer).

* Bilateral is an edge preserving noise reduction filter
* Mean (softest), Gaussian, Lanczos, and Sharpen (sharpest) are convolution filters
* Conserve is a min/max noise reduction filter
* Spatial is frequency-domain filter based on JND principles.

BILATERAL
MEAN
GAUSSIAN
LANCZOS
SHARPEN
CONSERVE
SPATIAL
**NoiseReductorFilterSettings**

**strength**
Relative strength of noise reducing filter. Higher values produce stronger filtering.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 3

**NoiseReductorSpatialFilterSettings**

**strength**
Relative strength of noise reducing filter. Higher values produce stronger filtering.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 16

**postFilterSharpenStrength**
Specify strength of post noise reduction sharpening filter, with 0 disabling the filter and 3 enabling it at maximum strength.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 3

**speed**
The speed of the filter, from -2 (lower speed) to 3 (higher speed), with 0 being the nominal value.

- **Type**: integer
- **Required**: False
- **Minimum**: -2
- **Maximum**: 3

**Order (Enum)**
When you request lists of resources, you can optionally specify whether they are sorted in ASCENDING or DESCENDING order. Default varies by resource.

- ASCENDING
- DESCENDING

**OutputChannelMapping**

**inputChannels**

- **Type**: Array of type integer
- **Required**: True
OutputSdt (Enum)

Selects method of inserting SDT information into output stream. "Follow input SDT" copies SDT information from input stream to output stream. "Follow input SDT if present" copies SDT information from input stream to output stream if SDT information is present in the input, otherwise it will fall back on the user-defined values. Enter "SDT Manually" means user will enter the SDT information. "No SDT" means output stream will not contain SDT information.

- SDT_FOLLOW
- SDT_FOLLOW_IF_PRESENT
- SDT_MANUAL
- SDT_NONE

Preset

settings

- Type: PresetSettings (p. 687)
- Required: True

lastUpdated

The timestamp in epoch seconds when the preset was last updated.

- Type: string
- Required: False
- Format: date-time

createdAt

The timestamp in epoch seconds for preset creation.

- Type: string
- Required: False
- Format: date-time

name

A name you create for each preset. Each name must be unique within your account.

- Type: string
- Required: True

description

An optional description you create for each preset.

- Type: string
- Required: False

category

An optional category you create to organize your presets.
Properties

**Type**
Type: string  
Required: False

**type**
Type: Type (p. 694)  
Required: False

**arn**
An identifier for this resource that is unique within all of AWS.
Type: string  
Required: False

**PresetListBy (Enum)**
Optional. When you request a list of presets, you can choose to list them alphabetically by NAME or chronologically by CREATION_DATE. If you don’t specify, the service will list them by name.

NAME  
CREATION_DATE  
SYSTEM

**PresetSettings**

**videoDescription**
(VideoDescription) contains a group of video encoding settings. The specific video settings depend on the video codec you choose when you specify a value for Video codec (codec). Include one instance of (VideoDescription) per output.

Type: VideoDescription (p. 695)  
Required: False

**audioDescriptions**
(AudioDescriptions) contains groups of audio encoding settings organized by audio codec. Include one instance of (AudioDescriptions) per output. (AudioDescriptions) can contain multiple groups of encoding settings.

Type: Array of type AudioDescription (p. 615)  
Required: False

**containerSettings**

Type: ContainerSettings (p. 625)  
Required: False

**captionDescriptions**
Caption settings for this preset. There can be multiple caption settings in a single output.
Type: Array of type CaptionDescriptionPreset (p. 622)
Required: False

ProresCodecProfile (Enum)
Use Profile (ProResCodecProfile) to specify the type of Apple ProRes codec to use for this output.

APPLE_PRORES_422
APPLE_PRORES_422_HQ
APPLE_PRORES_422_LT
APPLE_PRORES_422_PROXY

ProresFramerateControl (Enum)
Using the API, set FramerateControl to INITIALIZE_FROM_SOURCE if you want the service to use the framerate from the input. Using the console, do this by choosing INITIALIZE_FROM_SOURCE for Framerate.

INITIALIZE_FROM_SOURCE
SPECIFIED

ProresFramerateConversionAlgorithm (Enum)
When set to INTERPOLATE, produces smoother motion during framerate conversion.

DUPLICATE_DROP
INTERPOLATE

ProresInterlaceMode (Enum)
Use Interlace mode (InterlaceMode) to choose the scan line type for the output.

* Top Field First (TOP_FIELD) and Bottom Field First (BOTTOM_FIELD) produce interlaced output with the entire output having the same field polarity (top or bottom first).

* Follow, Default Top (FOLLOW_TOP_FIELD) and Follow, Default Bottom (FOLLOW_BOTTOM_FIELD) use the same field polarity as the source. Therefore, behavior depends on the input scan type. If the source is interlaced, the output will be interlaced with the same polarity as the source (it will follow the source). The output could therefore be a mix of "top field first" and "bottom field first". If the source is progressive, the output will be interlaced with "top field first" or "bottom field first" polarity, depending on which of the Follow options you chose.

PROGRESSIVE
TOP_FIELD
BOTTOM_FIELD
FOLLOW_TOP_FIELD
FOLLOW_BOTTOM_FIELD

ProresParControl (Enum)
Use (ProresParControl) to specify how the service determines the pixel aspect ratio. Set to Follow source (INITIALIZE_FROM_SOURCE) to use the pixel aspect ratio from the input. To specify a different pixel aspect ratio: Using the console, choose it from the dropdown menu. Using the API, set ProresParControl to (SPECIFIED) and provide for (ParNumerator) and (ParDenominator).
Properties

**INITIALIZE_FROM_SOURCE**
SPECIFIED

**ProresSettings**

**slowPal**

Type: ProresSlowPal (p. 690)
Required: False

**framerateControl**

Type: ProresFramerateControl (p. 688)
Required: False

**telecine**

Type: ProresTelecine (p. 690)
Required: False

**framerateDenominator**

Framerate denominator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

**framerateConversionAlgorithm**

Type: ProresFramerateConversionAlgorithm (p. 688)
Required: False

**interlaceMode**

Type: ProresInterlaceMode (p. 688)
Required: False

**codecProfile**

Type: ProresCodecProfile (p. 688)
Required: False

**parNumerator**

Pixel Aspect Ratio numerator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

parControl

Type: ProresParControl (p. 688)
Required: False

parDenominator

Pixel Aspect Ratio denominator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

framerateNumerator

When you use the API for transcode jobs that use framerate conversion, specify the framerate as a fraction. For example, 24000 / 1001 = 23.976 fps. Use FramerateNumerator to specify the numerator of this fraction. In this example, use 24000 for the value of FramerateNumerator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

ProresSlowPal (Enum)

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

DISABLED
ENABLED

ProresTelecine (Enum)

Only use Telecine (ProresTelecine) when you set Framerate (Framerate) to 29.970. Set Telecine (ProresTelecine) to Hard (hard) to produce a 29.97i output from a 23.976 input. Set it to Soft (soft) to produce 23.976 output and leave conversion to the player.

NONE
HARD

Rectangle

width

Width of rectangle in pixels.

Type: integer
Required: True
Minimum: -2147483648
Maximum: 2147483647
x
The distance, in pixels, between the rectangle and the left edge of the video frame.

- **Type:** integer
- **Required:** True
- **Minimum:** -2147483648
- **Maximum:** 2147483647

y
The distance, in pixels, between the rectangle and the top edge of the video frame.

- **Type:** integer
- **Required:** True
- **Minimum:** -2147483648
- **Maximum:** 2147483647

**height**
Height of rectangle in pixels.

- **Type:** integer
- **Required:** True
- **Minimum:** -2147483648
- **Maximum:** 2147483647

**RemixSettings**

**channelsOut**
Specify the number of channels in this output after remixing. Valid values: 1, 2, 4, 6, 8

- **Type:** integer
- **Required:** True
- **Minimum:** 1
- **Maximum:** 8

**channelMapping**

- **Type:** ChannelMapping (p. 623)
- **Required:** True

**channelsIn**
Specify the number of audio channels from your input that you want to use in your output. With remixing, you might combine or split the data in these channels, so the number of channels in your final output might be different.

- **Type:** integer
- **Required:** True
- **Minimum:** 1
- **Maximum:** 16
**RespondToAfd (Enum)**

Use Respond to AFD (RespondToAfd) to specify how the service changes the video itself in response to AFD values in the input.

* Choose Respond to clip the input video frame according to the AFD value, input display aspect ratio, and output display aspect ratio.

* Choose Passthrough to include the input AFD values. Do not choose this when AfdSignaling is set to (NONE). A preferred implementation of this workflow is to set RespondToAfd to (NONE) and set AfdSignaling to (AUTO).

* Choose None to remove all input AFD values from this output.

   NONE
   RESPOND
   PASSTHROUGH

**ScalingBehavior (Enum)**

Applies only if your input aspect ratio is different from your output aspect ratio. Enable Stretch to output (StretchToOutput) to have the service stretch your video image to fit. Leave this setting disabled to allow the service to letterbox your video instead. This setting overrides any positioning value you specify elsewhere in the job.

   DEFAULT
   STRETCH_TO_OUTPUT

**SccDestinationFramerate (Enum)**

Set Framerate (SccDestinationFramerate) to make sure that the captions and the video are synchronized in the output. Specify a framerate that matches the framerate of the associated video. If the video framerate is 29.97, choose 29.97 dropframe (FRAMERATE_29_97_DROPFRAME) only if the video has video_insertion=true and drop_frame_timecode=true; otherwise, choose 29.97 non-dropframe (FRAMERATE_29_97_NON_DROPFRAME).

   FRAMERATE_23_97
   FRAMERATE_24
   FRAMERATE_29_97_DROPFRAME
   FRAMERATE_29_97_NON_DROPFRAME

**SccDestinationSettings**

*framerate*

**Type:** SccDestinationFramerate (p. 692)

**Required:** False

**TeletextDestinationSettings**

*pageNumber*

Set pageNumber to the Teletext page number for the destination captions for this output. This value must be a three-digit hexadecimal string; strings ending in -FF are invalid. If you are passing through the entire set of Teletext data, do not use this field.
Properties

Type: string
Required: False
Pattern: ^[1-8][0-9a-fA-F][0-9a-eA-E]$ 

**TimecodeBurnin**

**prefix**

Use Prefix (Prefix) to place ASCII characters before any burned-in timecode. For example, a prefix of "EZ-" will result in the timecode "EZ-00:00:00:00". Provide either the characters themselves or the ASCII code equivalents. The supported range of characters is 0x20 through 0x7e. This includes letters, numbers, and all special characters represented on a standard English keyboard.

Type: string
Required: False
Pattern: ^[ -~]+$ 

**fontSize**

Use Font Size (FontSize) to set the font size of any burned-in timecode. Valid values are 10, 16, 32, 48.

Type: integer
Required: False
Minimum: 10
Maximum: 48

**position**

Type: \[TimecodeBurninPosition (p. 693)\]
Required: False

**TimecodeBurninPosition (Enum)**

Use Position (Position) under Timecode burn-in (TimecodeBurnIn) to specify the location the burned-in timecode on output video.

- TOP_CENTER
- TOP_LEFT
- TOP_RIGHT
- MIDDLE_LEFT
- MIDDLE_CENTER
- MIDDLE_RIGHT
- BOTTOM_LEFT
- BOTTOM_CENTER
- BOTTOM_RIGHT

**TimedMetadata (Enum)**

If PASSTHROUGH, inserts ID3 timed metadata from the timed_metadata REST command into this output. Only available for certain containers.

PASSTHROUGH
TtmlDestinationSettings

stylePassthrough

Type: TtmlStylePassthrough (p. 694)
Required: False

TtmlStylePassthrough (Enum)
Pass through style and position information from a TTML-like input source (TTML, SMPTE-TT, CFF-TT) to the CFF-TT output or TTML output.

ENABLED
DISABLED

Type (Enum)

SYSTEM
CUSTOM

VideoCodec (Enum)
Type of video codec

FRAME_CAPTURE
H_264
H_265
MPEG2
PRORES

VideoCodecSettings

h265Settings

Type: H265Settings (p. 650)
Required: False

codec

Type: VideoCodec (p. 694)
Required: True

proresSettings

Type: ProresSettings (p. 689)
Required: False

mpeg2Settings

Type: Mpeg2Settings (p. 679)
**Properties**

**Required**: False

**h264Settings**

**Type**: H264Settings (p. 642)

**Required**: False

**frameCaptureSettings**

**Type**: FrameCaptureSettings (p. 638)

**Required**: False

**VideoDescription**

**fixedAfd**

Applies only if you set AFD Signaling(AfdSignaling) to Fixed (FIXED). Use Fixed (FixedAfd) to specify a four-bit AFD value which the service will write on all frames of this video output.

**Type**: integer

**Required**: False

**Minimum**: 0

**Maximum**: 15

**scalingBehavior**

**Type**: ScalingBehavior (p. 692)

**Required**: False

**respondToAfd**

**Type**: RespondToAfd (p. 692)

**Required**: False

**codecSettings**

**Type**: VideoCodecSettings (p. 694)

**Required**: True

**afdSignaling**

**Type**: AfdSignaling (p. 613)

**Required**: False

**colorMetadata**

**Type**: ColorMetadata (p. 624)

**Required**: False

**timecodeInsertion**

**Type**: VideoTimecodeInsertion (p. 698)
**Properties**

**Required**: False

**width**

Use Width (Width) to define the video resolution width, in pixels, for this output. If you don't provide a value here, the service will use the input width.

- **Type**: integer
- **Required**: False
- **Minimum**: 32
- **Maximum**: 4096

**sharpness**

Use Sharpness (Sharpness)settling to specify the strength of anti-aliasing. This setting changes the width of the anti-alias filter kernel used for scaling. Sharpness only applies if your output resolution is different from your input resolution, and if you set Anti-alias (AntiAlias) to ENABLED. 0 is the softest setting, 100 the sharpest, and 50 recommended for most content.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 100

**antiAlias**

- **Type**: AntiAlias (p. 614)
- **Required**: False

**videoPreprocessors**

Find additional transcoding features under Preprocessors (VideoPreprocessors). Enable the features at each output individually. These features are disabled by default.

- **Type**: VideoPreprocessor (p. 697)
- **Required**: False

**position**

Use Position (Position) to point to a rectangle object to define your position. This setting overrides any other aspect ratio.

- **Type**: Rectangle (p. 690)
- **Required**: False

**dropFrameTimecode**

- **Type**: DropFrameTimecode (p. 627)
- **Required**: False

**crop**

Applies only if your input aspect ratio is different from your output aspect ratio. Use Input cropping rectangle (Crop) to specify the video area the service will include in the output. This will crop the input
source, causing video pixels to be removed on encode. Do not use this setting if you have enabled Stretch to output (stretchToOutput) in your output settings.

- **Type**: Rectangle (p. 690)
- **Required**: False

### height

Use the Height (Height) setting to define the video resolution height for this output. Specify in pixels. If you don’t provide a value here, the service will use the input height.

- **Type**: integer
- **Required**: False
- **Minimum**: 32
- **Maximum**: 2160

### VideoPreprocessor

#### timecodeBurnin

Timecode burn-in (TimecodeBurnIn)—Burns the output timecode and specified prefix into the output.

- **Type**: TimecodeBurnin (p. 693)
- **Required**: False

#### noiseReducer

Enable the Noise reducer (NoiseReducer) feature to remove noise from your video output if necessary. Enable or disable this feature for each output individually. This setting is disabled by default.

- **Type**: NoiseReducer (p. 684)
- **Required**: False

#### colorCorrector

Enable the Color corrector (ColorCorrector) feature if necessary. Enable or disable this feature for each output individually. This setting is disabled by default.

- **Type**: ColorCorrector (p. 624)
- **Required**: False

#### imageInserter

Enable the Image inserter (ImageInserter) feature to include a graphic overlay on your video. Enable or disable this feature for each output individually. This setting is disabled by default.

- **Type**: ImageInserter (p. 658)
- **Required**: False

#### deinterlacer

Use Deinterlacer (Deinterlacer) to produce smoother motion and a clearer picture.
**Type:** Deinterlacer (p. 627)  
**Required:** False

**VideoTimecodeInsertion (Enum)**

Enable Timecode insertion to include timecode information in this output. Do this in the API by setting (VideoTimecodeInsertion) to (PIC_TIMING_SEI). To get timecodes to appear correctly in your output, also set up the timecode configuration for your job in the input settings. Only enable Timecode insertion when the input framerate is identical to output framerate. Disable this setting to remove the timecode from the output. Default is disabled.

- DISABLED
- PIC_TIMING_SEI

**WavSettings**

**channels**

Set Channels to specify the number of channels in this output audio track. With WAV, valid values 1, 2, 4, and 8. In the console, these values are Mono, Stereo, 4-Channel, and 8-Channel, respectively.

- **Type:** integer  
  - **Required:** False  
  - **Minimum:** 1  
  - **Maximum:** 8

**bitDepth**

Specify Bit depth (BitDepth), in bits per sample, to choose the encoding quality for this audio track.

- **Type:** integer  
  - **Required:** False  
  - **Minimum:** 16  
  - **Maximum:** 24

**sampleRate**

Sample rate in Hz.

- **Type:** integer  
  - **Required:** False  
  - **Minimum:** 8000  
  - **Maximum:** 192000

**A Specific Preset**

**URI**

/presets/name
HTTP Methods

GET

Operation ID: GetPreset

Retrieve the JSON for a specific preset.

Path Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>True</td>
<td></td>
</tr>
</tbody>
</table>

Responses

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>GetPresetResponse (p. 710)</td>
<td>200: OkRresponse</td>
</tr>
<tr>
<td>400</td>
<td>ExceptionBody (p. 724)</td>
<td>400: BadRequestException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The conditional request failed. The service can't process your request because of a problem in the request. Please check your request form and syntax.</td>
</tr>
<tr>
<td>500</td>
<td>ExceptionBody (p. 724)</td>
<td>500: InternalServiceException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The service encountered an unexpected condition and cannot fulfill your request.</td>
</tr>
<tr>
<td>403</td>
<td>ExceptionBody (p. 724)</td>
<td>403: AccessDeniedException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>You don't have permissions for this action with the credentials you sent. Please check your authorization credentials. You should be sending credentials using the AWS Signature Version 4 signing process.</td>
</tr>
<tr>
<td>404</td>
<td>ExceptionBody (p. 724)</td>
<td>404: ResourceNotFoundException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The resource you requested does not exist.</td>
</tr>
<tr>
<td>429</td>
<td>ExceptionBody (p. 724)</td>
<td>429: LimitExceededException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Too many requests have been sent in too short of a time. The service limits the rate at which it will accept requests.</td>
</tr>
</tbody>
</table>
### Status Code | Response Model | Description
---|---|---
409 | ExceptionBody (p. 724) | For example, you may be hitting your account limits for preset creation or job submission.

**409: ResourceInUseException**
The service could not complete your request because there is a conflict with the current state of the resource. For example, you may be trying to delete a Queue that has jobs processing.

## PUT

Operation ID: UpdatePreset

Modify one of your existing presets.

**Path Parameters**

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>True</td>
<td></td>
</tr>
</tbody>
</table>

**Responses**

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>UpdatePresetResponse (p. 717)</td>
<td>200: OkResponse</td>
</tr>
<tr>
<td>400</td>
<td>ExceptionBody (p. 724)</td>
<td>400: BadRequestException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The conditional request failed. The service can't process your request because of a problem in the request. Please check your request form and syntax.</td>
</tr>
<tr>
<td>500</td>
<td>ExceptionBody (p. 724)</td>
<td>500: InternalServiceException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The service encountered an unexpected condition and cannot fulfill your request.</td>
</tr>
<tr>
<td>403</td>
<td>ExceptionBody (p. 724)</td>
<td>403: AccessDeniedException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>You don't have permissions for this action with the credentials you sent. Please check your authorization credentials. You should be sending credentials using the AWS Signature Version 4 signing process.</td>
</tr>
</tbody>
</table>
DELETE

Operation ID: DeletePreset

Permanently delete a preset you have created.

Path Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>True</td>
<td></td>
</tr>
</tbody>
</table>

Responses

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
</table>
| 400         | ExceptionBody (p. 724) | 400: BadRequestException  
The conditional request failed. The service can't process your request because of a problem in the request. Please check your request form and syntax. |
| 202         | DeletePresetResponse (p. 724) | 202: AcceptedResponse  
Your request has been accepted. Processing has not yet begun. |
<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>500</td>
<td>ExceptionBody (p. 724)</td>
<td>500: InternalServiceException. The service encountered an unexpected condition and cannot fulfill your request.</td>
</tr>
<tr>
<td>403</td>
<td>ExceptionBody (p. 724)</td>
<td>403: AccessDeniedException. You don't have permissions for this action with the credentials you sent. Please check your authorization credentials. You should be sending credentials using the AWS Signature Version 4 signing process.</td>
</tr>
<tr>
<td>404</td>
<td>ExceptionBody (p. 724)</td>
<td>404: ResourceNotFoundException. The resource you requested does not exist.</td>
</tr>
<tr>
<td>429</td>
<td>ExceptionBody (p. 724)</td>
<td>429: LimitExceededException. Too many requests have been sent in too short of a time. The service limits the rate at which it will accept requests. For example, you may be hitting your account limits for preset creation or job submission.</td>
</tr>
<tr>
<td>409</td>
<td>ExceptionBody (p. 724)</td>
<td>409: ResourceInUseException. The service could not complete your request because there is a conflict with the current state of the resource. For example, you may be trying to delete a Queue that has jobs processing.</td>
</tr>
</tbody>
</table>

