Table of Contents

Getting Started with the AWS SDKs and the AWS CLI ................................................................. 1
Getting Started with the API ........................................................................................................... 16
  Postman Collection Files ........................................................................................................... 17
    GET Collection ....................................................................................................................... 17
    POST Collection ..................................................................................................................... 19
Sample Settings in JSON ............................................................................................................... 23
  Simple Example ....................................................................................................................... 23
  Complex Example ..................................................................................................................... 25
Important Notes ........................................................................................................................ 32
  Account-Specific Endpoints .................................................................................................... 32
  Versioned Endpoints .............................................................................................................. 32
  Using the AWS CLI ................................................................................................................ 32
  Schemas and Example Job Settings ........................................................................................ 32
Resources .................................................................................................................................... 33
  Certificates ............................................................................................................................. 33
    URI ......................................................................................................................................... 33
    HTTP Methods ..................................................................................................................... 33
    Schemas ............................................................................................................................... 34
    Properties ............................................................................................................................. 34
    See Also ............................................................................................................................... 35
  Certificates arn ........................................................................................................................ 35
    URI ......................................................................................................................................... 35
    HTTP Methods ..................................................................................................................... 35
    Schemas ............................................................................................................................... 36
    Properties ............................................................................................................................. 37
    See Also ............................................................................................................................... 37
Endpoints .................................................................................................................................... 38
  URI ......................................................................................................................................... 38
  HTTP Methods ..................................................................................................................... 38
  Schemas ............................................................................................................................... 38
  Properties ............................................................................................................................. 39
  See Also ................................................................................................................................... 40
JobTemplates ............................................................................................................................. 41
  URI ......................................................................................................................................... 41
  HTTP Methods ..................................................................................................................... 41
  Schemas ............................................................................................................................... 43
  Properties ............................................................................................................................. 79
  See Also ............................................................................................................................... 238
JobTemplates name .................................................................................................................... 238
  URI ......................................................................................................................................... 238
  HTTP Methods ..................................................................................................................... 238
  Schemas ............................................................................................................................... 241
  Properties ............................................................................................................................. 277
  See Also ............................................................................................................................... 434
Jobs ........................................................................................................................................... 435
  URI ......................................................................................................................................... 435
  HTTP Methods ..................................................................................................................... 435
  Schemas ............................................................................................................................... 437
  Properties ............................................................................................................................. 475
  See Also ............................................................................................................................... 638
Jobs id ....................................................................................................................................... 638
  URI ......................................................................................................................................... 638
  HTTP Methods ..................................................................................................................... 638
  Schemas ............................................................................................................................... 640
Getting Started with AWS Elemental MediaConvert Using the AWS SDKs or the AWS CLI

To get started with AWS Elemental MediaConvert using one of the AWS SDKs or the AWS Command Line Interface (AWS CLI), follow this general procedure. For specific instructions and examples, choose a language in the final step of this procedure.

1. Set up AWS Identity and Access Management (IAM) permissions for both yourself and for the MediaConvert service to access your resources on your behalf:
   - For information about setting up permissions for yourself, see Overview of Identity Management: Users in the IAM User Guide.
   - For information about setting up permissions for the service to access your resources, see Set Up IAM Permissions in the MediaConvert User Guide.

2. In your client configuration, specify your authentication credentials and your AWS Region. For instructions that are specific to the programming language that you use, choose from this list of links to open the relevant topics in the AWS CLI or SDK guides:
   - AWS CLI
   - C++: credentials and region
   - Go
   - Java
   - JavaScript
   - .NET
   - PHP
   - Python: credentials and region
   - Ruby
   - Tools for PowerShell

3. Get your account-specific endpoint and send your MediaConvert requests to it. With most AWS services, you send your service request to a public endpoint. But with MediaConvert, you request an endpoint that is specific to your account, and then you send your service requests to that.

   For specific instructions and code samples, choose one of the following tabs:

AWS CLI

To create a transcoding job using the AWS CLI:

1. Use describe-endpoints to get your account endpoint and set your region. In this example, the region is set to ap-northeast-3:

   ```
   aws mediaconvert describe-endpoints --region ap-northeast-3
   ```

2. Use the --endpoint-url option to send your request to your account endpoint:

   ```
   aws --endpoint-url https://abcd1234.mediaconvert.region-name-1.amazonaws.com --region region-name-1 mediaconvert create-job --cli-input-json file://~/job.json
   ```
In the preceding example, `job.json` specifies your transcoding job settings. You can use the MediaConvert console to generate the JSON job specification by choosing your job settings, and then choosing **Show job JSON** at the bottom of the **Job** section. For sample job specifications, see **Sample Settings in JSON** (p. 23).

### C++

To send requests using the SDK for C++:

```cpp
// MediaConvertPort.cpp : Defines the entry point for the console application.

#include <aws/core/Aws.h>
#include <aws/core/utils/Outcome.h>
#include <aws/mediaconvert/MediaConvertClient.h>
#include <aws/mediaconvert/Model/DescribeEndpointsRequest.h>
#include <aws/mediaconvert/Model/CreateJobRequest.h>
#include <aws/mediaconvert/Model/CreateJobResult.h>

/* ----------------------------------------------
 * Permissions IAM user needs to run this example
 * ----------------------------------------------
 *
 * {
 *   "Version": "2012-10-17",
 *   "Statement": [
 *     {
 *       "Sid": "VisualEditor0",
 *       "Effect": "Allow",
 *       "Action": [
 *         "mediaconvert:DescribeEndpoints",
 *         "mediaconvert:CreateJob"
 *       ],
 *       "Resource": "*"
 *     }
 *   ]
 * }
 */

/* --------------------------------------
 * JSON job settings used in this example
 * --------------------------------------
 *
 * {
 *   "UserMetadata": {
 *     "Customer": "Amazon"
 *   },
 *   "Role": "Your AWS Elemental MediaConvert role ARN",
 *   "Settings": {
 *     "OutputGroups": [
 *       {
 *         "Name": "File Group",
 *         "OutputGroupSettings": {
 *           "Type": "FILE_GROUP_SETTINGS",
 *           "FileGroupSettings": {
 *             "Destination": "s3://youroutputdestination"
 *           }
 *         }
 *       }
 *     ]
 *   }
 * }
 */
```
", "Outputs": [
  {
    "VideoDescription": {
      "ScalingBehavior": "DEFAULT",
      "TimecodeInsertion": "DISABLED",
      "AntiAlias": "ENABLED",
      "Sharpness": 50,
      "CodecSettings": {
        "Codec": "H_264",
        "H264Settings": {
          "InterlaceMode": "PROGRESSIVE",
          "NumberReferenceFrames": 3,
          "Syntax": "DEFAULT",
          "Softness": 0,
          "GopClosedCadence": 1,
          "GopSize": 90,
          "Slices": 1,
          "GopBReference": "DISABLED",
          "SlowPal": "DISABLED",
          "SpatialAdaptiveQuantization": "ENABLED",
          "TemporalAdaptiveQuantization": "ENABLED",
          "FlickerAdaptiveQuantization": "DISABLED",
          "EntropyEncoding": "CABAC",
          "Bitrate": 5000000,
          "FramerateControl": "SPECIFIED",
          "RateControlMode": "CBR",
          "CodecProfile": "MAIN",
          "Telecine": "NONE",
          "MinIInterval": 0,
          "AdaptiveQuantization": "HIGH",
          "CodecLevel": "AUTO",
          "FieldEncoding": "PAFF",
          "SceneChangeDetect": "ENABLED",
          "QualityTuningLevel": "SINGLE_PASS",
          "FramerateConversionAlgorithm": "DUPLICATE_DROP",
          "UnregisteredSeiTimecode": "DISABLED",
          "GopSizeUnits": "FRAMES",
          "ParControl": "SPECIFIED",
          "NumberBFramesBetweenReferenceFrames": 2,
          "RepeatPps": "DISABLED",
          "FramerateDenominator": 30,
          "FramerateNumerator": 30,
          "ParNumerator": 1,
          "ParDenominator": 1
        }
      },
      "AfdSignaling": "NONE",
      "DropFrameTimecode": "ENABLED",
      "RespondToAfd": "NONE",
      "ColorMetadata": "INSERT"
    },
    "AudioDescriptions": [
      {
        "AudioTypeControl": "FOLLOW_INPUT",
        "CodecSettings": {
          "Codec": "AAC",
          "AacSettings": {
            "AudioDescriptionBroadcasterMix": "NORMAL",
            "RateControlMode": "CBR",
            "CodecProfile": "LC",
            "CodingMode": "CODING_MODE_2_0",
            "RawFormat": "NONE",
            "SampleRate": 48000,
            "Specification": "MPEG4",
            "Bitrate": 64000
          }
        }
      }
    ]
  }
]
int main()
{
  // Initialize the C++ SDK
  Aws::SDKOptions options;
  Aws::InitAPI(options);
  
  // If we do not have our customer-specific endpoint
  
  std::string mediaConvertRole = "arn:aws:iam::640773029566:role/media-convert-role";
  std::string fileInput = "s3://media-convert-sample/my-video.mp4";
  std::string fileOutput = "s3://media-convert-sample";
  std::string mediaConvertEndpoint = "";
}
if (mediaConvertEndpoint.empty())
{
    // Obtain the customer-specific MediaConvert endpoint
    Aws::Client::ClientConfiguration clientConfig;
    clientConfig.region = "us-west-2";
    Aws::MediaConvert::MediaConvertClient client(clientConfig);
    Aws::MediaConvert::Model::DescribeEndpointsRequest request;
    // need to strip https:// from endpoint for C++
    mediaConvertEndpoint = client.DescribeEndpoints(request).GetResult().GetEndpoints().at(0).GetUrl().substr(8);
}

    // Create MediaConvert client with the endpoints and region from above
    Aws::Client::ClientConfiguration mcClientConfig;
    // Also need to set region endpoint, must match endpoint embedded in custom endpoint
    mcClientConfig.region = "us-west-2";
    mcClientConfig.endpointOverride = mediaConvertEndpoint;
    Aws::MediaConvert::MediaConvertClient mcClient(mcClientConfig);

    // Create job request
    Aws::MediaConvert::Model::CreateJobRequest createJobRequest;
    createJobRequest.SetRole(mediaConvertRole);
    Aws::Http::HeaderValueCollection hvc;
    hvc.emplace("Customer", "Amazon");
    createJobRequest.SetUserMetadata(hvc);

    // Create job settings
    Aws::MediaConvert::Model::JobSettings jobSettings;
    jobSettings.SetAdAvailOffset(0);
    Aws::MediaConvert::Model::TimecodeConfig timecodeConfig;
    timecodeConfig.SetSource(Aws::MediaConvert::Model::TimecodeSource::EMBEDDED);
    jobSettings.SetTimecodeConfig(timecodeConfig);
    createJobRequest.SetSettings(jobSettings);

    // Output Group
    Aws::MediaConvert::Model::OutputGroup og;
    og.SetName("File Group");
    Aws::MediaConvert::Model::OutputGroupSettings ogs;
    ogs.SetType(Aws::MediaConvert::Model::OutputGroupType::FILE_GROUP_SETTINGS);
    Aws::MediaConvert::Model::FileGroupSettings fgs;
    fgs.SetDestination(fileOutput);
    ogs.SetFileGroupSettings(fgs);
    og.SetOutputGroupSettings(ogs);
    Aws::MediaConvert::Model::Output output;
    output.SetNameModifier("_1");

    Aws::MediaConvert::Model::VideoDescription vdes;
    vdes.SetScalingBehavior(Aws::MediaConvert::Model::ScalingBehavior::DEFAULT);
    vdes.SetTimecodeInsertion(Aws::MediaConvert::Model::VideoTimecodeInsertion::DISABLED);
    vdes.SetAntiAlias(Aws::MediaConvert::Model::AntiAlias::ENABLED);
    vdes.SetSharpness(50);
    vdes.SetAfdSignaling(Aws::MediaConvert::Model::AfdSignaling::NONE);
    vdes.SetRespondToAfd(Aws::MediaConvert::Model::RespondToAfd::NONE);
    vdes.SetColorMetadata(Aws::MediaConvert::Model::ColorMetadata::INSERT);
    Aws::MediaConvert::Model::VideoCodecSettings vcs;
    vcs.SetCodec(Aws::MediaConvert::Model::VideoCodec::H_264);
    Aws::MediaConvert::Model::H264Settings h264;
    h264.SetNumberReferenceFrames(3);
    h264.SetSyntax(Aws::MediaConvert::Model::H264Syntax::DEFAULT);
```cpp
h264.SetSoftness(0);
h264.SetGopClosedCadence(1);
h264.SetGopSize(90);
h264.SetSlices(1);
h264.SetGopBReference(Aws::MediaConvert::Model::H264GopBReference::DISABLED);
h264.SetSlowPal(Aws::MediaConvert::Model::H264SlowPal::DISABLED);
h264.SetSpatialAdaptiveQuantization(Aws::MediaConvert::Model::H264SpatialAdaptiveQuantization::ENABLED);
h264.SetTemporalAdaptiveQuantization(Aws::MediaConvert::Model::H264TemporalAdaptiveQuantization::ENABLED);
h264.SetFlickerAdaptiveQuantization(Aws::MediaConvert::Model::H264FlickerAdaptiveQuantization::DISABLED);
h264.SetEntropyEncoding(Aws::MediaConvert::Model::H264EntropyEncoding::CABAC);
h264.SetBitrate(5000000);
h264.SetFramerateControl(Aws::MediaConvert::Model::H264FramerateControl::SPECIFIED);
h264.SetRateControlMode(Aws::MediaConvert::Model::H264RateControlMode::CBR);
h264.SetCodecProfile(Aws::MediaConvert::Model::H264CodecProfile::MAIN);
h264.SetTelecine(Aws::MediaConvert::Model::H264Telecine::NONE);
h264.SetMinIInterval(0);
h264.SetAdaptiveQuantization(Aws::MediaConvert::Model::H264AdaptiveQuantization::HIGH);
h264.SetCodecLevel(Aws::MediaConvert::Model::H264CodecLevel::AUTO);
h264.SetFieldEncoding(Aws::MediaConvert::Model::H264FieldEncoding::PAFF);
h264.SetSceneChangeDetect(Aws::MediaConvert::Model::H264SceneChangeDetect::ENABLED);
h264.SetQualityTuningLevel(Aws::MediaConvert::Model::H264QualityTuningLevel::SINGLE_PASS);
h264.SetFramerateConversionAlgorithm(Aws::MediaConvert::Model::H264FramerateConversionAlgorithm::DUPLICATE_DROP);
```

```cpp
h264.SetUnregisteredSeiTimecode(Aws::MediaConvert::Model::H264UnregisteredSeiTimecode::DISABLED);
h264.SetGopSizeUnits(Aws::MediaConvert::Model::H264GopSizeUnits::FRAMES);
h264.SetParControl(Aws::MediaConvert::Model::H264ParControl::SPECIFIED);
h264.SetNumberBFramesBetweenReferenceFrames(2);
h264.SetRepeatPps(Aws::MediaConvert::Model::H264RepeatPps::DISABLED);
h264.SetFramerateNumerator(30);
h264.SetFramerateDenominator(1);
h264.SetParNumerator(1);
h264.SetParDenominator(1);
```

```cpp
vcs.SetH264Settings(h264);
```

```cpp
vdes.SetCodecSettings(vcs);
```

```cpp
output.SetVideoDescription(vdes);
```

```cpp
Aws::MediaConvert::Model::AudioDescription ades;
```

```cpp
ades.SetLanguageCodeControl(Aws::MediaConvert::Model::AudioLanguageCodeControl::FOLLOW_INPUT);
```

```cpp
// This name matches one specified in the Inputs below
```

```cpp
ades.SetAudioSourceName("Audio Selector 1");
```

```cpp
Aws::MediaConvert::Model::AudioCodecSettings acs;
acs.SetCodec(Aws::MediaConvert::Model::AudioCodec::AAC);
```

```cpp
acs.SetAacSettings(aac);
```

```cpp
aac.SetAudioDescriptionBroadcasterMix(Aws::MediaConvert::Model::AacAudioDescriptionBroadcasterMix::BROADCAST_DEFAULT);
aac.SetRateControlMode(Aws::MediaConvert::Model::AacRateControlMode::CBR);
aac.SetCodecProfile(Aws::MediaConvert::Model::AacCodecProfile::LC);
aac.SetCodingMode(Aws::MediaConvert::Model::AacCodingMode::CODING_MODE_2_0);
aac.SetRawFormat(Aws::MediaConvert::Model::AacRawFormat::NONE);
aac.SetSampleRate(48000);
aac.SetSpecification(Aws::MediaConvert::Model::AacSpecification::MPEG4);
aac.SetBitrate(64000);
aac.SetAacSettings(aac);
```

```cpp
output.SetAudioDescriptions(adess);
```
After you use the DescribeEndpoints method to request an account-specific endpoint, send your requests to it as described in Creating an Account Endpoint in the AWS SDK for Go Developer Guide.

Important
Make the DescribeEndpoints call only once in your application. Don’t use DescribeEndpoints to create your AWS client each time that you make a request.
to MediaConvert. Otherwise, you will reach the throttle maximum on the public API endpoint.

Java

For information about setting up your credentials and region in your client configuration, see Set up AWS Credentials and Region for Development in the AWS SDK for Java Developer Guide.

This procedure shows you how to get your account-specific endpoint and send MediaConvert requests to it.

1. Add the following import statements:

```java
import com.amazonaws.client.builder.AwsClientBuilder.EndpointConfiguration;
import com.amazonaws.regions.Region;
import com.amazonaws.services.mediaconvert.*;
```

2. Determine the region for your endpoint and create an AWSMediaConvert client object for it:

```java
String region = "us-west-2";
AWSMediaConvert mediaConvertClient =
AWSMediaConvertClientBuilder.standard()
 .withRegion(region)
 .build();
```

3. Call the describeEndpoints method to retrieve the endpoint and save the endpoint's URL:

```java
DescribeEndpointsRequest request = new DescribeEndpointsRequest();
String endpoint = mediaConvertClient.describeEndpoints(request)
 .getEndpoints()
 .get(0).getUrl();
```

**Important**
Make the DescribeEndpoints call only once in your application. Don’t use DescribeEndpoints to create your AWS client each time that you make a request to MediaConvert. Otherwise, you will reach the throttle maximum on the public API endpoint.

4. Create a job request and a submit job request object:

```java
CreateJobRequest jobParam = new CreateJobRequest()
 .withSettings(jobSettings);  
CreateJobResult mcResponse = new CreateJobResult();
mcResponse = mcClient.createJob(jobParam)
```

**Note**
The jobSettings object contains settings parameters. For sample job specifications in JSON format, see Sample Settings in JSON (p. 23).

JavaScript

Find information and examples for using JavaScript to access MediaConvert in the AWS Elemental MediaConvert Examples topic of the SDK for JavaScript Developer Guide.
**Important**
Make the `DescribeEndpoints` call only once in your application. Don't use `DescribeEndpoints` to create your AWS client each time that you make a request to MediaConvert. Otherwise, you will reach the throttle maximum on the public API endpoint.

**.NET**

To send requests using the AWS SDK for .NET:

```csharp
using System;
using Amazon.MediaConvert;
using Amazon.MediaConvert.Model;

namespace MediaConvertNET
{
    /* ----------------------------------------------
    * Permissions IAM user needs to run this example
    * ----------------------------------------------
    * */
    {
        "Version": "2012-10-17",
        "Statement": [
            {
                "Sid": "VisualEditor0",
                "Effect": "Allow",
                "Action": [
                    "mediaconvert:DescribeEndpoints",
                    "mediaconvert:CreateJob"
                ],
                "Resource": "*
            }
        ]
    }
    /* --------------------------------------
    * JSON job settings used in this example
    * --------------------------------------
    * */
    {
        "UserMetadata": {
            "Customer": "Amazon"
        },
        "Role": "Your AWS Elemental MediaConvert role ARN",
        "Settings": {
            "OutputGroups": [
                {
                    "Name": "File Group",
                    "OutputGroupSettings": {
                        "Type": "FILE_GROUP_SETTINGS",
                        "FileGroupSettings": {
                            "Destination": "s3://youroutputdestination"
                        }
                    }
                },
                "Outputs": [
                    {
                        "VideoDescription": {
                            "ScalingBehavior": "DEFAULT",
                            "TimecodeInsertion": "DISABLED",
                            "AntiAlias": "ENABLED",
                            "Sharpness": 50,
                        }
                    }
                ]
            ]
        }
    }
}
```
"CodecSettings": {
    "Codec": "H_264",
    "H264Settings": {
        "InterlaceMode": "PROGRESSIVE",
        "NumberReferenceFrames": 3,
        "Syntax": "DEFAULT",
        "Softness": 0,
        "GopClosedCadence": 1,
        "GopSize": 90,
        "Slices": 1,
        "GopBReference": "DISABLED",
        "SlowPal": "DISABLED",
        "SpatialAdaptiveQuantization": "ENABLED",
        "TemporalAdaptiveQuantization": "ENABLED",
        "FlickerAdaptiveQuantization": "DISABLED",
        "EntropyEncoding": "CABAC",
        "Bitrate": 5000000,
        "FrameRateControl": "SPECIFIED",
        "RateControlMode": "CBR",
        "CodecProfile": "MAIN",
        "Telecine": "NONE",
        "MinIInterval": 0,
        "AdaptiveQuantization": "HIGH",
        "CodecLevel": "AUTO",
        "FieldEncoding": "PAFF",
        "SceneChangeDetect": "ENABLED",
        "FrameRateConversionAlgorithm": "DUPLICATE_DROP",
        "UnregisteredSeiTimecode": "DISABLED",
        "GopSizeUnits": "FRAMES",
        "ParControl": "SPECIFIED",
        "NumberBFramesBetweenReferenceFrames": 2,
        "RepeatPps": "DISABLED",
        "FrameRateNumerator": 30,
        "FrameRateDenominator": 1,
        "ParNumerator": 1,
        "ParDenominator": 1
    },
    "AfdSignaling": "NONE",
    "DropFrameTimecode": "ENABLED",
    "RespondToAfd": "NONE",
    "ColorMetadata": "INSERT"
},
"AudioDescriptions": [
    {
        "AudioTypeControl": "FOLLOW_INPUT",
        "CodecSettings": {
            "Codec": "AAC",
            "AacSettings": {
                "AudioDescriptionBroadcasterMix": "NORMAL",
                "RateControlMode": "CBR",
                "CodecProfile": "LC",
                "CodingMode": "CODING_MODE_2_0",
                "RawFormat": "NONE",
                "SampleRate": 48000,
                "Specification": "MPEG4",
                "Bitrate": 64000
            }
        },
        "LanguageCodeControl": "FOLLOW_INPUT",
        "AudioSourceName": "Audio Selector 1"
    }]
},
"ContainerSettings": {
    "Container": "MP4",
}
"Mp4Settings": {
  "CslgAtom": "INCLUDE",
  "FreeSpaceBox": "EXCLUDE",
  "MoovPlacement": "PROGRESSIVE_DOWNLOAD"
}
},
"NameModifier": "_1"
},
],
"AdAvailOffset": 0,
"Inputs": [
  {
    "AudioSelectors": {
      "Audio Selector 1": {
        "Offset": 0,
        "DefaultSelection": "NOT_DEFAULT",
        "ProgramSelection": 1,
        "SelectorType": "TRACK",
        "Tracks": [
          1
        ]
      }
    },
    "VideoSelector": {
      "ColorSpace": "FOLLOW",
      "FilterEnable": "AUTO",
      "PsiControl": "USE_PSI",
      "FilterStrength": 0,
      "DeblockFilter": "DISABLED",
      "DenoiseFilter": "DISABLED",
      "TimecodeSource": "EMBEDDED",
      "FileInput": "s3://yourinputfile"
    }
  },
  "TimecodeConfig": {
    "Source": "EMBEDDED"
  }
}
]*

class Program
{
  static void Main(string[] args)
  {
    string mediaConvertRole = "Your AWS Elemental MediaConvert role ARN";
    string fileInput = "s3://yourinputfile";
    string fileOutput = "s3://youroutputdestination";
    // Once you know what your customer endpoint is, set it here
    string mediaConvertEndpoint = "";
    // If we do not have our customer-specific endpoint
    if (string.IsNullOrEmpty(mediaConvertEndpoint))
    {
      // Obtain the customer-specific MediaConvert endpoint
      AmazonMediaConvertClient client = new AmazonMediaConvertClient(Amazon.RegionEndpoint.USWest2);
      DescribeEndpointsRequest describeRequest = new DescribeEndpointsRequest();
      DescribeEndpointsResponse describeResponse = client.DescribeEndpoints(describeRequest);
      mediaConvertEndpoint = describeResponse.Endpoints[0].Url;
    }
  }
}
// Since we have a service url for MediaConvert, we do not
// need to set RegionEndpoint. If we do, the ServiceURL will
// be overwritten
AmazonMediaConvertConfig mcConfig = new AmazonMediaConvertConfig
{
    ServiceURL = mediaConvertEndpoint,
};