## Schemas

### Request Bodies

**Example GET**

```json
{
    "name (p. 753)": "string"
}
```
Example PUT

```json
{
    "settings (p. 807)": {
        "videoDescription (p. 800)": {
            "fixedAfd (p. 809)": integer,
            "scalingBehavior (p. 809)": enum,
            "respondToAfd (p. 809)": enum,
            "codecSettings (p. 809)": {
                "h265Settings (p. 808)": {
                    "slices (p. 765)": integer,
                    "minIInterval (p. 765)": integer,
                    "parNumerator (p. 765)": integer,
                    "flickerAdaptiveQuantization (p. 766)": enum,
                    "gopSizeUnits (p. 766)": enum,
                    "hrdBufferSize (p. 766)": integer,
                    "qualityTuningLevel (p. 766)": enum,
                    "maxBitrate (p. 766)": integer,
                    "bitrate (p. 766)": integer,
                    "spatialAdaptiveQuantization (p. 766)": enum,
                    "sampleAdaptiveOffsetFilterMode (p. 767)": enum,
                    "temporalIds (p. 767)": enum,
                    "slowPal (p. 767)": enum,
                    "tiles (p. 767)": enum,
                    "codecProfile (p. 767)": enum,
                    "alternateTransferFunctionSei (p. 767)": enum,
                    "unregisteredSeiTimecode (p. 767)": enum,
                    "framerateControl (p. 767)": enum,
                    "telecine (p. 767)": enum,
                    "framerateConversionAlgorithm (p. 767)": enum,
                    "codecLevel (p. 768)": enum,
                    "numberReferenceFrames (p. 768)": integer,
                    "temporalAdaptiveQuantization (p. 768)": enum,
                    "hrdBufferInitialFillPercentage (p. 768)": integer,
                    "framerateNumerator (p. 768)": integer,
                    "numberBFramesBetweenReferenceFrames (p. 768)": integer,
                    "gopClosedCadence (p. 768)": integer,
                    "framerateDenominator (p. 769)": integer,
                    "framerateConversionAlgorithm (p. 769)": enum,
                    "interlaceMode (p. 769)": enum,
                    "gopSize (p. 769)": number,
                    "gopBReference (p. 769)": enum,
                    "sceneChangeDetect (p. 769)": enum,
                    "parDenominator (p. 769)": integer,
                    "parControl (p. 770)": enum,
                    "rateControlMode (p. 770)": enum
                },
                "proresSettings (p. 808)": {
                    "slowPal (p. 802)": enum,
                    "framerateControl (p. 802)": enum,
                    "telecine (p. 802)": enum,
                    "framerateDenominator (p. 802)": integer,
                    "framerateConversionAlgorithm (p. 802)": enum,
                    "interlaceMode (p. 802)": enum,
                    "codecProfile (p. 803)": enum,
                    "parNumerator (p. 803)": integer,
                    "parControl (p. 803)": enum,
                    "parDenominator (p. 803)": integer,
                    "framerateNumerator (p. 803)": integer
                }
            },
            "mpeg2Settings (p. 808)": {
                "minIInterval (p. 793)": integer,
                "parNumerator (p. 793)": integer,
                "gopSizeUnits (p. 793)": enum
            }
        }
    }
}
```
"hrdBufferSize (p. 793)": integer,
"qualityTuningLevel (p. 793)": enum,
"maxBitrate (p. 793)": integer,
"bitrate (p. 794)": integer,
"spatialAdaptiveQuantization (p. 794)": enum,
"slowPal (p. 794)": enum,
"codecProfile (p. 794)": enum,
"intraDcPrecision (p. 794)": enum,
"softness (p. 794)": integer,
"framerateControl (p. 794)": enum,
"telecine (p. 794)": enum,
"framerateConversionAlgorithm (p. 795)": enum,
"codecLevel (p. 795)": enum,
"temporalAdaptiveQuantization (p. 795)": enum,
"hrdBufferInitialFillPercentage (p. 795)": integer,
"framerateNumerator (p. 795)": integer,
"numberBFramesBetweenReferenceFrames (p. 795)": integer,
"gopClosedCadence (p. 795)": integer,
"framerateDenominator (p. 796)": integer,
"adaptiveQuantization (p. 796)": enum,
"interlaceMode (p. 796)": enum,
"gopSize (p. 796)": number,
"sceneChangeDetect (p. 796)": enum,
"parDenominator (p. 796)": integer,
"parControl (p. 796)": enum,
"syntax (p. 796)": enum,
"rateControlMode (p. 797)": enum
},
"h264Settings (p. 808)": {
  "slices (p. 756)": integer,
  "minIInterval (p. 757)": integer,
  "parNumerator (p. 757)": integer,
  "flickerAdaptiveQuantization (p. 757)": enum,
  "gopSizeUnits (p. 757)": enum,
  "hrdBufferSize (p. 757)": integer,
  "qualityTuningLevel (p. 757)": enum,
  "maxBitrate (p. 757)": integer,
  "bitrate (p. 758)": integer,
  "spatialAdaptiveQuantization (p. 758)": enum,
  "slowPal (p. 758)": enum,
  "codecProfile (p. 758)": enum,
  "unregisteredSeiTimecode (p. 758)": enum,
  "softness (p. 758)": integer,
  "framerateControl (p. 758)": enum,
  "telecine (p. 758)": enum,
  "framerateConversionAlgorithm (p. 759)": enum,
  "codecLevel (p. 759)": enum,
  "numberReferenceFrames (p. 759)": integer,
  "temporalAdaptiveQuantization (p. 759)": enum,
  "repeatPps (p. 759)": enum,
  "hrdBufferInitialFillPercentage (p. 759)": integer,
  "framerateNumerator (p. 759)": integer,
  "numberBFramesBetweenReferenceFrames (p. 759)": integer,
  "gopClosedCadence (p. 760)": integer,
  "fieldEncoding (p. 760)": enum,
  "entropyEncoding (p. 760)": enum,
  "framerateDenominator (p. 760)": integer,
  "adaptiveQuantization (p. 760)": enum,
  "interlaceMode (p. 760)": enum,
  "gopSize (p. 760)": number,
  "gopBReference (p. 761)": enum,
  "sceneChangeDetect (p. 761)": enum,
  "parDenominator (p. 761)": integer,
  "parControl (p. 761)": enum,
  "syntax (p. 761)": enum,
  "rateControlMode (p. 761)": enum
}
"frameCaptureSettings (p. 808)": {
  "framerateDenominator (p. 752)": integer,
  "maxCaptures (p. 753)": integer,
  "framerateNumerator (p. 753)": integer,
  "quality (p. 753)": integer
},

"afdSignaling (p. 809)": enum,
"colorMetadata (p. 809)": enum,
"timecodeInsertion (p. 809)": enum,
"width (p. 809)": integer,
"sharpness (p. 810)": integer,
"antiAlias (p. 810)": enum,
"videoPreprocessors (p. 810)": {
  "timecodeBurnin (p. 811)": {
    "prefix (p. 806)": "string",
    "fontSize (p. 806)": integer,
    "position (p. 806)": enum
  },
  "noiseReducer (p. 811)": {
    "filter (p. 797)": enum,
    "filterSettings (p. 797)": {
      "strength (p. 798)": integer
    },
    "spatialFilterSettings (p. 798)": {
      "strength (p. 798)": integer,
      "postFilterSharpenStrength (p. 798)": integer,
      "speed (p. 799)": integer
    }
  },
  "colorCorrector (p. 811)": {
    "saturation (p. 738)": integer,
    "brightness (p. 739)": integer,
    "hdr10Metadata (p. 739)": {
      "redPrimaryY (p. 771)": integer,
      "greenPrimaryY (p. 771)": integer,
      "whitePointX (p. 771)": integer,
      "maxLuminance (p. 771)": integer,
      "greenPrimaryX (p. 772)": integer,
      "whitePointY (p. 772)": integer,
      "redPrimaryX (p. 772)": integer,
      "bluePrimaryX (p. 772)": integer,
      "maxFrameAverageLightLevel (p. 772)": integer,
      "bluePrimaryY (p. 772)": integer,
      "maxContentLightLevel (p. 773)": integer,
      "minLuminance (p. 773)": integer
    },
    "contrast (p. 739)": integer,
    "hue (p. 739)": integer,
    "colorSpaceConversion (p. 739)": enum
  },
  "imageInserter (p. 811)": {
    "insertableImages (p. 773)": [
      {
        "duration (p. 773)": integer,
        "fadeOut (p. 773)": integer,
        "imageY (p. 774)": integer,
        "fadeIn (p. 774)": integer,
        "imageX (p. 774)": integer,
        "width (p. 774)": integer,
        "startTime (p. 774)": "string",
        "opacity (p. 774)": integer,
        "layer (p. 775)": integer,
        "height (p. 775)": integer,
        "imageInserterInput (p. 775)": "string"
}{

},

"deinterlacer (p. 811)": {
  "mode (p. 741)": enum,
  "control (p. 741)": enum,
  "algorithm (p. 741)": enum
},

"position (p. 810)": {
  "width (p. 804)": integer,
  "x (p. 804)": integer,
  "y (p. 804)": integer,
  "height (p. 804)": integer
},

"dropFrameTimecode (p. 810)": enum,

"crop (p. 810)": {
  "width (p. 804)": integer,
  "x (p. 804)": integer,
  "y (p. 804)": integer,
  "height (p. 804)": integer
},

"height (p. 810)": integer
},

"audioDescriptions (p. 800)": [
  {
    "languageCodeControl (p. 730)": enum,
    "audioTypeControl (p. 730)": enum,
    "remixSettings (p. 730)": {
      "channelsOut (p. 804)": integer,
      "channelMapping (p. 804)": {
        "outputChannels (p. 738)": [
          {
            "inputChannels (p. 799)": [integer
          ]
        ]
      },
      "channelsIn (p. 805)": integer
    },
    "audioType (p. 731)": integer,
    "audioSourceName (p. 731)": "string",
    "codecSettings (p. 731)": {
      "codec (p. 729)": enum,
      "wavSettings (p. 730)": {
        "channels (p. 812)": integer,
        "bitDepth (p. 812)": integer,
        "sampleRate (p. 812)": integer
      },
      "ac3Settings (p. 730)": {
        "dynamicRangeCompressionProfile (p. 727)": enum,
        "dialnorm (p. 727)": integer,
        "codingMode (p. 728)": enum,
        "metadataControl (p. 728)": enum,
        "lfeFilter (p. 728)": enum,
        "bitrate (p. 728)": integer,
        "bitstreamMode (p. 728)": enum,
        "sampleRate (p. 728)": integer
      },
      "aacSettings (p. 730)": {
        "vbrQuality (p. 725)": enum,
        "codecProfile (p. 725)": enum,
        "codingMode (p. 725)": enum,
        "specification (p. 725)": enum,
        "bitrate (p. 725)": integer,
"rawFormat (p. 726)": enum,
"rateControlMode (p. 726)": enum,
"sampleRate (p. 726)": integer,
"audioDescriptionBroadcasterMix (p. 726)": enum
},
"aiffSettings (p. 730)": {
  "channels (p. 729)": integer,
  "bitDepth (p. 729)": integer,
  "sampleRate (p. 729)": integer
},
"eac3Settings (p. 730)": {
  "dialnorm (p. 749)": integer,
  "passthroughControl (p. 749)": enum,
  "metadataControl (p. 749)": enum,
  "bitrate (p. 749)": integer,
  "dynamicRangeCompressionRf (p. 749)": enum,
  "sampleRate (p. 749)": integer,
  "ltRtSurroundMixLevel (p. 750)": number,
  "surroundExMode (p. 750)": enum,
  "dynamicRangeCompressionLine (p. 750)": enum,
  "lfeControl (p. 750)": enum,
  "codingMode (p. 750)": enum,
  "surroundMode (p. 750)": enum,
  "attenuationControl (p. 750)": enum,
  "lfeFilter (p. 750)": enum,
  "ltRtCenterMixLevel (p. 751)": number,
  "surroundMode (p. 751)": enum,
  "dynamicRangeCompressionLine (p. 751)": number,
  "loRoSurroundMixLevel (p. 751)": number,
  "loRoCenterMixLevel (p. 751)": number
},
"mp2Settings (p. 730)": {
  "channels (p. 789)": integer,
  "bitrate (p. 789)": integer,
  "sampleRate (p. 789)": integer
},
"languageCode (p. 731)": enum,
"streamName (p. 731)": "string",
"audioNormalizationSettings (p. 731)": {
  "targetLkfs (p. 732)": number,
  "algorithmControl (p. 733)": enum,
  "peakCalculation (p. 733)": enum,
  "loudnessLogging (p. 733)": enum,
  "correctionGateLevel (p. 733)": integer,
  "algorithm (p. 733)": enum
}
},
"containerSettings (p. 801)": {
  "container (p. 740)": enum,
  "mp4Settings (p. 740)": {
    "mp4MajorBrand (p. 790)": "string",
    "moovPlacement (p. 790)": enum,
    "cs1gAtom (p. 790)": enum,
    "freeSpaceBox (p. 790)": enum
  },
  "m3u8Settings (p. 740)": {
    "pmtPid (p. 785)": integer,
    "pcrPid (p. 786)": integer,
    "audioPids (p. 786)": [integer
    ],
    "audioFramesPerPes (p. 786)": integer,
<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>scte35Pid</code> (p. 786)</td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td><code>transportStreamId</code> (p. 786)</td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td><code>videoPid</code> (p. 786)</td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td><code>pcrControl</code> (p. 787)</td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>privateMetadataPid</code> (p. 787)</td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td><code>pmtInterval</code> (p. 787)</td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td><code>patInterval</code> (p. 787)</td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td><code>programNumber</code> (p. 787)</td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td><code>timedMetadataPid</code> (p. 787)</td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td><code>timedMetadata</code> (p. 788)</td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>scte35Source</code> (p. 788)</td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>m2tsSettings</code> (p. 740)</td>
<td>{}</td>
<td></td>
</tr>
<tr>
<td><code>dvbTeletextPid</code> (p. 780)</td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td><code>bitrate</code> (p. 781)</td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td><code>segmentationTime</code> (p. 781)</td>
<td>number</td>
<td></td>
</tr>
<tr>
<td><code>audioPids</code> (p. 781)</td>
<td>[integer]</td>
<td></td>
</tr>
<tr>
<td><code>rateMode</code> (p. 781)</td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>ebpAudioInterval</code> (p. 781)</td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>fragmentTime</code> (p. 781)</td>
<td>number</td>
<td></td>
</tr>
<tr>
<td><code>audioFramesPerPes</code> (p. 781)</td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td><code>maxPcrInterval</code> (p. 782)</td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td><code>scte35Pid</code> (p. 782)</td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td><code>privateMetadataPid</code> (p. 782)</td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td><code>pmtInterval</code> (p. 782)</td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td><code>segmentationStyle</code> (p. 782)</td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>audioBufferModel</code> (p. 782)</td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>programNumber</code> (p. 783)</td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td><code>dvbNitSettings</code> (p. 783)</td>
<td>{}</td>
<td></td>
</tr>
<tr>
<td><code>networkName</code> (p. 742)</td>
<td>string</td>
<td></td>
</tr>
<tr>
<td><code>networkId</code> (p. 742)</td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td><code>nitInterval</code> (p. 742)</td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td><code>scte35Source</code> (p. 783)</td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>pmtPid</code> (p. 783)</td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td><code>bufferModel</code> (p. 783)</td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>ebpPlacement</code> (p. 783)</td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>dvbSdtSettings</code> (p. 783)</td>
<td>{}</td>
<td></td>
</tr>
<tr>
<td><code>sdInterval</code> (p. 742)</td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td><code>serviceName</code> (p. 743)</td>
<td>string</td>
<td></td>
</tr>
<tr>
<td><code>serviceProviderName</code> (p. 743)</td>
<td>string</td>
<td></td>
</tr>
<tr>
<td><code>outputSdt</code> (p. 743)</td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>nullPacketBitrate</code> (p. 783)</td>
<td>number</td>
<td></td>
</tr>
<tr>
<td><code>pcrPid</code> (p. 784)</td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td><code>minEbpInterval</code> (p. 784)</td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td><code>transportStreamId</code> (p. 784)</td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td><code>pcrControl</code> (p. 784)</td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>videoPid</code> (p. 784)</td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td><code>esRateInPes</code> (p. 784)</td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>segmentationMarkers</code> (p. 785)</td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>dvbTdtSettings</code> (p. 785)</td>
<td>{}</td>
<td></td>
</tr>
<tr>
<td><code>tdtInterval</code> (p. 747)</td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td><code>patInterval</code> (p. 785)</td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td><code>DvbSubPids</code> (p. 785)</td>
<td>[integer]</td>
<td></td>
</tr>
<tr>
<td><code>movSettings</code> (p. 740)</td>
<td>{}</td>
<td></td>
</tr>
<tr>
<td><code>reference</code> (p. 789)</td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>paddingControl</code> (p. 789)</td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>mpeg2FourCCControl</code> (p. 789)</td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>csigAtom</code> (p. 789)</td>
<td>enum</td>
<td></td>
</tr>
</tbody>
</table>

708
"clapAtom (p. 789)" : enum
,
"f4vSettings (p. 740)" : {
   "moovPlacement (p. 752)" : enum
}
,
"captionDescriptions (p. 801)" : [
   {
      "languageDescription (p. 737)" : "string",
      "languageCode (p. 737)" : enum,
      "destinationSettings (p. 737)" : {
         "burninDestinationSettings (p. 737)" : {
            "xPosition (p. 733)" : integer,
            "backgroundColor (p. 734)" : enum,
            "teletextSpacing (p. 734)" : enum,
            "yPosition (p. 734)" : integer,
            "backgroundOpacity (p. 734)" : integer,
            "fontOpacity (p. 734)" : integer,
            "shadowOpacity (p. 734)" : integer,
            "fontResolution (p. 735)" : integer,
            "shadowYOffset (p. 735)" : integer,
            "outlineSize (p. 735)" : integer,
            "outlineColor (p. 735)" : enum,
            "fontSize (p. 735)" : integer,
            "shadowXOffset (p. 735)" : integer,
            "alignment (p. 736)" : enum,
            "shadowColor (p. 736)" : enum,
            "fontColor (p. 736)" : enum
         },
         "teletextDestinationSettings (p. 738)" : {
            "pageNumber (p. 806)" : "string"
         },
         "ttmlDestinationSettings (p. 738)" : {
            "stylePassthrough (p. 807)" : enum
         },
         "destinationType (p. 738)" : enum,
         "dvbSubDestinationSettings (p. 738)" : {
            "xPosition (p. 743)" : integer,
            "backgroundColor (p. 743)" : enum,
            "teletextSpacing (p. 743)" : enum,
            "yPosition (p. 743)" : integer,
            "backgroundOpacity (p. 744)" : integer,
            "fontOpacity (p. 744)" : integer,
            "shadowOpacity (p. 744)" : integer,
            "fontResolution (p. 744)" : integer,
            "shadowYOffset (p. 744)" : integer,
            "outlineSize (p. 745)" : integer,
            "outlineColor (p. 745)" : enum,
            "fontSize (p. 745)" : integer,
            "shadowXOffset (p. 745)" : integer,
            "alignment (p. 745)" : enum,
            "shadowColor (p. 745)" : enum,
            "fontColor (p. 745)" : enum
         },
         "sccDestinationSettings (p. 738)" : {
            "framerate (p. 805)" : enum
         }
      }
   }
]
Example DELETE

```json
{
  "name (p. 742)": "string"
}
```

Response Bodies

Example GetPresetResponse

```json
{
  "preset (p. 753)": {
    "settings (p. 799)": {
      "videoDescription (p. 800)": {
        "fixedAfd (p. 809)": integer,
        "scalingBehavior (p. 809)": enum,
        "respondToAfd (p. 809)": enum,
        "codecSettings (p. 809)": {
          "h265Settings (p. 808)": {
            "slices (p. 765)": integer,
            "minIInterval (p. 765)": integer,
            "parNumerator (p. 765)": integer,
            "flickerAdaptiveQuantization (p. 766)": enum,
            "gopsSizeUnits (p. 766)": enum,
            "hrdBufferSize (p. 766)": integer,
            "qualityTuningLevel (p. 766)": enum,
            "maxBitrate (p. 766)": integer,
            "bitrate (p. 766)": integer,
            "spatialAdaptiveQuantization (p. 766)": enum,
            "sampleAdaptiveOffsetFilterMode (p. 767)": enum,
            "temporalIds (p. 767)": enum,
            "slowPal (p. 767)": enum,
            "tiles (p. 767)": enum,
            "codecProfile (p. 767)": enum,
            "alternateTransferFunctionSel (p. 767)": enum,
            "unregisteredSelTimecode (p. 767)": enum,
            "framerateControl (p. 767)": enum,
            "telecine (p. 767)": enum,
            "framerateConversionAlgorithm (p. 767)": enum,
            "codecLevel (p. 768)": enum,
            "numberReferenceFrames (p. 768)": integer,
            "temporalAdaptiveQuantization (p. 768)": enum,
            "hrdBufferInitialFillPercentage (p. 768)": integer,
            "framerateNumerator (p. 768)": integer,
            "framerateConversionAlgorithm (p. 768)": integer,
            "gopClosedCadence (p. 768)": integer,
            "framerateDenominator (p. 768)": integer,
            "adaptiveQuantization (p. 768)": enum,
            "interlaceMode (p. 769)": enum,
            "gopsSize (p. 769)": number,
            "gopBReference (p. 769)": enum,
            "sceneChangeDetect (p. 769)": enum,
            "parDenominator (p. 769)": integer,
            "parControl (p. 770)": enum,
            "rateControlMode (p. 770)": enum
          },
          "proresSettings (p. 808)": {
            "slowPal (p. 802)": enum,
            "framerateControl (p. 802)": enum,
            "telecine (p. 802)": enum,
            "framerateDenominator (p. 802)": integer,
            "framerateConversionAlgorithm (p. 802)": enum,
          }
        },
        "codec (p. 808)": enum,
        "proresSettings (p. 808)": {
          "slowPal (p. 802)": enum,
          "framerateControl (p. 802)": enum,
          "telecine (p. 802)": enum,
          "framerateDenominator (p. 802)": integer,
          "framerateConversionAlgorithm (p. 802)": enum,
        }
      },
      "h264Settings (p. 808)":{
        "slices (p. 765)": integer,
        "minIInterval (p. 765)": integer,
        "parNumerator (p. 765)": integer,
        "flickerAdaptiveQuantization (p. 766)": enum,
        "gopsSizeUnits (p. 766)": enum,
        "hrdBufferSize (p. 766)": integer,
        "qualityTuningLevel (p. 766)": enum,
        "maxBitrate (p. 766)": integer,
        "bitrate (p. 766)": integer,
        "spatialAdaptiveQuantization (p. 766)": enum,
        "sampleAdaptiveOffsetFilterMode (p. 767)": enum,
        "temporalIds (p. 767)": enum,
        "slowPal (p. 767)": enum,
        "tiles (p. 767)": enum,
        "codecProfile (p. 767)": enum,
        "alternateTransferFunctionSel (p. 767)": enum,
        "unregisteredSelTimecode (p. 767)": enum,
        "framerateControl (p. 767)": enum,
        "telecine (p. 767)": enum,
        "framerateConversionAlgorithm (p. 767)": enum,
        "codecLevel (p. 768)": enum,
        "numberReferenceFrames (p. 768)": integer,
        "temporalAdaptiveQuantization (p. 768)": enum,
        "hrdBufferInitialFillPercentage (p. 768)": integer,
        "framerateNumerator (p. 768)": integer,
        "framerateConversionAlgorithm (p. 768)": integer,
        "gopClosedCadence (p. 768)": integer,
        "framerateDenominator (p. 768)": integer,
        "adaptiveQuantization (p. 768)": enum,
        "interlaceMode (p. 769)": enum,
        "gopsSize (p. 769)": number,
        "gopBReference (p. 769)": enum,
        "sceneChangeDetect (p. 769)": enum,
        "parDenominator (p. 769)": integer,
        "parControl (p. 770)": enum,
        "rateControlMode (p. 770)": enum
      }
    }
  }
}
```
"interlaceMode (p. 802)": enum,
"codecProfile (p. 803)": enum,
"parNumerator (p. 803)": integer,
"parControl (p. 803)": enum,
"parDenominator (p. 803)": integer,
"framerateNumerator (p. 803)": integer
},
"mpeg2Settings (p. 808)": {
  "minIInterval (p. 793)": integer,
  "parNumerator (p. 793)": integer,
  "gopSizeUnits (p. 793)": enum,
  "hrdBufferSize (p. 793)": integer,
  "qualityTuningLevel (p. 793)": enum,
  "maxBitrate (p. 793)": integer,
  "bitrate (p. 794)": integer,
  "spatialAdaptiveQuantization (p. 794)": enum,
  "slowPal (p. 794)": enum,
  "codecProfile (p. 794)": enum,
  "intraDcPrecision (p. 794)": enum,
  "softness (p. 794)": integer,
  "framerateControl (p. 794)": enum,
  "telecine (p. 794)": enum,
  "framerateConversionAlgorithm (p. 795)": enum,
  "codecLevel (p. 795)": enum,
  "temporalAdaptiveQuantization (p. 795)": enum,
  "hrdBufferInitialFillPercentage (p. 795)": integer,
  "framerateNumerator (p. 795)": integer,
  "parDenominator (p. 796)": integer,
  "parControl (p. 796)": enum,
  "intraDcPrecision (p. 796)": enum,
  "syntax (p. 796)": enum,
  "rateControlMode (p. 797)": enum
},
"h264Settings (p. 808)": {
  "slices (p. 756)": integer,
  "minIInterval (p. 757)": integer,
  "parNumerator (p. 757)": integer,
  "flickerAdaptiveQuantization (p. 757)": enum,
  "gopSizeUnits (p. 757)": enum,
  "hrdBufferSize (p. 757)": integer,
  "qualityTuningLevel (p. 757)": enum,
  "maxBitrate (p. 757)": integer,
  "bitrate (p. 758)": integer,
  "spatialAdaptiveQuantization (p. 758)": enum,
  "slowPal (p. 758)": enum,
  "codecProfile (p. 758)": enum,
  "unregisteredSeiTimecode (p. 758)": enum,
  "softness (p. 758)": integer,
  "framerateControl (p. 758)": enum,
  "telecine (p. 758)": enum,
  "framerateConversionAlgorithm (p. 759)": enum,
  "codecLevel (p. 759)": enum,
  "numberReferenceFrames (p. 759)": integer,
  "temporalAdaptiveQuantization (p. 759)": enum,
  "repeatPps (p. 759)": enum,
  "hrdBufferInitialFillPercentage (p. 759)": integer,
  "framerateNumerator (p. 759)": integer,
  "parDenominator (p. 760)": integer,
  "fieldEncoding (p. 760)": enum,
"entropyEncoding (p. 760)": enum,
"framerateDenominator (p. 760)": integer,
"adaptiveQuantization (p. 760)": enum,
"interlaceMode (p. 760)": enum,
"gopSize (p. 760)": number,
"gopBReference (p. 761)": enum,
"sceneChangeDetect (p. 761)": enum,
"parDenominator (p. 761)": integer,
"parControl (p. 761)": enum,
"syntax (p. 761)": enum,
"rateControlMode (p. 761)": enum
},
"frameCaptureSettings (p. 808)": {
  "framerateDenominator (p. 752)": integer,
  "maxCaptures (p. 753)": integer,
  "framerateNumerator (p. 753)": integer,
  "quality (p. 753)": integer
}
},
"afdSignaling (p. 809)": enum,
"colorMetadata (p. 809)": enum,
"timecodeInsertion (p. 809)": enum,
"width (p. 809)": integer,
"sharpness (p. 810)": integer,
"antiAlias (p. 810)": enum,
"videoPreprocessors (p. 810)": {
  "timecodeBurnin (p. 811)": {
    "prefix (p. 806)": "string",
    "fontSize (p. 806)": integer,
    "position (p. 806)": enum
  },
  "noiseReducer (p. 811)": {
    "filter (p. 797)": enum,
    "filterSettings (p. 797)": {
      "strength (p. 798)": integer
    },
    "spatialFilterSettings (p. 798)": {
      "strength (p. 798)": integer,
      "postFilterSharpenStrength (p. 798)": integer,
      "speed (p. 799)": integer
    }
  },
  "colorCorrector (p. 811)": {
    "saturation (p. 738)": integer,
    "brightness (p. 739)": integer,
    "hdr10Metadata (p. 739)": {
      "redPrimaryY (p. 771)": integer,
      "greenPrimaryY (p. 771)": integer,
      "whitePointX (p. 771)": integer,
      "maxLuminance (p. 771)": integer,
      "greenPrimaryX (p. 772)": integer,
      "whitePointY (p. 772)": integer,
      "redPrimaryX (p. 772)": integer,
      "bluePrimaryX (p. 772)": integer,
      "maxFrameAverageLightLevel (p. 772)": integer,
      "bluePrimaryY (p. 772)": integer,
      "maxContentLightLevel (p. 773)": integer,
      "minLuminance (p. 773)": integer
    },
    "contrast (p. 739)": integer,
    "hue (p. 739)": integer,
    "colorSpaceConversion (p. 739)": enum
  },
  "imageInserter (p. 811)": {
    "insertableImages (p. 773)": [
"duration (p. 773)": integer,
"fadeOut (p. 773)": integer,
"imageY (p. 774)": integer,
"fadeIn (p. 774)": integer,
"imageX (p. 774)": integer,
"width (p. 774)": integer,
"startTime (p. 774)": "string",
"opacity (p. 774)": integer,
"layer (p. 775)": integer,
"height (p. 775)": integer,
"imageInserterInput (p. 775)": "string"
}
],
"deinterlacer (p. 811)": {
  "mode (p. 741)": enum,
  "control (p. 741)": enum,
  "algorithm (p. 741)": enum
}
],
"position (p. 810)": {
  "width (p. 804)": integer,
  "x (p. 804)": integer,
  "y (p. 804)": integer,
  "height (p. 804)": integer
},
"dropFrameTimecode (p. 810)": enum,
"crop (p. 810)": {
  "width (p. 804)": integer,
  "x (p. 804)": integer,
  "y (p. 804)": integer,
  "height (p. 804)": integer
},
"height (p. 810)": integer
],
"audioDescriptions (p. 800)": [
{
  "languageCodeControl (p. 730)": enum,
  "audioTypeControl (p. 730)": enum,
  "remixSettings (p. 730)": {
    "channelsOut (p. 804)": integer,
    "channelMapping (p. 804)": {
      "outputChannels (p. 738)": [
        {"inputChannels (p. 799)": [integer
          ]
      }
    }
  }
  },
  "channelsIn (p. 805)": integer
},
"audioType (p. 731)": integer,
"audioSourceName (p. 731)": "string",
"codecSettings (p. 731)": {
  "codec (p. 729)": enum,
  "wavSettings (p. 730)": {
    "channels (p. 812)": integer,
    "bitDepth (p. 812)": integer,
    "sampleRate (p. 812)": integer
  },
  "ac3Settings (p. 730)": {
    "dynamicRangeCompressionProfile (p. 727)": enum,
    "dialnorm (p. 727)": integer,
    "codingMode (p. 728)": enum,
    "metadataControl (p. 728)": enum,
"lfeFilter (p. 728)"
"bitrate (p. 728)"
"bitstreamMode (p. 728)"
"sampleRate (p. 728)"