AmazonMediaConvertClient mcClient = new
AmazonMediaConvertClient(mcConfig);
CreateJobRequest createJobRequest = new CreateJobRequest();
createJobRequest.Role = mediaConvertRole;
createJobRequest.UserMetadata.Add("Customer", "Amazon");

#region Create job settings
JobSettings jobSettings = new JobSettings();
jobSettings.AdAvailOffset = 0;
jobSettings.TimecodeConfig = new TimecodeConfig();
jobSettings.TimecodeConfig.Source = TimecodeSource.EMBEDDED;
createJobRequest.Settings = jobSettings;

#region OutputGroup
OutputGroup ofg = new OutputGroup();
ofg.Name = "File Group";
ofg.OutputGroupSettings = new OutputGroupSettings();
ofg.OutputGroupSettings.Type = OutputGroupType.FILE_GROUP_SETTINGS;
Output output = new Output();
output.NameModifier = "_1";

#region VideoDescription
VideoDescription vdes = new VideoDescription();
output.VideoDescription = vdes;
vdes.ScalingBehavior = ScalingBehavior.DEFAULT;
vdes.TimecodeInsertion = VideoTimecodeInsertion.DISABLED;
vdes.AntiAlias = AntiAlias.ENABLED;
vdes.Sharpness = 50;
vdes.AfdSignaling = AfdSignaling.NONE;
vdes.DropFrameTimecode = DropFrameTimecode.ENABLED;
vdes.RespondToAfd = RespondToAfd.NONE;
vdes.ColorMetadata = ColorMetadata.INSERT;
vdes.CodecSettings = new VideoCodecSettings();
H264Settings h264 = new H264Settings();
h264.InterlaceMode = H264InterlaceMode.PROGRESSIVE;
h264.NumberReferenceFrames = 3;
h264.Syntax = H264Syntax.DEFAULT;
h264.Softness = 0;
h264.GopClosedCadence = 1;
h264.GopSize = 90;
h264.Slices = 1;
h264.GopBReference = H264GopBReference.DISABLED;
h264.SlowPal = H264SlowPal.DISABLED;
h264.SpatialAdaptiveQuantization = H264SpatialAdaptiveQuantization.ENABLED;
h264.TemporalAdaptiveQuantization = H264TemporalAdaptiveQuantization.ENABLED;
h264.FlickerAdaptiveQuantization = H264FlickerAdaptiveQuantization.ENABLED;
h264.EntropyEncoding = H264EntropyEncoding.CABAC;
h264.Bitrate = 5000000;
h264.FramerateControl = H264FramerateControl.SPECIFIED;
h264.RateControlMode = H264RateControlMode.CBR;
h264.CodecProfile = H264CodecProfile.MAIN;
h264.Telecine = H264Telecine.NONE;
h264.MinIInterval = 0;
h264.AdaptiveQuantization = H264AdaptiveQuantization.HIGH;
h264.CodecLevel = H264CodecLevel.AUTO;
h264.FieldEncoding = H264FieldEncoding.PAFF;
h264.SceneChangeDetect = H264SceneChangeDetect.ENABLED;
h264.QualityTuningLevel = H264QualityTuningLevel.SINGLE_PASS;
h264.FramerateConversionAlgorithm = H264FramerateConversionAlgorithm.DUPLICATE_DROP;
h264.UnregisteredSeiTimecode = H264UnregisteredSeiTimecode.DISABLED;
h264.GopSizeUnits = H264GopSizeUnits.FRAMES;
h264.ParControl = H264ParControl.SPECIFIED;
h264.NumberBFramesBetweenReferenceFrames = 2;
h264.RepeatPps = H264RepeatPps.DISABLED;
h264.FramerateNumerator = 30;
h264.FramerateDenominator = 1;
h264.ParNumerator = 1;
h264.ParDenominator = 1;
output.VideoDescription.CodecSettings.H264Settings = h264;
#endregion VideoDescription

#region AudioDescription
AudioDescription ades = new AudioDescription();
// This name matches one specified in the Inputs below
ades.LanguageCodeControl = AudioLanguageCodeControl.FOLLOW_INPUT;
ades.AudioSourceName = "Audio Selector 1";
ades.CodecSettings = new AudioCodecSettings();
ades.CodecSettings.Codec = AudioCodec.AAC;
AacSettings aac = new AacSettings();
aac.AudioDescriptionBroadcasterMix = AacAudioDescriptionBroadcasterMix.NORMAL;
aac.RateControlMode = AacRateControlMode.CBR;
aac.CodecProfile = AacCodecProfile.LC;
aac.CodingMode = AacCodingMode.CODING_MODE_2_0;
aac.RawFormat = AacRawFormat.NONE;
aac.SampleRate = 48000;
aac.Specification = AacSpecification.MPEG4;
aac.Bitrate = 64000;
ades.CodecSettings.AacSettings = aac;
output.AudioDescriptions.Add(ades);
#endregion AudioDescription

#region Mp4 Container
output.ContainerSettings = new ContainerSettings();
output.ContainerSettings.Container = ContainerType.MP4;
Mp4Settings mp4 = new Mp4Settings();
mp4.CslgAtom = Mp4CslgAtom.INCLUDE;
mp4.FreeSpaceBox = Mp4FreeSpaceBox.EXCLUDE;
mp4.MoovPlacement = Mp4MoovPlacement.PROGRESSIVE_DOWNLOAD;
output.ContainerSettings.Mp4Settings = mp4;
#endregion Mp4 Container

ofg.Outputs.Add(output);
createJobRequest.Settings.OutputGroups.Add(ofg);
#endregion OutputGroup

#region Input
Input input = new Input();
input.FilterEnable = InputFilterEnable.AUTO;
input.PsiControl = InputPsiControl.USE_PSI;
input.FilterStrength = 0;
input.DeblockFilter = InputDeblockFilter.DISABLED;
input.DenoiseFilter = InputDenoiseFilter.DISABLED;
input.TimecodeSource = InputTimecodeSource.EMBEDDED;
input.FileInput = fileInput;

AudioSelector audsel = new AudioSelector();
audsel.Offset = 0;
audsel.DefaultSelection = AudioDefaultSelection.NOT_DEFAULT;
audsel.ProgramSelection = 1;
audsel.SelectorType = AudioSelectorType.TRACK;
audsel.Tracks.Add(1);
input.AudioSelectors.Add("Audio Selector 1", audsel);

input.VideoSelector = new VideoSelector();
input.VideoSelector.ColorSpace = ColorSpace.FOLLOW;

createJobRequest.Settings.Inputs.Add(input);

try {
    CreateJobResponse createJobResponse =
        mcClient.CreateJob(createJobRequest);
    Console.WriteLine("Job Id: {0}", createJobResponse.Job.Id);
} catch (BadRequestException bre) {
    // If the endpoint was bad
    if (bre.Message.StartsWith("You must use the customer-")) {
        // The exception contains the correct endpoint; extract it
        mediaConvertEndpoint = bre.Message.Split('\')[1];
        // Code to retry query
    }
}

For more information about sending requests to an account endpoint, see the Overriding Endpoints in the AWS SDK for .NET post in the AWS Developer Blog.

**PHP**

Find information and examples for using PHP to access MediaConvert in the AWS Elemental MediaConvert Examples topic of the AWS SDK for PHP Developer Guide.

**Important**

Make the DescribeEndpoints call only once in your application. Don't use DescribeEndpoints to create your AWS client each time that you make a request to MediaConvert. Otherwise, you will reach the throttle maximum on the public API endpoint.

**Python**

Follow these steps to send requests using the AWS SDK for Python (Boto):

1. Use the describe_endpoints method to request an account-specific endpoint.

**Important**

Make the DescribeEndpoints call only once in your application. Don't use DescribeEndpoints to create your AWS client each time that you make a request to MediaConvert. Otherwise, you will reach the throttle maximum on the public API endpoint.
2. Specify the transcoding settings for your job in a JSON file.

You can use the AWS Elemental MediaConvert console to generate the JSON job specification by choosing your job settings, and then choosing Show job JSON at the bottom of the Job section. For sample job specifications, see Sample Settings in JSON (p. 23).

3. Perform these steps in your Python code, as demonstrated in the example that follows these steps:
   a. Create the client with your account-specific endpoint specified.
   b. Load your settings JSON file.
   c. Create the transcoding job using create_job.

```python
import json
import boto3

# Create MediaConvert client
mediaconvert_client = boto3.client('mediaconvert', endpoint_url='https://abcd1234.mediaconvert.us-west-2.amazonaws.com')

# Load job.json from disk and store as Python object: job_object
with open("job.json", "r") as jsonfile:
    job_object = json.load(jsonfile)

# Create MediaConvert job by unpacking the arguments from job_object. The job object contains the required parameters
# for create_job. Pass these to create_job using Python's ** argument unpacking syntax.
mediaconvert_client.create_job(**job_object)
```

Ruby

After you use the DescribeEndpointsRequest method to request an account-specific endpoint, send your requests to it as described in Setting a Nonstandard Endpoint in the AWS SDK for Ruby Developer Guide.

**Important**
Make the DescribeEndpoints call only once in your application. Don't use DescribeEndpoints to create your AWS client each time that you make a request to MediaConvert. Otherwise, you will reach the throttle maximum on the public API endpoint.

Tools for Powershell

After you use the Get-EMCEndpoint Cmdlet to request an account-specific endpoint, send your requests to it as described inSpecifying a Custom or Nonstandard Endpoint in the AWS Tools for PowerShell User Guide. In brief, you specify your account-specific endpoint as a URL by adding the -EndpointUrl common parameter to your AWS Tools for PowerShell command.

In this example, replace https://abcd1234.mediaconvert.us-west-1.amazonaws.com with the endpoint that you get back from your Get-EMCEndpoint Cmdlet request:

```
AWS-PowerShellCmdlet -EndpointUrl "https://abcd1234.mediaconvert.us-west-1.amazonaws.com." -Other -Parameters
```

**Important**
Make the DescribeEndpoints call only once in your application. Don't use DescribeEndpoints to create your AWS client each time that you make a request to MediaConvert. Otherwise, you will reach the throttle maximum on the public API endpoint.
Getting Started with MediaConvert Using the API

This section shows you how to get started with the MediaConvert API to transcode media files using API calls.

**Important**
Unlike most AWS services, AWS Elemental MediaConvert requires that you send your requests to an endpoint that is specific to your account. Use the following steps to get going.

**To get set up to use the AWS Elemental MediaConvert API**

1. **Set up permissions:**
   - **Permissions that the AWS Elemental MediaConvert service can assume on your behalf.** These allow access to your Amazon S3 buckets and to Amazon API Gateway. For instructions, see Set Up IAM Permissions in the *AWS Elemental MediaConvert User Guide*.
   - **Your Signature Version 4 authentication for the requests that you send to AWS.** When you use the AWS Command Line Interface (AWS CLI) or one of the AWS SDKs, these tools automatically sign the requests for you with the access key and secret key that you specify in your client configuration.

2. **Set up S3 file locations.**

   The service reads your input files from and saves your output files to Amazon S3 buckets. For instructions on creating these buckets, see Create Storage for Files in the *AWS Elemental MediaConvert User Guide*.

3. **Request your account endpoint.**

   Send a `POST` request with an empty body to the following endpoint, replacing `<region>` with the name of your region. Usually, this is the region where you store your input and output files. This region must match the region that you set up in your client configuration:

   ```
   https://mediaconvert.<region>.amazonaws.com/2017-08-29/endpoints
   ```

   For example, for the region `ap-southeast-2`, the endpoint is the following:

   ```
   https://mediaconvert.ap-southeast-2.amazonaws.com/2017-08-29/endpoints
   ```

   For a full list of MediaConvert public endpoints, see AWS Elemental MediaConvert Regions and Endpoints.

   The service returns an endpoint similar to "https://abcd1234.mediaconvert.us-west-1.amazonaws.com". The first eight digits after "https://" are an alphanumeric combination that is unique to your account. The region matches the regional endpoint that you send the request to. This is the endpoint to use for your transcoding requests.

   **Important**
   If you request your account endpoint programmatically, do so only once in your application. Don't make a request to the public endpoint each time that you make a request to MediaConvert. Otherwise, you will reach the throttle maximum on the public API endpoint.
4. **Send your transcoding requests.**

Using the account endpoint that you received from your POST request, send your requests to manage transcoding jobs, queues, job templates, and presets. For general information about how these resources work, see the [AWS Elemental MediaConvert User Guide](#). For high-level information about each resource, including information about each transcoding setting, see the Resources portion of this guide.

If you use Postman to send your requests, you can optionally copy and import the preconfigured Postman Collection Files (p. 17).

### Postman Collection Files

You can use the following Postman collections to simplify the process of making calls to MediaConvert through the REST API. Copy the text in each of the following sections into a text editor, replace the example endpoint information, and save it as a *.json file. Then, import the collection files into Postman.

#### Topics

- GET Collection (p. 17)
- POST Collection (p. 19)

#### GET Collection

```json
{
  "owner": "2332976",
  "lastUpdatedBy": "2332976",
  "lastRevision": 1921667904,
  "team": null,
  "id": "87fac2df-dd0f-b54a-b1f9-5b138cb4147f",
  "name": "EMF Get",
  "description": "EMF Get Template",
  "folders_order": [],
  "order": [
    "bc671df5-4a85-54b6-f137-19cb70516fd2",
    "85318a0b-c490-3718-62eb-2a737de83af0",
    "1fd40def-ca4b-1842-c99a-778f62269010"
  ],
  "folders": [],
  "hasRequests": true,
  "requests": [
    {"id": "1fd40def-ca4b-1842-c99a-778f62269010",
     "headers": "Content-Type: application/json",
     "headerData": ["Content-Type", "application/json", "\"
     "enabled": true
     ]
    ],
    "url": "https://<custom-account-id>.mediaconvert.<region>.amazonaws.com/2017-08-29/
    queues",
    "folder": null,
    "queryParams": []
}
```

17
"preRequestScript": null,
"pathVariables": {},
"pathVariableData": [],
"method": "GET",
"data": null,
"dataMode": "params",
"tests": null,
"currentHelper": "awsSigV4",
"helperAttributes": {
  "accessKey": "AccessKey",
  "secretKey": "SecretKey",
  "region": "supported-region",
  "service": "mediaconvert",
  "saveToRequest": true
},
"time": 1513791262493,
"name": "GET List Queue ",
"description": "",
"collectionId": "87fac2df-dd0f-b54a-b1f9-5b138cb4147f",
"responses": []
},
{
  "id": "85318a0b-c490-3718-62eb-2a737de83af0",
  "headers": "Content-Type: application/json",
  "headerData": [
    {
      "key": "Content-Type",
      "value": "application/json",
      "description": "",
      "enabled": true
    }
  ],
  "url": "https://<custom-account-id>.mediaconvert.<region>.amazonaws.com/2017-08-29/queues/<QUEUE-NAME-HERE>",
  "folder": null,
  "queryParams": [],
  "preRequestScript": null,
  "pathVariables": {},
  "pathVariableData": [],
  "method": "GET",
  "data": null,
  "dataMode": "params",
  "tests": null,
  "currentHelper": "awsSigV4",
  "helperAttributes": {
    "accessKey": "AccessKey",
    "secretKey": "SecretKey",
    "region": "supported-region",
    "service": "mediaconvert",
    "saveToRequest": true
  },
  "time": 1507243078514,
  "name": "GET Queue Details",
  "description": "",
  "collectionId": "87fac2df-dd0f-b54a-b1f9-5b138cb4147f",
  "responses": []
},
{
  "id": "bc671df5-4a85-54b6-f137-19cb70516fd2",
  "headers": "Content-Type: application/json",
  "headerData": [
    {
      "key": "Content-Type",
      "value": "application/json",
      "description": "",
      "enabled": true
    }
  ],
  "url": "https://<custom-account-id>.mediaconvert.<region>.amazonaws.com/2017-08-29/queues/<QUEUE-NAME-HERE>",
  "folder": null,
  "queryParams": [],
  "preRequestScript": null,
  "pathVariables": {},
  "pathVariableData": [],
  "method": "GET",
  "data": null,
  "dataMode": "params",
  "tests": null,
  "currentHelper": "awsSigV4",
  "helperAttributes": {
    "accessKey": "AccessKey",
    "secretKey": "SecretKey",
    "region": "supported-region",
    "service": "mediaconvert",
    "saveToRequest": true
  },
  "time": 1507243078514,
  "name": "GET Queue Details",
  "description": "",
  "collectionId": "87fac2df-dd0f-b54a-b1f9-5b138cb4147f",
  "responses": []
}
POST Collection

{  
  "id": "a1be92f5-37d5-aaf0-06bb-14090d825c62",
  "name": "AWS Elemental MediaConvert POST",
  "description": "POST Template",
  "order": [
    "0fd3c4a5-fa08-2dbc-1f0a-955942664858",
    "d6ffaf05-0c3a-35ee-8a3d-4457a96f4926"
  ],
  "folders": [],
  "folders_order": [],
  "timestamp": 0,
  "owner": "2332976",
  "public": false,
  "requests": [
    {
      "id": "0fd3c4a5-fa08-2dbc-1f0a-955942664858",
      "headers": "Content-Type: application/json\n",
      "headerData": [
        {
          "key": "Content-Type",
          "value": "application/json",
          "description": "",
          "enabled": true
        }
      ],
      "url": "https://<custom-account-id>.mediaconvert.<region>.amazonaws.com/2017-08-29/"
    },
    {
      "id": "0fd3c4a5-fa08-2dbc-1f0a-955942664858",
      "headers": "Content-Type: application/json\n",
      "headerData": [
        {
          "key": "Content-Type",
          "value": "application/json",
          "description": "",
          "enabled": true
        }
      ],
      "url": "https://<custom-account-id>.mediaconvert.<region>.amazonaws.com/2017-08-29/"
    }
  ]
}
"queryParams": [
  {
    "key": "AWS_Region",
    "value": "eu-west-1",
    "equals": false,
    "description": "",
    "enabled": false
  },
  {
    "key": "AWS_Access_Key",
    "value": "KEY",
    "equals": false,
    "description": "",
    "enabled": false
  },
  {
    "key": "AWS_Secret_Key",
    "value": "KEY",
    "equals": false,
    "description": "",
    "enabled": false
  }
],
"preRequestScript": "",
"pathVariables": {},
"pathVariableData": [],
"method": "POST",
"data": [],
"dataMode": "raw",
"tests": "",
"currentHelper": "awsSigV4",
"helperAttributes": {
  "accessKey": "AccessKey",
  "secretKey": "SecretKey",
  "region": "supported-region",
  "service": "mediaconvert",
  "saveToRequest": true
},
"time": 1510272274641,
"name": "Post MP4 Job",
"description": "",
"collectionId": "a1be92f5-37d5-aaf0-06bb-14090d825c62",
"responses": [],
"rawModeData": "{|\n  "userMetadata": {},\n  "role": "ROLE ARN HERE\",
  "settings": {\n    "outputGroups": [{\n      "name": "File Group",
      "container": "MP4",
      "mp4Settings": {\n        "cslgAtom": "INCLUDE",
        "moovPlacement": "PROGRESSIVE_DOWNLOAD"
      },
      "videoDescription": {\n        "scalingBehavior": "DISABLED",
        "timecodeInsertion": "DISABLED",
        "antiAlias": "DISABLED",
        "sharpness": 50,
        "syntax": "DEFAULT",
        "softness": 0,
        "gopClosedCadence": 1,
        "gopSize": 90,
        "slices": 1,
        "adaptationpoint": "DISABLED",
        "temporalAdaptiveQuantization": "ENABLED",
        "entropyEncoding": "CABAC",
        "bitrate": 5000000,
        "framerateControl": "INITIALIZE_FROM_SOURCE",
        "rateControlMode": "CBR",
        "codecProfile": "MAIN",
        "telecine": "NONE",
        "adaptiveQuantization": "MEDIUM",
        "fieldEncoding": "PAFF",
      }
    }
  }
|\n|"
"": "ENABLED",
      "qualityTuningLevel": "SINGLE_PASS",
      "framerateConversionAlgorithm": "DUPLICATE_DROP",
      "unregisteredSeiTimecode": "DISABLED",
      "gopSizeUnits": "FRAMES",
      "parControl": "INITIALIZE_FROM_SOURCE",
      "numberBFramesBetweenReferenceFrames": 2,
      "repeatPps": "NONE",
      "dropFrameTimecode": "ENABLED",
      "respondToAfd": "NONE",
      "colorMetadata": "INSERT",
      "audioDescriptions": [
        {
          "audioTypeControl": "FOLLOW_INPUT",
          "codecSettings": {
            "codec": "AAC",
            "aacSettings": {
              "audioDescriptionBroadcasterMix": "NORMAL",
              "bitrate": 96000,
              "rateControlMode": "CBR",
              "rawFormat": "NONE",
              "specification": "MPEG4"
            }
          },
          "languageCodeControl": "FOLLOW_INPUT"
        }
      ],
      " outputs": [
        {
          "audioSelectors": {
            "Audio Selector 1": {
              "offset": 0,
              "defaultSelection": "DEFAULT",
              "programSelection": 1
            }
          },
          "videoSelector": {
            "colorSpace": "FOLLOW"
          },
          "filterEnable": "AUTO",
          "deblockFilter": "DISABLED",
          "denoiseFilter": "DISABLED",
          "timecodeSource": "EMBEDDED",
          "fileInput": "s3://bucket/file.mp4"
        }
      ]
    },
  }
}
"pathVariables": {},
"pathVariableData": [],
"method": "POST",
"data": [],
"dataMode": "raw",
"tests": "",
"currentHelper": "awsSigV4",
"helperAttributes": {
  "accessKey": "AccessKey",
  "secretKey": "SecretKey",
  "region": "Supported Region",
  "service": "mediaconvert",
  "saveToRequest": true
},
"time": 1510272153358,
"name": "POST Request Account Endpoint",
"description": "",
"collectionId": "a1be92f5-37d5-aaf0-06bb-14090d825c62",
"responses": [],
"rawModeData": ""}
Sample Job Settings Specifications in JSON

These sample AWS Elemental MediaConvert jobs are specified in JSON. When you use the Python SDK or the AWS CLI, you pass in your job settings directly as a JSON object. When you use the other SDKs, translate the job settings according to the SDK documentation.