"aacSettings (p. 730)"
"vbrQuality (p. 725)"
"codecProfile (p. 725)"
"codingMode (p. 725)"
"specification (p. 725)"
"bitrate (p. 725)"
"rawFormat (p. 726)"
"rateControlMode (p. 726)"
"sampleRate (p. 726)"
"audioDescriptionBroadcasterMix (p. 726)"

"aiffSettings (p. 730)"
"channels (p. 729)"
"bitDepth (p. 729)"
"sampleRate (p. 729)"

"eac3Settings (p. 730)"
"dialnorm (p. 749)"
"passthroughControl (p. 749)"
"metadataControl (p. 749)"
"bitrate (p. 749)"
"sampleRate (p. 749)"
"ltReSurroundMixLevel (p. 750)"
"surroundExMode (p. 750)"
"dynamicRangeCompressionRf (p. 749)"
"sampleRate (p. 749)"

"mp2Settings (p. 730)"
"channels (p. 789)"
"bitrate (p. 789)"
"sampleRate (p. 789)"

"languageCode (p. 731)"
"streamName (p. 731)"
"audioNormalizationSettings (p. 731)"
"targetLkfs (p. 732)"
"algorithmControl (p. 733)"
"peakCalculation (p. 733)"
"loudnessLogging (p. 733)"
"correctionGateLevel (p. 733)"

"containerSettings (p. 801)"
"container (p. 740)"
"mp4Settings (p. 740)"
"moovPlacement (p. 790)”: enum,
"csigAtom (p. 790)”: enum,
"freeSpaceBox (p. 790)”: enum
},
"m3u8Settings (p. 740)”: {
  "pmtPid (p. 785)”: integer,
  "pcrPid (p. 786)”: integer,
  "audioPids (p. 786)”: [
      integer
  ],
  "audioFramesPerPes (p. 786)”: integer,
  "scte35Pid (p. 786)”: integer,
  "transportStreamId (p. 786)”: integer,
  "videoPid (p. 786)”: integer,
  "pcrControl (p. 787)”: enum,
  "privateMetadataPid (p. 787)”: integer,
  "pmtInterval (p. 787)”: integer,
  "patInterval (p. 787)”: integer,
  "programNumber (p. 787)”: integer,
  "timedMetadataPid (p. 787)”: integer,
  "timedMetadata (p. 788)”: enum,
  "scte35Source (p. 788)”: enum
},
"m2tsSettings (p. 740)”: {
  "dvbTeletextPid (p. 780)”: integer,
  "bitrate (p. 781)”: integer,
  "segmentationTime (p. 781)”: number,
  "audioPids (p. 781)”: [
      integer
  ],
  "rateMode (p. 781)”: enum,
  "ebpAudioInterval (p. 781)”: enum,
  "fragmentTime (p. 781)”: number,
  "audioFramesPerPes (p. 781)”: integer,
  "maxPcrInterval (p. 782)”: integer,
  "scte35Pid (p. 782)”: integer,
  "privateMetadataPid (p. 782)”: integer,
  "pmtInterval (p. 782)”: integer,
  "segmentationStyle (p. 782)”: enum,
  "audioBufferModel (p. 782)”: enum,
  "programNumber (p. 783)”: integer,
  "dvbNitSettings (p. 783)”: {
    "networkName (p. 742)”: ”string”,
    "networkId (p. 742)”: integer,
    "nitInterval (p. 742)”: integer
  },
  "scte35Source (p. 783)”: enum,
  "pmtPid (p. 783)”: integer,
  "bufferModel (p. 783)”: enum,
  "ebpPlacement (p. 783)”: enum,
  "dvbSdtSettings (p. 783)”: {
    "sdtInterval (p. 742)": integer,
    "serviceName (p. 743)": ”string”,
    "serviceProviderName (p. 743)”: ”string”,
    "outputSdt (p. 743)”: enum
  },
  "nullPacketBitrate (p. 783)”: number,
  "pcrPid (p. 784)”: integer,
  "minEbpInterval (p. 784)”: integer,
  "transportStreamId (p. 784)”: integer,
  "pcrControl (p. 784)”: enum,
  "videoPid (p. 784)”: integer,
  "esRateInPes (p. 784)”: enum,
  "segmentationMarkers (p. 785)”: enum,
  "dvbTdtSettings (p. 785)”: {
    "tdtInterval (p. 747)": integer
  }
},
"patInterval (p. 785)" : integer,
"dvbSubPids (p. 785)" : [
  integer
],
"movSettings (p. 740)" : {
  "reference (p. 789)" : enum,
  "paddingControl (p. 789)" : enum,
  "mpeg2PourCControl (p. 789)" : enum,
  "cs1gAtom (p. 789)" : enum,
  "clapAtom (p. 789)" : enum
},
"f4vSettings (p. 740)" : {
  "moovPlacement (p. 752)" : enum
},
"captionDescriptions (p. 801)" : [
  {
    "languageDescription (p. 737)" : "string",
    "languageCode (p. 737)" : enum,
    "destinationSettings (p. 737)" : {
      "burninDestinationSettings (p. 737)" : {
        "xPosition (p. 733)" : integer,
        "backgroundColor (p. 734)" : enum,
        "teletextSpacing (p. 734)" : enum,
        "yPosition (p. 734)" : integer,
        "backgroundOpacity (p. 734)" : integer,
        "fontOpacity (p. 734)" : integer,
        "shadowOpacity (p. 734)" : integer,
        "fontResolution (p. 735)" : integer,
        "shadowYOffset (p. 735)" : integer,
        "outlineSize (p. 735)" : integer,
        "outlineColor (p. 735)" : enum,
        "fontSize (p. 735)" : integer,
        "shadowXOffset (p. 735)" : integer,
        "alignment (p. 736)" : enum,
        "shadowColor (p. 736)" : enum,
        "fontColor (p. 736)" : enum
      },
      "teletextDestinationSettings (p. 738)" : {
        "pageNumber (p. 806)" : "string"
      },
      "ttmlDestinationSettings (p. 738)" : {
        "stylePassthrough (p. 807)" : enum
      },
      "destinationType (p. 738)" : enum,
      "dvbSubDestinationSettings (p. 738)" : {
        "xPosition (p. 743)" : integer,
        "backgroundColor (p. 743)" : enum,
        "teletextSpacing (p. 743)" : enum,
        "yPosition (p. 743)" : integer,
        "backgroundOpacity (p. 744)" : integer,
        "fontOpacity (p. 744)" : integer,
        "shadowOpacity (p. 744)" : integer,
        "fontResolution (p. 744)" : integer,
        "shadowYOffset (p. 744)" : integer,
        "outlineSize (p. 745)" : integer,
        "outlineColor (p. 745)" : enum,
        "fontSize (p. 745)" : integer,
        "shadowXOffset (p. 745)" : integer,
        "alignment (p. 745)" : enum,
        "shadowColor (p. 745)" : enum,
        "fontColor (p. 745)" : enum
      },
      "sccDestinationSettings (p. 738)" : {
    },
Example UpdatePresetResponse

```
{
  "preset (p. 808)": {
    "settings (p. 799)": {
      "videoDescription (p. 800)": {
        "fixedAfd (p. 809)": integer,
        "scalingBehavior (p. 809)": enum,
        "respondToAfd (p. 809)": enum,
        "codecSettings (p. 809)": {
          "h265Settings (p. 808)": {
            "slices (p. 765)": integer,
            "minIInterval (p. 765)": integer,
            "parNumerator (p. 765)": integer,
            "flickerAdaptiveQuantization (p. 766)": enum,
            "gopSizeUnits (p. 766)": enum,
            "hrdBufferSize (p. 766)": integer,
            "qualityTuningLevel (p. 766)": enum,
            "maxBitrate (p. 766)": integer,
            "bitrate (p. 766)": integer,
            "spatialAdaptiveQuantization (p. 766)": enum,
            "sampleRateController (p. 767)": enum,
            "temporalIds (p. 767)": enum,
            "slowPA (p. 767)": enum,
            "tiles (p. 767)": enum,
            "codecsProfile (p. 767)": enum,
            "alternateTransferFunctionSei (p. 767)": enum,
            "unregisteredSeiTimecode (p. 767)": enum,
            "framerateControl (p. 767)": enum,
            "telecine (p. 767)": enum,
            "framerateConversionAlgorithm (p. 767)": enum,
            "codecLevel (p. 768)": enum,
            "numberReferenceFrames (p. 768)": integer,
            "adaptiveQuantization (p. 768)": enum,
            "hrdBufferInitialFillPercentage (p. 768)": integer,
            "framerateNumerator (p. 769)": integer,
            "numberBFramesBetweenReferenceFrames (p. 768)": integer,
            "gopClosedCadence (p. 769)": integer,
            "framerateDenominator (p. 769)": integer,
            "adaptiveQuantization (p. 769)": enum,
            "interlaceMode (p. 769)": enum,
            "gopSize (p. 769)": number,
            "gopBReference (p. 769)": enum,
            "sceneChangeDetect (p. 769)": enum,
            "parDenominator (p. 769)": integer,
            "parControl (p. 770)": enum,
            "rateControlMode (p. 770)": enum
          }
        }
      }
    }
  }
}
```
"codec (p. 808)": enum,
"proresSettings (p. 808)": {
  "slowPal (p. 802)": enum,
  "framerateControl (p. 802)": enum,
  "telecine (p. 802)": enum,
  "framerateDenominator (p. 802)": integer,
  "framerateConversionAlgorithm (p. 802)": enum,
  "codecProfile (p. 803)": enum,
  "parNumerator (p. 803)": integer,
  "parControl (p. 803)": enum,
  "parDenominator (p. 803)": integer,
  "framerateNumerator (p. 803)": integer
},
"mpeg2Settings (p. 808)": {
  "minInterval (p. 793)": integer,
  "parNumerator (p. 793)": integer,
  "gopSizeUnits (p. 793)": enum,
  "hrdBufferSize (p. 793)": integer,
  "qualityTuningLevel (p. 793)": enum,
  "maxBitrate (p. 793)": integer,
  "bitrate (p. 794)": integer,
  "spatialAdaptiveQuantization (p. 794)": enum,
  "slowPal (p. 794)": enum,
  "codecProfile (p. 794)": enum,
  "intraDcPrecision (p. 794)": enum,
  "softness (p. 794)": integer,
  "framerateControl (p. 794)": enum,
  "telecine (p. 794)": enum,
  "framerateConversionAlgorithm (p. 795)": enum,
  "codecLevel (p. 795)": enum,
  "temporalAdaptiveQuantization (p. 795)": enum,
  "hrdBufferInitialFillPercentage (p. 795)": integer,
  "framerateNumerator (p. 795)": integer,
  "gopClosedCadence (p. 795)": integer,
  "parDenominator (p. 796)": integer,
  "adaptiveQuantization (p. 796)": enum,
  "interlaceMode (p. 796)": enum,
  "gopSize (p. 796)": number,
  "sceneChangeDetect (p. 796)": enum,
  "parControl (p. 796)": integer,
  "syntax (p. 796)": enum,
  "rateControlMode (p. 797)": enum
},
"h264Settings (p. 808)": {
  "slices (p. 756)": integer,
  "minInterval (p. 757)": integer,
  "parNumerator (p. 757)": integer,
  "flickerAdaptiveQuantization (p. 757)": enum,
  "gopSizeUnits (p. 757)": enum,
  "hrdBufferSize (p. 757)": integer,
  "qualityTuningLevel (p. 757)": enum,
  "maxBitrate (p. 757)": integer,
  "bitrate (p. 758)": integer,
  "spatialAdaptiveQuantization (p. 758)": enum,
  "slowPal (p. 758)": enum,
  "codecProfile (p. 758)": enum,
  "unregisteredSeiTimecode (p. 758)": enum,
  "softness (p. 758)": integer,
  "framerateControl (p. 758)": enum,
  "telecine (p. 758)": enum,
  "framerateConversionAlgorithm (p. 759)": enum,
  "codecLevel (p. 759)": enum,
  "numberReferenceFrames (p. 759)": integer,
"temporalAdaptiveQuantization (p. 759)": enum,
"repeatPps (p. 759)": enum,
"hrdBufferInitialFillPercentage (p. 759)": integer,
"framerateNumerator (p. 759)": integer,
"numberBFramesBetweenReferenceFrames (p. 759)": integer,
"gopClosedCadence (p. 760)": integer,
"fieldEncoding (p. 760)": enum,
"entropyEncoding (p. 760)": enum,
"framerateDenominator (p. 760)": integer,
"adaptiveQuantization (p. 760)": enum,
"interlaceMode (p. 760)": enum,
"gopSize (p. 760)": number,
"gopBReference (p. 761)": enum,
"sceneChangeDetect (p. 761)": enum,
"parDenominator (p. 761)": integer,
"parControl (p. 761)": enum,
"syntax (p. 761)": enum,
"rateControlMode (p. 761)": enum
"frameCaptureSettings (p. 808)": {
"framerateDenominator (p. 752)": integer,
"maxCaptures (p. 753)": integer,
"framerateNumerator (p. 753)": integer,
"quality (p. 753)": integer
}

"afdSignaling (p. 809)": enum,
"colorMetadata (p. 809)": enum,
"timecodeInsertion (p. 809)": enum,
"width (p. 809)": integer,
"sharpness (p. 810)": integer,
"antiAlias (p. 810)": enum,
"videoPreprocessors (p. 810)": {
"timecodeBurnin (p. 811)": {
"prefix (p. 806)": "string",
"fontSize (p. 806)": integer,
"position (p. 806)": enum
},
"noiseReducer (p. 811)": {
"filter (p. 797)": enum,
"filterSettings (p. 797)": {
"strength (p. 798)": integer
}
},
"spatialFilterSettings (p. 798)": {
"strength (p. 798)": integer,
"postFilterSharpenStrength (p. 798)": integer,
"speed (p. 799)": integer
}

"colorCorrector (p. 811)": {
"saturation (p. 738)": integer,
"brightness (p. 739)": integer,
"hdr10Metadata (p. 739)": {
"redPrimaryY (p. 771)": integer,
"greenPrimaryY (p. 771)": integer,
"whitePointX (p. 771)": integer,
"maxLuminance (p. 771)": integer,
"greenPrimaryX (p. 772)": integer,
"whitePointY (p. 772)": integer,
"redPrimaryX (p. 772)": integer,
"bluePrimaryX (p. 772)": integer,
"maxFrameAverageLightLevel (p. 772)": integer,
"bluePrimaryY (p. 772)": integer,
"maxContentLightLevel (p. 773)": integer,
"minLuminance (p. 773)": integer
}
"width (p. 804)": integer,
"x (p. 804)": integer,
"y (p. 804)": integer,
"height (p. 804)": integer,
"height (p. 810)": integer
],
"audioDescriptions (p. 800)": [
{
"languageCodeControl (p. 730)": enum,
"audioTypeControl (p. 730)": enum,
"remixSettings (p. 730)": {
"channelsOut (p. 804)": integer,
"channelMapping (p. 804)": {
"outputChannels (p. 730)": [
{
"inputChannels (p. 799)": [ integer
]
}
],
"channelsIn (p. 805)": integer
],
"audioType (p. 731)": integer,
"audioSourceName (p. 731)": "string",
"codecSettings (p. 731)": {
"codec (p. 729)": enum,
"wavSettings (p. 730)": {
"channels (p. 812)": integer,
"bitDepth (p. 812)": integer,
"sampleRate (p. 812)": integer
},
"ac3Settings (p. 730)": {
  "dynamicRangeCompressionProfile (p. 727)": enum,
  "dialnorm (p. 727)": integer,
  "codingMode (p. 728)": enum,
  "metadataControl (p. 728)": enum,
  "lfeFilter (p. 728)": enum,
  "bitrate (p. 728)": integer,
  "bitstreamMode (p. 728)": enum,
  "sampleRate (p. 728)": integer
},
"aacSettings (p. 730)": {
  "vbrQuality (p. 725)": enum,
  "codecProfile (p. 725)": enum,
  "codingMode (p. 725)": enum,
  "specification (p. 725)": enum,
  "bitrate (p. 725)": integer,
  "rawFormat (p. 726)": enum,
  "rateControlMode (p. 726)": enum,
  "sampleRate (p. 726)": integer,
  "audioDescriptionBroadcasterMix (p. 726)": enum
},
"aiffSettings (p. 730)": {
  "channels (p. 729)": integer,
  "bitDepth (p. 729)": integer,
  "sampleRate (p. 729)": integer
},
"ac3Settings (p. 730)": {
  "dialnorm (p. 749)": integer,
  "passthroughControl (p. 749)": enum,
  "metadataControl (p. 749)": enum,
  "bitrate (p. 749)": integer,
  "dynamicRangeCompressionRf (p. 749)": enum,
  "sampleRate (p. 749)": integer,
  "ltRtSurroundMixLevel (p. 750)": number,
  "surroundExMode (p. 750)": enum,
  "dynamicRangeCompressionLine (p. 750)": enum,
  "lfeControl (p. 750)": enum,
  "codingMode (p. 750)": enum,
  "surroundMode (p. 750)": enum,
  "attenuationControl (p. 750)": enum,
  "lfeFilter (p. 750)": enum,
  "phaseControl (p. 750)": enum,
  "ltRtCenterMixLevel (p. 751)": number,
  "dcFilter (p. 751)": enum,
  "stereoDownmix (p. 751)": enum,
  "bitstreamMode (p. 751)": enum,
  "loRoSurroundMixLevel (p. 751)": number,
  "loRoCenterMixLevel (p. 751)": number
},
"mp2Settings (p. 730)": {
  "channels (p. 789)": integer,
  "bitrate (p. 789)": integer,
  "sampleRate (p. 789)": integer
},
"languageCode (p. 731)": enum,
"streamName (p. 731)": "string",
"audioNormalizationSettings (p. 731)": {
  "targetLkfs (p. 732)": number,
  "algorithmControl (p. 733)": enum,
  "peakCalculation (p. 733)": enum,
  "loudnessLogging (p. 733)": enum,
  "correctionGateLevel (p. 733)": integer,
  "algorithm (p. 733)": enum
}
Schemas

"containerSettings (p. 801)": {
"container (p. 740)": enum,
"mp4Settings (p. 740)": {
"mp4MajorBrand (p. 790)": "string",
"moovPlacement (p. 790)": enum,
"cslgAtom (p. 790)": enum,
"freeSpaceBox (p. 790)": enum
},
"m3u8Settings (p. 740)": {
"pmtPid (p. 785)": integer,
"pcrPid (p. 786)": integer,
"audioPids (p. 786)": [integer]
"audioFramesPerPes (p. 786)": integer,
"scte35Pid (p. 786)": integer,
"transportStreamId (p. 786)": integer,
"videoPid (p. 786)": integer,
"pcrControl (p. 787)": enum,
"privateMetadataPid (p. 787)": integer,
"pmtInterval (p. 787)": integer,
"patInterval (p. 787)": integer,
"programNumber (p. 787)": integer,
"timedMetadataPid (p. 787)": integer,
"timedMetadata (p. 788)": enum,
"scte35Source (p. 788)": enum
},
"m2tsSettings (p. 740)": {
"dvbTeletextPid (p. 780)": integer,
"bitrate (p. 781)": integer,
"segmentationTime (p. 781)": number,
"audioPids (p. 781)": [integer]
"rateMode (p. 781)": enum,
"ebpAudioInterval (p. 781)": enum,
"fragmentTime (p. 781)": number,
"audioFramesPerPes (p. 781)": integer,
"maxPcrInterval (p. 782)": integer,
"scte35Pid (p. 782)": integer,
"privateMetadataPid (p. 782)": integer,
"pmtInterval (p. 782)": integer,
"segmentationStyle (p. 782)": enum,
"audioBufferModel (p. 782)": enum,
"nullPacketBitrate (p. 783)": number,
"pcrPid (p. 784)": integer,
"minEbpInterval (p. 784)": integer,
"networkName (p. 742)": "string",
"networkId (p. 742)": integer,
"nitInterval (p. 742)": integer
},
"scte35Source (p. 783)": enum,
"transportStreamId (p. 784)" : integer,
"pcrControl (p. 784)" : enum,
"videoPid (p. 784)" : integer,
"esRateInPes (p. 784)" : enum,
"segmentationMarkers (p. 785)" : enum,
"dvbTdtSettings (p. 785)" : {
  "tdtInterval (p. 747)" : integer
},
"patInterval (p. 785)" : integer,
"dvbSubPids (p. 785)" : [integer]
},
"movSettings (p. 740)" : {
  "reference (p. 789)" : enum,
  "paddingControl (p. 789)" : enum,
  "mpeg2FourCCControl (p. 789)" : enum,
  "cslgAtom (p. 789)" : enum,
  "clapAtom (p. 789)" : enum
},
"f4vSettings (p. 740)" : {
  "moovPlacement (p. 752)" : enum
},
"captionDescriptions (p. 801)" : [
  {
    "languageDescription (p. 737)" : "string",
    "languageCode (p. 737)" : enum,
    "destinationSettings (p. 737)" : {
      "burninDestinationSettings (p. 737)" : {
        "xPosition (p. 733)" : integer,
        "backgroundColor (p. 734)" : enum,
        "teletextSpacing (p. 734)" : enum,
        "yPosition (p. 734)" : integer,
        "backgroundOpacity (p. 734)" : integer,
        "fontOpacity (p. 734)" : integer,
        "shadowOpacity (p. 734)" : integer,
        "fontResolution (p. 735)" : integer,
        "shadowYOffset (p. 735)" : integer,
        "outlineSize (p. 735)" : integer,
        "outlineColor (p. 735)" : enum,
        "fontSize (p. 735)" : integer,
        "shadowXOffset (p. 735)" : integer,
        "alignment (p. 736)" : enum,
        "shadowColor (p. 736)" : enum,
        "fontColor (p. 736)" : enum
      },
      "teletextDestinationSettings (p. 738)" : {
        "pageNumber (p. 806)" : "string"
      },
      "ttmlDestinationSettings (p. 738)" : {
        "stylePassthrough (p. 807)" : enum
      },
      "destinationType (p. 738)" : enum,
      "dvbSubDestinationSettings (p. 738)" : {
        "xPosition (p. 743)" : integer,
        "backgroundColor (p. 743)" : enum,
        "teletextSpacing (p. 743)" : enum,
        "yPosition (p. 743)" : integer,
        "backgroundOpacity (p. 744)" : integer,
        "fontOpacity (p. 744)" : integer,
        "shadowOpacity (p. 744)" : integer,
        "fontResolution (p. 744)" : integer,
        "shadowYOffset (p. 744)" : integer,
        "outlineSize (p. 745)" : integer,
        "outlineColor (p. 745)" : enum
      }
    }
  }
]
"fontSize (p. 745)": integer,
"shadowXOffset (p. 745)": integer,
"alignment (p. 745)": enum,
"shadowColor (p. 745)": enum,
"fontColor (p. 745)": enum
},
"sccDestinationSettings (p. 738)": {
  "framerate (p. 805)": enum
}

Example DeletePresetResponse

{
}

Example ExceptionBody

{
  "message (p. 752)": "string"
}

Properties

AacAudioDescriptionBroadcasterMix (Enum)

Choose BROADCASTER_MIXED_AD when the input contains pre-mixed main audio + audio description (AD) as a stereo pair. The value for AudioType will be set to 3, which signals to downstream systems that this stream contains “broadcaster mixed AD”. Note that the input received by the encoder must contain pre-mixed audio; the encoder does not perform the mixing. When you choose BROADCASTER_MIXED_AD, the encoder ignores any values you provide in AudioType and FollowInputAudioType. Choose NORMAL when the input does not contain pre-mixed audio + audio description (AD). In this case, the encoder will use any values you provide for AudioType and FollowInputAudioType.

  BROADCASTER_MIXED_AD
  NORMAL

AacCodecProfile (Enum)

AAC Profile.

  LC
  HEV1
  HEV2
AacCodingMode (Enum)

Mono (Audio Description), Mono, Stereo, or 5.1 channel layout. Valid values depend on rate control mode and profile. "1.0 - Audio Description (Receiver Mix)" setting receives a stereo description plus control track and emits a mono AAC encode of the description track, with control data emitted in the PES header as per ETSI TS 101 154 Annex E.

- AD_RECEIVER_MIX
- CODING_MODE_1_0
- CODING_MODE_1_1
- CODING_MODE_2_0
- CODING_MODE_5_1

AacRateControlMode (Enum)

Rate Control Mode.

- CBR
- VBR

AacRawFormat (Enum)

Enables LATM/LOAS AAC output. Note that if you use LATM/LOAS AAC in an output, you must choose "No container" for the output container.

- LATM_LOAS
- NONE

AacSettings

vbrQuality

Type: AacVbrQuality (p. 726)
Required: False

codecProfile

Type: AacCodecProfile (p. 724)
Required: False

codingMode

Type: AacCodingMode (p. 725)
Required: True

specification

Type: AacSpecification (p. 726)
Required: False

bitrate

Average bitrate in bits/second. Valid values depend on rate control mode and profile.
Properties

Type: integer
Required: False
Minimum: 6000
Maximum: 1024000

rawFormat

Type: AacRawFormat (p. 725)
Required: False

rateControlMode

Type: AacRateControlMode (p. 725)
Required: False

sampleRate

Sample rate in Hz. Valid values depend on rate control mode and profile.

Type: integer
Required: True
Minimum: 8000
Maximum: 96000

audioDescriptionBroadcasterMix

Type: AacAudioDescriptionBroadcasterMix (p. 724)
Required: False

AacSpecification (Enum)

Use MPEG-2 AAC instead of MPEG-4 AAC audio for raw or MPEG-2 Transport Stream containers.

MPEG2
MPEG4

AacVbrQuality (Enum)

VBR quality level. Only used if the rate control mode (AacRateControlMode) is VBR.

LOW
MEDIUM_LOW
MEDIUM_HIGH
HIGH

Ac3BitstreamMode (Enum)

Specifies the "Bitstream Mode" (bsmod) for the emitted AC-3 stream. See ATSC A/52-2012 for background on these values.

COMPLETE_MAIN
COMMENTARY
DIALOGUE
EMERGENCY
HEARING_IMPAIRED
MUSIC_AND_EFFECTS
VISUALLY_IMPAIRED
VOICE_OVER

**Ac3CodingMode (Enum)**

Dolby Digital coding mode. Determines number of channels.

- CODING_MODE_1_0
- CODING_MODE_1_1
- CODING_MODE_2_0
- CODING_MODE_3_2_LFE

**Ac3DynamicRangeCompressionProfile (Enum)**

If set to FILM_STANDARD, adds dynamic range compression signaling to the output bitstream as defined in the Dolby Digital specification.

- FILM_STANDARD
- NONE

**Ac3LfeFilter (Enum)**

Applies a 120Hz lowpass filter to the LFE channel prior to encoding. Only valid with 3_2_LFE coding mode.

- ENABLED
- DISABLED

**Ac3MetadataControl (Enum)**

When set to FOLLOW_INPUT, encoder metadata will be sourced from the DD, DD+, or DolbyE decoder that supplied this audio data. If audio was not supplied from one of these streams, then the static metadata settings will be used.

- FOLLOW_INPUT
- USE_CONFIGURED

**Ac3Settings**

**dynamicRangeCompressionProfile**

- **Type:** Ac3DynamicRangeCompressionProfile (p. 727)
- **Required:** False

**dialnorm**

Sets the dialnorm for the output. If blank and input audio is Dolby Digital, dialnorm will be passed through.

- **Type:** integer
- **Required:** False
- **Minimum:** 1
Maximum: 31

**codingMode**

- **Type:** Ac3CodingMode (p. 727)
- **Required:** False

**metadataControl**

- **Type:** Ac3MetadataControl (p. 727)
- **Required:** False

**lfeFilter**

- **Type:** Ac3LfeFilter (p. 727)
- **Required:** False

**bitrate**

Average bitrate in bits/second. Valid bitrates depend on the coding mode.

- **Type:** integer
- **Required:** False
- **Minimum:** 64000
- **Maximum:** 640000

**bitstreamMode**

- **Type:** Ac3BitstreamMode (p. 726)
- **Required:** False

**sampleRate**

Sample rate in hz. Sample rate is always 48000.

- **Type:** integer
- **Required:** False
- **Minimum:** 48000
- **Maximum:** 48000

**AfdSignaling (Enum)**

This setting only applies to H.264 and MPEG2 outputs. Use Insert AFD signaling (AfdSignaling) to whether there are AFD values in the output video data and what those values are.

* Choose None to remove all AFD values from this output.
* Choose Fixed to ignore input AFD values and instead encode the value specified in the job.
* Choose Auto to calculate output AFD values based on the input AFD scaler data.

NONE
AUTO
FIXED
**AiffSettings**

**channels**

Set Channels to specify the number of channels in this output audio track. Choosing Mono in the console will give you 1 output channel; choosing Stereo will give you 2. In the API, valid values are 1 and 2.

*Type:* integer  
*Required:* False  
*Minimum:* 1  
*Maximum:* 2

**bitDepth**

Specify Bit depth (BitDepth), in bits per sample, to choose the encoding quality for this audio track.

*Type:* integer  
*Required:* False  
*Minimum:* 16  
*Maximum:* 24

**sampleRate**

Sample rate in hz.

*Type:* integer  
*Required:* False  
*Minimum:* 8000  
*Maximum:* 192000

**AntiAlias (Enum)**

Enable Anti-alias (AntiAlias) to enhance sharp edges in video output when your input resolution is much larger than your output resolution. Default is enabled.

DISABLED  
ENABLED

**AudioCodec (Enum)**

Type of Audio codec.

AAC  
MP2  
WAV  
AIFF  
AC3  
EAC3  
PASSTHROUGH

**AudioCodecSettings**

**codec**

*Type: AudioCodec (p. 729)
Required: True

**wavSettings**

Type: WavSettings (p. 812)  
Required: False

**ac3Settings**

Type: Ac3Settings (p. 727)  
Required: False

**aacSettings**

Type: AacSettings (p. 725)  
Required: False

**aiffSettings**

Type: AiffSettings (p. 729)  
Required: False

**eac3Settings**

Type: Eac3Settings (p. 749)  
Required: False

**mp2Settings**

Type: Mp2Settings (p. 789)  
Required: False

**AudioDescription**

**languageCodeControl**

Type: AudioLanguageCodeControl (p. 732)  
Required: False

**audioTypeControl**

Type: AudioTypeControl (p. 733)  
Required: False

**remixSettings**

Advanced audio remixing settings.

Type: RemixSettings (p. 804)  
Required: False
Properties

audioType

Applies only if Follow Input Audio Type is unchecked (false). A number between 0 and 255. The following are defined in ISO-IEC 13818-1: 0 = Undefined, 1 = Clean Effects, 2 = Hearing Impaired, 3 = Visually Impaired Commentary, 4-255 = Reserved.

Type: integer
Required: False
Minimum: 0
Maximum: 255

audioSourceName

Specifies which audio data to use from each input. In the simplest case, specify an "Audio Selector":#inputs-audio_selector by name based on its order within each input. For example if you specify "Audio Selector 3", then the third audio selector will be used from each input. If an input does not have an "Audio Selector 3", then the audio selector marked as "default" in that input will be used. If there is no audio selector marked as "default", silence will be inserted for the duration of that input. Alternatively, an "Audio Selector Group":#inputs-audio_selector_group name may be specified, with similar default/silence behavior. If no audio_source_name is specified, then "Audio Selector 1" will be chosen automatically.

Type: string
Required: False

codecSettings

Type: AudioCodecSettings (p. 729)
Required: True

languageCode

Indicates the language of the audio output track. The ISO 639 language specified in the 'Language Code' drop down will be used when 'Follow Input Language Code' is not selected or when 'Follow Input Language Code' is selected but there is no ISO 639 language code specified by the input.

Type: LanguageCode (p. 775)
Required: False

streamName

Used for Microsoft Smooth Streaming and Apple HLS outputs. Indicates the name displayed by the player (eg. English, or Director Commentary). Alphanumeric characters, spaces, and underscore are legal.

Type: string
Required: False
Pattern: ^[\w\s]*$

audioNormalizationSettings

Advanced audio normalization settings.

Type: AudioNormalizationSettings (p. 732)
Required: False
AudioLanguageCodeControl (Enum)

Choosing FOLLOW_INPUT will cause the ISO 639 language code of the output to follow the ISO 639 language code of the input. The language specified for languageCode' will be used when USE_CONFIGURED is selected or when FOLLOW_INPUT is selected but there is no ISO 639 language code specified by the input.

- FOLLOW_INPUT
- USE_CONFIGURED

AudioNormalizationAlgorithm (Enum)

Audio normalization algorithm to use. 1770-1 conforms to the CALM Act specification, 1770-2 conforms to the EBU R-128 specification.

- ITU_BS_1770_1
- ITU_BS_1770_2

AudioNormalizationAlgorithmControl (Enum)

When enabled the output audio is corrected using the chosen algorithm. If disabled, the audio will be measured but not adjusted.

- CORRECT_AUDIO
- MEASURE_ONLY

AudioNormalizationLoudnessLogging (Enum)

If set to LOG, log each output's audio track loudness to a CSV file.

- LOG
- DONT_LOG

AudioNormalizationPeakCalculation (Enum)

If set to TRUE_PEAK, calculate and log the TruePeak for each output's audio track loudness.

- TRUE_PEAK
- NONE

AudioNormalizationSettings

**targetLkfs**

Target LKFS(loudness) to adjust volume to. If no value is entered, a default value will be used according to the chosen algorithm. The CALM Act (1770-1) recommends a target of -24 LKFS. The EBU R-128 specification (1770-2) recommends a target of -23 LKFS.

- **Type:** number
- **Required:** False
- **Format:** float
- **Minimum:** -59.0
- **Maximum:** 0.0
algorithmControl

Type: AudioNormalizationAlgorithmControl (p. 732)
Required: False

peakCalculation

Type: AudioNormalizationPeakCalculation (p. 732)
Required: False

loudnessLogging

Type: AudioNormalizationLoudnessLogging (p. 732)
Required: False

correctionGateLevel

Content measuring above this level will be corrected to the target level. Content measuring below this level will not be corrected. Gating only applies when not using real_time_correction.

Type: integer
Required: False
Minimum: -70
Maximum: 0

algorithm

Type: AudioNormalizationAlgorithm (p. 732)
Required: False

AudioTypeControl (Enum)

When set to FOLLOW_INPUT, if the input contains an ISO 639 audio_type, then that value is passed through to the output. If the input contains no ISO 639 audio_type, the value in Audio Type is included in the output. Otherwise the value in Audio Type is included in the output. Note that this field and audioType are both ignored if audioDescriptionBroadcasterMix is set to BROADCASTER_MIXED_AD.

FOLLOW_INPUT
USE_CONFIGURED

BurninDestinationSettings

xPosition

Specifies the horizontal position of the caption relative to the left side of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the left of the output. If no explicit x_position is provided, the horizontal caption position will be determined by the alignment parameter. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: 0
**Maximum**: 2147483647

**backgroundColor**

- **Type**: BurninSubtitleBackgroundColor (p. 736)
- **Required**: False

**teletextSpacing**

- **Type**: BurninSubtitleTeletextSpacing (p. 737)
- **Required**: False

**yPosition**

Specifies the vertical position of the caption relative to the top of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the top of the output. If no explicit y_position is provided, the caption will be positioned towards the bottom of the output. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 2147483647

**backgroundOpacity**

Specifies the opacity of the background rectangle. 255 is opaque; 0 is transparent. Leaving this parameter blank is equivalent to setting it to 0 (transparent). All burn-in and DVB-Sub font settings must match.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 255

**fontOpacity**

Specifies the opacity of the burned-in captions. 255 is opaque; 0 is transparent. All burn-in and DVB-Sub font settings must match.

- **Type**: integer
- **Required**: True
- **Minimum**: 0
- **Maximum**: 255

**shadowOpacity**

Specifies the opacity of the shadow. 255 is opaque; 0 is transparent. Leaving this parameter blank is equivalent to setting it to 0 (transparent). All burn-in and DVB-Sub font settings must match.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 255
**fontResolution**

Font resolution in DPI (dots per inch); default is 96 dpi. All burn-in and DVB-Sub font settings must match.

- **Type:** integer
- **Required:** False
- **Minimum:** 96
- **Maximum:** 600

**shadowYOffset**

Specifies the vertical offset of the shadow relative to the captions in pixels. A value of -2 would result in a shadow offset 2 pixels above the text. All burn-in and DVB-Sub font settings must match.

- **Type:** integer
- **Required:** False
- **Minimum:** -2147483648
- **Maximum:** 2147483647

**outlineSize**

Specifies font outline size in pixels. This option is not valid for source captions that are either 608/ embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

- **Type:** integer
- **Required:** True
- **Minimum:** 0
- **Maximum:** 10

**outlineColor**

- **Type:** BurninSubtitleOutlineColor (p. 736)
- **Required:** True

**fontSize**

A positive integer indicates the exact font size in points. Set to 0 for automatic font size selection. All burn-in and DVB-Sub font settings must match.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 96

**shadowXOffset**

Specifies the horizontal offset of the shadow relative to the captions in pixels. A value of -2 would result in a shadow offset 2 pixels to the left. All burn-in and DVB-Sub font settings must match.

- **Type:** integer
- **Required:** False
- **Minimum:** -2147483648
- **Maximum:** 2147483647
alignment

Type: BurninSubtitleAlignment (p. 736)
Required: True

shadowColor

Type: BurninSubtitleShadowColor (p. 737)
Required: False

fontColor

Type: BurninSubtitleFontColor (p. 736)
Required: False

BurninSubtitleAlignment (Enum)

If no explicit x_position or y_position is provided, setting alignment to centered will place the captions at the bottom center of the output. Similarly, setting a left alignment will align captions to the bottom left of the output. If x and y positions are given in conjunction with the alignment parameter, the font will be justified (either left or centered) relative to those coordinates. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

CENTERED
LEFT

BurninSubtitleBackgroundColor (Enum)

Specifies the color of the rectangle behind the captions. All burn-in and DVB-Sub font settings must match.

NONE
BLACK
WHITE

BurninSubtitleFontColor (Enum)

Specifies the color of the burned-in captions. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

WHITE
BLACK
YELLOW
RED
GREEN
BLUE

BurninSubtitleOutlineColor (Enum)

Specifies font outline color. This option is not valid for source captions that are either 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.
BLACK
WHITE
YELLOW
RED
GREEN
BLUE

**BurninSubtitleShadowColor (Enum)**

Specifies the color of the shadow cast by the captions. All burn-in and DVB-Sub font settings must match.

- NONE
- BLACK
- WHITE

**BurninSubtitleTeletextSpacing (Enum)**

Controls whether a fixed grid size or proportional font spacing will be used to generate the output subtitles bitmap. Only applicable for Teletext inputs and DVB-Sub/Burn-in outputs.

- FIXED_GRID
- PROPORTIONAL

**CaptionDescriptionPreset**

**LanguageDescription**

Human readable information to indicate captions available for players (e.g., English, or Spanish). Alphanumeric characters, spaces, and underscore are legal.

- **Type:** string
- **Required:** False
- **Pattern:** `^[\w\ ]*$`

**LanguageCode**

Indicates the language of the caption output track.

- **Type:** `LanguageCode (p. 775)`
- **Required:** False

**DestinationSettings**

- **Type:** `CaptionDestinationSettings (p. 737)`
- **Required:** False

**CaptionDestinationSettings**

**BurninDestinationSettings**

- **Type:** `BurninDestinationSettings (p. 733)`
**Required**: False

**teletextDestinationSettings**

*Type*: TeletextDestinationSettings (p. 806)
*Required*: False

**ttmlDestinationSettings**

*Type*: TtmlDestinationSettings (p. 807)
*Required*: False

**destinationType**

*Type*: CaptionDestinationType (p. 738)
*Required*: True

**dvbSubDestinationSettings**

*Type*: DvbSubDestinationSettings (p. 743)
*Required*: False

**sccDestinationSettings**

*Type*: SccDestinationSettings (p. 805)
*Required*: False

**CaptionDestinationType (Enum)**

- BURN_IN
- DVB_SUB
- EMBEDDED
- SCC
- SRT
- TELETEXT
- TTML
- WEBVTT

**ChannelMapping**

**outputChannels**

*Type*: Array of type OutputChannelMapping (p. 799)
*Required*: True

**ColorCorrector**

**saturation**

Saturation level.

*Type*: integer
brightness

Brightness level.

Type: integer
Required: False
Minimum: 1
Maximum: 100

hdr10Metadata

Type: Hdr10Metadata (p. 771)
Required: False

contrast

Contrast level.

Type: integer
Required: False
Minimum: 1
Maximum: 100

hue

Hue in degrees.

Type: integer
Required: False
Minimum: -180
Maximum: 180

colorSpaceConversion

Type: ColorSpaceConversion (p. 739)
Required: False

ColorMetadata (Enum)

Enable Insert color metadata (ColorMetadata) to include color metadata in this output. This setting is enabled by default.

IGNORE
INSERT

ColorSpaceConversion (Enum)

Determines if colorspace conversion will be performed. If set to _None_, no conversion will be performed. If _Force 601_ or _Force 709_ are selected, conversion will be performed for inputs with
differing colorspaces. An input's colorspace can be specified explicitly in the "Video Selector":#inputs-
video_selector if necessary.

| NONE          |
| FORCE_601    |
| FORCE_709    |
| FORCE_HDR10  |
| FORCE_HLG_2020 |

**ContainerSettings**

**container**

<table>
<thead>
<tr>
<th>Type: ContainerType (p. 740)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Required: True</td>
</tr>
</tbody>
</table>

**mp4Settings**

<table>
<thead>
<tr>
<th>Type: Mp4Settings (p. 790)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Required: False</td>
</tr>
</tbody>
</table>

**m3u8Settings**

<table>
<thead>
<tr>
<th>Type: M3u8Settings (p. 785)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Required: False</td>
</tr>
</tbody>
</table>

**m2tsSettings**

<table>
<thead>
<tr>
<th>Type: M2tsSettings (p. 780)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Required: False</td>
</tr>
</tbody>
</table>

**movSettings**

<table>
<thead>
<tr>
<th>Type: MovSettings (p. 789)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Required: False</td>
</tr>
</tbody>
</table>

**f4vSettings**

<table>
<thead>
<tr>
<th>Type: F4vSettings (p. 752)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Required: False</td>
</tr>
</tbody>
</table>

**ContainerType (Enum)**

Container for this output. Some containers require a container settings object. If not specified, the
default object will be created.