Simple Example

This example JSON job settings specification describes a job that creates a single file output:

```json
{
    "UserMetadata": {
        "Customer": "Amazon"
    },
    "Role": "arn:aws:iam::505474453218:role/EMFRoleSPNames",
    "Settings": {
        "OutputGroups": [
            {
                "Name": "File Group",
                "OutputGroupSettings": {
                    "Type": "FILE_GROUP_SETTINGS",
                    "FileGroupSettings": {
                        "Destination": "s3://testbucket/output"
                    }
                },
                "Outputs": [
                    {
                        "VideoDescription": {
                            "ScalingBehavior": "DEFAULT",
                            "TimecodeInsertion": "DISABLED",
                            "AntiAlias": "ENABLED",
                            "Sharpness": 50,
                            "CodecSettings": {
                                "Codec": "H_264",
                                "H264Settings": {
                                    "InterlaceMode": "PROGRESSIVE",
                                    "NumberReferenceFrames": 3,
                                    "Syntax": "DEFAULT",
                                    "Softness": 0,
                                    "GopClosedCadence": 1,
                                    "GopSize": 90,
                                    "Slices": 1,
                                    "GopBReference": "DISABLED",
                                    "SlowPal": "DISABLED",
                                    "SpatialAdaptiveQuantization": "ENABLED",
                                    "TemporalAdaptiveQuantization": "ENABLED",
                                    "FlickerAdaptiveQuantization": "DISABLED",
                                    "EntropyEncoding": "CABAC",
                                    "Bitrate": 5000000,
                                    "FramerateControl": "SPECIFIED",
                                    "RateControlMode": "CBR",
                                    "CodecProfile": "MAIN",
                                    "Telecine": "NONE"
                                }
                            }
                        }
                    }
                ]
            }
        ]
    }
}
```
"MinInterval": 0,
"AdaptiveQuantization": "HIGH",
"CodecLevel": "AUTO",
"FieldEncoding": "PAFF",
"SceneChangeDetect": "ENABLED",
"QualityTuningLevel": "SINGLE_PASS",
"FramerateConversionAlgorithm": "DUPLICATE_DROP",
"UnregisteredSeiTimecode": "DISABLED",
"GopSizeUnits": "FRAMES",
"ParControl": "SPECIFIED",
"NumberBFramesBetweenReferenceFrames": 2,
"RepeatPps": "DISABLED",
"FramerateNumerator": 30,
"FramerateDenominator": 1,
"ParNumerator": 1,
"ParDenominator": 1
}
},
"AdSignaling": "NONE",
"DropFrameTimecode": "ENABLED",
"RespondToAd": "NONE",
"ColorMetadata": "INSERT"
},
"AudioDescriptions": [
{
"AudioTypeControl": "FOLLOW_INPUT",
"CodecSettings": {
"Codec": "AAC",
"AacSettings": {
"AudioDescriptionBroadcastermix": "NORMAL",
"RateControlMode": "CBR",
"CodecProfile": "LC",
"CodingMode": "CODING_MODE_2_0",
"RawFormat": "NONE",
"SampleRate": 48000,
"Specification": "MPEG4",
"Bitrate": 64000
}
},
"LanguageCodeControl": "FOLLOW_INPUT",
"AudioSourceName": "Audio Selector 1"
}
],
"ContainerSettings": {
"Container": "MP4",
"Mp4Settings": {
"CsrlgAtom": "INCLUDE",
"FreeSpaceBox": "EXCLUDE",
"MoovPlacement": "PROGRESSIVE_DOWNLOAD"
}
},
"NameModifier": "_1"
]}],
"AdAvailOffset": 0,
"Inputs": [
{
"AudioSelectors": {
"Audio Selector 1": {
"Offset": 0,
"DefaultSelection": "NOT_DEFAULT",
"ProgramSelection": 1,
"SelectorType": "TRACK",
"Tracks": [
Complex Example

This example JSON job settings specification describes a job that creates output in multiple packages for viewing on different device types. It uses encryption for digital rights management:

```json
{
    "UserMetadata": {
        "Customer": "Amazon"
    },
    "Role": "arn:aws:iam::111122223333:role/MediaConvertRole",
    "Settings": {
        "OutputGroups": [
            {
                "CustomName": "",
                "Name": "DASH ISO",
                "Outputs": [
                    {
                        "ContainerSettings": {
                            "Container": "MPD"
                        },
                        "VideoDescription": {
                            "Width": 1080,
                            "ScalingBehavior": "DEFAULT",
                            "Height": 720,
                            "TimecodeInsertion": "DISABLED",
                            "AntiAlias": "ENABLED",
                            "Sharpness": 50,
                            "CodecSettings": {
                                "Codec": "H_264",
                                "H264Settings": {
                                    "InterlaceMode": "PROGRESSIVE",
                                    "NumberReferenceFrames": 3,
                                    "Syntax": "DEFAULT",
                                    "Softness": 0,
                                    "GopClosedCadence": 1,
                                    "GopSize": 30,
                                    "Slices": 1,
                                    "GopBReference": "DISABLED",
                                    "SlowPal": "DISABLED"....
```
"SpatialAdaptiveQuantization": "ENABLED",
"TemporalAdaptiveQuantization": "ENABLED",
"FlickerAdaptiveQuantization": "DISABLED",
"EntropyEncoding": "CABAC",
"Bitrate": 5000000,
"FramerateControl": "INITIALIZE_FROM_SOURCE",
"RateControlMode": "CBR",
"CodecProfile": "MAIN",
"Telecine": "NONE",
"MinIInterval": 0,
"AdaptiveQuantization": "HIGH",
"CodecLevel": "AUTO",
"FieldEncoding": "PAFF",
"SceneChangeDetect": "ENABLED",
"QualityTuningLevel": "SINGLE_PASS",
"FramerateConversionAlgorithm": "DUPLICATE_DROP",
"UnregisteredSeiTimecode": "DISABLED",
"GopSizeUnits": "FRAMES",
"ParControl": "INITIALIZE_FROM_SOURCE",
"NumberBFramesBetweenReferenceFrames": 2,
"RepeatPps": "DISABLED",
"HrdBufferSize": 10000000
},
"AfdSignaling": "NONE",
"DropFrameTimecode": "ENABLED",
"RespondToAfd": "NONE",
"ColorMetadata": "INSERT"
},
"NameModifier": "-1080"
],
{
"ContainerSettings": {
"Container": "MPD"
},
"AudioDescriptions": [
{
"AudioTypeControl": "FOLLOW_INPUT",
"AudioSourceName": "Audio Selector 1",
"CodecSettings": {
"Codec": "AAC",
"AacSettings": {
"AudioDescriptionBroadcasterMix": "NORMAL",
"Bitrate": 96000,
"RateControlMode": "CBR",
"CodecProfile": "LC",
"CodingMode": "CODING_MODE_2_0",
"RawFormat": "NONE",
"SampleRate": 48000,
"Specification": "MPEG4"
}
},
"LanguageCodeControl": "FOLLOW_INPUT"
},
"NameModifier": "-audio"
]
"OutputGroupSettings": {
"Type": "DASH_ISO_GROUP_SETTINGS",
"DashIsoGroupSettings": {
"SegmentLength": 30,
"Destination": "s3://testbucket/drm/10/dash-drm/master",
"Encryption": {
"SpekeKeyProvider": {
"ResourceId": "drm-test-1",
"Resourceid": "drm-test-1",
"RepeatPps": "DISABLED",
"HrdBufferSize": 10000000
},
"AfdSignaling": "NONE",
"DropFrameTimecode": "ENABLED",
"RespondToAfd": "NONE",
"ColorMetadata": "INSERT"
},
"NameModifier": "-1080"
}
]
"SystemIds": [ 
  "edef8ba9-79d6-4ace-a3c8-27dc51d21ed"
],
"Url": "https://abcdefg123.execute-api.us-west-2.amazonaws.com/live/speke/v1.0/copyProtection"
},
"FragmentLength": 2,
"SegmentControl": "SINGLE_FILE",
"HbbtvCompliance": "NONE"
}
],
"Name": "Apple HLS",
"OutputGroupSettings": {
  "Type": "HLS_GROUP_SETTINGS",
  "HlsGroupSettings": {
    "ManifestDurationFormat": "INTEGER",
    "SegmentLength": 10,
    "TimedMetadataId3Period": 10,
    "CaptionLanguageSetting": "OMIT",
    "TimedMetadataId3Frame": "PRIV",
    "CodecSpecification": "RFC_4281",
    "OutputSelection": "MANIFESTS_AND_SEGMENTS",
    "ProgramDateTimePeriod": 600,
    "MinSegmentLength": 0,
    "DirectoryStructure": "SINGLE_DIRECTORY",
    "ProgramDateTime": "EXCLUDE",
    "SegmentControl": "SEGMENTED_FILES",
    "ManifestCompression": "NONE",
    "ClientCache": "ENABLED",
    "StreamInfResolution": "INCLUDE",
    "Destination": "s3://testbucket/hls/no-drm/master"
  }
},
"Outputs": [
  {
    "VideoDescription": {
      "ScalingBehavior": "DEFAULT",
      "TimecodeInsertion": "DISABLED",
      "AntiAlias": "ENABLED",
      "Sharpness": 50,
      "CodecSettings": {
        "Codec": "H_264",
        "H264Settings": {
          "InterlaceMode": "PROGRESSIVE",
          "NumberReferenceFrames": 3,
          "Syntax": "DEFAULT",
          "Softness": 0,
          "GopClosedCadence": 1,
          "GopSize": 90,
          "Slices": 1,
          "GopBReference": "DISABLED",
          "SlowPal": "DISABLED",
          "SpatialAdaptiveQuantization": "ENABLED",
          "TemporalAdaptiveQuantization": "ENABLED",
          "FlickerAdaptiveQuantization": "DISABLED",
          "EntropyEncoding": "CABAC",
          "Bitrate": 5000000,
          "FrameRateControl": "INITIALIZE_FROM_SOURCE",
          "RateControlMode": "CBR",
          "CodecProfile": "MAIN",
          "Telecine": "NONE",
          "MinIInterval": 0,
          "AdaptiveQuantization": "HIGH",
        }
      }
    }
  }
]
"CodecLevel": "AUTO",
"FieldEncoding": "PAFF",
"SceneChangeDetect": "ENABLED",
"QualityTuningLevel": "SINGLE_PASS",
"FramerateConversionAlgorithm": "DUPLICATE_DROP",
"UnregisteredSeiTimecode": "DISABLED",
"GopSizeUnits": "FRAMES",
"ParControl": "INITIALIZE_FROM_SOURCE",
"NumberBFramesBetweenReferenceFrames": 2,
"RepeatPps": "DISABLED"
}
},
"Afdsignaling": "NONE",
"DropFrameTimecode": "ENABLED",
"RespondToAfds": "NONE",
"ColorMetadata": "INSERT"
},
"AudioDescriptions": [
{
"AudioTypeControl": "FOLLOW_INPUT",
"CodecSettings": {
"Codec": "AAC",
"AacSettings": {
"AudioDescriptionBroadcasterMix": "NORMAL",
"RateControlMode": "CBR",
"CodecProfile": "LC",
"CodingMode": "CODING_MODE_2_0",
"RawFormat": "NONE",
"SampleRate": 48000,
"Specification": "MPEG4"
}
},
"LanguageCodeControl": "FOLLOW_INPUT"
}
],
"OutputSettings": {
"HlsSettings": {
"AudioGroupId": "program_audio",
"AudioRenditionSets": "program_audio",
"IFrameOnlyManifest": "EXCLUDE"
}
},
"ContainerSettings": {
"Container": "M3U8",
"M3U8Settings": {
"AudioFramesPerPes": 4,
"PcrControl": "PCR_EVERY_PES_PACKET",
"PmtPid": 480,
"PrivateMetadataPid": 503,
"ProgramNumber": 1,
"PmtInterval": 0,
"PmtInterval": 0,
"Scte35Source": "NONE",
"NielsenId3": "NONE",
"TimedMetadata": "NONE",
"VideoPid": 481,
"AudioPids": [482, 483, 484, 485, 486, 487, 488, 489, 490]
491,
492
],
"NameModifier": "_v1"
},
{
  "AudioDescriptions": [
    {
      "AudioTypeControl": "FOLLOW_INPUT",
      "CodecSettings": {
        "Codec": "AAC",
        "AacSettings": {
          "AudioDescriptionBroadcasterMix": "NORMAL",
          "RateControlMode": "CBR",
          "CodecProfile": "LC",
          "CodingMode": "CODING_MODE_2_0",
          "RawFormat": "NONE",
          "SampleRate": 48000,
          "Specification": "MPEG4",
          "Bitrate": 64000
        }
      },
      "LanguageCodeControl": "FOLLOW_INPUT",
      "AudioSourceName": "Audio Selector 1",
      "StreamName": "English",
      "LanguageCode": "ENG"
    }
  ],
  "OutputSettings": {
    "HlsSettings": {
      "AudioGroupId": "program_audio",
      "AudioRenditionSets": "program_audio",
      "IFrameOnlyManifest": "EXCLUDE",
      "AudioTrackType": "ALTERNATE_AUDIO_AUTO_SELECT_DEFAULT"
    }
  },
  "ContainerSettings": {
    "Container": "M3U8",
    "M3u8Settings": {
      "AudioFramesPerPes": 4,
      "PcrControl": "PCR_EVERY_PES_PACKET",
      "PmtPid": 480,
      "PrivateMetadataPid": 503,
      "ProgramNumber": 1,
      "PatInterval": 0,
      "PmtInterval": 0,
      "Scte35Source": "NONE",
      "NielsenId3": "NONE",
      "TimedMetadata": "NONE",
      "VideoPid": 481,
      "AudioPids": [482, 483, 484, 485, 486, 487, 488, 489, 490, 491, 492]}
  }
}
{
  "NameModifier": "_a1"
},
{
  "AudioDescriptions": [
    {
      "AudioTypeControl": "FOLLOW_INPUT",
      "CodecSettings": {
        "Codec": "AAC",
        "AacSettings": {
          "AudioDescriptionBroadcasterMix": "NORMAL",
          "RateControlMode": "CBR",
          "CodecProfile": "LC",
          "CodingMode": "CODING_MODE_2_0",
          "RawFormat": "NONE",
          "SampleRate": 48000,
          "Specification": "MPEG4",
          "Bitrate": 64000
        }
      },
      "LanguageCodeControl": "FOLLOW_INPUT",
      "AudioSourceName": "Audio Selector 2",
      "StreamName": "Spanish",
      "LanguageCode": "SPA"
    }
  ],
  "OutputSettings": {
    "HlsSettings": {
      "AudioGroupId": "program_audio",
      "AudioRenditionSets": "program_audio",
      "IFrameOnlyManifest": "EXCLUDE",
      "AudioTrackType": "ALTERNATE_AUDIO_AUTO_SELECT"
    }
  },
  "ContainerSettings": {
    "Container": "M3U8",
    "M3u8Settings": {
      "AudioFramesPerPes": 4,
      "PcrControl": "PCP_EVERY_PES_PACKET",
      "PmtPid": 480,
      "PrivateMetadataPid": 503,
      "ProgramNumber": 1,
      "PatInterval": 0,
      "PmtInterval": 0,
      "Scte35Source": "NONE",
      "NielsenId3": "NONE",
      "TimedMetadata": "NONE",
      "VideoPid": 481,
      "AudioPids": [482, 483, 484, 485, 486, 487, 488, 489, 490, 491, 492]
    }
  },
  "NameModifier": "_a2"
}
},
"AdAvailOffset": 0,
"Inputs": [
{
"AudioSelectors": {
"Audio Selector 1": {
"Offset": 0,
"DefaultSelection": "DEFAULT",
"ProgramSelection": 1,
"SelectorType": "TRACK",
"Tracks": [1]
},
"Audio Selector 2": {
"Offset": 0,
"DefaultSelection": "NOT_DEFAULT",
"ProgramSelection": 1,
"SelectorType": "TRACK",
"Tracks": [2]
}
},
"VideoSelector": {
"ColorSpace": "FOLLOW"
},
"FilterEnable": "AUTO",
"PsiControl": "USE_PSI",
"FilterStrength": 0,
"DeblockFilter": "DISABLED",
"DenoiseFilter": "DISABLED",
"TimecodeSource": "EMBEDDED",
"FileInput": "s3://testbucket-input/720/test_file.mp4"
}
],
"TimecodeConfig": {
"Source": "ZEROBASED"
}
}
Important Notes

Account-Specific Endpoints

Unlike most AWS services, AWS Elemental MediaConvert requires that you send your requests to an endpoint that is specific to your account. For more information, see Getting Started with MediaConvert Using the API (p. 16) or Getting Started with AWS Elemental MediaConvert Using the AWS SDKs or the AWS CLI (p. 1).

Versioned Endpoints

When you send requests programmatically, you must specify the API version that you send the requests to. The current API version is 2017-08-29.

Therefore, you must construct your endpoint like this:

```
https://<account-specific-digits>.mediaconvert.<region>.amazonaws.com/2017-08-29/<resource>/<parameter>
```

For example:

```
https://abcd1234.mediaconvert.region-name-1.amazonaws.com/2017-08-29/queues/my-queue
```

Using the AWS CLI

When you send requests using the AWS CLI, use PascalCase for all properties. For example, if you used the properties `settings` and `timecodeConfig` in your API call, you must change those to `Settings` and `TimecodeConfig` for your CLI call. This is required because the CLI is built on Python, which uses PascalCase for properties.

Schemas and Example Job Settings

The schemas provided in this guide are not working examples. Instead, they provide information about how to structure your job settings specification. For example job settings in JSON format, see example job settings in the AWS Elemental MediaConvert User Guide. You pass in job settings as the payload when you create jobs, custom output presets, and custom job templates.

If you access AWS Elemental MediaConvert directly through the API, using the AWS CLI, or using the AWS SDK for Python (Boto), you submit your job settings as a JSON file. The simplest way to generate this file is to set up your job using the MediaConvert console and then, on the Create job page, choose Show job JSON.

If you access AWS Elemental MediaConvert through one of the AWS SDKs other than Python, consult the documentation for those SDKs for information about the syntax to use to for your job settings specification. For more information about using the AWS SDKs to access MediaConvert, see Getting Started with AWS Elemental MediaConvert Using the AWS SDKs or the AWS CLI (p. 1).
Resources

The AWS Elemental MediaConvert REST API includes the following resources.

Topics
- Certificates (p. 33)
- Certificates arn (p. 35)
- Endpoints (p. 38)
- JobTemplates (p. 41)
- JobTemplates name (p. 238)
- Jobs (p. 435)
- Jobs id (p. 638)
- Presets (p. 815)
- Presets name (p. 952)
- Queues (p. 1089)
- Queues name (p. 1100)
- Tags (p. 1111)
- Tags arn (p. 1115)

Certificates

URI

/2017-08-29/certificates

HTTP Methods

POST

Operation ID: AssociateCertificate

Associates an AWS Certificate Manager (ACM) Amazon Resource Name (ARN) with AWS Elemental MediaConvert.

Responses

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>201</td>
<td>AssociateCertificateResponse</td>
<td>201 response</td>
</tr>
<tr>
<td>400</td>
<td>ExceptionBody (p. 34)</td>
<td>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>403</td>
<td>ExceptionBody (p. 34)</td>
<td>You don't have permissions for this action with the credentials you sent.</td>
</tr>
</tbody>
</table>
### Status Code

<table>
<thead>
<tr>
<th>Code</th>
<th>Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>404</td>
<td>ExceptionBody</td>
<td>The resource you requested does not exist.</td>
</tr>
<tr>
<td>409</td>
<td>ExceptionBody</td>
<td>The service could not complete your request because there is a conflict with the current state of the resource.</td>
</tr>
<tr>
<td>429</td>
<td>ExceptionBody</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>
<tr>
<td>500</td>
<td>ExceptionBody</td>
<td>The service encountered an unexpected condition and cannot fulfill your request.</td>
</tr>
</tbody>
</table>

### Schemas

#### Request Bodies

**Example POST**

```json
{
    "arn": "string"
}
```

#### Response Bodies

**Example AssociateCertificateResponse**

```json
{}
```

**Example ExceptionBody**

```json
{
    "message": "string"
}
```

### Properties

**AssociateCertificateRequest**

Associates the Amazon Resource Name (ARN) of an AWS Certificate Manager (ACM) certificate with an AWS Elemental MediaConvert resource.

**arn**

The ARN of the ACM certificate that you want to associate with your MediaConvert resource.
Type: string  
Required: True

**AssociateCertificateResponse**

Successful association of Certificate Manager Amazon Resource Name (ARN) with Mediaconvert returns an OK message.

**ExceptionBody**

message  
Type: string  
Required: False

**See Also**

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

**AssociateCertificate**

- AWS Command Line Interface  
- AWS SDK for .NET  
- AWS SDK for C++  
- AWS SDK for Go  
- AWS SDK for Go - Pilot  
- AWS SDK for Java  
- AWS SDK for JavaScript  
- AWS SDK for PHP V3  
- AWS SDK for Python  
- AWS SDK for Ruby V2

**Certificates arn**

**URI**

/2017-08-29/certificates/arn

**HTTP Methods**

**DELETE**

Operation ID: DisassociateCertificate

Removes an association between the Amazon Resource Name (ARN) of an AWS Certificate Manager (ACM) certificate and an AWS Elemental MediaConvert resource.
Path Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>arn</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>202</td>
<td>DisassociateCertificateResponse</td>
<td>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>400</td>
<td>ExceptionBody (p. 37)</td>
<td>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>403</td>
<td>ExceptionBody (p. 37)</td>
<td>You don't have permissions for this action with the credentials you sent.</td>
</tr>
<tr>
<td>404</td>
<td>ExceptionBody (p. 37)</td>
<td>The resource you requested does not exist.</td>
</tr>
<tr>
<td>409</td>
<td>ExceptionBody (p. 37)</td>
<td>The service could not complete your request because there is a conflict with the current state of the resource.</td>
</tr>
<tr>
<td>429</td>
<td>ExceptionBody (p. 37)</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>
<tr>
<td>500</td>
<td>ExceptionBody (p. 37)</td>
<td>The service encountered an unexpected condition and cannot fulfill your request.</td>
</tr>
</tbody>
</table>

Schemas

Request Bodies

Example DELETE

```json
{
  "arn": "string"
}
```

Response Bodies

Example DisassociateCertificateResponse

```json
{}
```
**Properties**

**DisassociateCertificateRequest**

Removes an association between the Amazon Resource Name (ARN) of an AWS Certificate Manager (ACM) certificate and an AWS Elemental MediaConvert resource.

**arn**

The ARN of the ACM certificate that you want to disassociate from your MediaConvert resource.

*Type:* string  
*Required:* False

**DisassociateCertificateResponse**

Successful disassociation of Certificate Manager Amazon Resource Name (ARN) with Mediaconvert returns an OK message.

**ExceptionBody**

**message**

*Type:* string  
*Required:* False

**See Also**

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

**DisassociateCertificate**

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Go - Pilot
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
Endpoints

URI

/2017-08-29/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 response</td>
</tr>
<tr>
<td>400</td>
<td>ExceptionBody (p. 39)</td>
<td>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>403</td>
<td>ExceptionBody (p. 39)</td>
<td>You don't have permissions for this action with the credentials you sent.</td>
</tr>
<tr>
<td>404</td>
<td>ExceptionBody (p. 39)</td>
<td>The resource you requested does not exist.</td>
</tr>
<tr>
<td>409</td>
<td>ExceptionBody (p. 39)</td>
<td>The service could not complete your request because there is a conflict with the current state of the resource.</td>
</tr>
<tr>
<td>429</td>
<td>ExceptionBody (p. 39)</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>
<tr>
<td>500</td>
<td>ExceptionBody (p. 39)</td>
<td>The service encountered an unexpected condition and cannot fulfill your request.</td>
</tr>
</tbody>
</table>

Schemas

Request Bodies

Example POST

```{json}
{
```
"nextToken": "string",
"maxResults": integer,
"mode": enum
}

Response Bodies

Example DescribeEndpointsResponse

{
   "endpoints": [
      { "url": "string"
   },
   "nextToken": "string"
}

Example ExceptionBody

{
   "message": "string"
}

Properties

DescribeEndpointsMode

Optional field, defaults to DEFAULT. Specify DEFAULT for this operation to return your endpoints if any exist, or to create an endpoint for you and return it if one doesn’t already exist. Specify GET_ONLY to return your endpoints if any exist, or an empty list if none exist.

   DEFAULT
   GET_ONLY

DescribeEndpointsRequest

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

nextToken

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

   Type: string
   Required: False

maxResults

Optional. Max number of endpoints, up to twenty, that will be returned at one time.

   Type: integer
   Required: False
   Format: int32
mode
Optional field, defaults to DEFAULT. Specify DEFAULT for this operation to return your endpoints if any exist, or to create an endpoint for you and return it if one doesn't already exist. Specify GET_ONLY to return your endpoints if any exist, or an empty list if none exist.

  Type: DescribeEndpointsMode (p. 39)
  Required: False

DescribeEndpointsResponse
Successful describe endpoints requests will return your account API endpoint.

endpoints
List of endpoints

  Type: Array of type Endpoint (p. 40)
  Required: False

nextToken
Use this string to request the next batch of endpoints.