<table>
<thead>
<tr>
<th>F4V</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISMV</td>
</tr>
<tr>
<td>M2TS</td>
</tr>
<tr>
<td>M3U8</td>
</tr>
<tr>
<td>MOV</td>
</tr>
<tr>
<td>MP4</td>
</tr>
<tr>
<td>MPD</td>
</tr>
</tbody>
</table>
DeinterlaceAlgorithm (Enum)

Only applies when you set Deinterlace (DeinterlaceMode) to Deinterlace (DEINTERLACE) or Adaptive (ADAPTIVE). Motion adaptive interpolate (INTERPOLATE) produces sharper pictures, while blend (BLEND) produces smoother motion. Use (INTERPOLATE_TICKER) OR (BLEND_TICKER) if your source file includes a ticker, such as a scrolling headline at the bottom of the frame.

INTERPOLATE
INTERPOLATE_TICKER
BLEND
BLEND_TICKER

Deinterlacer

mode

Type: DeinterlaceMode (p. 741)
Required: False

control

Type: DeinterlacerControl (p. 741)
Required: False

algorithm

Type: DeinterlaceAlgorithm (p. 741)
Required: False

DeinterlacerControl (Enum)

- When set to NORMAL (default), the deinterlacer does not convert frames that are tagged in metadata as progressive. It will only convert those that are tagged as some other type.
- When set to FORCE_ALL_FRAMES, the deinterlacer converts every frame to progressive - even those that are already tagged as progressive. Turn Force mode on only if there is a good chance that the metadata has tagged frames as progressive when they are not progressive. Do not turn on otherwise; processing frames that are already progressive into progressive will probably result in lower quality video.

FORCE_ALL_FRAMES
NORMAL

DeinterlacerMode (Enum)

Use Deinterlace (DeinterlaceMode) to choose how the service will do deinterlacing. Default is Deinterlace. - Deinterlace converts interlaced to progressive. - Inverse telecine converts Hard Telecine 29.97i to progressive 23.976p. - Adaptive auto-detects and converts to progressive.

DEINTERLACE
INVERSE_TELECINE
ADAPTIVE
DeletePresetRequest

name

Type: string
Required: True

DeletePresetResponse

DropFrameTimecode (Enum)
Applies only to 29.97 fps outputs. When this feature is enabled, the service will use drop-frame timecode on outputs. If it is not possible to use drop-frame timecode, the system will fall back to non-drop-frame. This setting is enabled by default when Timecode insertion (TimecodeInsertion) is enabled.

DISABLED
ENABLED

DvbNitSettings

networkName
The network name text placed in the network_name_descriptor inside the Network Information Table. Maximum length is 256 characters.

Type: string
Required: True

networkId
The numeric value placed in the Network Information Table (NIT).

Type: integer
Required: True
Minimum: 0
Maximum: 65535

nitInterval
The number of milliseconds between instances of this table in the output transport stream.

Type: integer
Required: True
Minimum: 25
Maximum: 10000

DvbSdtSettings

sdtInterval
The number of milliseconds between instances of this table in the output transport stream.

Type: integer
Required: False
Minimum: 25  
Maximum: 2000

**serviceName**

The service name placed in the service_descriptor in the Service Description Table. Maximum length is 256 characters.

  Type: string  
  Required: False

**serviceProviderName**

The service provider name placed in the service_descriptor in the Service Description Table. Maximum length is 256 characters.

  Type: string  
  Required: False

**outputSdt**

  Type: OutputSdt (p. 799)  
  Required: False

**DvbSubDestinationSettings**

**xPosition**

Specifies the horizontal position of the caption relative to the left side of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the left of the output. If no explicit x_position is provided, the horizontal caption position will be determined by the alignment parameter. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

  Type: integer  
  Required: False  
  Minimum: 0  
  Maximum: 2147483647

**backgroundColor**

  Type: DvbSubtitleBackgroundColor (p. 746)  
  Required: False

**teletextSpacing**

  Type: DvbSubtitleTeletextSpacing (p. 747)  
  Required: False

**yPosition**

Specifies the vertical position of the caption relative to the top of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the top of the output. If no explicit y_position is
provided, the caption will be positioned towards the bottom of the output. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

**Type**: integer  
**Required**: False  
**Minimum**: 0  
**Maximum**: 2147483647

**backgroundColor**
Specifies the opacity of the background rectangle. 255 is opaque; 0 is transparent. Leaving this parameter blank is equivalent to setting it to 0 (transparent). All burn-in and DVB-Sub font settings must match.

**Type**: integer  
**Required**: False  
**Minimum**: 0  
**Maximum**: 255

**fontOpacity**
Specifies the opacity of the burned-in captions. 255 is opaque; 0 is transparent. All burn-in and DVB-Sub font settings must match.

**Type**: integer  
**Required**: True  
**Minimum**: 0  
**Maximum**: 255

**shadowOpacity**
Specifies the opacity of the shadow. 255 is opaque; 0 is transparent. Leaving this parameter blank is equivalent to setting it to 0 (transparent). All burn-in and DVB-Sub font settings must match.

**Type**: integer  
**Required**: False  
**Minimum**: 0  
**Maximum**: 255

**fontResolution**
Font resolution in DPI (dots per inch); default is 96 dpi. All burn-in and DVB-Sub font settings must match.

**Type**: integer  
**Required**: False  
**Minimum**: 96  
**Maximum**: 600

**shadowYOffset**
Specifies the vertical offset of the shadow relative to the captions in pixels. A value of -2 would result in a shadow offset 2 pixels above the text. All burn-in and DVB-Sub font settings must match.
### Properties

**Type**: integer  
**Required**: False  
**Minimum**: -2147483648  
**Maximum**: 2147483647

**outlineSize**

Specifies font outline size in pixels. This option is not valid for source captions that are either 608/ embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

**Type**: integer  
**Required**: True  
**Minimum**: 0  
**Maximum**: 10

**outlineColor**

- **Type**: DvbSubtitleOutlineColor (p. 746)  
- **Required**: True

**fontSize**

A positive integer indicates the exact font size in points. Set to 0 for automatic font size selection. All burn-in and DVB-Sub font settings must match.

**Type**: integer  
**Required**: False  
**Minimum**: 0  
**Maximum**: 96

**shadowXOffset**

Specifies the horizontal offset of the shadow relative to the captions in pixels. A value of -2 would result in a shadow offset 2 pixels to the left. All burn-in and DVB-Sub font settings must match.

**Type**: integer  
**Required**: False  
**Minimum**: -2147483648  
**Maximum**: 2147483647

**alignment**

- **Type**: DvbSubtitleAlignment (p. 746)  
- **Required**: True

**shadowColor**

- **Type**: DvbSubtitleShadowColor (p. 746)  
- **Required**: False

**fontColor**

- **Type**: DvbSubtitleFontColor (p. 746)
Required: False

DvbSubtitleAlignment (Enum)

If no explicit x_position or y_position is provided, setting alignment to centered will place the captions at the bottom center of the output. Similarly, setting a left alignment will align captions to the bottom left of the output. If x and y positions are given in conjunction with the alignment parameter, the font will be justified (either left or centered) relative to those coordinates. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

- CENTERED
- LEFT

DvbSubtitleBackgroundColor (Enum)

Specifies the color of the rectangle behind the captions. All burn-in and DVB-Sub font settings must match.

- NONE
- BLACK
- WHITE

DvbSubtitleFontColor (Enum)

Specifies the color of the burned-in captions. This option is not valid for source captions that are STL, 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

- WHITE
- BLACK
- YELLOW
- RED
- GREEN
- BLUE

DvbSubtitleOutlineColor (Enum)

Specifies font outline color. This option is not valid for source captions that are either 608/embedded or teletext. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

- BLACK
- WHITE
- YELLOW
- RED
- GREEN
- BLUE

DvbSubtitleShadowColor (Enum)

Specifies the color of the shadow cast by the captions. All burn-in and DVB-Sub font settings must match.

- NONE
DvbSubtitleTeletextSpacing (Enum)

Controls whether a fixed grid size or proportional font spacing will be used to generate the output subtitles bitmap. Only applicable for Teletext inputs and DVB-Sub/Burn-in outputs.

- FIXED_GRID
- PROPORTIONAL

DvbTdtSettings

tdtInterval

The number of milliseconds between instances of this table in the output transport stream.

- **Type**: integer
- **Required**: True
- **Minimum**: 1000
- **Maximum**: 30000

Eac3AttenuationControl (Enum)

If set to ATTENUATE_3_DB, applies a 3 dB attenuation to the surround channels. Only used for 3/2 coding mode.

- ATTENUATE_3_DB
- NONE

Eac3BitstreamMode (Enum)

Specifies the "Bitstream Mode" (bsmod) for the emitted E-AC-3 stream. See ATSC A/52-2012 (Annex E) for background on these values.

- COMPLETE_MAIN
- COMMENTARY
- EMERGENCY
- HEARING_IMPAIRED
- VISUALLY_IMPAIRED

Eac3CodingMode (Enum)

Dolby Digital Plus coding mode. Determines number of channels.

- CODING_MODE_1_0
- CODING_MODE_2_0
- CODING_MODE_3_2

Eac3DcFilter (Enum)

Activates a DC highpass filter for all input channels.

- ENABLED
DISABLED

Eac3DynamicRangeCompressionLine (Enum)
Enables Dynamic Range Compression that restricts the absolute peak level for a signal.

- NONE
- FILM_STANDARD
- FILM_LIGHT
- MUSIC_STANDARD
- MUSIC_LIGHT
- SPEECH

Eac3DynamicRangeCompressionRf (Enum)
Enables Heavy Dynamic Range Compression, ensures that the instantaneous signal peaks do not exceed specified levels.

- NONE
- FILM_STANDARD
- FILM_LIGHT
- MUSIC_STANDARD
- MUSIC_LIGHT
- SPEECH

Eac3LfeControl (Enum)
When encoding 3/2 audio, controls whether the LFE channel is enabled

- LFE
- NO_LFE

Eac3LfeFilter (Enum)
Applies a 120Hz lowpass filter to the LFE channel prior to encoding. Only valid with 3_2_LFE coding mode.

- ENABLED
- DISABLED

Eac3MetadataControl (Enum)
When set to FOLLOW_INPUT, encoder metadata will be sourced from the DD, DD+, or DolbyE decoder that supplied this audio data. If audio was not supplied from one of these streams, then the static metadata settings will be used.

- FOLLOW_INPUT
- USE_CONFIGURED

Eac3PassthroughControl (Enum)
When set to WHEN_POSSIBLE, input DD+ audio will be passed through if it is present on the input. This detection is dynamic over the life of the transcode. Inputs that alternate between DD+ and non-DD+ content will have a consistent DD+ output as the system alternates between passthrough and encoding.
Eac3PhaseControl (Enum)

Controls the amount of phase-shift applied to the surround channels. Only used for 3/2 coding mode.

- **SHIFT_90_DEGREES**
- **NO_SHIFT**

Eac3Settings

dialnorm

Sets the dialnorm for the output. If blank and input audio is Dolby Digital Plus, dialnorm will be passed through.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 31

passthroughControl

- **Type**: Eac3PassthroughControl (p. 748)
- **Required**: False

metadataControl

- **Type**: Eac3MetadataControl (p. 748)
- **Required**: False

bitrate

Average bitrate in bits/second. Valid bitrates depend on the coding mode.

- **Type**: integer
- **Required**: False
- **Minimum**: 64000
- **Maximum**: 640000

dynamicRangeCompressionRf

- **Type**: Eac3DynamicRangeCompressionRf (p. 748)
- **Required**: False

sampleRate

Sample rate in hertz. Sample rate is always 48000.

- **Type**: integer
- **Required**: False
- **Minimum**: 48000

749
**Properties**

**Maximum:** 48000

**LtRtSurroundMixLevel**

Left total/Right total surround mix level. Only used for 3/2 coding mode. Valid values: -1.5 -3.0 -4.5 -6.0 -60

Type: number  
Required: False  
Format: float  
Minimum: -60.0  
Maximum: -1.5

**surroundExMode**

Type: Eac3SurroundExMode (p. 752)  
Required: False

**dynamicRangeCompressionLine**

Type: Eac3DynamicRangeCompressionLine (p. 748)  
Required: False

**lfeControl**

Type: Eac3LfeControl (p. 748)  
Required: False

**codingMode**

Type: Eac3CodingMode (p. 747)  
Required: False

**surroundMode**

Type: Eac3SurroundMode (p. 752)  
Required: False

**attenuationControl**

Type: Eac3AttenuationControl (p. 747)  
Required: False

**lfeFilter**

Type: Eac3LfeFilter (p. 748)  
Required: False

**phaseControl**

Type: Eac3PhaseControl (p. 749)  
Required: False
ltRtCenterMixLevel

Left total/Right total center mix level. Only used for 3/2 coding mode. Valid values: 3.0, 1.5, 0.0, -1.5 -3.0 -4.5 -6.0 -60

- Type: number
- Required: False
- Format: float
- Minimum: -60.0
- Maximum: 3.0

dcFilter

- Type: Eac3DcFilter (p. 747)
- Required: False

stereoDownmix

- Type: Eac3StereoDownmix (p. 751)
- Required: False

bitstreamMode

- Type: Eac3BitstreamMode (p. 747)
- Required: False

loRoSurroundMixLevel

Left only/Right only surround mix level. Only used for 3/2 coding mode. Valid values: -1.5 -3.0 -4.5 -6.0 -60

- Type: number
- Required: False
- Format: float
- Minimum: -60.0
- Maximum: -1.5

loRoCenterMixLevel

Left only/Right only center mix level. Only used for 3/2 coding mode. Valid values: 3.0, 1.5, 0.0, -1.5 -3.0 -4.5 -6.0 -60

- Type: number
- Required: False
- Format: float
- Minimum: -60.0
- Maximum: 3.0

Eac3StereoDownmix (Enum)

Stereo downmix preference. Only used for 3/2 coding mode.

- NOT_INDICATED
- LO_RO
Eac3SurroundExMode (Enum)

When encoding 3/2 audio, sets whether an extra center back surround channel is matrix encoded into the left and right surround channels.

- NOT_INDICATED
- ENABLED
- DISABLED

Eac3SurroundMode (Enum)

When encoding 2/0 audio, sets whether Dolby Surround is matrix encoded into the two channels.

- NOT_INDICATED
- ENABLED
- DISABLED

ExceptionBody

message

Type: string
Required: False

F4vMoovPlacement (Enum)

If set to PROGRESSIVE_DOWNLOAD, the MOOV atom is relocated to the beginning of the archive as required for progressive downloading. Otherwise it is placed normally at the end.

- PROGRESSIVE_DOWNLOAD
- NORMAL

F4vSettings

moovPlacement

Type: F4vMoovPlacement (p. 752)
Required: False

FrameCaptureSettings

framerateDenominator

Frame capture will encode the first frame of the output stream, then one frame every framerateDenominator/framerateNumerator seconds. For example, settings of framerateNumerator = 1 and framerateDenominator = 3 (a rate of 1/3 frame per second) will capture the first frame, then 1 frame every 3s. Files will be named as filename.n.jpg where n is the 0-based sequence number of each Capture.

Type: integer
**maxCaptures**

Maximum number of captures (encoded jpg output files).

**Type:** integer  
**Required:** False  
**Minimum:** 1  
**Maximum:** 2147483647

**framerateNumerator**

Frame capture will encode the first frame of the output stream, then one frame every \( \frac{\text{framerateDenominator}}{\text{framerateNumerator}} \) seconds. For example, settings of \( \text{framerateNumerator} = 1 \) and \( \text{framerateDenominator} = 3 \) (a rate of 1/3 frame per second) will capture the first frame, then 1 frame every 3s. Files will be named as \( \text{filename.NNNNNNN.jpg} \) where \( N \) is the 0-based frame sequence number zero padded to 7 decimal places.

**Type:** integer  
**Required:** False  
**Minimum:** 1  
**Maximum:** 2147483647

**quality**

JPEG Quality - a higher value equals higher quality.

**Type:** integer  
**Required:** False  
**Minimum:** 1  
**Maximum:** 100

**GetPresetRequest**

**name**

**Type:** string  
**Required:** True

**GetPresetResponse**

**preset**

**Type:** Preset (p. 799)  
**Required:** False

**H264AdaptiveQuantization (Enum)**

Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.
OFF
LOW
MEDIUM
HIGH
HIGHER
MAX

**H264CodecLevel (Enum)**

H.264 Level.

- AUTO
- LEVEL_1
- LEVEL_1_1
- LEVEL_1_2
- LEVEL_1_3
- LEVEL_2
- LEVEL_2_1
- LEVEL_2_2
- LEVEL_3
- LEVEL_3_1
- LEVEL_3_2
- LEVEL_4
- LEVEL_4_1
- LEVEL_4_2
- LEVEL_5
- LEVEL_5_1
- LEVEL_5_2

**H264CodecProfile (Enum)**

H.264 Profile. High 4:2:2 and 10-bit profiles are only available with the AVC-I License.

- BASELINE
- HIGH
- HIGH_10BIT
- HIGH_422
- HIGH_422_10BIT
- MAIN

**H264EntropyEncoding (Enum)**

Entropy encoding mode. Use CABAC (must be in Main or High profile) or CAVLC.

- CABAC
- CAVLC

**H264FieldEncoding (Enum)**

Choosing FORCE_FIELD disables PAFF encoding for interlaced outputs.

- PAFF
- FORCE_FIELD
H264FlickerAdaptiveQuantization (Enum)

Adjust quantization within each frame to reduce flicker or 'pop' on I-frames.

   DISABLED
   ENABLED

H264FramerateControl (Enum)

Using the API, set FramerateControl to INITIALIZE_FROM_SOURCE if you want the service to use the framerate from the input. Using the console, do this by choosing INITIALIZE_FROM_SOURCE for Framerate.

   INITIALIZE_FROM_SOURCE
   SPECIFIED

H264FramerateConversionAlgorithm (Enum)

When set to INTERPOLATE, produces smoother motion during framerate conversion.

   DUPLICATE_DROP
   INTERPOLATE

H264GopBReference (Enum)

If enable, use reference B frames for GOP structures that have B frames > 1.

   DISABLED
   ENABLED

H264GopSizeUnits (Enum)

Indicates if the GOP Size in H264 is specified in frames or seconds. If seconds the system will convert the GOP Size into a frame count at run time.

   FRAMES
   SECONDS

H264InterlaceMode (Enum)

Use Interlace mode (InterlaceMode) to choose the scan line type for the output.

* Top Field First (TOP_FIELD) and Bottom Field First (BOTTOM_FIELD) produce interlaced output with the entire output having the same field polarity (top or bottom first).

* Follow, Default Top (FOLLOW_TOP_FIELD) and Follow, Default Bottom (FOLLOW_BOTTOM_FIELD) use the same field polarity as the source. Therefore, behavior depends on the input scan type. If the source is interlaced, the output will be interlaced with the same polarity as the source (it will follow the source). The output could therefore be a mix of "top field first" and "bottom field first". If the source is progressive, the output will be interlaced with "top field first" or "bottom field first" polarity, depending on which of the Follow options you chose.

   PROGRESSIVE
   TOP_FIELD
   BOTTOM_FIELD
FOLLOW_TOP_FIELD
FOLLOW.Bottom_FIELD

H264ParControl (Enum)
Using the API, enable ParFollowSource if you want the service to use the pixel aspect ratio from the input. Using the console, do this by choosing Follow source for Pixel aspect ratio.

INITIALIZE_FROM_SOURCE
SPECIFIED

H264QualityTuningLevel (Enum)
Use Quality tuning level (H264QualityTuningLevel) to specify whether to use fast single-pass, high-quality singlepass, or high-quality multipass video encoding.

SINGLE_PASS
SINGLE_PASS_HQ
MULTI_PASS_HQ

H264RateControlMode (Enum)
Rate control mode. CQ uses constant quantizer (qp), ABR (average bitrate) does not write HRD parameters.

VBR
CBR

H264RepeatPps (Enum)
Places a PPS header on each encoded picture, even if repeated.

DISABLED
ENABLED

H264SceneChangeDetect (Enum)
Scene change detection (inserts I-frames on scene changes).

DISABLED
ENABLED

H264Settings

slices
Number of slices per picture. Must be less than or equal to the number of macroblock rows for progressive pictures, and less than or equal to half the number of macroblock rows for interlaced pictures.

Type: integer
Required: False
Minimum: 1
Maximum: 32

minIInterval

Enforces separation between repeated (cadence) I-frames and I-frames inserted by Scene Change Detection. If a scene change I-frame is within I-interval frames of a cadence I-frame, the GOP is shrunk and/or stretched to the scene change I-frame. GOP stretch requires enabling lookahead as well as setting I-interval. The normal cadence resumes for the next GOP. This setting is only used when Scene Change Detect is enabled. Note: Maximum GOP stretch = GOP size + Min-I-interval - 1

Type: integer
Required: False
Minimum: 0
Maximum: 30

parNumerator

Pixel Aspect Ratio numerator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

flickerAdaptiveQuantization

Type: H264FlickerAdaptiveQuantization (p. 755)
Required: False

gopSizeUnits

Type: H264GopSizeUnits (p. 755)
Required: False

hrdBufferSize

Size of buffer (HRD buffer model). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

qualityTuningLevel

Type: H264QualityTuningLevel (p. 756)
Required: False

maxBitrate

Maximum bitrate in bits/second (for VBR mode only). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.
Properties

**Type**: integer  
**Required**: False
**Minimum**: -2147483648  
**Maximum**: 2147483647

**bitrate**
Average bitrate in bits/second. Required for VBR, CBR, and ABR. Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.

**Type**: integer  
**Required**: False
**Minimum**: 1000  
**Maximum**: 2147483647

**spatialAdaptiveQuantization**

**Type**: H264SpatialAdaptiveQuantization (p. 761)  
**Required**: False

**slowPal**

**Type**: H264SlowPal (p. 761)  
**Required**: False

**codecProfile**

**Type**: H264CodecProfile (p. 754)  
**Required**: False

**unregisteredSeiTImecode**

**Type**: H264UnregisteredSeiTImecode (p. 762)  
**Required**: False

**softness**
Softness. Selects quantizer matrix, larger values reduce high-frequency content in the encoded image.

**Type**: integer  
**Required**: False
**Minimum**: 0  
**Maximum**: 128

**framerateControl**

**Type**: H264FramerateControl (p. 755)  
**Required**: False

**telecine**

**Type**: H264Telecine (p. 762)
Properties

Required: False

framerateConversionAlgorithm

Type: H264FramerateConversionAlgorithm (p. 755)
Required: False

codecLevel

Type: H264CodecLevel (p. 754)
Required: False

numberReferenceFrames

Number of reference frames to use. The encoder may use more than requested if using B-frames and/or interlaced encoding.

Type: integer
Required: False
Minimum: 1
Maximum: 6

temporalAdaptiveQuantization

Type: H264TemporalAdaptiveQuantization (p. 762)
Required: False

repeatPps

Type: H264RepeatPps (p. 756)
Required: False

hrdBufferInitialFillPercentage

Percentage of the buffer that should initially be filled (HRD buffer model).

Type: integer
Required: False
Minimum: 0
Maximum: 100

framerateNumerator

Framerate numerator - framerate is a fraction, e.g. 24000 / 1001 = 23.976 fps.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

numberBFramesBetweenReferenceFrames

Number of B-frames between reference frames.
Type: integer
Required: False
Minimum: 0
Maximum: 7

gopClosedCadence

Frequency of closed GOPs. In streaming applications, it is recommended that this be set to 1 so a decoder joining mid-stream will receive an IDR frame as quickly as possible. Setting this value to 0 will break output segmenting.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

fieldEncoding

Type: H264FieldEncoding (p. 754)
Required: False

entropyEncoding

Type: H264EntropyEncoding (p. 754)
Required: False

framerateDenominator

When you use the API for transcode jobs that use framerate conversion, specify the framerate as a fraction. For example, 24000 / 1001 = 23.976 fps. Use FramerateDenominator to specify the denominator of this fraction. In this example, use 1001 for the value of FramerateDenominator. When you use the console for transcode jobs that use framerate conversion, provide the value as a decimal number for Framerate. In this example, specify 23.976.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

adaptiveQuantization

Type: H264AdaptiveQuantization (p. 753)
Required: False

interlaceMode

Type: H264InterlaceMode (p. 755)
Required: False

gopSize

GOP Length (keyframe interval) in frames or seconds. Must be greater than zero.
**Properties**

- **Type**: number
  - **Required**: False
  - **Format**: float
  - **Minimum**: 0.0

**gopBReference**

- **Type**: H264GopBReference (p. 755)
- **Required**: False

**sceneChangeDetect**

- **Type**: H264SceneChangeDetect (p. 756)
- **Required**: False

**parDenominator**

Pixel Aspect Ratio denominator.