  Type: string
  Required: False

Endpoint
Describes an account-specific API endpoint.

url
URL of endpoint

  Type: string
  Required: False

ExceptionBody

message

  Type: string
  Required: False

See Also
For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

DescribeEndpoints
• AWS Command Line Interface
JobTemplates

URI

/2017-08-29/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>200 response</td>
</tr>
<tr>
<td>400</td>
<td>ExceptionBody (p. 79)</td>
<td>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>403</td>
<td>ExceptionBody (p. 79)</td>
<td>You don't have permissions for this action with the credentials you sent.</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>404</td>
<td>ExceptionBody (p. 79)</td>
<td>The resource you requested does not exist.</td>
</tr>
<tr>
<td>409</td>
<td>ExceptionBody (p. 79)</td>
<td>The service could not complete your request because there is a conflict with the current state of the resource.</td>
</tr>
<tr>
<td>429</td>
<td>ExceptionBody (p. 79)</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>
<tr>
<td>500</td>
<td>ExceptionBody (p. 79)</td>
<td>The service encountered an unexpected condition and cannot fulfill your request.</td>
</tr>
</tbody>
</table>

### 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>
<tbody>
<tr>
<td>201</td>
<td>CreateJobTemplateResponse</td>
<td>201 response</td>
</tr>
<tr>
<td>400</td>
<td>ExceptionBody (p. 79)</td>
<td>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>403</td>
<td>ExceptionBody (p. 79)</td>
<td>You don’t have permissions for this action with the credentials you sent.</td>
</tr>
<tr>
<td>404</td>
<td>ExceptionBody (p. 79)</td>
<td>The resource you requested does not exist.</td>
</tr>
<tr>
<td>409</td>
<td>ExceptionBody (p. 79)</td>
<td>The service could not complete your request because there is a conflict with the current state of the resource.</td>
</tr>
<tr>
<td>429</td>
<td>ExceptionBody (p. 79)</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>
<tr>
<td>500</td>
<td>ExceptionBody (p. 79)</td>
<td>The service encountered an unexpected condition and cannot fulfill your request.</td>
</tr>
</tbody>
</table>
Schemas

Request Bodies

Example GET

```json
{
    "listBy": "enum",
    "category": "string",
    "order": "enum",
    "nextToken": "string",
    "maxResults": "integer"
}
```

Example POST

```json
{
    "description": "string",
    "category": "string",
    "queue": "string",
    "name": "string",
    "settings": {
        "timecodeConfig": {
            "anchor": "string",
            "source": "enum",
            "start": "string",
            "timestampOffset": "string"
        },
        "outputGroups": [
            {
                "customName": "string",
                "name": "string",
                "outputs": [
                    {
                        "containerSettings": {
                            "container": "enum",
                            "m3u8Settings": {
                                "audioFramesPerPes": "integer",
                                "pcrControl": "enum",
                                "pcrPid": "integer",
                                "pmtpId": "integer",
                                "privateMetadataPid": "integer",
                                "programNumber": "integer",
                                "patInterval": "integer",
                                "pmtInterval": "integer",
                                "scte35Source": "enum",
                                "scte35Pid": "integer",
                                "nielsenId3": "enum",
                                "timedMetadata": "enum",
                                "timedMetadataPid": "integer",
                                "transportStreamId": "integer",
                                "videoId": "integer",
                                "audioPids": ["integer"]
                            },
                            "f4vSettings": {
                                "moovPlacement": "enum"
                            },
                            "m2tsSettings": {
                                "audioBufferSize": "integer",
                                "minEbpInterval": "integer",
                            }
                        }
                    }
                ]
            }
        ]
    }
}
```
"esRateInPes": enum,
"patInterval": integer,
"dvbNitSettings": {
  "nitInterval": integer,
  "networkId": integer,
  "networkName": "string"
},
"dvbSdtSettings": {
  "outputSdt": enum,
  "sdtInterval": integer,
  "serviceName": "string",
  "serviceProviderName": "string"
},
"scte35Source": enum,
"scte35Pid": integer,
"scte35Esam": {
  "scte35EsamPid": integer
},
"videoPid": integer,
"dvbTdtSettings": {
  "tdtInterval": integer
},
"pmtInterval": integer,
"segmentationStyle": enum,
"segmentationTime": number,
"pmtPid": integer,
"bitrate": integer,
"audioPids": [ integer
],
"privateMetadataPid": integer,
"nielsenId3": enum,
"timedMetadataPid": integer,
"maxPcrInterval": integer,
"transportStreamId": integer,
"dvbSubPids": [ integer
],
"rateMode": enum,
"audioFramesPerPes": integer,
"pcrControl": enum,
"segmentationMarkers": enum,
"ebpAudioInterval": enum,
"forceTsVideoEbpOrder": enum,
"programNumber": integer,
"pcrPid": integer,
"bufferModel": enum,
"dvbTeletextPid": integer,
"fragmentTime": number,
"ebpPlacement": enum,
"nullPacketBitrate": number
},
"movSettings": {
  "clapAtom": enum,
  "cs1gAtom": enum,
  "paddingControl": enum,
  "reference": enum,
  "mpeg2FourCCControl": enum
},
"mp4Settings": {
  "cs1gAtom": enum,
  "freeSpaceBox": enum,
  "mp4MajorBrand": "string",
  "moovPlacement": enum
}
"preset": "string",
"videoDescription": {
  "fixedAfd": integer,
  "width": integer,
  "scalingBehavior": enum,
  "crop": {
    "height": integer,
    "width": integer,
    "x": integer,
    "y": integer
  },
  "height": integer,
  "videoPreprocessors": {
    "colorCorrector": {
      "brightness": integer,
      "colorSpaceConversion": enum,
      "contrast": integer,
      "hue": integer,
      "saturation": integer,
      "hdr10Metadata": {
        "redPrimaryX": integer,
        "redPrimaryY": integer,
        "greenPrimaryX": integer,
        "greenPrimaryY": integer,
        "bluePrimaryX": integer,
        "bluePrimaryY": integer,
        "whitePointX": integer,
        "whitePointY": integer,
        "maxFrameAverageLightLevel": integer,
        "maxContentLightLevel": integer,
        "maxLuminance": integer,
        "minLuminance": integer
      }
    },
    "deinterlacer": {
      "algorithm": enum,
      "mode": enum,
      "control": enum
    },
    "imageInserter": {
      "insertableImages": [
        {
          "width": integer,
          "height": integer,
          "imageX": integer,
          "imageY": integer,
          "duration": integer,
          "fadeIn": integer,
          "layer": integer,
          "imageInserterInput": "string",
          "startTime": "string",
          "fadeOut": integer,
          "opacity": integer
        }
      ]
    },
    "noiseReducer": {
      "filter": enum,
      "filterSettings": {
        "strength": integer
      },
      "spatialFilterSettings": {
        "strength": integer,
        "speed": integer,
        "postFilterSharpenStrength": integer
      }
    }
  }
}
"timecodeBurnin": {
  "fontSize": integer,
  "position": enum,
  "prefix": "string"
}
"timecodeInsertion": enum,
"antiAlias": enum,
"position": {
  "height": integer,
  "width": integer,
  "x": integer,
  "y": integer
}
"sharpness": integer,
"codecSettings": {
  "codec": enum,
  "frameCaptureSettings": {
    "framerateNumerator": integer,
    "framerateDenominator": integer,
    "maxCaptures": integer,
    "quality": integer
  },
  "h264Settings": {
    "interlaceMode": enum,
    "parNumerator": integer,
    "numberReferenceFrames": integer,
    "syntax": enum,
    "softness": integer,
    "framerateDenominator": integer,
    "gopClosedCadence": integer,
    "hrdBufferInitialFillPercentage": integer,
    "gopSize": number,
    "slices": integer,
    "gopReference": enum,
    "hrdBufferSize": integer,
    "maxBitrate": integer,
    "slowPal": enum,
    "parDenominator": integer,
    "spatialAdaptiveQuantization": enum,
    "temporalAdaptiveQuantization": enum,
    "flickerAdaptiveQuantization": enum,
    "entropyEncoding": enum,
    "bitrate": integer,
    "framerateControl": enum,
    "rateControlMode": enum,
    "qvbrSettings": {
      "qvbrQualityLevel": integer,
      "maxAverageBitrate": integer
    },
    "codecProfile": enum,
    "telecine": enum,
    "framerateNumerator": integer,
    "minIInterval": integer,
    "adaptiveQuantization": enum,
    "codecLevel": enum,
    "fieldEncoding": enum,
    "sceneChangeDetect": enum,
    "qualityTuningLevel": enum,
    "framerateConversionAlgorithm": enum,
    "unregisteredSeiTimecode": enum,
    "gopSizeUnits": enum,
    "parControl": enum,
    "numberBFramesBetweenReferenceFrames": integer,
    "repeatFps": enum,
"dynamicSubGop": enum,
"telecine": enum,
"framerateNumerator": integer,
"minIInterval": integer,
"adaptiveQuantization": enum,
"codecLevel": enum,
"sceneChangeDetect": enum,
"qualityTuningLevel": enum,
"framerateConversionAlgorithm": enum,
"gopSizeUnits": enum,
"parControl": enum,
"numberBFramesBetweenReferenceFrames": integer,
"dynamicSubGop": enum
},
"proresSettings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "framerateDenominator": integer,
  "codecProfile": enum,
  "slowPal": enum,
  "parDenominator": integer,
  "framerateControl": enum,
  "telecine": enum,
  "framerateNumerator": integer,
  "framerateConversionAlgorithm": enum,
  "parControl": enum
}
},
"afdSignaling": enum,
"dropFrameTimecode": enum,
"respondToAfd": enum,
"colorMetadata": enum
},
"audioDescriptions": [
{
  "audioTypeControl": enum,
  "audioSourceName": "string",
  "audioNormalizationSettings": {
    "algorithm": enum,
    "algorithmControl": enum,
    "correctionGateLevel": integer,
    "loudnessLogging": enum,
    "targetLkfs": number,
    "peakCalculation": enum
  }
},
"codecSettings": {
  "codec": enum,
  "aacSettings": {
    "audioDescriptionBroadcasterMix": enum,
    "vbrQuality": enum,
    "bitrate": integer,
    "rateControlMode": enum,
    "codecProfile": enum,
    "codingMode": enum,
    "rawFormat": enum,
    "sampleRate": integer,
    "specification": enum
  }
},
"ac3Settings": {
  "bitrate": integer,
  "bitstreamMode": enum,
  "codingMode": enum,
  "dialnorm": integer,
  "dynamicRangeCompressionProfile": enum,
  "metadataControl": enum,
  "lfeFilter": enum,
  "sampleRate": integer
}
"aiffSettings": {  "bitDepth": integer,  "channels": integer,  "sampleRate": integer  },  "eac3Settings": {  "metadataControl": enum,  "surroundExMode": enum,  "loRoSurroundMixLevel": number,  "phaseControl": enum,  "dialnorm": integer,  "ltRtSurroundMixLevel": number,  "bitrate": integer,  "ltRtCenterMixLevel": number,  "passthroughControl": enum,  "lfeControl": enum,  "loRoCenterMixLevel": number,  "attenuationControl": enum,  "codingMode": enum,  "surroundMode": enum,  "bitstreamMode": enum,  "lfeFilter": enum,  "stereoDownmix": enum,  "dynamicRangeCompressionRf": enum,  "sampleRate": integer,  "dynamicRangeCompressionLine": enum,  "dcFilter": enum  },  "mp2Settings": {  "bitrate": integer,  "channels": integer,  "sampleRate": integer  },  "wavSettings": {  "bitDepth": integer,  "channels": integer,  "sampleRate": integer,  "format": enum  },  "remixSettings": {  "channelMapping": {  "outputChannels": [  {  "inputChannels": [  integer  ]  }  ]  },  "channelsIn": integer,  "channelsOut": integer  },  "streamName": "string",  "languageCodeControl": enum,  "audioType": integer,  "customLanguageCode": "string",  "languageCode": enum  },  "outputSettings": {  "hlsSettings": {  "audioGroupId": "string",  "audioRenditionSets": "string",  "audioTrackType": enum,  "hlsSettings": {  "audioGroupID": "string",  "audioRenditionSets": "string",  "audioTrackType": enum,  "languageCode": enum  }}
"iFrameOnlyManifest": enum,
"segmentModifier": "string"
}
"extension": "string",
"nameModifier": "string",
"captionDescriptions": [
{
"captionSelectorName": "string",
"destinationSettings": {
"destinationType": enum,
"burninDestinationSettings": {
"backgroundOpacity": integer,
"shadowXOffset": integer,
"teletextSpacing": enum,
"alignment": enum,
"outlineSize": integer,
"yPosition": integer,
"shadowColor": enum,
"fontOpacity": integer,
"fontSize": integer,
"fontScript": enum,
"fontColor": enum,
"backgroundColor": enum,
"fontResolution": integer,
"outlineColor": enum,
"shadowYOffset": integer,
"xPosition": integer,
"shadowOpacity": integer
},
"dvbSubDestinationSettings": {
"backgroundOpacity": integer,
"shadowXOffset": integer,
"teletextSpacing": enum,
"alignment": enum,
"outlineSize": integer,
"yPosition": integer,
"shadowColor": enum,
"fontOpacity": integer,
"fontSize": integer,
"fontScript": enum,
"fontColor": enum,
"backgroundColor": enum,
"fontResolution": integer,
"outlineColor": enum,
"shadowYOffset": integer,
"xPosition": integer,
"shadowOpacity": integer
},
"sccDestinationSettings": {
"framerate": enum
},
"teletextDestinationSettings": {
"pageNumber": "string"
},
"ttmlDestinationSettings": {
"stylePassthrough": enum
},
"embeddedDestinationSettings": {
"destination608ChannelNumber": integer
}
"customLanguageCode": "string",
"languageCode": enum,
"languageDescription": "string"}
"outputGroupSettings": {
  "type": enum,
  "hlsGroupSettings": {
    "manifestDurationFormat": enum,
    "segmentLength": integer,
    "timedMetadataId3Period": integer,
    "captionLanguageSetting": enum,
    "captionLanguageMappings": [
      {
        "captionChannel": integer,
        "customLanguageCode": "string",
        "languageCode": enum,
        "languageDescription": "string"
      }
    ],
    "destination": "string",
    "destinationSettings": {
      "s3Settings": {
        "encryption": {
          "encryptionType": enum,
          "kmsKeyArn": "string"
        }
      }
    },
    "encryption": {
      "encryptionMethod": enum,
      "constantInitializationVector": "string",
      "initializationVectorInManifest": enum,
      "offlineEncrypted": enum,
      "spekeKeyProvider": {
        "resourceId": "string",
        "systemIds": [
          "string"
        ],
        "url": "string",
        "certificateArn": "string"
      },
      "staticKeyProvider": {
        "staticKeyValue": "string",
        "keyFormat": "string",
        "keyFormatVersions": "string",
        "url": "string"
      },
      "type": enum
    },
    "timedMetadataId3Frame": enum,
    "baseUrl": "string",
    "codecSpecification": enum,
    "outputSelection": enum,
    "programDateTimePeriod": integer,
    "segmentsPerSubdirectory": integer,
    "minSegmentLength": integer,
    "minFinalSegmentLength": number,
    "directoryStructure": enum,
    "programDateTime": enum,
    "adMarkers": [
      enum
    ],
    "segmentControl": enum,
    "timestampDeltaMilliseconds": integer,
    "manifestCompression": enum,
    "clientCache": enum,
    "streamInfResolution": enum
  }
},
"encryption": {
  "encryptionMethod": enum,
  "constantInitializationVector": "string",
  "initializationVectorInManifest": enum,
  "offlineEncrypted": enum,
  "spekeKeyProvider": {
    "resourceId": "string",
    "systemIds": [
      "string"
    ],
    "url": "string",
    "certificateArn": "string"
  },
  "staticKeyProvider": {
    "staticKeyValue": "string",
    "keyFormat": "string",
    "keyFormatVersions": "string",
    "url": "string"
  },
  "type": enum
},
"timedMetadataId3Frame": enum,
"baseUrl": "string",
"codecSpecification": enum,
"outputSelection": enum,
"programDateTimePeriod": integer,
"segmentsPerSubdirectory": integer,
"minSegmentLength": integer,
"minFinalSegmentLength": number,
"directoryStructure": enum,
"programDateTime": enum,
"adMarkers": [
  enum
],
"segmentControl": enum,
"timestampDeltaMilliseconds": integer,
"manifestCompression": enum,
"clientCache": enum,
"streamInfResolution": enum
"dashIsoGroupSettings": {
  "segmentLength": integer,
  "destination": string,
  "destinationSettings": {
    "s3Settings": {
      "encryption": {
        "encryptionType": enum,
        "kmsKeyArn": string
      }
    }
  }
},
"encryption": {
  "spekeKeyProvider": {
    "resourceId": string,
    "systemIds": [string]
  },
  "url": string,
  "certificateArn": string
},
"minBufferTime": integer,
"fragmentLength": integer,
"baseUrl": string,
"segmentControl": enum,
"hbbtvCompliance": enum,
"writeSegmentTimelineInRepresentation": enum
},
"fileGroupSettings": {
  "destination": string,
  "destinationSettings": {
    "s3Settings": {
      "encryption": {
        "encryptionType": enum,
        "kmsKeyArn": string
      }
    }
  }
},
"msSmoothGroupSettings": {
  "destination": string,
  "destinationSettings": {
    "s3Settings": {
      "encryption": {
        "encryptionType": enum,
        "kmsKeyArn": string
      }
    }
  }
},
"fragmentLength": integer,
"encryption": {
  "spekeKeyProvider": {
    "resourceId": string,
    "systemIds": [string]
  },
  "url": string,
  "certificateArn": string
},
"manifestEncoding": enum,
"audioDeduplication": enum
},
"cmafGroupSettings": {
  "writeHlsManifest": enum,
"writeDashManifest": enum,
"segmentLength": integer,
"minFinalSegmentLength": number,
"destination": "string",
"destinationSettings": {
  "s3Settings": {
    "encryption": {
      "encryptionType": enum,
      "kmsKeyArn": "string"
    }
  },
  "encryption": {
    "encryptionMethod": enum,
    "constantInitializationVector": "string",
    "initializationVectorInManifest": enum,
    "staticKeyProvider": {
      "staticKeyValue": "string",
      "keyFormat": "string",
      "keyFormatVersions": "string",
      "url": "string"
    },
    "type": enum
  },
  "minBufferTime": integer,
  "fragmentLength": integer,
  "baseUrl": "string",
  "segmentControl": enum,
  "manifestDurationFormat": enum,
  "streamInfResolution": enum,
  "clientCache": enum,
  "manifestCompression": enum,
  "codecSpecification": enum
}
},
"adAvailOffset": integer,
"availBlanking": {
  "availBlankingImage": "string"
},
"timedMetadataInsertion": {
  "id3Insertions": [
    {
      "timecode": "string",
      "id3": "string"
    }
  ]
},
"nielsenConfiguration": {
  "breakoutCode": integer,
  "distributorId": "string"
},
"motionImageInserter": {
  "insertionMode": enum,
  "input": "string",
  "offset": {
    "imageX": integer,
    "imageY": integer
  },
  "startTime": "string",
  "playback": enum,
  "framerate": {
    "framerateNumerator": integer,
    "framerateDenominator": integer
  }
}
AWS Elemental MediaConvert API Reference API Reference
Schemas
},
"esam": {
"signalProcessingNotification": {
"sccXml": "string"
},
"manifestConfirmConditionNotification": {
"mccXml": "string"
},
"responseSignalPreroll": integer
},
"inputs": [
{
"inputClippings": [
{
"endTimecode": "string",
"startTimecode": "string"
}
],
"audioSelectors": {
},
"audioSelectorGroups": {
},
"programNumber": integer,
"videoSelector": {
"colorSpace": enum,
"rotate": enum,
"pid": integer,
"programNumber": integer,
"colorSpaceUsage": enum,
"hdr10Metadata": {
"redPrimaryX": integer,
"redPrimaryY": integer,
"greenPrimaryX": integer,
"greenPrimaryY": integer,
"bluePrimaryX": integer,
"bluePrimaryY": integer,
"whitePointX": integer,
"whitePointY": integer,
"maxFrameAverageLightLevel": integer,
"maxContentLightLevel": integer,
"maxLuminance": integer,
"minLuminance": integer
}
},
"filterEnable": enum,
"psiControl": enum,
"filterStrength": integer,
"deblockFilter": enum,
"denoiseFilter": enum,
"timecodeSource": enum,
"captionSelectors": {
},
"imageInserter": {
"insertableImages": [
{
"width": integer,
"height": integer,
"imageX": integer,
"imageY": integer,
"duration": integer,
"fadeIn": integer,
"layer": integer,
"imageInserterInput": "string",
"startTime": "string",
"fadeOut": integer,
"opacity": integer

54


Response Bodies

Example ListJobTemplatesResponse

```json
{
  "jobTemplates": [
    {
      "arn": "string",
      "createdAt": "string",
      "lastUpdated": "string",
      "description": "string",
      "category": "string",
      "queue": "string",
      "name": "string",
      "type": enum,
      "settings": {
        "timecodeConfig": {
          "anchor": "string",
          "source": enum,
          "start": "string",
          "timestampOffset": "string"
        },
        "outputGroups": [
          {
            "customName": "string",
            "name": "string",
            "outputs": [
              {
                "containerSettings": {
                  "container": enum,
                  "m3u8Settings": {
                    "audioFramesPerPES": integer,
                    "pcrControl": enum,
                    "pcrPid": integer,
                    "pmtPid": integer,
                    "privateMetadataPid": integer,
                    "programNumber": integer,
                    "patInterval": integer,
                    "pmtInterval": integer,
                    "scte35Source": enum,
                    "scte35Pid": integer,
                    "nielsenId3": enum,
                    "timedMetadata": enum,
                    "timedMetadataPID": integer,
                    "transportStreamId": integer,
                    "videoPid": integer,
                    "audioPids": [ integer ]
                }
              }
            ]
          }
        ]
    }
  ]
}
```
"f4vSettings": {
  "moovPlacement": enum
},
"m2tsSettings": {
  "audioBufferModel": enum,
  "minEbpInterval": integer,
  "esRateInPes": enum,
  "patInterval": integer,
  "dvbNitSettings": {
    "nitInterval": integer,
    "networkId": integer,
    "networkName": "string"
  },
  "dvbSdtSettings": {
    "outputSdt": enum,
    "sdtInterval": integer,
    "serviceName": "string",
    "serviceProviderName": "string"
  },
  "scte35Source": enum,
  "scte35Pid": integer,
  "scte35Esam": {
    "scte35EsamPid": integer
  },
  "videoPid": integer,
  "dvbTdtSettings": {
    "tdtInterval": integer
  },
  "pmtInterval": integer,
  "segmentationStyle": enum,
  "segmentationTime": number,
  "pmtPid": integer,
  "bitrate": integer,
  "audioPids": [
    integer
  ],
  "privateMetadataPid": integer,
  "nielsenId3": enum,
  "timedMetadataPid": integer,
  "maxPcrInterval": integer,
  "transportStreamId": integer,
  "dvbSubPids": [
    integer
  ],
  "rateMode": enum,
  "audioFramesPerPes": integer,
  "pcrControl": enum,
  "segmentationMarkers": enum,
  "ebpAudioInterval": enum,
  "forceTsVideoEbpOrder": enum,
  "programNumber": integer,
  "pcrPid": integer,
  "bufferModel": enum,
  "dvbTeletextPid": integer,
  "fragmentTime": number,
  "ebpPlacement": enum,
  "nullPacketBitrate": number
},
"movSettings": {
  "clapAtom": enum,
  "cslgAtom": enum,
  "paddingControl": enum,
  "reference": enum,
  "mpeg2FourCCControl": enum
},
"mp4Settings": {

"cslgAtom": enum,
"freeSpaceBox": enum,
"mp4MajorBrand": "string",
"moovPlacement": enum
}
},
"preset": "string",
"videoDescription": {
"fixedAfD": integer,
"width": integer,
"scalingBehavior": enum,
"crop": {
"height": integer,
"width": integer,
"x": integer,
"y": integer
},
"height": integer,
"videoPreprocessors": {
"colorCorrector": {
"brightness": integer,
"colorSpaceConversion": enum,
"contrast": integer,
"hue": integer,
"saturation": integer,
"hdr10Metadata": {
"redPrimaryX": integer,
"redPrimaryY": integer,
"greenPrimaryX": integer,
"greenPrimaryY": integer,
"bluePrimaryX": integer,
"bluePrimaryY": integer,
"whitePointX": integer,
"whitePointY": integer,
"maxFrameAverageLightLevel": integer,
"maxContentLightLevel": integer,
"maxLuminance": integer,
"minLuminance": integer
}
},
"deinterlacer": {
"algorithm": enum,
"mode": enum,
"control": enum
},
"imageInserter": {
"insertableImages": [
{
"width": integer,
"height": integer,
"imageX": integer,
"imageY": integer,
"duration": integer,
"fadeIn": integer,
"layer": integer,
"imageInserterInput": "string",
"startTime": "string",
"fadeOut": integer,
"opacity": integer
}
]
},
"noiseReducer": {
"filter": enum,
"filterSettings": {
"strength": integer
}
},
  "spatialFilterSettings": {
    "strength": integer,
    "speed": integer,
    "postFilterSharpenStrength": integer
  },
},
  "timecodeBurnin": {
    "fontSize": integer,
    "position": enum,
    "prefix": "string"
  },
},
  "timecodeInsertion": enum,
  "antiAlias": enum,
  "position": {
    "height": integer,
    "width": integer,
    "x": integer,
    "y": integer
  },
  "sharpness": integer,
  "codecSettings": {
    "codec": enum,
    "frameCaptureSettings": {
      "framerateNumerator": integer,
      "framerateDenominator": integer,
      "maxCaptures": integer,
      "quality": integer
    },
    "h264Settings": {
      "interlaceMode": enum,
      "parNumerator": integer,
      "numberReferenceFrames": integer,
      "syntax": enum,
      "softness": integer,
      "framerateDenominator": integer,
      "gopClosedCadence": integer,
      "hrdBufferInitialFillPercentage": integer,
      "gopSize": number,
      "slices": integer,
      "gopBReference": enum,
      "hrdBufferSize": integer,
      "maxBitrate": integer,
      "slowPal": enum,
      "parDenominator": integer,
      "spatialAdaptiveQuantization": enum,
      "temporalAdaptiveQuantization": enum,
      "flickerAdaptiveQuantization": enum,
      "entropyEncoding": enum,
      "bitrate": integer,
      "framerateControl": enum,
      "rateControlMode": enum,
      "gbrSettings": {
        "gbrQualityLevel": integer,
        "maxAverageBitrate": integer
      },
      "codecProfile": enum,
      "telecine": enum,
      "framerateNumerator": integer,
      "minIInterval": integer,
      "adaptiveQuantization": enum,
      "codecLevel": enum,
      "fieldEncoding": enum,
      "sceneChangeDetect": enum,
      "qualityTuningLevel": enum,
"framerateConversionAlgorithm": enum,
"unregisteredSeiTimecode": enum,
"gopSizeUnits": enum,
"parControl": enum,
"numberBFramesBetweenReferenceFrames": integer,
"repeatPps": enum,
"dynamicSubGop": enum
},
<h265Settings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "numberReferenceFrames": integer,
  "framerateDenominator": integer,
  "gopClosedCadence": integer,
  "alternateTransferFunctionSei": enum,
  "hrdBufferInitialFillPercentage": integer,
  "gopSize": number,
  "slices": integer,
  "gopReference": enum,
  "hrdBufferSize": integer,
  "maxBitrate": integer,
  "slowPal": enum,
  "parDenominator": integer,
  "spatialAdaptiveQuantization": enum,
  "temporalAdaptiveQuantization": enum,
  "flickerAdaptiveQuantization": enum,
  "bitrate": integer,
  "framerateControl": enum,
  "rateControlMode": enum,
  "qvbrSettings": {
    "qvbrQualityLevel": integer,
    "maxAverageBitrate": integer
  },
  "codecProfile": enum,
  "tiles": enum,
  "telecine": enum,
  "framerateNumerator": integer,
  "minIInterval": integer,
  "adaptiveQuantization": enum,
  "codecLevel": enum,
  "sceneChangeDetect": enum,
  "qualityTuningLevel": enum,
  "framerateConversionAlgorithm": enum,
  "unregisteredSeiTimecode": enum,
  "gopSizeUnits": enum,
  "parControl": enum,
  "numberBFramesBetweenReferenceFrames": integer,
  "temporalIds": enum,
  "sampleAdaptiveOffsetFilterMode": enum,
  "writeMp4PackagingType": enum,
  "dynamicSubGop": enum
},
"mpeg2Settings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "syntax": enum,
  "softness": integer,
  "framerateDenominator": integer,
  "gopClosedCadence": integer,
  "hrdBufferInitialFillPercentage": integer,
  "gopSize": number,
  "hrdBufferSize": integer,
  "maxBitrate": integer,
  "slowPal": enum,
  "parDenominator": integer,
  "spatialAdaptiveQuantization": enum,
"temporalAdaptiveQuantization": enum,
"bitrate": integer,
"intraDcPrecision": enum,
"framerateControl": enum,
"rateControlMode": enum,
"codecProfile": enum,
"telecine": enum,
"framerateNumerator": integer,
"minIInterval": integer,
"adaptiveQuantization": enum,
"codecLevel": enum,
"sceneChangeDetect": enum,
"qualityTuningLevel": enum,
"framerateConversionAlgorithm": enum,
"gopSizeUnits": enum,
"parControl": enum,
"numberBFramesBetweenReferenceFrames": integer,
"dynamicSubGop": enum
}

"proresSettings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "framerateDenominator": integer,
  "codecProfile": enum,
  "slowPal": enum,
  "parDenominator": integer,
  "framerateControl": enum,
  "telecine": enum,
  "framerateNumerator": integer,
  "framerateConversionAlgorithm": enum,
  "parControl": enum
}

"afdsignaling": enum,
"dropFrameTimecode": enum,
"respondToAfd": enum,
"colorMetadata": enum

"audioDescriptions": [
  {
    "audioTypeControl": enum,
    "audioSourceName": "string",
    "audioNormalizationSettings": {
      "algorithm": enum,
      "algorithmControl": enum,
      "correctionGateLevel": integer,
      "loudnessLogging": enum,
      "targetLkfs": number,
      "peakCalculation": enum
    },
    "codecSettings": {
      "codec": enum,
      "ac3Settings": {
        "audioDescriptionBroadcasterMix": enum,
        "vbrQuality": enum,
        "bitrate": integer,
        "rateControlMode": enum,
        "codecProfile": enum,
        "codingMode": enum,
        "rawFormat": enum,
        "sampleRate": integer,
        "specification": enum
      },
      "ac3Settings": {
        "bitrate": integer,
        "bitstreamMode": enum,
        "audioTypeControl": enum,
        "audioSourceName": "string",
        "audioNormalizationSettings": {
          "algorithm": enum,
          "algorithmControl": enum,
          "correctionGateLevel": integer,
          "loudnessLogging": enum,
          "targetLkfs": number,
          "peakCalculation": enum
        },
        "codecSettings": {
          "codec": enum,
          "ac3Settings": {
            "audioDescriptionBroadcasterMix": enum,
            "vbrQuality": enum,
            "bitrate": integer,
            "rateControlMode": enum,
            "codecProfile": enum,
            "codingMode": enum,
            "rawFormat": enum,
            "sampleRate": integer,
            "specification": enum
          }
        }
      }
    }
  }
]
"codingMode": enum,
"dialnorm": integer,
"dynamicRangeCompressionProfile": enum,
"metadataControl": enum,
"lfeFilter": enum,
"sampleRate": integer
},
"aiffSettings": {
"bitDepth": integer,
"channels": integer,
"sampleRate": integer
},
"eac3Settings": {
"metadataControl": enum,
"surroundExMode": enum,
"loRoSurroundMixLevel": number,
"phaseControl": enum,
"dialnorm": integer,
"ltRtSurroundMixLevel": number,
"bitrate": integer,
"ltRtCenterMixLevel": number,
"passthroughControl": enum,
"lfeControl": enum,
"loRoCenterMixLevel": number,
"attenuationControl": enum,
"codingMode": enum,
"surroundMode": enum,
"bitstreamMode": enum,
"lfeFilter": enum,
"stereoDownmix": enum,
"dynamicRangeCompressionRf": enum,
"sampleRate": integer,
"dynamicRangeCompressionLine": enum,
"dcFilter": enum
},
"mp2Settings": {
"bitrate": integer,
"channels": integer,
"sampleRate": integer
},
"wavSettings": {
"bitDepth": integer,
"channels": integer,
"sampleRate": integer,
"format": enum
},
"remixSettings": {
"channelMapping": {
"outputChannels": [
{
"inputChannels": [
integer
]
]
}
},
"channelsIn": integer,
"channelsOut": integer
},
"streamName": "string",
"languageCodeControl": enum,
"audioType": integer,
"customLanguageCode": "string",
"languageCode": enum
}
"outputSettings": {
  "hlsSettings": {
    "audioGroupId": "string",
    "audioRenditionSets": "string",
    "audioTrackType": enum,
    "iFrameOnlyManifest": enum,
    "segmentModifier": "string"
  }
},
"extension": "string",
"nameModifier": "string",
"captionDescriptions": [
  {
    "captionSelectorName": "string",
    "destinationSettings": {
      "destinationType": enum,
      "burninDestinationSettings": {
        "backgroundOpacity": integer,
        "shadowXOffset": integer,
        "teletextSpacing": enum,
        "alignment": enum,
        "outlineSize": integer,
        "yPosition": integer,
        "shadowColor": enum,
        "fontOpacity": integer,
        "fontSize": integer,
        "fontScript": enum,
        "fontColor": enum,
        "backgroundColor": enum,
        "fontResolution": integer,
        "outlineColor": enum,
        "shadowYOffset": integer,
        "xPosition": integer,
        "shadowOpacity": integer
      },
      "dvbSubDestinationSettings": {
        "backgroundOpacity": integer,
        "shadowXOffset": integer,
        "teletextSpacing": enum,
        "alignment": enum,
        "outlineSize": integer,
        "yPosition": integer,
        "shadowColor": enum,
        "fontOpacity": integer,
        "fontSize": integer,
        "fontScript": enum,
        "fontColor": enum,
        "backgroundColor": enum,
        "fontResolution": integer,
        "outlineColor": enum,
        "shadowYOffset": integer,
        "xPosition": integer,
        "shadowOpacity": integer
      },
      "sccDestinationSettings": {
        "framerate": enum
      },
      "teletextDestinationSettings": {
        "pageNumber": "string"
      },
      "ttmlDestinationSettings": {
        "stylePassthrough": enum
      },
      "embeddedDestinationSettings": {
        "destination608ChannelNumber": integer
      }
    }
  }
]
"outputGroupSettings": {
    "type": enum,
    "hlsGroupSettings": {
        "manifestDurationFormat": enum,
        "segmentLength": integer,
        "timedMetadataId3Period": integer,
        "captionLanguageSetting": enum,
        "captionLanguageMappings": [
            {
                "captionChannel": integer,
                "customLanguageCode": "string",
                "languageCode": enum,
                "languageDescription": "string"
            }
        ],
        "destination": "string",
        "destinationSettings": {
            "s3Settings": {
                "encryption": {
                    "encryptionType": enum,
                    "kmsKeyArn": "string"
                }
            }
        }
    }
},
"encryption": {
    "encryptionMethod": enum,
    "constantInitializationVector": "string",
    "initializationVectorInManifest": enum,
    "offlineEncrypted": enum,
    "spekeKeyProvider": {
        "resourceId": "string",
        "systemIds": [
            "string"
        ],
        "url": "string",
        "certificateArn": "string"
    },
    "staticKeyProvider": {
        "staticKeyValue": "string",
        "keyFormat": "string",
        "keyFormatVersions": "string",
        "url": "string"
    }
},
"type": enum,
"timedMetadataId3Frame": enum,
"baseUrl": "string",
"codecSpecification": enum,
"outputSelection": enum,
"programDateTimePeriod": integer,
"segmentsPerSubdirectory": integer,
"minSegmentLength": integer,
"minFinalSegmentLength": number,
"directoryStructure": enum,
"programDateTime": enum,
"adMarkers": [
    enum
"segmentControl": enum,
"timestampDeltaMilliseconds": integer,
"manifestCompression": enum,
"clientCache": enum,
"streamInfResolution": enum
},
"dashIsoGroupSettings": {
"segmentLength": integer,
"destination": "string",
"destinationSettings": {
"s3Settings": {
"encryption": {
"encryptionType": enum,
"kmsKeyArn": "string"
}
}
},
"encryption": {
"spekeKeyProvider": {
"resourceId": "string",
"systemIds": [
"string"
],
"url": "string",
"certificateArn": "string"
}
},
"minBufferTime": integer,
"fragmentLength": integer,
"baseUrl": "string",
"segmentControl": enum,
"hbbtvCompliance": enum,
"writeSegmentTimelineInRepresentation": enum
},
"fileGroupSettings": {
"destination": "string",
"destinationSettings": {
"s3Settings": {
"encryption": {
"encryptionType": enum,
"kmsKeyArn": "string"
}
}
},
"msSmoothGroupSettings": {
"destination": "string",
"destinationSettings": {
"s3Settings": {
"encryption": {
"encryptionType": enum,
"kmsKeyArn": "string"
}
}
},
"fragmentLength": integer,
"encryption": {
"spekeKeyProvider": {
"resourceId": "string",
"systemIds": [
"string"
],
"url": "string",
"certificateArn": "string"
}
}
"manifestEncoding": enum,
"audioDeduplication": enum
},
"cmafGroupSettings": {
  "writeHlsManifest": enum,
  "writeDashManifest": enum,
  "segmentLength": integer,
  "minFinalSegmentLength": number,
  "destination": "string",
  "destinationSettings": {
    "s3Settings": {
      "encryption": {
        "encryptionType": enum,
        "kmsKeyArn": "string"
      }
    },
    "encryption": {
      "encryptionMethod": enum,
      "constantInitializationVector": "string",
      "initializationVectorInManifest": enum,
      "staticKeyProvider": {
        "staticKeyValue": "string",
        "keyFormat": "string",
        "keyFormatVersions": "string",
        "url": "string"
      },
      "type": enum
    },
    "minBufferTime": integer,
    "fragmentLength": integer,
    "baseUrl": "string",
    "segmentControl": enum,
    "manifestDurationFormat": enum,
    "streamInfResolution": enum,
    "clientCache": enum,
    "manifestCompression": enum,
    "codecSpecification": enum
  }
},
"adAvailOffset": integer,
"availBlanking": {
  "availBlankingImage": "string"
},
"timedMetadataInsertion": {
  "id3Insertions": [
    {"timecode": "string",
     "id3": "string"
    }
  ],
  "nielsenConfiguration": {
    "breakoutCode": integer,
    "distributorId": "string"
  },
  "motionImageInserter": {
    "insertionMode": enum,
    "input": "string",
    "offset": {
      "imageX": integer,
      "imageY": integer
    }
  }
}
"startTime": "string",
"playback": enum,
"framerate": {
  "framerateNumerator": integer,
  "framerateDenominator": integer
},
"esam": {
  "signalProcessingNotification": {
    "mccXml": "string"
  },
  "manifestConfirmConditionNotification": {
    "mccXml": "string"
  },
  "responseSignalPreroll": integer
},
"inputs": [
  {
    "inputClippings": [
      {
        "endTimecode": "string",
        "startTimecode": "string"
      }
    ],
    "audioSelectors": {
    },
    "audioSelectorGroups": {
    },
    "programNumber": integer,
    "videoSelector": {
      "colorSpace": enum,
      "rotate": enum,
      "pid": integer,
      "programNumber": integer,
      "colorSpaceUsage": enum,
      "hdr10Metadata": {
        "redPrimaryX": integer,
        "redPrimaryY": integer,
        "greenPrimaryX": integer,
        "greenPrimaryY": integer,
        "bluePrimaryX": integer,
        "bluePrimaryY": integer,
        "whitePointX": integer,
        "whitePointY": integer,
        "maxFrameAverageLightLevel": integer,
        "maxContentLightLevel": integer,
        "maxLuminance": integer,
        "minLuminance": integer
      }
    },
    "filterEnable": enum,
    "psiControl": enum,
    "filterStrength": integer,
    "deblockFilter": enum,
    "denoiseFilter": enum,
    "timecodeSource": enum,
    "captionSelectors": {
    },
    "imageInserter": {
      "insertableImages": [
        {
          "width": integer,
          "height": integer,
          "imageX": integer,
          "imageY": integer,
          "duration": integer,
          
        }]}},

Example CreateJobTemplateResponse

```json
{
  "jobTemplate": {
    "arn": "string",
    "createdAt": "string",
    "lastUpdated": "string",
    "description": "string",
    "category": "string",
    "queue": "string",
    "name": "string",
    "type": enum,
    "settings": {
      "timecodeConfig": {
        "anchor": "string",
        "source": enum,
        "start": "string",
        "timestampOffset": "string"
      },
      "outputGroups": [
        {
          "customName": "string",
          "name": "string",
          "outputs": [
            {
              "containerSettings": {
                "container": enum,
                "m3u8Settings": {
                  "audioFramesPerPes": integer,
                  "pcrControl": enum,
                  "pcrPid": integer,
                  "pmtPid": integer,
                  "privateMetadataPid": integer,
                  "programNumber": integer,
                  "patInterval": integer,
                  "pmtInterval": integer,
                  "scte35Source": enum,
                  "scte35Pid": integer,
                  "nielsenId3": enum,
                  "timedMetadata": enum,
                  "timedMetadataPid": integer,
                  "transportStreamId": integer,
                  "videoPid": integer,
```
"audioPids": [
    integer
],
"f4vSettings": {
    "moovPlacement": enum
},
"m2tsSettings": {
    "audioBufferModel": enum,
    "minEbpInterval": integer,
    "esRateInPes": enum,
    "patInterval": integer,
    "dvbNitSettings": {
        "nitInterval": integer,
        "networkId": integer,
        "networkName": "string"
    },
    "dvbSdtSettings": {
        "outputSdt": enum,
        "sdtInterval": integer,
        "serviceName": "string",
        "serviceProviderName": "string"
    },
    "scte35Source": enum,
    "scte35Pid": integer,
    "scte35Esam": {
        "scte35EsamPid": integer
    },
    "videoPid": integer,
    "dvbTdtSettings": {
        "tdtInterval": integer
    },
    "pmtInterval": integer,
    "segmentationStyle": enum,
    "segmentationTime": number,
    "pmtPid": integer,
    "bitrate": integer,
    "audioPids": [
        integer
    ],
    "privateMetadataPid": integer,
    "nielsenId3": enum,
    "timedMetadataPid": integer,
    "maxPcrInterval": integer,
    "transportStreamId": integer,
    "dvbSubPids": [
        integer
    ],
    "rateMode": enum,
    "audioFramesPerPes": integer,
    "pcrControl": enum,
    "segmentationMarkers": enum,
    "ebpAudioInterval": enum,
    "forceTsVideoEbpOrder": enum,
    "programNumber": integer,
    "pcrPid": integer,
    "bufferModel": enum,
    "dvbTeletextPid": integer,
    "fragmentTime": number,
    "ebpPlacement": enum,
    "nullPacketBitrate": number
},
"movSettings": {
    "clapAtom": enum,
    "cslgAtom": enum,
    "paddingControl": enum,
"reference": enum,
"mpeg2FourCCControl": enum
},
"mp4Settings": {
  "cslgAtom": enum,
  "freeSpaceBox": enum,
  "mp4MajorBrand": "string",
  "moovPlacement": enum
}
},
"preset": "string",
"videoDescription": {
  "fixedAfd": integer,
  "width": integer,
  "scalingBehavior": enum,
  "crop": {
    "height": integer,
    "width": integer,
    "x": integer,
    "y": integer
  },
  "height": integer,
  "videoPreprocessors": {
    "colorCorrector": {
      "brightness": integer,
      "colorSpaceConversion": enum,
      "contrast": integer,
      "hue": integer,
      "saturation": integer,
      "hdr10Metadata": {
        "redPrimaryX": integer,
        "redPrimaryY": integer,
        "greenPrimaryX": integer,
        "greenPrimaryY": integer,
        "bluePrimaryX": integer,
        "bluePrimaryY": integer,
        "whitePointX": integer,
        "whitePointY": integer,
        "maxFrameAverageLightLevel": integer,
        "maxContentLightLevel": integer,
        "maxLuminance": integer,
        "minLuminance": integer
      }
    },
    "deinterlacer": {
      "algorithm": enum,
      "mode": enum,
      "control": enum
    },
    "imageInserter": {
      "insertableImages": [
        {
          "width": integer,
          "height": integer,
          "imageX": integer,
          "imageY": integer,
          "duration": integer,
          "fadeIn": integer,
          "layer": integer,
          "imageInserterInput": "string",
          "startTime": "string",
          "fadeOut": integer,
          "opacity": integer
        }
      ]
    }
  }
},

69
"noiseReducer": {
    "filter": enum,
    "filterSettings": {
        "strength": integer
    },
    "spatialFilterSettings": {
        "strength": integer,
        "speed": integer,
        "postFilterSharpenStrength": integer
    }
},
"timecodeBurnin": {
    "fontSize": integer,
    "position": enum,
    "prefix": "string"
},
"timecodeInsertion": enum,
"antiAlias": enum,
"position": {
    "height": integer,
    "width": integer,
    "x": integer,
    "y": integer
},
"sharpness": integer,
"codecSettings": {
    "codec": enum,
    "frameCaptureSettings": {
        "framerateNumerator": integer,
        "framerateDenominator": integer,
        "maxCaptures": integer,
        "quality": integer
    },
    "h264Settings": {
        "interlaceMode": enum,
        "parNumerator": integer,
        "numberReferenceFrames": integer,
        "syntax": enum,
        "softness": integer,
        "framerateDenominator": integer,
        "gopClosedCadence": integer,
        "hrdBufferInitialFillPercentage": integer,
        "gopSize": number,
        "slices": integer,
        "gopReference": enum,
        "hrdBufferSize": integer,
        "maxBitrate": integer,
        "slowPal": enum,
        "parDenominator": integer,
        "spatialAdaptiveQuantization": enum,
        "temporalAdaptiveQuantization": enum,
        "flickerAdaptiveQuantization": enum,
        "entropyEncoding": enum,
        "bitrate": integer,
        "framerateControl": enum,
        "rateControlMode": enum,
        "qvbrSettings": {
            "qvbrQualityLevel": integer,
            "maxAverageBitrate": integer
        },
        "codecProfile": enum,
        "telecine": enum,
        "framerateNumerator": integer,
        "minInterval": integer,
        "adaptiveQuantization": enum,
"codecLevel": enum,
"fieldEncoding": enum,
"sceneChangeDetect": enum,
"qualityTuningLevel": enum,
"framerateConversionAlgorithm": enum,
"unregisteredSeiTimecode": enum,
"gopSizeUnits": enum,
"parControl": enum,
"numberBFramesBetweenReferenceFrames": integer,
"repeatPps": enum,
"dynamicSubGop": enum
},
"h265Settings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "numberReferenceFrames": integer,
  "framerateDenominator": integer,
  "gopClosedCadence": integer,
  "alternateTransferFunctionSei": enum,
  "hrdBufferInitialFillPercentage": integer,
  "gopSize": number,
  "slices": integer,
  "gopBRReference": enum,
  "hrdBufferSize": integer,
  "maxBitrate": integer,
  "slowPal": enum,
  "parDenominator": integer,
  "spatialAdaptiveQuantization": enum,
  "temporalAdaptiveQuantization": enum,
  "flickerAdaptiveQuantization": enum,
  "bitrate": integer,
  "framerateControl": enum,
  "rateControlMode": enum,
  "qvbrSettings": {
    "qvbrQualityLevel": integer,
    "maxAverageBitrate": integer
  },
  "codecProfile": enum,
  "tiles": enum,
  "telecine": enum,
  "framerateNumerator": integer,
  "minIInterval": integer,
  "adaptiveQuantization": enum,
  "codecLevel": enum,
  "sceneChangeDetect": enum,
  "qualityTuningLevel": enum,
  "framerateConversionAlgorithm": enum,
  "unregisteredSeiTimecode": enum,
  "gopSizeUnits": enum,
  "parControl": enum,
  "numberBFramesBetweenReferenceFrames": integer,
  "temporalIds": enum,
  "sampleAdaptiveOffsetFilterMode": enum,
  "writeMp4PackagingType": enum,
  "dynamicSubGop": enum
},
"mpeg2Settings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "syntax": enum,
  "softness": integer,
  "framerateDenominator": integer,
  "gopClosedCadence": integer,
  "hrdBufferInitialFillPercentage": integer,
  "gopSize": number,
  "hrdBufferSize": integer,
Schemas