- **Type**: integer
- **Required**: False
  - **Minimum**: 1
  - **Maximum**: 2147483647

**parControl**

- **Type**: H264ParControl (p. 756)
- **Required**: False

**syntax**

- **Type**: H264Syntax (p. 762)
- **Required**: False

**rateControlMode**

- **Type**: H264RateControlMode (p. 756)
- **Required**: False

**H264SlowPal (Enum)**

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

- **DISABLED**
- **ENABLED**

**H264SpatialAdaptiveQuantization (Enum)**

Adjust quantization within each frame based on spatial variation of content complexity.

- **DISABLED**
**ENABLED**

**H264Syntax (Enum)**

Produces a bitstream compliant with SMPTE RP-2027.

- DEFAULT
- RP2027

**H264Telecine (Enum)**

This field applies only if the Streams > Advanced > Framerate (framerate) field is set to 29.970. This field works with the Streams > Advanced > Preprocessors > Deinterlacer field (deinterlace_mode) and the Streams > Advanced > Interlaced Mode field (interlace_mode) to identify the scan type for the output:
- Progressive, Interlaced, Hard Telecine or Soft Telecine.
- Hard: produces 29.97i output from 23.976 input.
- Soft: produces 23.976; the player converts this output to 29.97i.

- NONE
- SOFT
- HARD

**H264TemporalAdaptiveQuantization (Enum)**

Adjust quantization within each frame based on temporal variation of content complexity.

- DISABLED
- ENABLED

**H264UnregisteredSeiTImecode (Enum)**

Inserts timecode for each frame as 4 bytes of an unregistered SEI message.

- DISABLED
- ENABLED

**H265AdaptiveQuantization (Enum)**

Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

- OFF
- LOW
- MEDIUM
- HIGH
- HIGHER
- MAX

**H265AlternateTransferFunctionSei (Enum)**

Enables Alternate Transfer Function SEI message for outputs using Hybrid Log Gamma (HLG) Electro-Optical Transfer Function (EOTF).

- DISABLED
- ENABLED
H265CodecLevel (Enum)

H.265 Level.

AUTO
LEVEL_1
LEVEL_2
LEVEL_2_1
LEVEL_3
LEVEL_3_1
LEVEL_4
LEVEL_4_1
LEVEL_5
LEVEL_5_1
LEVEL_5_2
LEVEL_6
LEVEL_6_1
LEVEL_6_2

H265CodecProfile (Enum)

Represents the Profile and Tier, per the HEVC (H.265) specification. Selections are grouped as [Profile] / [Tier], so "Main/High" represents Main Profile with High Tier. 4:2:2 profiles are only available with the HEVC 4:2:2 License.

MAIN_MAIN
MAIN_HIGH
MAIN10_MAIN
MAIN10_HIGH
MAIN_422_8BIT_MAIN
MAIN_422_8BIT_HIGH
MAIN_422_10BIT_MAIN
MAIN_422_10BIT_HIGH

H265FlickerAdaptiveQuantization (Enum)

Adjust quantization within each frame to reduce flicker or 'pop' on I-frames.

DISABLED
ENABLED

H265FramerateControl (Enum)

Using the API, set FramerateControl to INITIALIZE_FROM_SOURCE if you want the service to use the framerate from the input. Using the console, do this by choosing INITIALIZE_FROM_SOURCE for Framerate.

INITIALIZE_FROM_SOURCE
SPECIFIED

H265FramerateConversionAlgorithm (Enum)

When set to INTERPOLATE, produces smoother motion during framerate conversion.
DUPLICATE_DROP
INTERPOLATE

**H265GopBReference (Enum)**

If enable, use reference B frames for GOP structures that have B frames > 1.

DISABLED
ENABLED

**H265GopSizeUnits (Enum)**

Indicates if the GOP Size in H265 is specified in frames or seconds. If seconds the system will convert the GOP Size into a frame count at run time.

FRAMES
SECONDS

**H265InterlaceMode (Enum)**

Use Interlace mode (InterlaceMode) to choose the scan line type for the output.

* Top Field First (TOP_FIELD) and Bottom Field First (BOTTOM_FIELD) produce interlaced output with the entire output having the same field polarity (top or bottom first).

* Follow, Default Top (FOLLOW_TOP_FIELD) and Follow, Default Bottom (FOLLOW_BOTTOM_FIELD) use the same field polarity as the source. Therefore, behavior depends on the input scan type. If the source is interlaced, the output will be interlaced with the same polarity as the source (it will follow the source). The output could therefore be a mix of "top field first" and "bottom field first". If the source is progressive, the output will be interlaced with "top field first" or "bottom field first" polarity, depending on which of the Follow options you chose.

PROGRESSIVE
TOP_FIELD
BOTTOM_FIELD
FOLLOW_TOP_FIELD
FOLLOW_BOTTOM_FIELD

**H265ParControl (Enum)**

Using the API, enable ParFollowSource if you want the service to use the pixel aspect ratio from the input. Using the console, do this by choosing Follow source for Pixel aspect ratio.

INITIALIZE_FROM_SOURCE
SPECIFIED

**H265QualityTuningLevel (Enum)**

Use Quality tuning level (H265QualityTuningLevel) to specify whether to use fast single-pass, high-quality singlepass, or high-quality multipass video encoding.

SINGLE_PASS
SINGLE_PASS_HQ
MULTI_PASS_HQ
H265RateControlMode (Enum)

Rate control mode. CQ uses constant quantizer (qp), ABR (average bitrate) does not write HRD parameters.

- VBR
- CBR

H265SampleAdaptiveOffsetFilterMode (Enum)

Specify Sample Adaptive Offset (SAO) filter strength. Adaptive mode dynamically selects best strength based on content.

- DEFAULT
- ADAPTIVE
- OFF

H265SceneChangeDetect (Enum)

Scene change detection (inserts I-frames on scene changes).

- DISABLED
- ENABLED

H265Settings

slices

Number of slices per picture. Must be less than or equal to the number of macroblock rows for progressive pictures, and less than or equal to half the number of macroblock rows for interlaced pictures.

- Type: integer
- Required: False
- Minimum: 1
- Maximum: 32

minIInterval

Enforces separation between repeated (cadence) I-frames and I-frames inserted by Scene Change Detection. If a scene change I-frame is within I-interval frames of a cadence I-frame, the GOP is shrunk and/or stretched to the scene change I-frame. GOP stretch requires enabling lookahead as well as setting I-interval. The normal cadence resumes for the next GOP. This setting is only used when Scene Change Detect is enabled. Note: Maximum GOP stretch = GOP size + Min-I-interval - 1

- Type: integer
- Required: False
- Minimum: 0
- Maximum: 30

parNumerator

Pixel Aspect Ratio numerator.

- Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

flickerAdaptiveQuantization

Type: H265FlickerAdaptiveQuantization (p. 763)
Required: False

gopSizeUnits

Type: H265GopSizeUnits (p. 764)
Required: False

hrdBufferSize

Size of buffer (HRD buffer model). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

qualityTuningLevel

Type: H265QualityTuningLevel (p. 764)
Required: False

maxBitrate

Maximum bitrate in bits/second (for VBR mode only). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

bitrate

Average bitrate in bits/second. Required for VBR, CBR, and ABR. Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.

Type: integer
Required: False
Minimum: 1000
Maximum: 2147483647

spatialAdaptiveQuantization

Type: H265SpatialAdaptiveQuantization (p. 770)
Required: False

sampleAdaptiveOffsetFilterMode

Type: H265SampleAdaptiveOffsetFilterMode (p. 765)
Required: False

temporalIds

Type: H265TemporalIds (p. 770)
Required: False

slowPal

Type: H265SlowPal (p. 770)
Required: False

tiles

Type: H265Tiles (p. 771)
Required: False

codecProfile

Type: H265CodecProfile (p. 763)
Required: False

alternateTransferFunctionSei

Type: H265AlternateTransferFunctionSei (p. 762)
Required: False

unregisteredSeiTimecode

Type: H265UnregisteredSeiTimecode (p. 771)
Required: False

framerateControl

Type: H265FramerateControl (p. 763)
Required: False

telecine

Type: H265Telecine (p. 770)
Required: False

framerateConversionAlgorithm

Type: H265FramerateConversionAlgorithm (p. 763)
Required: False
codecLevel

Type: H265CodecLevel (p. 763)
Required: False

numberReferenceFrames

Number of reference frames to use. The encoder may use more than requested if using B-frames and/or interlaced encoding.

Type: integer
Required: False
Minimum: 1
Maximum: 6

temporalAdaptiveQuantization

Type: H265TemporalAdaptiveQuantization (p. 770)
Required: False

hrdBufferInitialFillPercentage

Percentage of the buffer that should initially be filled (HRD buffer model).

Type: integer
Required: False
Minimum: 0
Maximum: 100

framerateNumerator

Framerate numerator - framerate is a fraction, e.g. 24000 / 1001 = 23.976 fps.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

numberBFramesBetweenReferenceFrames

Number of B-frames between reference frames.

Type: integer
Required: False
Minimum: 0
Maximum: 7

gopClosedCadence

Frequency of closed GOPs. In streaming applications, it is recommended that this be set to 1 so a decoder joining mid-stream will receive an IDR frame as quickly as possible. Setting this value to 0 will break output segmenting.
Properties

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

framerateDenominator

Framerate denominator.
Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

adaptiveQuantization

Type: H265AdaptiveQuantization (p. 762)
Required: False

interlaceMode

Type: H265InterlaceMode (p. 764)
Required: False

gopSize

GOP Length (keyframe interval) in frames or seconds. Must be greater than zero.
Type: number
Required: False
Format: float
Minimum: 0.0


gopBReference

Type: H265GopBReference (p. 764)
Required: False

sceneChangeDetect

Type: H265SceneChangeDetect (p. 765)
Required: False

parDenominator

Pixel Aspect Ratio denominator.
Type: integer
Required: False
Minimum: 1
Maximum: 2147483647
parControl

Type: H265ParControl (p. 764)
Required: False

rateControlMode

Type: H265RateControlMode (p. 765)
Required: False

H265SlowPal (Enum)

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

DISABLED
ENABLED

H265SpatialAdaptiveQuantization (Enum)

Adjust quantization within each frame based on spatial variation of content complexity.

DISABLED
ENABLED

H265Telecine (Enum)

This field applies only if the Streams > Advanced > Framerate (framerate) field is set to 29.970. This field works with the Streams > Advanced > Preprocessors > Deinterlacer field (deinterlace_mode) and the Streams > Advanced > Interlaced Mode field (interlace_mode) to identify the scan type for the output: Progressive, Interlaced, Hard Telecine or Soft Telecine. - Hard: produces 29.97i output from 23.976 input.
- Soft: produces 23.976; the player converts this output to 29.97i.

NONE
SOFT
HARD

H265TemporalAdaptiveQuantization (Enum)

Adjust quantization within each frame based on temporal variation of content complexity.

DISABLED
ENABLED

H265TemporalIds (Enum)

Enables temporal layer identifiers in the encoded bitstream. Up to 3 layers are supported depending on GOP structure: I- and P-frames form one layer, reference B-frames can form a second layer and non-reference b-frames can form a third layer. Decoders can optionally decode only the lower temporal layers to generate a lower frame rate output. For example, given a bitstream with temporal IDs and with b-frames = 1 (i.e. IbPbPb display order), a decoder could decode all the frames for full frame rate output or only the I and P frames (lowest temporal layer) for a half frame rate output.

DISABLED
ENABLED
**H265Tiles (Enum)**

Enable use of tiles, allowing horizontal as well as vertical subdivision of the encoded pictures.

- DISABLED
- ENABLED

**H265UnregisteredSeiTimecode (Enum)**

Inserts timecode for each frame as 4 bytes of an unregistered SEI message.

- DISABLED
- ENABLED

**Hdr10Metadata**

**redPrimaryY**

HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

- Type: integer
- Required: False
- Minimum: 0
- Maximum: 50000

**greenPrimaryY**

HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

- Type: integer
- Required: False
- Minimum: 0
- Maximum: 50000

**whitePointX**

HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

- Type: integer
- Required: False
- Minimum: 0
- Maximum: 50000

**maxLuminance**

Nominal maximum mastering display luminance in units of 0.0001 candelas per square meter.

- Type: integer
- Required: False
- Minimum: 0
- Maximum: 2147483647
greenPrimaryX

HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 50000

whitePointY

HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 50000

redPrimaryX

HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 50000

bluePrimaryX

HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 50000

maxFrameAverageLightLevel

Maximum average light level of any frame in the coded video sequence, in units of candelas per square meter.

- **Type**: integer
- **Required**: True
- **Minimum**: 0
- **Maximum**: 65535

bluePrimaryY

HDR Master Display Information comes from the color grader and the color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate.
**Type**
integer

**Required**
False

**Minimum**
0

**Maximum**
50000

---

**maxContentLightLevel**

Maximum light level among all samples in the coded video sequence, in units of candela per square meter.

**Type**
integer

**Required**
True

**Minimum**
0

**Maximum**
65535

---

**minLuminance**

Nominal minimum mastering display luminance in units of candela per square meter

**Type**
integer

**Required**
False

**Minimum**
0

**Maximum**
2147483647

---

**ImageInserter**

**insertableImages**

Image to insert. Must be 32 bit windows BMP, PNG, or TGA file. Must not be larger than the output frames.

**Type**
Array of type `InsertableImage` (p. 773)

**Required**
True

---

**InsertableImage**

**duration**

Use Duration (Duration) to set the time, in milliseconds, for the image to remain on the output video.

**Type**
integer

**Required**
False

**Minimum**
-2147483648

**Maximum**
2147483647

---

**fadeOut**

Use Fade out (FadeOut) to set the length, in milliseconds, of the inserted image fade out. If you don't specify a value for Fade out, the image will disappear abruptly at the end of the inserted image duration.

**Type**
integer

**Required**
False

**Minimum**
-2147483648

**Maximum**
2147483647
**imageY**

Use Top (ImageY) to set the distance, in pixels, between the inserted image and the top edge of the video frame. Required for BMP, PNG and TGA input.

- **Type**: integer
- **Required**: True
- **Minimum**: -2147483648
- **Maximum**: 2147483647

**fadeIn**

Use Fade in (FadeIn) to set the length, in milliseconds, of the inserted image fade in. If you don't specify a value for Fade in, the image will appear abruptly at the Start time.

- **Type**: integer
- **Required**: False
- **Minimum**: -2147483648
- **Maximum**: 2147483647

**imageX**

Use Left (ImageX) to set the distance, in pixels, between the inserted image and the left edge of the frame. Required for BMP, PNG and TGA input.

- **Type**: integer
- **Required**: True
- **Minimum**: -2147483648
- **Maximum**: 2147483647

**width**

Specify the Width (Width) of the inserted image. Use a value that is less than or equal to the video resolution width. Leave this setting blank to use the native width of the image.

- **Type**: integer
- **Required**: False
- **Minimum**: -2147483648
- **Maximum**: 2147483647

**startTime**

Use Start time (StartTime) to specify the video timecode when the image is inserted in the output. This must be in timecode format (HH:MM:SS:FF)

- **Type**: string
- **Required**: False

**opacity**

Use Opacity (Opacity) to specify how much of the underlying video shows through the inserted image. 0 is transparent and 100 is fully opaque. Default is 50.

- **Type**: integer
- **Required**: True
Properties

Minimum: 0
Maximum: 100

layer

Use Layer (Layer) to specify how overlapping inserted images appear. Images with higher values of layer appear on top of images with lower values of layer.

Type: integer
Required: True
Minimum: 0
Maximum: 100

height

Specify the Height (Height) of the inserted image. Use a value that is less than or equal to the video resolution height. Leave this setting blank to use the native height of the image.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

imageInserterInput

Use Image location (imageInserterInput) to specify the Amazon S3 location of the image to be inserted into the output. Use a 32 bit BMP, PNG, or TGA file that fits inside the video frame.

Type: string
Required: True
Pattern: ^(s3://)(.*)\.(bmp|BMP|png|PNG|tga|TGA)$

LanguageCode (Enum)

Indicates the language of the audio output track. The ISO 639 language specified in the 'Language Code' drop down will be used when 'Follow Input Language Code' is not selected or when 'Follow Input Language Code' is selected but there is no ISO 639 language code specified by the input.

ENG
SPA
FRA
DEU
GER
ZHO
ARA
HIN
JPN
RUS
POR
ITA
URD
VIE
KOR
PAN
ABK
AAR
AFR
AKA
SQI
AMH
ARG
HYE
ASM
AVA
AVE
AYM
AZE
BAM
BAK
EUS
BEL
BEN
BIH
BIS
BOS
BRE
BUL
MYA
CAT
KHM
CHA
CHE
NYA
CHU
CHV
COR
COS
CRE
HRV
CES
DAN
DIV
NLD
DZO
ENM
EPO
EST
EWE
FAO
FIJ
FIN
FRM
FUL
GLA
GLG
LUG
KAT
ELL
GRN
GUJ
HAT
HAU
HEB
HER
HMO
HUN
ISL
IDO
IBO
IND
INA
ILE
IKU
IPK
GLE
JAV
KAL
KAN
KAU
KAS
KAZ
KIK
KIN
KIR
KOM
KON
KUA
KUR
LAO
LAT
LAV
LIM
LIN
LIT
LUB
LTZ
MKD
MLG
MSA
MAL
MLT
GLV
MRI
MAR
MAH
MON
NAU
NAV
NDE
NBL
NDO
NEP
SME
NOR
NOB
NNO
OCI
OJI
ORI
M2tsAudioBufferModel (Enum)
Selects between the DVB and ATSC buffer models for Dolby Digital audio.

DVB
ATSC

M2tsBufferModel (Enum)
Controls what buffer model to use for accurate interleaving. If set to MULTIPLEX, use multiplex buffer model. If set to NONE, this can lead to lower latency, but low-memory devices may not be able to play back the stream without interruptions.

MULTIPLEX
NONE

M2tsEbpAudioInterval (Enum)
When set to VIDEO_AND_FIXED_INTERVALS, audio EBP markers will be added to partitions 3 and 4. The interval between these additional markers will be fixed, and will be slightly shorter than the video EBP marker interval. When set to VIDEO_INTERVAL, these additional markers will not be inserted. Only applicable when EBP segmentation markers are is selected (segmentationMarkers is EBP or EBP_LEGACY).

VIDEO_AND_FIXED_INTERVALS
VIDEO_INTERVAL

M2tsEbpPlacement (Enum)
Selects which PIDs to place EBP markers on. They can either be placed only on the video PID, or on both the video PID and all audio PIDs. Only applicable when EBP segmentation markers are is selected (segmentationMarkers is EBP or EBP_LEGACY).

VIDEO_AND_AUDIO_PIDS
VIDEO_PID

M2tsEsRateInPes (Enum)
Controls whether to include the ES Rate field in the PES header.

INCLUDE
EXCLUDE

M2tsPcrControl (Enum)
When set to PCR_EVERY_PES_PACKET, a Program Clock Reference value is inserted for every Packetized Elementary Stream (PES) header. This is effective only when the PCR PID is the same as the video or audio elementary stream.

PCR_EVERY_PES_PACKET
CONFIGURED_PCR_PERIOD
M2tsRateMode (Enum)

When set to CBR, inserts null packets into transport stream to fill specified bitrate. When set to VBR, the bitrate setting acts as the maximum bitrate, but the output will not be padded up to that bitrate.

- VBR
- CBR

M2tsScte35Source (Enum)

Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output. This is only available for certain containers.

- PASSTHROUGH
- NONE

M2tsSegmentationMarkers (Enum)

Inserts segmentation markers at each segmentation_time period. rai_segstart sets the Random Access Indicator bit in the adaptation field. rai_adapt sets the RAI bit and adds the current timecode in the private data bytes. psi_segstart inserts PAT and PMT tables at the start of segments. ebp adds Encoder Boundary Point information to the adaptation field as per OpenCable specification OC-SP-EBP-I01-130118. ebp_legacy adds Encoder Boundary Point information to the adaptation field using a legacy proprietary format.

- NONE
- RAI_SEGSTART
- RAI_ADAPT
- PSI_SEGSTART
- EBP
- EBP_LEGACY

M2tsSegmentationStyle (Enum)

The segmentation style parameter controls how segmentation markers are inserted into the transport stream. With avails, it is possible that segments may be truncated, which can influence where future segmentation markers are inserted. When a segmentation style of "reset_cadence" is selected and a segment is truncated due to an avail, we will reset the segmentation cadence. This means the subsequent segment will have a duration of $segmentation_time seconds. When a segmentation style of "maintain_cadence" is selected and a segment is truncated due to an avail, we will not reset the segmentation cadence. This means the subsequent segment will likely be truncated as well. However, all segments after that will have a duration of $segmentation_time seconds. Note that EBP lookahead is a slight exception to this rule.

- MAINTAIN_CADENCE
- RESET_CADENCE

M2tsSettings

dvbTeletextPid

Packet Identifier (PID) for input source DVB Teletext data to this output. Can be entered as a decimal or hexadecimal value.

Type: integer
Properties

required: False
minimum: 32
maximum: 8182

bitrate
The output bitrate of the transport stream in bits per second. Setting to 0 lets the muxer automatically determine the appropriate bitrate. Other common values are 3750000, 7500000, and 15000000.

type: integer
required: false
minimum: 0
maximum: 2147483647

segmentationTime
The length in seconds of each segment. Required unless markers is set to _none_.

type: number
required: false
format: float
minimum: 0.0

audioPids
Packet Identifier (PID) of the elementary audio stream(s) in the transport stream. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values.

type: Array of type integer
required: false

rateMode

type: M2tsRateMode (p. 780)
required: false

ebpAudioInterval

type: M2tsEbpAudioInterval (p. 779)
required: false

fragmentTime
The length in seconds of each fragment. Only used with EBP markers.

type: number
required: false
format: float
minimum: 0.0

audioFramesPerPes
The number of audio frames to insert for each PES packet.
maxPcrInterval

Maximum time in milliseconds between Program Clock References (PCRs) inserted into the transport stream.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

scte35Pid

Packet Identifier (PID) of the SCTE-35 stream in the transport stream. Can be entered as a decimal or hexadecimal value.

Type: integer
Required: False
Minimum: 32
Maximum: 8182

privateMetadataPid

Packet Identifier (PID) of the private metadata stream in the transport stream. Can be entered as a decimal or hexadecimal value.

Type: integer
Required: False
Minimum: 32
Maximum: 8182

pmtInterval

The number of milliseconds between instances of this table in the output transport stream.

Type: integer
Required: False
Minimum: 0
Maximum: 1000

segmentationStyle

Type: M2tsSegmentationStyle (p. 780)
Required: False

audioBufferModel

Type: M2tsAudioBufferModel (p. 779)
Required: False
programNumber

The value of the program number field in the Program Map Table.

Type: integer
Required: False
Minimum: 0
Maximum: 65535

dvbNitSettings

Type: DvbNitSettings (p. 742)
Required: False

scte35Source

Type: M2tsScte35Source (p. 780)
Required: False

pmtPid

Packet Identifier (PID) for the Program Map Table (PMT) in the transport stream. Can be entered as a decimal or hexadecimal value.

Type: integer
Required: False
Minimum: 32
Maximum: 8182

bufferModel

Type: M2tsBufferModel (p. 779)
Required: False

ebpPlacement

Type: M2tsEbpPlacement (p. 779)
Required: False

dvbSdtSettings

Type: DvbSdtSettings (p. 742)
Required: False

nullPacketBitrate

Value in bits per second of extra null packets to insert into the transport stream. This can be used if a downstream encryption system requires periodic null packets.

Type: number
Required: False
Format: float
Properties

Minimum: 0.0

**pcrPid**

Packet Identifier (PID) of the Program Clock Reference (PCR) in the transport stream. When no value is given, the encoder will assign the same value as the Video PID. Can be entered as a decimal or hexadecimal value.

- **Type:** integer
- **Required:** False
- **Minimum:** 32
- **Maximum:** 8182

**minEbpInterval**

When set, enforces that Encoder Boundary Points do not come within the specified time interval of each other by looking ahead at input video. If another EBP is going to come in within the specified time interval, the current EBP is not emitted, and the segment is "stretched" to the next marker. The lookahead value does not add latency to the system. The Live Event must be configured elsewhere to create sufficient latency to make the lookahead accurate.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 10000

**transportStreamId**

The value of the transport stream ID field in the Program Map Table.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 65535

**pcrControl**

- **Type:** M2tsPcrControl (p. 779)
- **Required:** False

**videoPid**

Packet Identifier (PID) of the elementary video stream in the transport stream. Can be entered as a decimal or hexadecimal value.

- **Type:** integer
- **Required:** False
- **Minimum:** 32
- **Maximum:** 8182

**esRateInPes**

- **Type:** M2tsEsRateInPes (p. 779)
- **Required:** False
segmentationMarkers

Type: M2tsSegmentationMarkers (p. 780)
Required: False

dvbTdtSettings

Type: DvbTdtSettings (p. 747)
Required: False

patInterval

The number of milliseconds between instances of this table in the output transport stream.

Type: integer
Required: False
Minimum: 0
Maximum: 1000

dvbSubPids

Packet Identifier (PID) for input source DVB Subtitle data to this output. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values.

Type: Array of type integer
Required: False

M3u8PcrControl (Enum)

When set to PCR_EVERY_PES_PACKET a Program Clock Reference value is inserted for every Packetized Elementary Stream (PES) header. This parameter is effective only when the PCR PID is the same as the video or audio elementary stream.

PCR_EVERY_PES_PACKET
CONFIGURED_PCR_PERIOD

M3u8Scte35Source (Enum)

Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output. This is only available for certain containers.

PASSTHROUGH
NONE

M3u8Settings

pmtPid

Packet Identifier (PID) for the Program Map Table (PMT) in the transport stream. Can be entered as a decimal or hexadecimal value.

Type: integer
Required: False
**Properties**

**Minimum**: 32  
**Maximum**: 8182

**pcrPid**

Packet Identifier (PID) of the Program Clock Reference (PCR) in the transport stream. When no value is given, the encoder will assign the same value as the Video PID. Can be entered as a decimal or hexadecimal value.

**Type**: integer  
**Required**: False  
**Minimum**: 32  
**Maximum**: 8182

**audioPids**

Packet Identifier (PID) of the elementary audio stream(s) in the transport stream. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values.

**Type**: Array of type integer  
**Required**: False

**audioFramesPerPes**

The number of audio frames to insert for each PES packet.

**Type**: integer  
**Required**: False  
**Minimum**: 0  
**Maximum**: 2147483647

**scte35Pid**

Packet Identifier (PID) of the SCTE-35 stream in the transport stream. Can be entered as a decimal or hexadecimal value.

**Type**: integer  
**Required**: False  
**Minimum**: 32  
**Maximum**: 8182

**transportStreamId**

The value of the transport stream ID field in the Program Map Table.

**Type**: integer  
**Required**: False  
**Minimum**: 0  
**Maximum**: 65535

**videoPid**

Packet Identifier (PID) of the elementary video stream in the transport stream. Can be entered as a decimal or hexadecimal value.
Properties

**Type**: integer
**Required**: False
**Minimum**: 32
**Maximum**: 8182

**pcrControl**

**Type**: M3u8PcrControl (p. 785)
**Required**: False

**privateMetadataPid**

Packet Identifier (PID) of the private metadata stream in the transport stream. Can be entered as a decimal or hexadecimal value.

**Type**: integer
**Required**: False
**Minimum**: 32
**Maximum**: 8182

**pmtInterval**

The number of milliseconds between instances of this table in the output transport stream.

**Type**: integer
**Required**: False
**Minimum**: 0
**Maximum**: 1000

**patInterval**

The number of milliseconds between instances of this table in the output transport stream.

**Type**: integer
**Required**: False
**Minimum**: 0
**Maximum**: 1000

**programNumber**

The value of the program number field in the Program Map Table.

**Type**: integer
**Required**: False
**Minimum**: 0
**Maximum**: 65535

**timedMetadataPid**

Packet Identifier (PID) of the timed metadata stream in the transport stream. Can be entered as a decimal or hexadecimal value.

**Type**: integer
Properties

**Required**: False  
**Minimum**: 32  
**Maximum**: 8182

**timedMetadata**
- **Type**: TimedMetadata (p. 807)  
- **Required**: False

**scte35Source**
- **Type**: M3u8Scte35Source (p. 785)  
- **Required**: False

**MovClapAtom (Enum)**
When enabled, include 'clap' atom if appropriate for the video output settings.

- INCLUDE  
- EXCLUDE

**MovCslgAtom (Enum)**
When enabled, file composition times will start at zero, composition times in the 'ctts' (composition time to sample) box for B-frames will be negative, and a 'cslg' (composition shift least greatest) box will be included per 14496-1 amendment 1. This improves compatibility with Apple players and tools.

- INCLUDE  
- EXCLUDE

**MovMpeg2FourCCControl (Enum)**
When set to XDCAM, writes MPEG2 video streams into the QuickTime file using XDCAM fourcc codes. This increases compatibility with Apple editors and players, but may decrease compatibility with other players. Only applicable when the video codec is MPEG2.

- XDCAM  
- MPEG

**MovPaddingControl (Enum)**
If set to OMNEON, inserts Omneon-compatible padding

- OMNEON  
- NONE

**MovReference (Enum)**
A value of 'external' creates separate media files and the wrapper file (.mov) contains references to these media files. A value of 'self_contained' creates only a wrapper (.mov) file and this file contains all of the media.

- SELF_CONTAINED  
- EXTERNAL
MovSettings

reference

Type: MovReference (p. 788)
Required: False

paddingControl

Type: MovPaddingControl (p. 788)
Required: False

mpeg2FourCCControl

Type: MovMpeg2FourCCControl (p. 788)
Required: False

cslgAtom

Type: MovCslgAtom (p. 788)
Required: False

clapAtom

Type: MovClapAtom (p. 788)
Required: False

Mp2Settings

channels

Set Channels to specify the number of channels in this output audio track. Choosing Mono in the console will give you 1 output channel; choosing Stereo will give you 2. In the API, valid values are 1 and 2.

Type: integer
Required: False
Minimum: 1
Maximum: 2

bitrate

Average bitrate in bits/second.

Type: integer
Required: False
Minimum: 32000
Maximum: 384000

sampleRate

Sample rate in hz.
Properties

Type: integer
Required: False
Minimum: 32000
Maximum: 48000

Mp4CslgAtom (Enum)
When enabled, file composition times will start at zero, composition times in the 'ctts' (composition time to sample) box for B-frames will be negative, and a 'cslg' (composition shift least greatest) box will be included per 14496-1 amendment 1. This improves compatibility with Apple players and tools.

INCLUDE
EXCLUDE

Mp4FreeSpaceBox (Enum)
Inserts a free-space box immediately after the moov box.

INCLUDE
EXCLUDE

Mp4MoovPlacement (Enum)
If set to PROGRESSIVE_DOWNLOAD, the MOOV atom is relocated to the beginning of the archive as required for progressive downloading. Otherwise it is placed normally at the end.

PROGRESSIVE_DOWNLOAD
NORMAL

Mp4Settings

mp4MajorBrand
Overrides the "Major Brand" field in the output file. Usually not necessary to specify.

Type: string
Required: False

type: Mp4MoovPlacement
Required: False

cslg Atom
Type: Mp4CslgAtom
Required: False

freeSpaceBox
Type: Mp4FreeSpaceBox
Required: False
**Mpeg2AdaptiveQuantization (Enum)**

Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

- OFF
- LOW
- MEDIUM
- HIGH

**Mpeg2CodecLevel (Enum)**

Use Level (Mpeg2CodecLevel) to set the MPEG-2 level for the video output.

- AUTO
- LOW
- MAIN
- HIGH1440
- HIGH

**Mpeg2CodecProfile (Enum)**

Use Profile (Mpeg2CodecProfile) to set the MPEG-2 profile for the video output.

- MAIN
- PROFILE_422

**Mpeg2FramerateControl (Enum)**

Using the API, set FramerateControl to INITIALIZE_FROM_SOURCE if you want the service to use the framerate from the input. Using the console, do this by choosing INITIALIZE_FROM_SOURCE for Framerate.

- INITIALIZE_FROM_SOURCE
- SPECIFIED

**Mpeg2FramerateConversionAlgorithm (Enum)**

When set to INTERPOLATE, produces smoother motion during framerate conversion.

- DUPLICATE_DROP
- INTERPOLATE

**Mpeg2GopSizeUnits (Enum)**

Indicates if the GOP Size in MPEG2 is specified in frames or seconds. If seconds the system will convert the GOP Size into a frame count at run time.

- FRAMES
- SECONDS

**Mpeg2InterlaceMode (Enum)**

Use Interlace mode (InterlaceMode) to choose the scan line type for the output.
* Top Field First (TOP_FIELD) and Bottom Field First (BOTTOM_FIELD) produce interlaced output with the entire output having the same field polarity (top or bottom first).

* Follow, Default Top (FOLLOW_TOP_FIELD) and Follow, Default Bottom (FOLLOW_BOTTOM_FIELD) use the same field polarity as the source. Therefore, behavior depends on the input scan type. If the source is interlaced, the output will be interlaced with the same polarity as the source (it will follow the source). The output could therefore be a mix of "top field first" and "bottom field first". If the source is progressive, the output will be interlaced with "top field first" or "bottom field first" polarity, depending on which of the Follow options you chose.

| PROGRESSIVE | TOP_FIELD | BOTTOM_FIELD | FOLLOW_TOP_FIELD | FOLLOW_BOTTOM_FIELD |

Mpeg2IntraDcPrecision (Enum)

Use Intra DC precision (Mpeg2IntraDcPrecision) to set quantization precision for intra-block DC coefficients. If you choose the value auto, the service will automatically select the precision based on the per-frame compression ratio.

AUTO
INTRA_DC_PRECISION_8
INTRA_DC_PRECISION_9
INTRA_DC_PRECISION_10
INTRA_DC_PRECISION_11

Mpeg2ParControl (Enum)

Using the API, enable ParFollowSource if you want the service to use the pixel aspect ratio from the input. Using the console, do this by choosing Follow source for Pixel aspect ratio.

INITIALIZE_FROM_SOURCE
SPECIFIED

Mpeg2QualityTuningLevel (Enum)

Use Quality tuning level (Mpeg2QualityTuningLevel) to specify whether to use single-pass or multipass video encoding.

SINGLE_PASS
MULTI_PASS

Mpeg2RateControlMode (Enum)

Use Rate control mode (Mpeg2RateControlMode) to specify whether the bitrate is variable (vbr) or constant (cbr).

VBR
CBR

Mpeg2SceneChangeDetect (Enum)

Scene change detection (inserts I-frames on scene changes).
DISABLED
ENABLED

**Mpeg2Settings**

**minIInterval**

Enforces separation between repeated (cadence) I-frames and I-frames inserted by Scene Change Detection. If a scene change I-frame is within I-interval frames of a cadence I-frame, the GOP is shrunk and/or stretched to the scene change I-frame. GOP stretch requires enabling lookahead as well as setting I-interval. The normal cadence resumes for the next GOP. This setting is only used when Scene Change Detect is enabled. Note: Maximum GOP stretch = GOP size + Min-I-interval - 1

  - **Type**: integer
  - **Required**: False
  - **Minimum**: 0
  - **Maximum**: 30

**parNumerator**

Pixel Aspect Ratio numerator.

  - **Type**: integer
  - **Required**: False
  - **Minimum**: 1
  - **Maximum**: 2147483647

**gopSizeUnits**

  - **Type**: Mpeg2GopSizeUnits (p. 791)
  - **Required**: False

**hrdBufferSize**

Size of buffer (HRD buffer model). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.

  - **Type**: integer
  - **Required**: False
  - **Minimum**: -2147483648
  - **Maximum**: 2147483647

**qualityTuningLevel**

  - **Type**: Mpeg2QualityTuningLevel (p. 792)
  - **Required**: False

**maxBitrate**

Maximum bitrate in bits/second (for VBR mode only). Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m.

  - **Type**: integer
### bitrate

Average bitrate in bits/second. Required for VBR, CBR, and ABR. Five megabits can be entered as 5000000 or 5m. Five hundred kilobits can be entered as 500000 or 0.5m. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.

- **Type**: integer
- **Required**: False
- **Minimum**: 1000
- **Maximum**: 2147483647

### spatialAdaptiveQuantization

- **Type**: Mpeg2SpatialAdaptiveQuantization (p. 797)
- **Required**: False

### slowPal

- **Type**: Mpeg2SlowPal (p. 797)
- **Required**: False

### codecProfile

- **Type**: Mpeg2CodecProfile (p. 791)
- **Required**: False

### intraDcPrecision

- **Type**: Mpeg2IntraDcPrecision (p. 792)
- **Required**: False

### softness

Softness. Selects quantizer matrix, larger values reduce high-frequency content in the encoded image.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 128

### framerateControl

- **Type**: Mpeg2FramerateControl (p. 791)
- **Required**: False

### telecine

- **Type**: Mpeg2Telecine (p. 797)
Required: False

framerateConversionAlgorithm

Type: Mpeg2FramerateConversionAlgorithm (p. 791)
Required: False

codecLevel

Type: Mpeg2CodecLevel (p. 791)
Required: False

temporalAdaptiveQuantization

Type: Mpeg2TemporalAdaptiveQuantization (p. 797)
Required: False

hrdBufferInitialFillPercentage

Percentage of the buffer that should initially be filled (HRD buffer model).

Type: integer
Required: False
Minimum: 0
Maximum: 100

framerateNumerator

Framerate numerator - framerate is a fraction, e.g. 24000 / 1001 = 23.976 fps.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

numberBFramesBetweenReferenceFrames

Number of B-frames between reference frames.

Type: integer
Required: False
Minimum: 0
Maximum: 7

gopClosedCadence

Frequency of closed GOPs. In streaming applications, it is recommended that this be set to 1 so a decoder joining mid-stream will receive an IDR frame as quickly as possible. Setting this value to 0 will break output segmenting.

Type: integer
Required: False
Minimum: 0
Properties

Maximum: 2147483647

framerateDenominator

Framerate denominator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

adaptiveQuantization

Type: Mpeg2AdaptiveQuantization (p. 791)
Required: False

interlaceMode

Type: Mpeg2InterlaceMode (p. 791)
Required: False

gopSize

GOP Length (keyframe interval) in frames or seconds. Must be greater than zero.

Type: number
Required: False
Format: float
Minimum: 0.0

sceneChangeDetect

Type: Mpeg2SceneChangeDetect (p. 792)
Required: False

parDenominator

Pixel Aspect Ratio denominator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

parControl

Type: Mpeg2ParControl (p. 792)
Required: False

syntax

Type: Mpeg2Syntax (p. 797)
Required: False
rateControlMode

Type: Mpeg2RateControlMode (p. 792)
Required: False

Mpeg2SlowPal (Enum)

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

- DISABLED
- ENABLED

Mpeg2SpatialAdaptiveQuantization (Enum)

Adjust quantization within each frame based on spatial variation of content complexity.

- DISABLED
- ENABLED

Mpeg2Syntax (Enum)

Produces a Type D-10 compatible bitstream (SMPTE 356M-2001).

- DEFAULT
- D_10

Mpeg2Telecine (Enum)

Only use Telecine (Mpeg2Telecine) when you set Framerate (Framerate) to 29.970. Set Telecine (Mpeg2Telecine) to Hard (hard) to produce a 29.97i output from a 23.976 input. Set it to Soft (soft) to produce 23.976 output and leave conversion to the player.

- NONE
- SOFT
- HARD

Mpeg2TemporalAdaptiveQuantization (Enum)

Adjust quantization within each frame based on temporal variation of content complexity.

- DISABLED
- ENABLED

NoiseReducer

filter

Type: NoiseReducerFilter (p. 798)
Required: True

filterSettings

Type: NoiseReducerFilterSettings (p. 798)
**spatialFilterSettings**

* Required: False

- **Type**: NoiseReducerSpatialFilterSettings (p. 798)
- * Required: False

**NoiseReducerFilter (Enum)**

Use Noise reducer filter (NoiseReducerFilter) to select one of the following spatial image filtering functions. To use this setting, you must also enable Noise reducer (NoiseReducer).

* Bilateral is an edge preserving noise reduction filter
* Mean (softest), Gaussian, Lanczos, and Sharpen (sharpest) are convolution filters
* Conserve is a min/max noise reduction filter
* Spatial is frequency-domain filter based on JND principles.

`BILATERAL`
`MEAN`
`GAUSSIAN`
`LANCZOS`
`SHARPEN`
`CONSERVE`
`SPATIAL`

**NoiseReducerFilterSettings**

- **strength**
  - Relative strength of noise reducing filter. Higher values produce stronger filtering.
  - * Type: integer
  - * Required: False
  - * Minimum: 0
  - * Maximum: 3

**NoiseReducerSpatialFilterSettings**

- **strength**
  - Relative strength of noise reducing filter. Higher values produce stronger filtering.
  - * Type: integer
  - * Required: False
  - * Minimum: 0
  - * Maximum: 16

**postFilterSharpenStrength**

Specify strength of post noise reduction sharpening filter, with 0 disabling the filter and 3 enabling it at maximum strength.
**Type: integer**
**Required: False**
**Minimum: 0**
**Maximum: 3**

**speed**

The speed of the filter, from -2 (lower speed) to 3 (higher speed), with 0 being the nominal value.

**Type: integer**
**Required: False**
**Minimum: -2**
**Maximum: 3**

**OutputChannelMapping**

**inputChannels**

**Type: Array of type integer**
**Required: True**

**OutputSdt (Enum)**

Selects method of inserting SDT information into output stream. "Follow input SDT" copies SDT information from input stream to output stream. "Follow input SDT if present" copies SDT information from input stream to output stream if SDT information is present in the input, otherwise it will fall back on the user-defined values. Enter "SDT Manually" means user will enter the SDT information. "No SDT" means output stream will not contain SDT information.

- SDT_FOLLOW
- SDT_FOLLOW_IF_PRESENT
- SDT_MANUAL
- SDT_NONE

**Preset**

**settings**

**Type: PresetSettings (p. 800)**
**Required: True**

**lastUpdated**

The timestamp in epoch seconds when the preset was last updated.

**Type: string**
**Required: False**
**Format: date-time**

**createdAt**

The timestamp in epoch seconds for preset creation.
Type: string  
Required: False  
Format: date-time

name
A name you create for each preset. Each name must be unique within your account.

Type: string  
Required: True

description
An optional description you create for each preset.

Type: string  
Required: False

category
An optional category you create to organize your presets.

Type: string  
Required: False

type

Type: Type (p. 807)  
Required: False

arn
An identifier for this resource that is unique within all of AWS.

Type: string  
Required: False

PresetSettings

videoDescription
(VideoDescription) contains a group of video encoding settings. The specific video settings depend on the video codec you choose when you specify a value for Video codec (codec). Include one instance of (VideoDescription) per output.

Type: VideoDescription (p. 809)  
Required: False

audioDescriptions
(AudioDescriptions) contains groups of audio encoding settings organized by audio codec. Include one instance of (AudioDescriptions) per output. (AudioDescriptions) can contain multiple groups of encoding settings.
Properties

Type: Array of type AudioDescription (p. 730)
Required: False

captionDescriptions

Caption settings for this preset. There can be multiple caption settings in a single output.

Type: Array of type CaptionDescriptionPreset (p. 737)
Required: False

ProresCodecProfile (Enum)

Use Profile (ProResCodecProfile) to specify the type of Apple ProRes codec to use for this output.

APPLE_PRORES_422
APPLE_PRORES_422_HQ
APPLE_PRORES_422_LT
APPLE_PRORES_422_PROXY

ProresFramerateControl (Enum)

Using the API, set FramerateControl to INITIALIZE_FROM_SOURCE if you want the service to use the framerate from the input. Using the console, do this by choosing INITIALIZE_FROM_SOURCE for Framerate.

INITIALIZE_FROM_SOURCE
SPECIFIED

ProresFramerateConversionAlgorithm (Enum)

When set to INTERPOLATE, produces smoother motion during framerate conversion.

DUPLICATE_DROP
INTERPOLATE

ProresInterlaceMode (Enum)

Use Interlace mode (InterlaceMode) to choose the scan line type for the output.

* Top Field First (TOP_FIELD) and Bottom Field First (BOTTOM_FIELD) produce interlaced output with the entire output having the same field polarity (top or bottom first).

* Follow, Default Top (FOLLOW_TOP_FIELD) and Follow, Default Bottom (FOLLOW_BOTTOM_FIELD) use the same field polarity as the source. Therefore, behavior depends on the input scan type. If the source is interlaced, the output will be interlaced with the same polarity as the source (it will follow the source). The output could therefore be a mix of "top field first" and "bottom field first". If the source is progressive, the output will be interlaced with "top field first" or "bottom field first" polarity, depending on which of the Follow options you chose.
PROGRESSIVE
TOP_FIELD
BOTTOM_FIELD
FOLLOW_TOP_FIELD
FOLLOW_BOTTOM_FIELD

ProresParControl (Enum)

Use (ProresParControl) to specify how the service determines the pixel aspect ratio. Set to Follow source (INITIALIZE_FROM_SOURCE) to use the pixel aspect ratio from the input. To specify a different pixel aspect ratio: Using the console, choose it from the dropdown menu. Using the API, set ProresParControl to (SPECIFIED) and provide for (ParNumerator) and (ParDenominator).

INITIALIZE_FROM_SOURCE
SPECIFIED

ProresSettings

slowPal
Type: ProresSlowPal (p. 803)
Required: False

framerateControl
Type: ProresFramerateControl (p. 801)
Required: False

telecine
Type: ProresTelecine (p. 803)
Required: False

framerateDenominator
Framerate denominator.
Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

framerateConversionAlgorithm
Type: ProresFramerateConversionAlgorithm (p. 801)
Required: False

interlaceMode
Type: ProresInterlaceMode (p. 801)
Required: False
**codecProfile**

*Type*: ProresCodecProfile (p. 801)

*Required*: False

**parNumerator**

Pixel Aspect Ratio numerator.

*Type*: integer

*Required*: False

*Minimum*: 1

*Maximum*: 2147483647

**parControl**

*Type*: ProresParControl (p. 802)

*Required*: False

**parDenominator**

Pixel Aspect Ratio denominator.

*Type*: integer

*Required*: False

*Minimum*: 1

*Maximum*: 2147483647

**framerateNumerator**

When you use the API for transcode jobs that use framerate conversion, specify the framerate as a fraction. For example, 24000 / 1001 = 23.976 fps. Use FramerateNumerator to specify the numerator of this fraction. In this example, use 24000 for the value of FramerateNumerator.

*Type*: integer

*Required*: False

*Minimum*: 1

*Maximum*: 2147483647

**ProresSlowPal (Enum)**

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

- DISABLED
- ENABLED

**ProresTelecine (Enum)**

Only use Telecine (ProresTelecine) when you set Framerate (Framerate) to 29.970. Set Telecine (ProresTelecine) to Hard (hard) to produce a 29.97i output from a 23.976 input. Set it to Soft (soft) to produce 23.976 output and leave conversion to the player.

- NONE
- HARD
Rectangle

**width**

Width of rectangle in pixels.

- **Type:** integer
- **Required:** True
- **Minimum:** -2147483648
- **Maximum:** 2147483647

**x**

The distance, in pixels, between the rectangle and the left edge of the video frame.