"maxBitrate": integer,
"slowPal": enum,
"parDenominator": integer,
"spatialAdaptiveQuantization": enum,
"temporalAdaptiveQuantization": enum,
"bitrate": integer,
"intraDcPrecision": enum,
"framerateControl": enum,
"rateControlMode": enum,
"codecProfile": enum,
"telecine": enum,
"framerateNumerator": integer,
"minIInterval": integer,
"adaptiveQuantization": enum,
"codecLevel": enum,
"sceneChangeDetect": enum,
"qualityTuningLevel": enum,
"framerateConversionAlgorithm": enum,
"gopSizeUnits": enum,
"parControl": enum,
"numberBFramesBetweenReferenceFrames": integer,
"dynamicSubGop": enum
},
"proresSettings": {

"interlaceMode": enum,
"parNumerator": integer,
"framerateDenominator": integer,
"codecProfile": enum,
"slowPal": enum,
"parDenominator": integer,
"framerateControl": enum,
"telecine": enum,
"framerateNumerator": integer,
"framerateConversionAlgorithm": enum,
"parControl": enum
}
},
"afdSignaling": enum,
"dropFrameTimecode": enum,
"respondToAfd": enum,
"colorMetadata": enum
},
"audioDescriptions": [
{
"audioTypeControl": enum,
"audioSourceName": "string",
"audioNormalizationSettings": {

"algorithm": enum,
"algorithmControl": enum,
"correctionGateLevel": integer,
"loudnessLogging": enum,
"targetLkfs": number,
"peakCalculation": enum
},
"codecSettings": {

"codec": enum,
"aacSettings": {

"audioDescriptionBroadcasterMix": enum,
"vbrQuality": enum,
"bitrate": integer,
"rateControlMode": enum,
"codecProfile": enum,
"codingMode": enum,
"rawFormat": enum,
"sampleRate": integer,
"specification": enum
}
"ac3Settings": {
    "bitrate": integer,
    "bitstreamMode": enum,
    "codingMode": enum,
    "dialnorm": integer,
    "dynamicRangeCompressionProfile": enum,
    "metadataControl": enum,
    "lfeFilter": enum,
    "sampleRate": integer
},
"aiffSettings": {
    "bitDepth": integer,
    "channels": integer,
    "sampleRate": integer
},
"eac3Settings": {
    "metadataControl": enum,
    "surroundExMode": enum,
    "loRoSurroundMixLevel": number,
    "phaseControl": enum,
    "dialnorm": integer,
    "lRtSurroundMixLevel": number,
    "bitrate": integer,
    "lRtCenterMixLevel": number,
    "passThroughControl": enum,
    "lfeControl": enum,
    "loRoCenterMixLevel": number,
    "attenuationControl": enum,
    "codingMode": enum,
    "surroundMode": enum,
    "bitstreamMode": enum,
    "lfeFilter": enum,
    "stereoDownmix": enum,
    "dynamicRangeCompressionRf": enum,
    "sampleRate": integer,
    "dynamicRangeCompressionLine": enum,
    "dcFilter": enum
},
"mp2Settings": {
    "bitrate": integer,
    "channels": integer,
    "sampleRate": integer
},
"wavSettings": {
    "bitDepth": integer,
    "channels": integer,
    "sampleRate": integer,
    "format": enum
},
"remixSettings": {
    "channelMapping": {
        "outputChannels": [
            {
                "inputChannels": [integer]
            }
        ],
    },
    "channelsIn": integer,
    "channelsOut": integer
},
"streamName": "string",
"languageCodeControl": enum,
"audioType": integer,
"customLanguageCode": "string",
"languageCode": enum
}
],
"outputSettings": {
"hlsSettings": {
"audioGroupId": "string",
"audioRenditionSets": "string",
"audioTrackType": enum,
"iFrameOnlyManifest": enum,
"segmentModifier": "string"
}
},
"extension": "string",
"nameModifier": "string",
"captionDescriptions": [
{"captionSelectorName": "string",
"destinationSettings": {
"burninDestinationSettings": {
"backgroundOpacity": integer,
"shadowXOffset": integer,
"teletextSpacing": enum,
"alignment": enum,
"outlineSize": integer,
"yPosition": integer,
"shadowColor": enum,
"fontOpacity": integer,
"fontSize": integer,
"fontScript": enum,
"foregroundColor": enum,
"backgroundColor": enum,
"fontResolution": integer,
"outlineColor": enum,
"shadowYOffset": integer,
"xPosition": integer,
"shadowOpacity": integer
},
"dvbSubDestinationSettings": {
"backgroundOpacity": integer,
"shadowXOffset": integer,
"teletextSpacing": enum,
"alignment": enum,
"outlineSize": integer,
"yPosition": integer,
"shadowColor": enum,
"fontOpacity": integer,
"fontSize": integer,
"fontScript": enum,
"foregroundColor": enum,
"backgroundColor": enum,
"fontResolution": integer,
"outlineColor": enum,
"shadowYOffset": integer,
"xPosition": integer,
"shadowOpacity": integer
},
"sccDestinationSettings": {
"framerate": enum
},
"teletextDestinationSettings": {
"pageNumber": "string"
},
"ttmlDestinationSettings": {
"framerate": enum
}
"stylePassthrough": enum,
"embeddedDestinationSettings": {
  "destination608ChannelNumber": integer
},
"customLanguageCode": "string",
"languageCode": enum,
"languageDescription": "string"
}
],
"outputGroupSettings": {
"type": enum,
"hlsGroupSettings": {
  "manifestDurationFormat": enum,
  "segmentLength": integer,
  "timedMetadataId3Period": integer,
  "captionLanguageSetting": enum,
  "captionLanguageMappings": [
    {
      "captionChannel": integer,
      "customLanguageCode": "string",
      "languageCode": enum,
      "languageDescription": "string"
    }
  ],
  "destination": "string",
  "destinationSettings": {
    "s3Settings": {
      "encryption": {
        "encryptionType": enum,
        "kmsKeyArn": "string"
      }
    },
    "encryption": {
      "encryptionMethod": enum,
      "constantInitializationVector": "string",
      "initializationVectorInManifest": enum,
      "offlineEncrypted": enum,
      "spekeKeyProvider": {
        "resourceId": "string",
        "systemIds": ["string"
      ],
      "url": "string",
      "certificateArn": "string"
    },
    "staticKeyProvider": {
      "staticKeyValue": "string",
      "keyFormat": "string",
      "keyFormatVersions": "string",
      "url": "string"
    }
  },
  "type": enum
},
"timedMetadataId3Frame": enum,
"baseUrl": "string",
"codecSpecification": enum,
"outputSelection": enum,
"programDateTimePeriod": integer,
"segmentsPerSubdirectory": integer,
"minSegmentLength": integer,
"minFinalSegmentLength": number,
"directoryStructure": enum,
"programDateTime": enum,
"adMarkers": [ enum
],
"segmentControl": enum,
"timestampDeltaMilliseconds": integer,
"manifestCompression": enum,
"clientCache": enum,
"streamInfResolution": enum
},
"dashIsoGroupSettings": {
"segmentLength": integer,
"destination": "string",
"destinationSettings": {
"s3Settings": {
"encryption": {
"encryptionType": enum,
"kmsKeyArn": "string"
}
}
},
"encryption": {
"spekeKeyProvider": {
"resourceId": "string",
"systemIds": [ "string"
],
"url": "string",
"certificateArn": "string"
}
},
"minBufferTime": integer,
"fragmentLength": integer,
"baseUrl": "string",
"segmentControl": enum,
"hbbtvCompliance": enum,
"writeSegmentTimelineInRepresentation": enum
},
"fileGroupSettings": {
"destination": "string",
"destinationSettings": {
"s3Settings": {
"encryption": {
"encryptionType": enum,
"kmsKeyArn": "string"
}
}
},
"msSmoothGroupSettings": {
"destination": "string",
"destinationSettings": {
"s3Settings": {
"encryption": {
"encryptionType": enum,
"kmsKeyArn": "string"
}
}
},
"fragmentLength": integer,
"encryption": {
"spekeKeyProvider": {
"resourceId": "string",
"systemIds": [ "string"
]
AWS Elemental MediaConvert API Reference API Reference
Schemas

},
"url": "string",
"certificateArn": "string"
}
},
"manifestEncoding": enum,
"audioDeduplication": enum,
},
"cmafGroupSettings": {
"writeHlsManifest": enum,
"writeDashManifest": enum,
"segmentLength": integer,
"minFinalSegmentLength": number,
"destination": "string",
"destinationSettings": {
"s3Settings": {
"encryption": {
"encryptionType": enum,
"kmsKeyArn": "string"
}
}
},
"encryption": {
"encryptionMethod": enum,
"constantInitializationVector": "string",
"initializationVectorInManifest": enum,
"staticKeyProvider": {
"staticKeyValue": "string",
"keyFormat": "string",
"keyFormatVersions": "string",
"url": "string"
},
"type": enum
},
"minBufferTime": integer,
"fragmentLength": integer,
"baseUrl": "string",
"segmentControl": enum,
"manifestDurationFormat": enum,
"streamInfResolution": enum,
"clientCache": enum,
"manifestCompression": enum,
"codecSpecification": enum
}
}
,"adAvailOffset": integer,
"availBlanking": {
"availBlankingImage": "string"
},
"timedMetadataInsertion": {
"id3Insertions": [
{
"timecode": "string",
"id3": "string"
}
],
"nielsenConfiguration": {
"breakoutCode": integer,
"distributorId": "string"
},
"motionImageInserter": {
"insertionMode": enum,
"input": "string"}
"offset": {
  "imageX": integer,
  "imageY": integer
},
"startTime": "string",
"playback": enum,
"framerate": {
  "framerateNumerator": integer,
  "framerateDenominator": integer
}
},
"esam": {
  "signalProcessingNotification": {
    "sccXml": "string"
  },
  "manifestConfirmConditionNotification": {
    "mccXml": "string"
  },
  "responseSignalPreroll": integer
},
"inputs": [
  {
    "inputClippings": [
      {
        "endTimecode": "string",
        "startTimecode": "string"
      }
    ],
    "audioSelectors": {
    },
    "audioSelectorGroups": {
    },
    "programNumber": integer,
    "videoSelector": {
      "colorSpace": enum,
      "rotate": enum,
      "pid": integer,
      "programNumber": integer,
      "colorSpaceUsage": enum,
      "hdr10Metadata": {
        "redPrimaryX": integer,
        "redPrimaryY": integer,
        "greenPrimaryX": integer,
        "greenPrimaryY": integer,
        "bluePrimaryX": integer,
        "bluePrimaryY": integer,
        "whitePointX": integer,
        "whitePointY": integer,
        "maxFrameAverageLightLevel": integer,
        "maxContentLightLevel": integer,
        "maxLuminance": integer,
        "minLuminance": integer
      }
    },
    "filterEnable": enum,
    "psiControl": enum,
    "filterStrength": integer,
    "deblockFilter": enum,
    "denoiseFilter": enum,
    "timecodeSource": enum,
    "captionSelectors": {
    },
    "imageInserter": {
      "insertableImages": [
        {
          "width": integer,
          "height": integer
        }
      ]
    }
  }
]
Properties

AacAudioDescriptionBroadcasterMix

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

AAC Profile.

LC
HEV1
HEV2

AacCodingMode

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**

Rate Control Mode.

- CBR
- VBR

**AacRawFormat**

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**

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value AAC. The service accepts one of two mutually exclusive groups of AAC settings--VBR and CBR. To select one of these modes, set the value of Bitrate control mode (rateControlMode) to "VBR" or "CBR". In VBR mode, you control the audio quality with the setting VBR quality (vbrQuality). In CBR mode, you use the setting Bitrate (bitrate). Defaults and valid values depend on the rate control mode.

**audioDescriptionBroadcasterMix**

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.

Type: AacAudioDescriptionBroadcasterMix (p. 79)
Required: False

**vbrQuality**

VBR Quality Level - Only used if rate_control_mode is VBR.

Type: AacVbrQuality (p. 82)
Required: False
**bitrate**

Average bitrate in bits/second. The set of valid values for this setting is: 6000, 8000, 10000, 12000, 14000, 16000, 20000, 24000, 28000, 32000, 40000, 48000, 56000, 64000, 80000, 96000, 112000, 128000, 160000, 192000, 224000, 256000, 288000, 320000, 384000, 448000, 512000, 576000, 640000, 768000, 896000, 1024000. The value you set is also constrained by the values you choose for Profile (codecProfile), Bitrate control mode (codingMode), and Sample rate (sampleRate). Default values depend on Bitrate control mode and Profile.

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

**rateControlMode**

Rate Control Mode.

- **Type:** AacRateControlMode (p. 80)
- **Required:** False

**codecProfile**

AAC Profile.

- **Type:** AacCodecProfile (p. 79)
- **Required:** False

**codingMode**

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.

- **Type:** AacCodingMode (p. 79)
- **Required:** False

**rawFormat**

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.

- **Type:** AacRawFormat (p. 80)
- **Required:** False

**sampleRate**

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

- **Type:** integer
- **Required:** False
- **Minimum:** 8000
- **Maximum:** 96000
specification

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

Type: AacSpecification (p. 82)
Required: False

AacSpecification

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

MPEG2
MPEG4

AacVbrQuality

VBR Quality Level - Only used if rate_control_mode is VBR.

LOW
MEDIUM_LOW
MEDIUM_HIGH
HIGH

Ac3BitstreamMode

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

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

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**

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

ENABLED
DISABLED

**Ac3MetadataControl**

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**

Required when you set (Codec) under (AudioDescriptions)> CodecSettings) to the value AC3.

**bitrate**

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

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

**bitstreamMode**

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

*Type:* Ac3BitstreamMode (p. 82)
*Required:* False

**codingMode**

Dolby Digital coding mode. Determines number of channels.

*Type:* Ac3CodingMode (p. 82)
*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

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

Type: Ac3DynamicRangeCompressionProfile (p. 82)
Required: False

metadataControl
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.

Type: Ac3MetadataControl (p. 83)
Required: False

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

Type: Ac3LfeFilter (p. 83)
Required: False

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

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

AccelerationMode
Enable Acceleration (AccelerationMode) on any job that you want processed with accelerated transcoding.

DISABLED
ENABLED

AccelerationSettings
Accelerated transcoding can significantly speed up jobs with long, visually complex content. Outputs that use this feature incur pro-tier pricing. For information about feature limitations, see the AWS Elemental MediaConvert User Guide.
mode

Acceleration configuration for the job.

Type: AccelerationMode (p. 84)
Required: True

AfdSignaling

This setting only applies to H.264, H.265, and MPEG2 outputs. Use Insert AFD signaling (AfdSignaling) to specify whether the service includes 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

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value AIFF.

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

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

sampleRate

Sample rate in hz.

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

AncillarySourceSettings

Settings for ancillary captions source.
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

The anti-alias filter is automatically applied to all outputs. The service no longer accepts the value DISABLED for AntiAlias. If you specify that in your job, the service will ignore the setting.

- DISABLED
- ENABLED

AudioCodec

Type of Audio codec.

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

AudioCodecSettings

Audio codec settings (CodecSettings) under (AudioDescriptions) contains the group of settings related to audio encoding. The settings in this group vary depending on the value you choose for Audio codec (Codec). For each codec enum you choose, define the corresponding settings object. The following lists the codec enum, settings object pairs.  

- AAC, AacSettings  
- MP2, Mp2Settings  
- WAV, WavSettings  
- AIFF, AiffSettings  
- AC3, Ac3Settings  
- EAC3, Eac3Settings

**codec**

Type of Audio codec.

**Type:** AudioCodec  
**Required:** False

**aacSettings**

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value AAC. The service accepts one of two mutually exclusive groups of AAC settings--VBR and CBR. To select one of these modes, set the value of Bitrate control mode (rateControlMode) to "VBR" or "CBR". In VBR mode, you control the audio quality with the setting VBR quality (vbrQuality). In CBR mode, you use the setting Bitrate (bitrate). Defaults and valid values depend on the rate control mode.

**Type:** AacSettings  
**Required:** False
**ac3Settings**
Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value AC3.

*Type:* Ac3Settings (p. 83)  
*Required:* False

**aiffSettings**
Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value AIFF.

*Type:* AiffSettings (p. 85)  
*Required:* False

**eac3Settings**
Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value EAC3.

*Type:* Eac3Settings (p. 125)  
*Required:* False

**mp2Settings**
Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value MP2.

*Type:* Mp2Settings (p. 200)  
*Required:* False

**wavSettings**
Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value WAV.

*Type:* WavSettings (p. 237)  
*Required:* False

**AudioDefaultSelection**
Enable this setting on one audio selector to set it as the default for the job. The service uses this default for outputs where it can't find the specified input audio. If you don't set a default, those outputs have no audio.

*DEFAULT*  
*NOT_DEFAULT*

**AudioDescription**
Description of audio output

**audioTypeControl**
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.
Properties

Type: AudioTypeControl (p. 93)
Required: False

audioSourceName

Specifies which audio data to use from each input. In the simplest case, specify an "Audio Selector" as #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" as #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

audioNormalizationSettings

Advanced audio normalization settings.

Type: AudioNormalizationSettings (p. 90)
Required: False

codecSettings

Audio codec settings (CodecSettings) under (AudioDescriptions) contains the group of settings related to audio encoding. The settings in this group vary depending on the value you choose for Audio codec (Codec). For each codec enum you choose, define the corresponding settings object. The following lists the codec enum, settings object pairs. * AAC, AacSettings * MP2, Mp2Settings * WAV, WavSettings * AIFF, AiffSettings * AC3, Ac3Settings * EAC3, Eac3Settings

Type: AudioCodecSettings (p. 86)
Required: False

remixSettings

Advanced audio remixing settings.

Type: RemixSettings (p. 222)
Required: False

streamName

Used for MS Smooth 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]*$

languageCodeControl

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.

**Type:** AudioLanguageCodeControl (p. 89)
**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

**customLanguageCode**
Specify the language for this audio output track, using the ISO 639-2 or ISO 639-3 three-letter language code. The language specified 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:** string
**Required:** False
**Pattern:** ^[A-Za-z]{3}$
**MinLength:** 3
**MaxLength:** 3

**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. 179)
**Required:** False

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

LOG
DONT_LOG

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

TRUE_PEAK
NONE

AudioNormalizationSettings
Advanced audio normalization settings.

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

Type: AudioNormalizationAlgorithm (p. 89)
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
**loudnessLogging**

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

*Type:* AudioNormalizationLoudnessLogging (p. 90)  
*Required:* False

**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

**peakCalculation**

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

*Type:* AudioNormalizationPeakCalculation (p. 90)  
*Required:* False

**AudioSelector**

Selector for Audio

**tracks**

Identify a track from the input audio to include in this selector by entering the track index number. To include several tracks in a single audio selector, specify multiple tracks as follows. Using the console, enter a comma-separated list. For example, type "1,2,3" to include tracks 1 through 3. Specifying directly in your JSON job file, provide the track numbers in an array. For example, "tracks": [1,2,3].

*Type:* Array of type integer  
*Required:* False  
*Minimum:* 1  
*Maximum:* 2147483647

**offset**

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

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

**defaultSelection**

Enable this setting on one audio selector to set it as the default for the job. The service uses this default for outputs where it can't find the specified input audio. If you don't set a default, those outputs have no audio.
**Properties**

**Type**: AudioDefaultSelection (p. 87)
**Required**: False

**selectorType**

Specifies the type of the audio selector.

**Type**: AudioSelectorType (p. 93)
**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
**Minimum**: 1
**Maximum**: 2147483647

**externalAudioFileInput**

Specifies audio data from an external file source.

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

**programSelection**

Use this setting for input streams that contain Dolby E, to have the service extract specific program data from the track. To select multiple programs, create multiple selectors with the same Track and different Program numbers. In the console, this setting is visible when you set Selector type to Track. Choose the program number from the dropdown list. If you are sending a JSON file, provide the program ID, which is part of the audio metadata. If your input file has incorrect metadata, you can choose All channels instead of a program number to have the service ignore the program IDs and include all the programs in the track.

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

**customLanguageCode**

Selects a specific language code from within an audio source, using the ISO 639-2 or ISO 639-3 three-letter language code
Properties

**Type**: string  
**Required**: False  
**Pattern**: ^[A-Za-z]{3}$  
**MinLength**: 3  
**MaxLength**: 3

**languageCode**

Selects a specific language code from within an audio source.

- **Type**: LanguageCode (p. 179)  
- **Required**: False

**remixSettings**

Use these settings to reorder the audio channels of one input to match those of another input. This allows you to combine the two files into a single output, one after the other.

- **Type**: RemixSettings (p. 222)  
- **Required**: False

**AudioSelectorGroup**

Group of Audio Selectors

**audioSelectorNames**

Name of an 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**: False  
- **MinLength**: 1

**AudioSelectorType**

Specifies the type of the audio selector.

- **PID**  
- **TRACK**  
- **LANGUAGE_CODE**

**AudioTypeControl**

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**

93
AvailBlanking

Settings for Avail Blanking

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$
- **MinLength**: 14

BurninDestinationSettings

Burn-In Destination Settings.

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

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

teletextSpacing

Only applies to jobs with input captions in Teletext or STL formats. Specify whether the spacing between letters in your captions is set by the captions grid or varies depending on letter width. Choose fixed grid to conform to the spacing specified in the captions file more accurately. Choose proportional to make the text easier to read if the captions are closed caption.

- **Type**: BurninSubtitleTeletextSpacing (p. 98)
- **Required**: False

alignment

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.

**Type:** BurninSubtitleAlignment (p. 97)
**Required:** False

### 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:** False
**Minimum:** 0
**Maximum:** 10

### 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

### shadowColor

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

**Type:** BurninSubtitleShadowColor (p. 98)
**Required:** False

### 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:** False
**Minimum:** 0
**Maximum:** 255

### 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
Properties

Minimum: 0
Maximum: 96

**fontScript**

Provide the font script, using an ISO 15924 script code, if the LanguageCode is not sufficient for determining the script type. Where LanguageCode or CustomLanguageCode is sufficient, use "AUTOMATIC" or leave unset. This is used to help determine the appropriate font for rendering burn-in captions.

_Type: FontScript (p. 133)_
_Required: False_

**fontColor**

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.

_Type: BurninSubtitleFontColor (p. 97)_
_Required: False_

**backgroundColor**

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

_Type: BurninSubtitleBackgroundColor (p. 97)_
_Required: False_

**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_

**outlineColor**

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.

_Type: BurninSubtitleOutlineColor (p. 98)_
_Required: False_

**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_
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: -2147483648
Maximum: 2147483647

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

BurninSubtitleAlignment

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

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

NONE
BLACK
WHITE

BurninSubtitleFontColor

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**

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**

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

NONE
BLACK
WHITE

**BurninSubtitleTeletextSpacing**

Only applies to jobs with input captions in Teletext or STL formats. Specify whether the spacing between letters in your captions is set by the captions grid or varies depending on letter width. Choose fixed grid to conform to the spacing specified in the captions file more accurately. Choose proportional to make the text easier to read if the captions are closed caption.

FIXED_GRID
PROPORTIONAL

**CaptionDescription**

Description of Caption output

**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: False  
MinLength: 1
destinationSettings
Specific settings required by destination type. Note that burnin_destination_settings are not available if the source of the caption data is Embedded or Teletext.

Type: CaptionDestinationSettings (p. 99)
Required: False

customLanguageCode
Indicates the language of the caption output track, using the ISO 639-2 or ISO 639-3 three-letter language code. For most captions output formats, the encoder puts this language information in the output captions metadata. If your output captions format is DVB-Sub or Burn in, the encoder uses this language information to choose the font language for rendering the captions text.

Type: string
Required: False
Pattern: ^[A-Za-z]{3}$
MinLength: 3
MaxLength: 3

languageCode
Specify the language of this captions output track. For most captions output formats, the encoder puts this language information in the output captions metadata. If your output captions format is DVB-Sub or Burn in, the encoder uses this language information to choose the font language for rendering the captions text.

Type: LanguageCode (p. 179)
Required: False

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

CaptionDestinationSettings
Specific settings required by destination type. Note that burnin_destination_settings are not available if the source of the caption data is Embedded or Teletext.

destinationType
Specify the format for this set of captions on this output. The default format is embedded without SCTE-20. Other options are embedded with SCTE-20, burn-in, DVB-sub, SCC, SRT, teletext, TTML, and web-VTT. If you are using SCTE-20, choose SCTE-20 plus embedded (SCTE20_PLUS_EMBEDDED) to create an output that complies with the SCTE-43 spec. To create a non-compliant output where the embedded captions come first, choose Embedded plus SCTE-20 (EMBEDDED_PLUS_SCTE20).

Type: CaptionDestinationType (p. 100)
Required: False
burninDestinationSettings
Burn-In Destination Settings.
  
  **Type:** BurninDestinationSettings (p. 94)
  **Required:** False

dvbSubDestinationSettings
DVB-Sub Destination Settings
  
  **Type:** DvbSubDestinationSettings (p. 118)
  **Required:** False

cscDestinationSettings
Settings for SCC caption output.
  
  **Type:** SccDestinationSettings (p. 224)
  **Required:** False

teletextDestinationSettings
Settings for Teletext caption output
  
  **Type:** TeletextDestinationSettings (p. 226)
  **Required:** False

ttmlDestinationSettings
Settings specific to TTML caption outputs, including Pass style information (TtmlStylePassthrough).
  
  **Type:** TtmlDestinationSettings (p. 230)
  **Required:** False

embeddedDestinationSettings
Settings specific to embedded/ancillary caption outputs, including 608/708 Channel destination number.
  
  **Type:** EmbeddedDestinationSettings (p. 129)
  **Required:** False

CaptionDestinationType
Specify the format for this set of captions on this output. The default format is embedded without SCTE-20. Other options are embedded with SCTE-20, burn-in, DVB-sub, SCC, SRT, teletext, TTML, and web-VTT. If you are using SCTE-20, choose SCTE-20 plus embedded (SCTE20_PLUS_EMBEDDED) to create an output that complies with the SCTE-43 spec. To create a non-compliant output where the embedded captions come first, choose Embedded plus SCTE-20 (EMBEDDED_PLUS_SCTE20).

  BURN_IN
  DVB_SUB
  EMBEDDED
**CaptionSelector**

Set up captions in your outputs by first selecting them from your input here.

**customLanguageCode**

The specific language to extract from source, using the ISO 639-2 or ISO 639-3 three-letter language code. 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**: string
- **Required**: False
- **Pattern**: `^[A-Za-z]{3}$`
- **MinLength**: 3
- **MaxLength**: 3

**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. 179)`
- **Required**: False

**sourceSettings**

Source settings (SourceSettings) contains the group of settings for captions in the input.

- **Type**: `CaptionSourceSettings (p. 101)`
- **Required**: False

**CaptionSourceSettings**

Source settings (SourceSettings) contains the group of settings for captions in the input.

**sourceType**

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

- **Type**: `CaptionSourceType (p. 102)`
**ancillarySourceSettings**

Settings for ancillary captions source.

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

**dvbSubSourceSettings**

DVB Sub Source Settings

- **Type:** DvbSubSourceSettings (p. 122)
- **Required:** False

**embeddedSourceSettings**

Settings for embedded captions Source

- **Type:** EmbeddedSourceSettings (p. 130)
- **Required:** False

**fileSourceSettings**

Settings for File-based Captions in Source

- **Type:** FileSourceSettings (p. 132)
- **Required:** False

**teletextSourceSettings**

Settings specific to Teletext caption sources, including Page number.

- **Type:** TeletextSourceSettings (p. 227)
- **Required:** False

**trackSourceSettings**

Settings specific to caption sources that are specified by track number. Sources include IMSC in IMF.

- **Type:** TrackSourceSettings (p. 230)
- **Required:** False

**CaptionSourceType**

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

- ANCILLARY
- DVB_SUB
- EMBEDDED
- SCTE20
ChannelMapping

Channel mapping (ChannelMapping) contains the group of fields that hold the remixing value for each channel. Units are in dB. Acceptable values are within the range from -60 (mute) through 6. A setting of 0 passes the input channel unchanged to the output channel (no attenuation or amplification).

outputChannels

List of output channels

Type: Array of type OutputChannelMapping (p. 215)
Required: False

CmafClientCache

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

DISABLED
ENABLED

CmafCodecSpecification

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

RFC_6381
RFC_4281

CmafEncryptionSettings

Settings for CMAF encryption

encryptionMethod

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

Type: CmafEncryptionType (p. 104)
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.
**Properties**

- **Type**: string  
  **Required**: False  
  **Pattern**: ^[0-9a-fA-F]{32}$  
  **MinLength**: 32  
  **MaxLength**: 32

**initializationVectorInManifest**

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.

- **Type**: CmafInitializationVectorInManifest (p. 107)  
  **Required**: False

**staticKeyProvider**

Use these settings to set up encryption with a static key provider.

- **Type**: StaticKeyProvider (p. 225)  
  **Required**: False

**type**

Indicates which type of key provider is used for encryption.

- **Type**: CmafKeyProviderType (p. 107)  
  **Required**: False

**CmafEncryptionType**

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

- **SAMPLE_AES**

**CmafGroupSettings**

Required when you set (Type) under (OutputGroups)>>(OutputGroupSettings) to CMAF_GROUP_SETTINGS. Each output in a CMAF Output Group may only contain a single video, audio, or caption output.

**writeHlsManifest**

When set to ENABLED, an Apple HLS manifest will be generated for this output.

- **Type**: CmafWriteHLSManifest (p. 108)  
  **Required**: False

**writeDashManifest**

When set to ENABLED, a DASH MPD manifest will be generated for this output.

- **Type**: CmafWriteDASHManifest (p. 108)
**Properties**

**Required**: False

**segmentLength**

Use this setting to specify the length, in seconds, of each individual CMAF segment. This value applies to the whole package; that is, to every output in the output group. Note that segments end on the first keyframe after this number of seconds, so the actual segment length might be slightly longer. If you set Segment control (CmafSegmentControl) to single file, the service puts the content of each output in a single file that has metadata that marks these segments. If you set it to segmented files, the service creates multiple files for each output, each with the content of one segment.

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

**minFinalSegmentLength**

Keep this setting at the default value of 0, unless you are troubleshooting a problem with how devices play back the end of your video asset. If you know that player devices are hanging on the final segment of your video because the length of your final segment is too short, use this setting to specify a minimum final segment length, in seconds. Choose a value that is greater than or equal to 1 and less than your segment length. When you specify a value for this setting, the encoder will combine any final segment that is shorter than the length that you specify with the previous segment. For example, your segment length is 3 seconds and your final segment is .5 seconds without a minimum final segment length; when you set the minimum final segment length to 1, your final segment is 3.5 seconds.

- **Type**: number
- **Required**: False
- **Format**: float
- **Minimum**: 0.0
- **Maximum**: 2147483647

**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**: False
- **Pattern**: `^s3://\/*\/*$/`

**destinationSettings**

Settings associated with the destination. Will vary based on the type of destination

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

**encryption**

DRM settings.
Properties

**Type**: CmafEncryptionSettings (p. 103)
**Required**: False

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

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

**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**: False
**Minimum**: 1
**Maximum**: 2147483647

**baseUrl**
A partial URI prefix that will be put in the manifest 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

**segmentControl**
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.

**Type**: CmafSegmentControl (p. 108)
**Required**: False

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

**Type**: CmafManifestDurationFormat (p. 107)
**Required**: False

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

**Type**: CmafStreamInfResolution (p. 108)
**Required**: False

**clientCache**

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

*Type*: CmafClientCache (p. 103)
*Required*: False

**manifestCompression**

When set to GZIP, compresses HLS playlist.

*Type*: CmafManifestCompression (p. 107)
*Required*: False

**codecSpecification**

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

*Type*: CmafCodecSpecification (p. 103)
*Required*: False

**CmafInitializationVectorInManifest**

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*

**CmafKeyProviderType**

Indicates which type of key provider is used for encryption.

*STATIC_KEY*

**CmafManifestCompression**

When set to GZIP, compresses HLS playlist.

*GZIP*
*NONE*

**CmafManifestDurationFormat**

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

*FLOATING_POINT*
*INTEGER*
CmafSegmentControl

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

CmafStreamInfResolution

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

INCLUDE
EXCLUDE

CmafWriteDASHManifest

When set to ENABLED, a DASH MPD manifest will be generated for this output.

DISABLED
ENABLED

CmafWriteHLSManifest

When set to ENABLED, an Apple HLS manifest will be generated for this output.

DISABLED
ENABLED

ColorCorrector

Settings for color correction.

brightness

Brightness level.

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

colorSpaceConversion

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.

Type: ColorSpaceConversion (p. 110)
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

saturation

Saturation level.

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

hdr10Metadata

Use the HDR master display (Hdr10Metadata) settings to correct HDR metadata or to provide missing metadata. Note that these settings are not color correction.

Type: Hdr10Metadata (p. 157)  
Required: False

ColorMetadata

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

IGNORE  
INSERT

ColorSpace

If your input video has accurate color space metadata, or if you don't know about color space, leave this set to the default value FOLLOW. The service will automatically detect your input color space. If your input video has metadata indicating the wrong color space, or if your input video is missing color space metadata that should be there, specify the accurate color space here. If you choose HDR10, you can also correct inaccurate color space coefficients, using the HDR master display information controls. You must also set Color space usage (ColorSpaceUsage) to FORCE for the service to use these values.

FOLLOW  
REC_601
REC_709
HDR10
HLG_2020

**ColorSpaceConversion**

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**

There are two sources for color metadata, the input file and the job configuration (in the Color space and HDR master display information settings). The Color space usage setting controls which takes precedence. FORCE: The 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: The 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 specific settings.

*container*

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

**Type:** ContainerType (p. 111)
**Required:** False

**m3u8Settings**

Settings for TS segments in HLS

**Type:** M3u8Settings (p. 193)
**Required:** False

**f4vSettings**

Settings for F4v container

**Type:** F4vSettings (p. 132)
**Required**: False

**m2tsSettings**

MPEG-2 TS container settings. These apply to outputs in a File output group when the output's container (ContainerType) is MPEG-2 Transport Stream (M2TS). In these assets, data is organized by the program map table (PMT). Each transport stream program contains subsets of data, including audio, video, and metadata. Each of these subsets of data has a numerical label called a packet identifier (PID). Each transport stream program corresponds to one MediaConvert output. The PMT lists the types of data in a program along with their PID. Downstream systems and players use the program map table to look up the PID for each type of data it accesses and then uses the PIDs to locate specific data within the asset.

**Type**: M2tsSettings (p. 186)
**Required**: False

**movSettings**

Settings for MOV Container.

**Type**: MovSettings (p. 199)
**Required**: False

**mp4Settings**

Settings for MP4 Container

**Type**: Mp4Settings (p. 201)
**Required**: False

**ContainerType**

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

- F4V
- ISMV
- M2TS
- M3U8
- CMFC
- MOV
- MP4
- MPD
- MXF
- RAW

**CreateJobTemplateRequest**

Send your create job template request with the name of the template and the JSON for the template. The template JSON should include everything in a valid job, except for input location and filename, IAM role, and user metadata.

**description**

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

**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

**name**
The name of the job template you are creating.

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

**settings**
JobTemplateSettings contains all the transcode settings saved in the template that will be applied to jobs created from it.

**Type**: JobTemplateSettings (p. 177)  
**Required**: True

**tags**
The tags that you want to add to the resource. You can tag resources with a key-value pair or with only a key.

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

**accelerationSettings**
Accelerated transcoding can significantly speed up jobs with long, visually complex content. Outputs that use this feature incur pro-tier pricing. For information about feature limitations, see the AWS Elemental MediaConvert User Guide.

**Type**: AccelerationSettings (p. 84)  
**Required**: False

**statusUpdateInterval**
Specify how often MediaConvert sends STATUS_UPDATE events to Amazon CloudWatch Events. Set the interval, in seconds, between status updates. MediaConvert sends an update at this interval from
the time the service begins processing your job to the time it completes the transcode or encounters an error.

**Type**: StatusUpdateInterval (p. 226)

**Required**: False

### CreateJobTemplateResponse

Successful create job template requests will return the template JSON.

**jobTemplate**

A job template is a pre-made set of encoding instructions that you can use to quickly create a job.

**Type**: JobTemplate (p. 176)

**Required**: False

### DashIsoEncryptionSettings

Specifies DRM settings for DASH outputs.

**spekeKeyProvider**

Settings for use with a SPEKE key provider

**Type**: SpekeKeyProvider (p. 224)

**Required**: False

### DashIsoGroupSettings

Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to DASH_ISO_GROUP_SETTINGS.

**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**: False

**Minimum**: 1

**Maximum**: 2147483647

**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**: False

**Pattern**: ^s3:/

**destinationSettings**

Settings associated with the destination. Will vary based on the type of destination

**Type:** DestinationSettings (p. 117)
**Required:** False

**encryption**

DRM settings.

**Type:** DashIsoEncryptionSettings (p. 113)
**Required:** False

**minBufferTime**

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

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

**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:** False
**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

**segmentControl**

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.

**Type:** DashIsoSegmentControl (p. 115)
**Required:** False

**hbbtvCompliance**

Supports HbbTV specification as indicated
**writeSegmentTimelineInRepresentation**

When you enable Precise segment duration in manifests (writeSegmentTimelineInRepresentation), your DASH manifest shows precise segment durations. The segment duration information appears inside the SegmentTimeline element, inside SegmentTemplate at the Representation level. When this feature isn't enabled, the segment durations in your DASH manifest are approximate. The segment duration information appears in the duration attribute of the SegmentTemplate element.

**DashIsoHbbtvCompliance**

Supports HbbTV specification as indicated

- HBBTV_1_5
- NONE

**DashIsoSegmentControl**

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

**DashIsoWriteSegmentTimelineInRepresentation**

When you enable Precise segment duration in manifests (writeSegmentTimelineInRepresentation), your DASH manifest shows precise segment durations. The segment duration information appears inside the SegmentTimeline element, inside SegmentTemplate at the Representation level. When this feature isn't enabled, the segment durations in your DASH manifest are approximate. The segment duration information appears in the duration attribute of the SegmentTemplate element.

- ENABLED
- DISABLED

**DeinterlaceAlgorithm**

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

Settings for deinterlacer

**algorithm**

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.

*Type: DeinterlaceAlgorithm (p. 115)*

*Required: False*

**mode**

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.

*Type: DeinterlaceMode (p. 116)*

*Required: False*

**control**

- 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.

*Type: DeinterlacerControl (p. 116)*

*Required: False*

**DeinterlacerControl**

- 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**

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
Properties

INVERSE_TELECINE
ADAPTIVE

DestinationSettings

Settings associated with the destination. Will vary based on the type of destination

s3Settings

Settings associated with S3 destination

Type: S3DestinationSettings (p. 223)
Required: False

DropFrameTimecode

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

Inserts DVB Network Information Table (NIT) at the specified table repetition interval.

nitInterval

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

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

networkId

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

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

networkName

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

Type: string
Required: False
MinLength: 1
MaxLength: 256
**DvbSdtSettings**

Inserts DVB Service Description Table (NIT) at the specified table repetition interval.

**outputSdt**

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.

   Type: OutputSdt (p. 217)
   Required: False

**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
   MinLength: 1
   MaxLength: 256

**serviceProviderName**

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

   Type: string
   Required: False
   MinLength: 1
   MaxLength: 256

**DvbSubDestinationSettings**

DVB-Sub Destination Settings

**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

**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

**teletextSpacing**

Only applies to jobs with input captions in Teletext or STL formats. Specify whether the spacing between letters in your captions is set by the captions grid or varies depending on letter width. Choose fixed grid to conform to the spacing specified in the captions file more accurately. Choose proportional to make the text easier to read if the captions are closed caption.

- **Type**: DvbSubtitleTeletextSpacing (p. 123)  
- **Required**: False

**alignment**

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.

- **Type**: DvbSubtitleAlignment (p. 122)  
- **Required**: False

**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**: False  
- **Minimum**: 0  
- **Maximum**: 10

**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
Properties

**shadowColor**

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

Type: DvbSubtitleShadowColor (p. 123)
Required: False

**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: False
Minimum: 0
Maximum: 255

**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

**fontScript**

Provide the font script, using an ISO 15924 script code, if the LanguageCode is not sufficient for determining the script type. Where LanguageCode or CustomLanguageCode is sufficient, use "AUTOMATIC" or leave unset. This is used to help determine the appropriate font for rendering DVB-Sub captions.

Type: FontScript (p. 133)
Required: False

**fontColor**

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.

Type: DvbSubtitleFontColor (p. 122)
Required: False

**backgroundColor**

Specifies the color of the rectangle behind the captions. All burn-in and DVB-Sub font settings must match.
Type: DvbSubtitleBackgroundColor (p. 122)
Required: False

**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

**outlineColor**

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.

Type: DvbSubtitleOutlineColor (p. 122)
Required: False

**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

**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

**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
DvbSubSourceSettings

DVB Sub Source Settings

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

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

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

- NONE
- BLACK
- WHITE

DvbSubtitleFontColor

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

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.
DvbSubtitleShadowColor

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

- NONE
- BLACK
- WHITE

DvbSubtitleTeletextSpacing

Only applies to jobs with input captions in Teletext or STL formats. Specify whether the spacing between letters in your captions is set by the captions grid or varies depending on letter width. Choose fixed grid to conform to the spacing specified in the captions file more accurately. Choose proportional to make the text easier to read if the captions are closed caption.

- FIXED_GRID
- PROPORTIONAL

DvbTdtSettings

Inserts DVB Time and Date Table (TDT) at the specified table repetition interval.

tdtInterval

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

- Type: integer
- Required: False
- Minimum: 1000
- Maximum: 30000

Eac3AttenuationControl

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

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

COMPLETE_MAIN
COMMENTARY
EMERGENCY
HEARING_IMPAIRED
VISUALLY_IMPAIRED

Eac3CodingMode
Dolby Digital Plus coding mode. Determines number of channels.

CODING_MODE_1_0
CODING_MODE_2_0
CODING_MODE_3_2

Eac3DcFilter
Activates a DC highpass filter for all input channels.

ENABLED
DISABLED

Eac3DynamicRangeCompressionLine
Enables Dynamic Range Compression that restricts the absolute peak level for a signal.

NONE
FILM_STANDARD
FILM_LIGHT
MUSIC_STANDARD
MUSIC_LIGHT
SPEECH

Eac3DynamicRangeCompressionRf
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
When encoding 3/2 audio, controls whether the LFE channel is enabled

LFE
NO_LFE
**Eac3LfeFilter**

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

- ENABLED
- DISABLED

**Eac3MetadataControl**

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**

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**

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

- SHIFT_90_DEGREES
- NO_SHIFT

**Eac3Settings**

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value EAC3.

**metadataControl**

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.

- **Type**: Eac3MetadataControl (p. 125)
- **Required**: False

**surroundExMode**

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

- **Type**: Eac3SurroundExMode (p. 129)
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

**phaseControl**
Controls the amount of phase-shift applied to the surround channels. Only used for 3/2 coding mode.

- **Type**: Eac3PhaseControl (p. 125)
- **Required**: False

**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

**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

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

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

**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
Properties

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

**passthroughControl**

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.

Type: Eac3PassthroughControl (p. 125)
Required: False

**lfeControl**

When encoding 3/2 audio, controls whether the LFE channel is enabled

Type: Eac3LfeControl (p. 124)
Required: False

**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

**attenuationControl**

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

Type: Eac3AttenuationControl (p. 123)
Required: False

**codingMode**

Dolby Digital Plus coding mode. Determines number of channels.

Type: Eac3CodingMode (p. 124)
Required: False

**surroundMode**

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

Type: Eac3SurroundMode (p. 129)
Properties

Required: False

bitstreamMode

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

  Type: Eac3BitstreamMode (p. 123)
  Required: False

lfeFilter

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

  Type: Eac3LfeFilter (p. 125)
  Required: False

stereoDownmix

Stereo downmix preference. Only used for 3/2 coding mode.

  Type: Eac3StereoDownmix (p. 129)
  Required: False

dynamicRangeCompressionRf

Enables Heavy Dynamic Range Compression, ensures that the instantaneous signal peaks do not exceed specified levels.

  Type: Eac3DynamicRangeCompressionRf (p. 124)
  Required: False

sampleRate

Sample rate in hz. Sample rate is always 48000.

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

dynamicRangeCompressionLine

Enables Dynamic Range Compression that restricts the absolute peak level for a signal.

  Type: Eac3DynamicRangeCompressionLine (p. 124)
  Required: False

dcFilter

Activates a DC highpass filter for all input channels.

  Type: Eac3DcFilter (p. 124)
  Required: False
Properties

Eac3StereoDownmix

Stereo downmix preference. Only used for 3/2 coding mode.

- NOT_INDICATED
- LO_RO
- LT_RT
- DPL2

Eac3SurroundExMode

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

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

- NOT_INDICATED
- ENABLED
- DISABLED

EmbeddedConvert608To708

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

EmbeddedDestinationSettings

Settings specific to embedded/ancillary caption outputs, including 608/708 Channel destination number.

destination608ChannelNumber

Ignore this setting unless your input captions are SCC format and your output container is MXF. With this combination of input captions format and output container, you can optionally use this setting to replace the input channel number with the track number that you specify. Specify a different number for each output captions track. If you don't specify an output track number, the system uses the input channel number for the output channel number. This setting applies to each output individually. You can optionally combine two captions channels in your output. The two output channel numbers can be one of the following pairs: 1,3; 2,4; 1,4; or 2,3.

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

Settings for embedded captions Source

**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

**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

**convert608To708**

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.

*Type:* EmbeddedConvert608To708 (p. 129)  
*Required:* False

**EsamManifestConfirmConditionNotification**

ESAM ManifestConfirmConditionNotification defined by OC-SP-ESAM-API-I03-131025.

**mccXml**

Provide your ESAM ManifestConfirmConditionNotification XML document inside your JSON job settings. Form the XML document as per OC-SP-ESAM-API-I03-131025. The transcoder will use the Manifest Conditioning instructions in the message that you supply.

*Type:* string  
*Required:* False  
*Pattern:* `\s*<(.|\n)*ManifestConfirmConditionNotification(.|\n)*\s*>\s*`  

**EsamSettings**

Settings for Event Signaling And Messaging (ESAM). If you don't do ad insertion, you can ignore these settings.

**signalProcessingNotification**

Specifies an ESAM SignalProcessingNotification XML as per OC-SP-ESAM-API-I03-131025. The transcoder uses the signal processing instructions that you provide in the setting SCC XML (sccXml).
**manifestConfirmConditionNotification**

Specifies an ESAM ManifestConfirmConditionNotification XML as per OC-SP-ESAM-API-I03-131025. The transcoder uses the manifest conditioning instructions that you provide in the setting MCC XML (mccXml).

- **Type:** EsamManifestConfirmConditionNotification (p. 130)
- **Required:** False

**responseSignalPreroll**

Specifies the stream distance, in milliseconds, between the SCTE 35 messages that the transcoder places and the splice points that they refer to. If the time between the start of the asset and the SCTE-35 message is less than this value, then the transcoder places the SCTE-35 marker at the beginning of the stream.

- **Type:** integer
  - **Required:** False
  - **Minimum:** 0
  - **Maximum:** 30000

**EsamSignalProcessingNotification**

ESAM SignalProcessingNotification data defined by OC-SP-ESAM-API-I03-131025.

**sccXml**

Provide your ESAM SignalProcessingNotification XML document inside your JSON job settings. Form the XML document as per OC-SP-ESAM-API-I03-131025. The transcoder will use the signal processing instructions in the message that you supply. Provide your ESAM SignalProcessingNotification XML document inside your JSON job settings. If you want the service to place SCTE-35 markers at the insertion points you specify in the XML document, you must also enable SCTE-35 ESAM (scte35Esam). Note that you can either specify an ESAM XML document or enable SCTE-35 passthrough. You can't do both.