- **Type:** integer
- **Required:** True
- **Minimum:** -2147483648
- **Maximum:** 2147483647

**y**

The distance, in pixels, between the rectangle and the top edge of the video frame.

- **Type:** integer
- **Required:** True
- **Minimum:** -2147483648
- **Maximum:** 2147483647

**height**

Height of rectangle in pixels.

- **Type:** integer
- **Required:** True
- **Minimum:** -2147483648
- **Maximum:** 2147483647

RemixSettings

**channelsOut**

Specify the number of channels in this output after remixing. Valid values: 1, 2, 4, 6, 8

- **Type:** integer
- **Required:** True
- **Minimum:** 1
- **Maximum:** 8

**channelMapping**

- **Type:** ChannelMapping (p. 738)
- **Required:** True
channelsIn

Specify the number of audio channels from your input that you want to use in your output. With remixing, you might combine or split the data in these channels, so the number of channels in your final output might be different.

- **Type**: integer
- **Required**: True
- **Minimum**: 1
- **Maximum**: 16

**RespondToAfd (Enum)**

Use Respond to AFD (RespondToAfd) to specify how the service changes the video itself in response to AFD values in the input.

- Choose **Respond** to clip the input video frame according to the AFD value, input display aspect ratio, and output display aspect ratio.
- Choose **Passthrough** to include the input AFD values. Do not choose this when AfdSignaling is set to (NONE). A preferred implementation of this workflow is to set RespondToAfd to (NONE) and set AfdSignaling to (AUTO).
- Choose **None** to remove all input AFD values from this output.

  - NONE
  - RESPOND
  - PASSTHROUGH

**ScalingBehavior (Enum)**

Applies only if your input aspect ratio is different from your output aspect ratio. Enable **Stretch** to output (StretchToOutput) to have the service stretch your video image to fit. Leave this setting disabled to allow the service to letterbox your video instead. This setting overrides any positioning value you specify elsewhere in the job.

  - DEFAULT
  - STRETCH_TO_OUTPUT

**SccDestinationFramerate (Enum)**

Set Framerate (SccDestinationFramerate) to make sure that the captions and the video are synchronized in the output. Specify a framerate that matches the framerate of the associated video. If the video framerate is 29.97, choose 29.97 dropframe (FRAMERATE_29_97_DROPFRAME) only if the video has video_insertion=true and drop_frame_timecode=true; otherwise, choose 29.97 non-dropframe (FRAMERATE_29_97_NON_DROPFRAME).

  - FRAMERATE_23_97
  - FRAMERATE_24
  - FRAMERATE_29_97_DROPFRAME
  - FRAMERATE_29_97_NON_DROPFRAME

**SccDestinationSettings**

**framerate**

- **Type**: SccDestinationFramerate (p. 805)
Required: False

**TeletextDestinationSettings**

**pageNumber**

Set pageNumber to the Teletext page number for the destination captions for this output. This value must be a three-digit hexadecimal string; strings ending in -FF are invalid. If you are passing through the entire set of Teletext data, do not use this field.

Type: string
Required: False
Pattern: ^[1-8][0-9a-fA-F][0-9a-eA-E]$

**TimecodeBurnin**

**prefix**

Use Prefix (Prefix) to place ASCII characters before any burned-in timecode. For example, a prefix of "EZ-" will result in the timecode "EZ-00:00:00:00". Provide either the characters themselves or the ASCII code equivalents. The supported range of characters is 0x20 through 0x7e. This includes letters, numbers, and all special characters represented on a standard English keyboard.

Type: string
Required: False
Pattern: ^[ -~]+$

**fontSize**

Use Font Size (FontSize) to set the font size of any burned-in timecode. Valid values are 10, 16, 32, 48.

Type: integer
Required: False
Minimum: 10
Maximum: 48

**position**

Type: TimecodeBurninPosition (p. 806)
Required: False

**TimecodeBurninPosition (Enum)**

Use Position (Position) under Timecode burn-in (TimecodeBurnIn) to specify the location the burned-in timecode on output video.

- TOP_CENTER
- TOP_LEFT
- TOP_RIGHT
- MIDDLE_LEFT
- MIDDLE_CENTER
- MIDDLE_RIGHT
- BOTTOM_LEFT
- BOTTOM_CENTER
- BOTTOM_RIGHT
TimedMetadata (Enum)

If PASSTHROUGH, inserts ID3 timed metadata from the timed_metadata REST command into this output. Only available for certain containers.

   PASSTHROUGH
   NONE

TtmlDestinationSettings

stylePassthrough

   Type: TtmlStylePassthrough (p. 807)
   Required: False

TtmlStylePassthrough (Enum)

Pass through style and position information from a TTML-like input source (TTML, SMPTE-TT, CFF-TT) to the CFF-TT output or TTML output.

   ENABLED
   DISABLED

Type (Enum)

   SYSTEM
   CUSTOM

UpdatePresetRequest

settings

   Type: PresetSettings (p. 800)
   Required: False

name

The name of the preset you are modifying.

   Type: string
   Required: True

description

The new description for the preset, if you are changing it.

   Type: string
   Required: False

category

The new category for the preset, if you are changing it.
Type: string
Required: False

UpdatePresetResponse

preset

Type: Preset (p. 799)
Required: False

VideoCodec (Enum)

Type of video codec

- FRAME_CAPTURE
- H_264
- H_265
- MPEG2
- PRORES

VideoCodecSettings

h265Settings

Type: H265Settings (p. 765)
Required: False

codec

Type: VideoCodec (p. 808)
Required: True

proresSettings

Type: ProresSettings (p. 802)
Required: False

mpeg2Settings

Type: Mpeg2Settings (p. 793)
Required: False

h264Settings

Type: H264Settings (p. 756)
Required: False

frameCaptureSettings

Type: FrameCaptureSettings (p. 752)
Properties

Required: False

**VideoDescription**

**fixedAfd**

Applies only if you set AFD Signaling(AfdSignaling) to Fixed (FIXED). Use Fixed (FixedAfd) to specify a four-bit AFD value which the service will write on all frames of this video output.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 15

**scalingBehavior**

- **Type:** ScalingBehavior (p. 805)
- **Required:** False

**respondToAfd**

- **Type:** RespondToAfd (p. 805)
- **Required:** False

**codecSettings**

- **Type:** VideoCodecSettings (p. 808)
- **Required:** True

**afdSignaling**

- **Type:** AfdSignaling (p. 728)
- **Required:** False

**colorMetadata**

- **Type:** ColorMetadata (p. 739)
- **Required:** False

**timecodeInsertion**

- **Type:** VideoTimecodeInsertion (p. 811)
- **Required:** False

**width**

Use Width (Width) to define the video resolution width, in pixels, for this output. If you don’t provide a value here, the service will use the input width.

- **Type:** integer
- **Required:** False
- **Minimum:** 32
Maximum: 4096

**sharpness**

Use Sharpness (Sharpness) setting to specify the strength of anti-aliasing. This setting changes the width of the anti-alias filter kernel used for scaling. Sharpness only applies if your output resolution is different from your input resolution, and if you set Anti-alias (AntiAlias) to ENABLED. 0 is the softest setting, 100 the sharpest, and 50 recommended for most content.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 100

**antiAlias**

- **Type:** AntiAlias (p. 729)
- **Required:** False

**videoPreprocessors**

Find additional transcoding features under Preprocessors (VideoPreprocessors). Enable the features at each output individually. These features are disabled by default.

- **Type:** VideoPreprocessor (p. 811)
- **Required:** False

**position**

Use Position (Position) to point to a rectangle object to define your position. This setting overrides any other aspect ratio.

- **Type:** Rectangle (p. 804)
- **Required:** False

**dropFrameTimecode**

- **Type:** DropFrameTimecode (p. 742)
- **Required:** False

**crop**

Applies only if your input aspect ratio is different from your output aspect ratio. Use Input cropping rectangle (Crop) to specify the video area the service will include in the output. This will crop the input source, causing video pixels to be removed on encode. Do not use this setting if you have enabled Stretch to output (stretchToOutput) in your output settings.

- **Type:** Rectangle (p. 804)
- **Required:** False

**height**

Use the Height (Height) setting to define the video resolution height for this output. Specify in pixels. If you don’t provide a value here, the service will use the input height.
**Type**: integer  
**Required**: False  
**Minimum**: 32  
**Maximum**: 2160

### VideoPreprocessor

#### timecodeBurnin

Timecode burn-in (TimecodeBurnIn)--Burns the output timecode and specified prefix into the output.

- **Type**: TimecodeBurnin (p. 806)  
- **Required**: False

#### noiseReducer

Enable the Noise reducer (NoiseReducer) feature to remove noise from your video output if necessary. Enable or disable this feature for each output individually. This setting is disabled by default.

- **Type**: NoiseReducer (p. 797)  
- **Required**: False

#### colorCorrector

Enable the Color corrector (ColorCorrector) feature if necessary. Enable or disable this feature for each output individually. This setting is disabled by default.

- **Type**: ColorCorrector (p. 738)  
- **Required**: False

#### imageInserter

Enable the Image inserter (ImageInserter) feature to include a graphic overlay on your video. Enable or disable this feature for each output individually. This setting is disabled by default.

- **Type**: ImageInserter (p. 773)  
- **Required**: False

#### deinterlacer

Use Deinterlacer (Deinterlacer) to produce smoother motion and a clearer picture.

- **Type**: Deinterlacer (p. 741)  
- **Required**: False

### VideoTimecodeInsertion (Enum)

Enable Timecode insertion to include timecode information in this output. Do this in the API by setting (VideoTimecodeInsertion) to (PIC_TIMING_SEI). To get timecodes to appear correctly in your output, also set up the timecode configuration for your job in the input settings. Only enable Timecode insertion when the input framerate is identical to output framerate. Disable this setting to remove the timecode from the output. Default is disabled.

- **DISABLED**
- **PIC_TIMING_SEI**
WavSettings

channels

Set Channels to specify the number of channels in this output audio track. With WAV, valid values 1, 2, 4, and 8. In the console, these values are Mono, Stereo, 4-Channel, and 8-Channel, respectively.

Type: integer  
Required: False 
Minimum: 1  
Maximum: 8

bitDepth

Specify Bit depth (BitDepth), in bits per sample, to choose the encoding quality for this audio track.

Type: integer  
Required: False 
Minimum: 16  
Maximum: 24

sampleRate

Sample rate in Hz.

Type: integer  
Required: False 
Minimum: 8000  
Maximum: 192000

Queues

URI

/queues

HTTP Methods

GET

Operation ID: ListQueues

Retrieve a JSON array of up to twenty of your queues. This will return the queues themselves, not just a list of them. To retrieve the next twenty queues, use the nextToken string returned with the array.

Query Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>listBy</td>
<td>String</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>nextToken</td>
<td>String</td>
<td>False</td>
<td></td>
</tr>
<tr>
<td>maxResults</td>
<td>String</td>
<td>False</td>
<td></td>
</tr>
</tbody>
</table>
## AWS Elemental MediaConvert API Reference
### HTTP Methods

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>order</td>
<td>String</td>
<td>False</td>
<td></td>
</tr>
</tbody>
</table>

### Responses

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>ListQueuesResponse (p. 815)</td>
<td>200: OkRresponse</td>
</tr>
<tr>
<td>400</td>
<td>ExceptionBody (p. 816)</td>
<td>400: BadRequestException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The conditional request failed. The service can't process your request because of a problem in the request. Please check your request form and syntax.</td>
</tr>
<tr>
<td>500</td>
<td>ExceptionBody (p. 816)</td>
<td>500: InternalServiceException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The service encountered an unexpected condition and cannot fulfill your request.</td>
</tr>
<tr>
<td>403</td>
<td>ExceptionBody (p. 816)</td>
<td>403: AccessDeniedException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>You don't have permissions for this action with the credentials you sent. Please check your authorization credentials. You should be sending credentials using the AWS Signature Version 4 signing process.</td>
</tr>
<tr>
<td>404</td>
<td>ExceptionBody (p. 816)</td>
<td>404: ResourceNotFoundException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The resource you requested does not exist.</td>
</tr>
<tr>
<td>429</td>
<td>ExceptionBody (p. 816)</td>
<td>429: LimitExceededException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Too many requests have been sent in too short of a time. The service limits the rate at which it will accept requests. For example, you may be hitting your account limits for preset creation or job submission.</td>
</tr>
<tr>
<td>409</td>
<td>ExceptionBody (p. 816)</td>
<td>409: ResourceInUseException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The service could not complete your request because there is a conflict with the current state of the resource. For example, you</td>
</tr>
</tbody>
</table>
**POST**

Operation ID: CreateQueue

Create a new transcoding queue. For information about job templates see the User Guide at http://docs.aws.amazon.com/mediaconvert/latest/ug/what-is.html

**Responses**

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>201</td>
<td>CreateQueueResponse (p. 816)</td>
<td>Your resource has been successfully created.</td>
</tr>
</tbody>
</table>
| 400         | ExceptionBody (p. 816) | 400: BadRequestException
The conditional request failed. The service can't process your request because of a problem in the request. Please check your request form and syntax. |
| 500         | ExceptionBody (p. 816) | 500: InternalServiceException
The service encountered an unexpected condition and cannot fulfill your request. |
| 403         | ExceptionBody (p. 816) | 403: AccessDeniedException
You don't have permissions for this action with the credentials you sent. Please check your authorization credentials. You should be sending credentials using the AWS Signature Version 4 signing process. |
| 404         | ExceptionBody (p. 816) | 404: ResourceNotFoundException
The resource you requested does not exist. |
| 429         | ExceptionBody (p. 816) | 429: LimitExceededException
Too many requests have been sent in too short of a time. The service limits the rate at which it will accept requests. |
Schemas

Request Bodies

Example GET

```json
{
    "nextToken (p. 817)": "string",
    "maxResults (p. 817)": integer,
    "listBy (p. 817)": enum,
    "order (p. 817)": enum
}
```

Example POST

```json
{
    "name (p. 816)": "string",
    "description (p. 816)": "string"
}
```

Response Bodies

Example ListQueuesResponse

```json
{
    "queues (p. 817)": [
        {
            "lastUpdated (p. 818)": "string",
            "createdAt (p. 818)": "string",
            "name (p. 818)": "string",
            "description (p. 818)": "string",
            "arn (p. 818)": "string",
            "type (p. 818)": enum,
            "status (p. 818)": enum
        }
    ],
    "nextToken (p. 817)": "string"
}
```
Example CreateQueueResponse

```json
{
    "queue (p. 816)": {
        "lastUpdated (p. 818)": "string",
        "createdAt (p. 818)": "string",
        "name (p. 818)": "string",
        "description (p. 818)": "string",
        "arn (p. 818)": "string",
        "type (p. 818)": enum,
        "status (p. 818)": enum
    }
}
```

Example ExceptionBody

```json
{
    "message (p. 816)": "string"
}
```

Properties

CreateQueueRequest

name

The name of the queue you are creating.

- **Type:** string
- **Required:** True

description

Optional. A description of the queue you are creating.

- **Type:** string
- **Required:** False

CreateQueueResponse

queue

- **Type:** Queue (p. 818)
- **Required:** False

ExceptionBody

message

- **Type:** string
- **Required:** False
**ListQueuesRequest**

**nextToken**

Use this string, provided with the response to a previous request, to request the next batch of queues.

*Type*: string  
*Required*: False

**maxResults**

Optional. Number of queues, up to twenty, that will be returned at one time.

*Type*: integer  
*Required*: False  
*Format*: int32

**listBy**

*Type*: QueueListBy (p. 818)  
*Required*: False

**order**

*Type*: Order (p. 817)  
*Required*: False

**ListQueuesResponse**

**queues**

*Type*: Array of type Queue (p. 818)  
*Required*: False

**nextToken**

Use this string to request the next batch of queues.

*Type*: string  
*Required*: False

**Order (Enum)**

When you request lists of resources, you can optionally specify whether they are sorted in ASCENDING or DESCENDING order. Default varies by resource.

ASCENDING  
DESCENDING
Queue

lastUpdated
The timestamp in epoch seconds when the queue was last updated.

Type: string  
Required: False  
Format: date-time

createdAt
The timestamp in epoch seconds for queue creation.

Type: string  
Required: False  
Format: date-time

name
A name you create for each queue. Each name must be unique within your account.

Type: string  
Required: True

description
An optional description you create for each queue.

Type: string  
Required: False

arn
An identifier for this resource that is unique within all of AWS.

Type: string  
Required: False

type

Type: Type (p. 819)  
Required: False

status

Type: QueueStatus (p. 819)  
Required: False

QueueListBy (Enum)
Optional. When you request a list of queues, you can choose to list them alphabetically by NAME or chronologically by CREATION_DATE. If you don't specify, the service will list them by creation date.
A Specific Queue

URI

/queues/name

HTTP Methods

GET

Operation ID: GetQueue

Retrieve the JSON for a specific queue.

Path Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>True</td>
<td></td>
</tr>
</tbody>
</table>

Responses

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>GetQueueResponse (p. 823)</td>
<td>200: OkHttpResponse</td>
</tr>
<tr>
<td>400</td>
<td>ExceptionBody (p. 824)</td>
<td>400: BadRequestException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The conditional request failed. The service can't process your request because of a problem in the request. Please check your request form and syntax.</td>
</tr>
<tr>
<td>500</td>
<td>ExceptionBody (p. 824)</td>
<td>500: InternalServiceException</td>
</tr>
</tbody>
</table>
### Status Code

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>403</td>
<td>ExceptionBody (p. 824)</td>
<td>The service encountered an unexpected condition and cannot fulfill your request.</td>
</tr>
</tbody>
</table>
| 404         | ExceptionBody (p. 824) | 404: ResourceNotFoundException
The resource you requested does not exist. |
| 429         | ExceptionBody (p. 824) | 429: LimitExceededException
Too many requests have been sent in too short of a time. The service limits the rate at which it will accept requests. For example, you may be hitting your account limits for preset creation or job submission. |
| 409         | ExceptionBody (p. 824) | 409: ResourceInUseException
The service could not complete your request because there is a conflict with the current state of the resource. For example, you may be trying to delete a Queue that has jobs processing. |

### PUT

**Operation ID: UpdateQueue**

Modify one of your existing queues.

**Path Parameters**

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>True</td>
<td></td>
</tr>
</tbody>
</table>
## Responses

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>UpdateQueueResponse (p. 824)</td>
<td>200: OkResponse</td>
</tr>
<tr>
<td>400</td>
<td>ExceptionBody (p. 824)</td>
<td>400: BadRequestException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The conditional request failed. The service can't process your request because of a problem in the request. Please check your request form and syntax.</td>
</tr>
<tr>
<td>500</td>
<td>ExceptionBody (p. 824)</td>
<td>500: InternalServiceException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The service encountered an unexpected condition and cannot fulfill your request.</td>
</tr>
<tr>
<td>403</td>
<td>ExceptionBody (p. 824)</td>
<td>403: AccessDeniedException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>You don't have permissions for this action with the credentials you sent. Please check your authorization credentials. You should be sending credentials using the AWS Signature Version 4 signing process.</td>
</tr>
<tr>
<td>404</td>
<td>ExceptionBody (p. 824)</td>
<td>404: ResourceNotFoundException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The resource you requested does not exist.</td>
</tr>
<tr>
<td>429</td>
<td>ExceptionBody (p. 824)</td>
<td>429: LimitExceededException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Too many requests have been sent in too short of a time. The service limits the rate at which it will accept requests. For example, you may be hitting your account limits for preset creation or job submission.</td>
</tr>
<tr>
<td>409</td>
<td>ExceptionBody (p. 824)</td>
<td>409: ResourceInUseException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The service could not complete your request because there is a conflict with the current state of the resource. For example, you may be trying to delete a Queue that has jobs processing.</td>
</tr>
</tbody>
</table>
DELETE

Operation ID: DeleteQueue

Permanently delete a queue you have created.

Path Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>True</td>
<td></td>
</tr>
</tbody>
</table>

Responses

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>400</td>
<td>ExceptionBody (p. 824)</td>
<td>400: BadRequestException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The conditional request failed.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The service can't process your request because of a problem in the request.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Please check your request form and syntax.</td>
</tr>
<tr>
<td>202</td>
<td>DeleteQueueResponse (p. 824)</td>
<td>202: AcceptedResponse</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Your request has been accepted.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Processing has not yet begun.</td>
</tr>
<tr>
<td>500</td>
<td>ExceptionBody (p. 824)</td>
<td>500: InternalServiceException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The service encountered an unexpected condition and cannot fulfill your request.</td>
</tr>
<tr>
<td>403</td>
<td>ExceptionBody (p. 824)</td>
<td>403: AccessDeniedException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>You don't have permissions for this action with the credentials you sent.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Please check your authorization credentials. You should be sending credentials</td>
</tr>
<tr>
<td></td>
<td></td>
<td>using the AWS Signature Version 4 signing process.</td>
</tr>
<tr>
<td>404</td>
<td>ExceptionBody (p. 824)</td>
<td>404: ResourceNotFoundException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The resource you requested does not exist.</td>
</tr>
<tr>
<td>429</td>
<td>ExceptionBody (p. 824)</td>
<td>429: LimitExceededException</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Too many requests have been sent in too short of a time.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The service limits the rate at which it will accept requests.</td>
</tr>
</tbody>
</table>
### Status Code | Response Model | Description
--- | --- | ---
409 | ExceptionBody (p. 824) | For example, you may be hitting your account limits for preset creation or job submission.

#### Schemas

##### Request Bodies

**Example GET**

```json
{
"name (p. 824)" : "string"
}
```

**Example PUT**

```json
{
"name (p. 826)" : "string",
"description (p. 826)" : "string",
"status (p. 826)" : enum
}
```

**Example DELETE**

```json
{
"name (p. 824)" : "string"
}
```

##### Response Bodies

**Example GetQueueResponse**

```json
{
"queue (p. 825)" : {
"lastUpdated (p. 825)" : "string",
"createdAt (p. 825)" : "string",
"name (p. 825)" : "string",
"description (p. 825)" : "string",
"arn (p. 825)" : "string",
"type (p. 825)" : enum,
"status (p. 825)" : enum
}
```
Example UpdateQueueResponse

```json
{
  "queue (p. 826)": {
    "lastUpdated (p. 825)": "string",
    "createdAt (p. 825)": "string",
    "name (p. 825)": "string",
    "description (p. 825)": "string",
    "arn (p. 825)": "string",
    "type (p. 825)": enum,
    "status (p. 825)": enum
  }
}
```

Example DeleteQueueResponse

```json
{
}
```

Example ExceptionBody

```json
{
  "message (p. 824)": "string"
}
```

Properties

**DeleteQueueRequest**

`name`

- **Type:** string
- **Required:** True

**DeleteQueueResponse**

**ExceptionBody**

`message`

- **Type:** string
- **Required:** False

**GetQueueRequest**

`name`

- **Type:** string
- **Required:** True
GetQueueResponse

queue

  Type: Queue (p. 825)
  Required: False

Queue

lastUpdated

The timestamp in epoch seconds when the queue was last updated.

  Type: string
  Required: False
  Format: date-time

createdAt

The timestamp in epoch seconds for queue creation.

  Type: string
  Required: False
  Format: date-time

name

A name you create for each queue. Each name must be unique within your account.

  Type: string
  Required: True

description

An optional description you create for each queue.

  Type: string
  Required: False

arn

An identifier for this resource that is unique within all of AWS.

  Type: string
  Required: False

type

  Type: Type (p. 826)
  Required: False

status

  Type: QueueStatus (p. 826)
Required: False

QueueStatus (Enum)

Queues can be ACTIVE or PAUSED. If you pause a queue, jobs in that queue will not begin. Jobs running when a queue is paused continue to run until they finish or error out.

ACTIVE
PAUSED

Type (Enum)

SYSTEM
CUSTOM

UpdateQueueRequest

name

The name of the queue you are modifying.

Type: string
Required: True

description

The new description for the queue, if you are changing it.

Type: string
Required: False

status

Type: QueueStatus (p. 826)
Required: False

UpdateQueueResponse

queue

Type: Queue (p. 825)
Required: False