- **Type:** string
  - **Required:** False
  - **Pattern:** ^\s*<(.|
)*SignalProcessingNotification(.|
)*>\s*$

**ExceptionBody**

**message**

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

**F4vMoovPlacement**

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.
Properties

**PROGRESSIVE_DOWNLOAD**

NORMAL

### F4vSettings

Settings for F4v container

**moovPlacement**

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.

- **Type:** F4vMoovPlacement (p. 131)
- **Required:** False

### FileGroupSettings

Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to FILE_GROUP_SETTINGS.

**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:** False
- **Pattern:** ^s3://

**destinationSettings**

Settings associated with the destination. Will vary based on the type of destination

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

### FileSourceConvert608To708

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

Settings for File-based Captions in Source

**sourceFile**

External caption file used for loading captions. Accepted file extensions are 'scc', 'ttml', 'dfxp', 'stl', 'srt', and 'smi'.

132
Properties

Type: string
Required: False
Pattern: `^s3:\/(.*)\.(scc|SCC|ttml|TTML|dfxp|DFXP|stl|STL|srt|SRT|smi|SMI)\$`
MinLength: 14

timeDelta

Specifies a time delta in seconds to offset the captions from the source file.

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

convert608To708

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.

Type: FileSourceConvert608To708 (p. 132)
Required: False

FontScript

Provide the font script, using an ISO 15924 script code, if the LanguageCode is not sufficient for determining the script type. Where LanguageCode or CustomLanguageCode is sufficient, use "AUTOMATIC" or leave unset.

AUTOMATIC
HANS
HANT

FrameCaptureSettings

Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value FRAME_CAPTURE.

framerateNumerator

Frame capture will encode the first frame of the output stream, then one frame every framerateDenominator/framerateNumerator seconds. For example, settings of framerateNumerator = 133 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

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

### quality

JPEG Quality - a higher value equals higher quality.

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

### H264AdaptiveQuantization

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

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

### H264CodecLevel

Specify an H.264 level that is consistent with your output video settings. If you aren't sure what level to specify, choose Auto (AUTO).

- AUTO  
- LEVEL_1  
- LEVEL_1_1  
- LEVEL_1_2  
- LEVEL_1_3  
- LEVEL_2  
- LEVEL_2_1  
- LEVEL_2_2  
- LEVEL_3
H264CodecProfile

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

H264DynamicSubGop

Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting B frames between reference frames (numberBFramesBetweenReferenceFrames).

ADAPTIVE
STATIC

H264EntropyEncoding

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

CABAC
CAVLC

H264FieldEncoding

Choosing FORCE_FIELD disables PAFF encoding for interlaced outputs.

PAFF
FORCE_FIELD

H264FlickerAdaptiveQuantization

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

DISABLED
ENABLED
H264FramerateControl

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

INITIALIZE_FROM_SOURCE
SPECIFIED

H264FramerateConversionAlgorithm

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

DUPLICATE_DROP
INTERPOLATE

H264GopBReference

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

DISABLED
ENABLED

H264GopSizeUnits

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

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, as follows. - 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**

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**

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

**H264QvbrSettings**

Settings for quality-defined variable bitrate encoding with the H.264 codec. Required when you set Rate control mode to QVBR. Not valid when you set Rate control mode to a value other than QVBR, or when you don’t define Rate control mode.

**qvbrQualityLevel**

Required when you use QVBR rate control mode. That is, when you specify qvbrSettings within h264Settings. Specify the target quality level for this output, from 1 to 10. Use higher numbers for greater quality. Level 10 results in nearly lossless compression. The quality level for most broadcast-quality transcodes is between 6 and 9.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 10

**maxAverageBitrate**

Use this setting only when Rate control mode is QVBR and Quality tuning level is Multi-pass HQ. For Max average bitrate values suited to the complexity of your input video, the service limits the average bitrate of the video part of this output to the value you choose. That is, the total size of the video element is less than or equal to the value you set multiplied by the number of seconds of encoded output.

- **Type**: integer
- **Required**: False
- **Minimum**: 1000
- **Maximum**: 1152000000

**H264RateControlMode**

Use this setting to specify whether this output has a variable bitrate (VBR), constant bitrate (CBR) or quality-defined variable bitrate (QVBR).

- VBR
CBR
QVBR

H264RepeatPps
Places a PPS header on each encoded picture, even if repeated.

DISABLED
ENABLED

H264SceneChangeDetect
Scene change detection (inserts I-frames on scene changes).

DISABLED
ENABLED

H264Settings
Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value H_264.

interlaceMode
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, as follows. - 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.

Type: H264InterlaceMode (p. 136)
Required: False

parNumerator
Pixel Aspect Ratio numerator.

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

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
**syntax**

Produces a bitstream compliant with SMPTE RP-2027.

- **Type:** H264Syntax (p. 145)
- **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

**framerateDenominator**

When you use the API for transcode jobs that use frame rate conversion, specify the frame rate 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 frame rate conversion, provide the value as a decimal number for Framerate. In this example, specify 23.976.

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

**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

**hrdBufferInitialFillPercentage**

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

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

**gopSize**

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

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

**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

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

**Type**: H264GopBReference (p. 136)
**Required**: False

**hrdBufferSize**
Size of buffer (HRD buffer model) in bits. For example, enter five megabits as 5000000.

**Type**: integer
**Required**: False
**Minimum**: 0
**Maximum**: 1152000000

**maxBitrate**
Maximum bitrate in bits/second. For example, enter five megabits per second as 5000000. Required when Rate control mode is QVBR.

**Type**: integer
**Required**: False
**Minimum**: 1000
**Maximum**: 1152000000

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

**Type**: H264SlowPal (p. 144)
**Required**: False

**parDenominator**
Pixel Aspect Ratio denominator.

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

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

Type: H264SpatialAdaptiveQuantization (p. 144)
Required: False

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

Type: H264TemporalAdaptiveQuantization (p. 145)
Required: False

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

Type: H264FlickerAdaptiveQuantization (p. 135)
Required: False

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

Type: H264EntropyEncoding (p. 135)
Required: False

**bitrate**
Average bitrate in bits/second. Required for VBR and CBR. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.

Type: integer
Required: False
Minimum: 1000
Maximum: 1152000000

**framerateControl**
If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

Type: H264FramerateControl (p. 136)
Required: False

rateControlMode
Use this setting to specify whether this output has a variable bitrate (VBR), constant bitrate (CBR) or quality-defined variable bitrate (QVBR).

Type: H264RateControlMode (p. 137)
Required: False

qvbrSettings
Settings for quality-defined variable bitrate encoding with the H.264 codec. Required when you set Rate control mode to QVBR. Not valid when you set Rate control mode to a value other than QVBR, or when you don’t define Rate control mode.

Type: H264QvbrSettings (p. 137)
Required: False

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

Type: H264CodecProfile (p. 135)
Required: False

telecine
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.

Type: H264Telecine (p. 145)
Required: False

framerateNumerator
Frame rate numerator - frame rate is a fraction, e.g. 24000 / 1001 = 23.976 fps.

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

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
Properties

Required: False
Minimum: 0
Maximum: 30

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

Type: H264AdaptiveQuantization (p. 134)
Required: False

codecLevel
Specify an H.264 level that is consistent with your output video settings. If you aren't sure what level to specify, choose Auto (AUTO).

Type: H264CodecLevel (p. 134)
Required: False

fieldEncoding
Choosing FORCE_FIELD disables PAFF encoding for interlaced outputs.

Type: H264FieldEncoding (p. 135)
Required: False

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

Type: H264SceneChangeDetect (p. 138)
Required: False

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

Type: H264QualityTuningLevel (p. 137)
Required: False

framerateConversionAlgorithm
When set to INTERPOLATE, produces smoother motion during frame rate conversion.

Type: H264FramerateConversionAlgorithm (p. 136)
Required: False

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

Type: H264UnregisteredSeiTimecode (p. 145)
Required: False
**gopSizeUnits**
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.

- **Type:** H264GopSizeUnits (p. 136)
- **Required:** False

**parControl**
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.

- **Type:** H264ParControl (p. 137)
- **Required:** False

**numberBFramesBetweenReferenceFrames**
Number of B-frames between reference frames.

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

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

- **Type:** H264RepeatPps (p. 138)
- **Required:** False

**dynamicSubGop**
Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting B frames between reference frames (numberBFramesBetweenReferenceFrames).

- **Type:** H264DynamicSubGop (p. 135)
- **Required:** False

**H264SlowPal**
Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

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

**H264SpatialAdaptiveQuantization**
Adjust quantization within each frame based on spatial variation of content complexity.
**H264Syntax**

Produces a bitstream compliant with SMPTE RP-2027.

- DEFAULT
- RP2027

**H264Telecine**

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**

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

- DISABLED
- ENABLED

**H264UnregisteredSeiTimecode**

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

- DISABLED
- ENABLED

**H265AdaptiveQuantization**

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

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

**H265AlternateTransferFunctionSei**

Enables Alternate Transfer Function SEI message for outputs using Hybrid Log Gamma (HLG) Electro-Optical Transfer Function (EOTF).
H265CodecLevel

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

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

H265DynamicSubGop

Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting B frames between reference frames (numberBFramesBetweenReferenceFrames).

- ADAPTIVE
- STATIC

H265FlickerAdaptiveQuantization

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

- DISABLED
- ENABLED
H265FramerateControl

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

  INITIALIZE_FROM_SOURCE
  SPECIFIED

H265FramerateConversionAlgorithm

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

  DUPLICATE_DROP
  INTERPOLATE

H265GopBReference

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

  DISABLED
  ENABLED

H265GopSizeUnits

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

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
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
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

H265QvbrSettings
Settings for quality-defined variable bitrate encoding with the H.265 codec. Required when you set Rate control mode to QVBR. Not valid when you set Rate control mode to a value other than QVBR, or when you don’t define Rate control mode.

qvbrQualityLevel
Required when you use QVBR rate control mode. That is, when you specify qvbrSettings within h265Settings. Specify the target quality level for this output, from 1 to 10. Use higher numbers for greater quality. Level 10 results in nearly lossless compression. The quality level for most broadcast-quality transcodes is between 6 and 9.

- Type: integer
- Required: False
- Minimum: 1
- Maximum: 10

maxAverageBitrate
Use this setting only when Rate control mode is QVBR and Quality tuning level is Multi-pass HQ. For Max average bitrate values suited to the complexity of your input video, the service limits the average bitrate of the video part of this output to the value you choose. That is, the total size of the video element is less than or equal to the value you set multiplied by the number of seconds of encoded output.

- Type: integer
- Required: False
- Minimum: 1000
- Maximum: 1466400000

H265RateControlMode
Use this setting to specify whether this output has a variable bitrate (VBR), constant bitrate (CBR) or quality-defined variable bitrate (QVBR).

- VBR
CBR
QVBR

**H265SampleAdaptiveOffsetFilterMode**

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

- DEFAULT
- ADAPTIVE
- OFF

**H265SceneChangeDetect**

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

- DISABLED
- ENABLED

**H265Settings**

Settings for H265 codec

**interlaceMode**

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.

- **Type:** H265InterlaceMode (p. 147)
- **Required:** False

**parNumerator**

Pixel Aspect Ratio numerator.

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

**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
Properties

Minimum: 1
Maximum: 6

framerateDenominator

Frame rate denominator.

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

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

alternateTransferFunctionSei

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

Type: H265AlternateTransferFunctionSei (p. 145)
Required: False

hrdBufferInitialFillPercentage

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

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

gopSize

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

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

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

gopBReference

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

Type: H265GopBReference (p. 147)  
Required: False

hrdBufferSize

Size of buffer (HRD buffer model) in bits. For example, enter five megabits as 5000000.

Type: integer  
Required: False  
Minimum: 0  
Maximum: 1466400000

maxBitrate

Maximum bitrate in bits/second. For example, enter five megabits per second as 5000000. Required when Rate control mode is QVBR.

Type: integer  
Required: False  
Minimum: 1000  
Maximum: 1466400000

slowPal

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

Type: H265SlowPal (p. 155)  
Required: False

parDenominator

Pixel Aspect Ratio denominator.

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

spatialAdaptiveQuantization

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

Type: H265SpatialAdaptiveQuantization (p. 156)  
Required: False
temporalAdaptiveQuantization

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

Type: H265TemporalAdaptiveQuantization (p. 156)
Required: False

flickerAdaptiveQuantization

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

Type: H265FlickerAdaptiveQuantization (p. 146)
Required: False

bitrate

Average bitrate in bits/second. Required for VBR and CBR. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.

Type: integer
Required: False
Minimum: 1000
Maximum: 1466400000

framerateControl

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

Type: H265FramerateControl (p. 147)
Required: False

rateControlMode

Use this setting to specify whether this output has a variable bitrate (VBR), constant bitrate (CBR) or quality-defined variable bitrate (QVBR).

Type: H265RateControlMode (p. 148)
Required: False

qvbrSettings

Settings for quality-defined variable bitrate encoding with the H.265 codec. Required when you set Rate control mode to QVBR. Not valid when you set Rate control mode to a value other than QVBR, or when you don't define Rate control mode.

Type: H265QvbrSettings (p. 148)
Required: False
**codecProfile**

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.

- **Type:** H265CodecProfile (p. 146)
- **Required:** False

**tiles**

Enable use of tiles, allowing horizontal as well as vertical subdivision of the encoded pictures.

- **Type:** H265Tiles (p. 156)
- **Required:** False

**telecine**

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.

- **Type:** H265Telecine (p. 156)
- **Required:** False

**framerateNumerator**

Frame rate numerator - frame rate is a fraction, e.g. 24000 / 1001 = 23.976 fps.

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

**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

**adaptiveQuantization**

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

- **Type:** H265AdaptiveQuantization (p. 145)
- **Required:** False
**codecLevel**

H.265 Level.

*Type:* H265CodecLevel (p. 146)
*Required:* False

**sceneChangeDetect**

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

*Type:* H265SceneChangeDetect (p. 149)
*Required:* False

**qualityTuningLevel**

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

*Type:* H265QualityTuningLevel (p. 148)
*Required:* False

**framerateConversionAlgorithm**

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

*Type:* H265FramerateConversionAlgorithm (p. 147)
*Required:* False

**unregisteredSeiTimecode**

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

*Type:* H265UnregisteredSeiTimecode (p. 156)
*Required:* False

**gopSizeUnits**

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.

*Type:* H265GopSizeUnits (p. 147)
*Required:* False

**parControl**

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.

*Type:* H265ParControl (p. 148)
*Required:* False

**numberBFramesBetweenReferenceFrames**

Number of B-frames between reference frames.
**temporalIds**

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.

- **Type:** `H265TemporalIds (p. 156)`
- **Required:** False

**sampleAdaptiveOffsetFilterMode**

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

- **Type:** `H265SampleAdaptiveOffsetFilterMode (p. 149)`
- **Required:** False

**writeMp4PackagingType**

Use this setting only for outputs encoded with H.265 that are in CMAF or DASH output groups. If you include `writeMp4PackagingType` in your JSON job specification for other outputs, your video might not work properly with downstream systems and video players. If the location of parameter set NAL units don't matter in your workflow, ignore this setting. The service defaults to marking your output as HEV1. Choose HVC1 to mark your output as HVC1. This makes your output compliant with this specification: ISO IECJTC1 SC29 N13798 Text ISO/IEC FDIS 14496-15 3rd Edition. For these outputs, the service stores parameter set NAL units in the sample headers but not in the samples directly. Keep the default HEV1 to mark your output as HEV1. For these outputs, the service writes parameter set NAL units directly into the samples.

- **Type:** `H265WriteMp4PackagingType (p. 157)`
- **Required:** False

**dynamicSubGop**

Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting `B frames between reference frames`.

- **Type:** `H265DynamicSubGop (p. 146)`
- **Required:** False

**H265SlowPal**

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

DISABLED
ENABLED

H265SpatialAdaptiveQuantization
Adjust quantization within each frame based on spatial variation of content complexity.

DISABLED
ENABLED

H265Telecine
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
Adjust quantization within each frame based on temporal variation of content complexity.

DISABLED
ENABLED

H265TemporalIds
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. lbPbPb 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
Enable use of tiles, allowing horizontal as well as vertical subdivision of the encoded pictures.

DISABLED
ENABLED

H265UnregisteredSeiTimecode
Inserts timecode for each frame as 4 bytes of an unregistered SEI message.
DISABLED
ENABLED

H265WriteMp4PackagingType

Use this setting only for outputs encoded with H.265 that are in CMAF or DASH output groups. If you include writeMp4PackagingType in your JSON job specification for other outputs, your video might not work properly with downstream systems and video players. If the location of parameter set NAL units don't matter in your workflow, ignore this setting. The service defaults to marking your output as HEV1. Choose HVC1 to mark your output as HVC1. This makes your output compliant with this specification: ISO IECJTC1 SC29 N13798 Text ISO/IEC FDIS 14496-15 3rd Edition. For these outputs, the service stores parameter set NAL units in the sample headers but not in the samples directly. Keep the default HEV1 to mark your output as HEV1. For these outputs, the service writes parameter set NAL units directly into the samples.

HVC1
HEV1

Hdr10Metadata

Use the "HDR master display information" (Hdr10Metadata) settings to correct HDR metadata or to provide missing metadata. These values vary depending on the input video and must be provided by a color grader. Range is 0 to 50,000; each increment represents 0.00002 in CIE1931 color coordinate. Note that these settings are not color correction. Note that if you are creating HDR outputs inside of an HLS CMAF package, to comply with the Apple specification, you must use the following settings. Set "MP4 packaging type" (writeMp4PackagingType) to HVC1 (HVC1). Set "Profile" (H265Settings > codecProfile) to Main10/High (MAIN10_HIGH). Set "Level" (H265Settings > codecLevel) to 5 (LEVEL_5).

redPrimaryX

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

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

redPrimaryY

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

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

greenPrimaryX

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.
Type: integer  
Required: False  
Minimum: 0  
Maximum: 50000

greenPrimaryY

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

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

bluePrimaryX

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

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

bluePrimaryY

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

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

whitePointX

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

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

whitePointY

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.
**maxFrameAverageLightLevel**

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

<table>
<thead>
<tr>
<th>Type</th>
<th>integer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Required</td>
<td>False</td>
</tr>
<tr>
<td>Minimum</td>
<td>0</td>
</tr>
<tr>
<td>Maximum</td>
<td>50000</td>
</tr>
</tbody>
</table>

**maxContentLightLevel**

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

<table>
<thead>
<tr>
<th>Type</th>
<th>integer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Required</td>
<td>False</td>
</tr>
<tr>
<td>Minimum</td>
<td>0</td>
</tr>
<tr>
<td>Maximum</td>
<td>65535</td>
</tr>
</tbody>
</table>

**maxLuminance**

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

<table>
<thead>
<tr>
<th>Type</th>
<th>integer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Required</td>
<td>False</td>
</tr>
<tr>
<td>Minimum</td>
<td>0</td>
</tr>
<tr>
<td>Maximum</td>
<td>2147483647</td>
</tr>
</tbody>
</table>

**minLuminance**

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

<table>
<thead>
<tr>
<th>Type</th>
<th>integer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Required</td>
<td>False</td>
</tr>
<tr>
<td>Minimum</td>
<td>0</td>
</tr>
<tr>
<td>Maximum</td>
<td>2147483647</td>
</tr>
</tbody>
</table>

**HlsAdMarkers**

<table>
<thead>
<tr>
<th>ELEMENTAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>ELEMENTAL_SCTE35</td>
</tr>
</tbody>
</table>

**HlsAudioTrackType**

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

Caption Language Mapping

captionChannel

Caption channel.

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

customLanguageCode

Specify the language for this caption channel, using the ISO 639-2 or ISO 639-3 three-letter language code

Type: string
Required: False
Pattern: ^[A-Za-z]{3}$
MinLength: 3
MaxLength: 3

languageCode


Type: LanguageCode (p. 179)
Required: False

languageDescription

Caption language description.

Type: string
Required: False

HlsCaptionLanguageSetting

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**

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

DISABLED
ENABLED

**HlsCodecSpecification**

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

RFC_6381
RFC_4281

**HlsDirectoryStructure**

Indicates whether segments should be placed in subdirectories.

SINGLE_DIRECTORY
SUBDIRECTORY_PER_STREAM

**HlsEncryptionSettings**

Settings for HLS encryption

**encryptionMethod**

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

*Type*: HlsEncryptionType (p. 162)
*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}\$
*MinLength*: 32
*MaxLength*: 32
initializationVectorInManifest

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.

  Type: HlsInitializationVectorInManifest (p. 166)
  Required: False

offlineEncrypted

Enable this setting to insert the EXT-X-SESSION-KEY element into the master playlist. This allows for offline Apple HLS FairPlay content protection.

  Type: HlsOfflineEncrypted (p. 167)
  Required: False

spekeKeyProvider

Settings for use with a SPEKE key provider

  Type: SpekeKeyProvider (p. 224)
  Required: False

staticKeyProvider

Use these settings to set up encryption with a static key provider.

  Type: StaticKeyProvider (p. 225)
  Required: False

type

Indicates which type of key provider is used for encryption.

  Type: HlsKeyProviderType (p. 167)
  Required: False

HlsEncryptionType

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

Required when you set (Type) under (OutputGroups)->(OutputGroupSettings) to HLS_GROUP_SETTINGS.

manifestDurationFormat

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

  Type: HlsManifestDurationFormat (p. 167)
**Properties**

**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:** False
- **Minimum:** 1
- **Maximum:** 2147483647

**timedMetadataId3Period**
Timed Metadata interval in seconds.

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

**captionLanguageSetting**
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.

- **Type:** HlsCaptionLanguageSetting (p. 160)
- **Required:** False

**captionLanguageMappings**
Language to be used on Caption outputs

- **Type:** Array of type HlsCaptionLanguageMapping (p. 160)
- **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:** False
- **Pattern:** ^s3://\/

**destinationSettings**
Settings associated with the destination. Will vary based on the type of destination
Type: DestinationSettings (p. 117)
Required: False

encryption

DRM settings.

Type: HlsEncryptionSettings (p. 161)
Required: False

timedMetadataId3Frame

Indicates ID3 frame that has the timecode.

Type: HlsTimedMetadataId3Frame (p. 169)
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

codecSpecification

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

Type: HlsCodecSpecification (p. 161)
Required: False

outputSelection

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

Type: HlsOutputSelection (p. 167)
Required: False

programDateTimePeriod

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

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

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
**Properties**

**Required**: False

**Minimum**: 1

**Maximum**: 2147483647

**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**: False

**Minimum**: 0

**Maximum**: 2147483647

**minFinalSegmentLength**

Keep this setting at the default value of 0, unless you are troubleshooting a problem with how devices play back the end of your video asset. If you know that player devices are hanging on the final segment of your video because the length of your final segment is too short, use this setting to specify a minimum final segment length, in seconds. Choose a value that is greater than or equal to 1 and less than your segment length. When you specify a value for this setting, the encoder will combine any final segment that is shorter than the length that you specify with the previous segment. For example, your segment length is 3 seconds and your final segment is .5 seconds without a minimum final segment length; when you set the minimum final segment length to 1, your final segment is 3.5 seconds.

**Type**: number

**Required**: False

**Format**: float

**Minimum**: 0.0

**Maximum**: 2147483647

**directoryStructure**

Indicates whether segments should be placed in subdirectories.

**Type**: HlsDirectoryStructure (p. 161)

**Required**: False

**programDateTime**

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.

**Type**: HlsProgramDateTime (p. 167)

**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. 159)

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

- **Type**: HlsSegmentControl (p. 167)
- **Required**: False

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

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

**manifestCompression**
When set to GZIP, compresses HLS playlist.

- **Type**: HlsManifestCompression (p. 167)
- **Required**: False

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

- **Type**: HlsClientCache (p. 161)
- **Required**: False

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

- **Type**: HlsStreamInfResolution (p. 168)
- **Required**: False

**HlsIFrameOnlyManifest**
When set to INCLUDE, writes I-Frame Only Manifest in addition to the HLS manifest

- INCLUDE
- EXCLUDE

**HlsInitializationVectorInManifest**
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
Indicates which type of key provider is used for encryption.

SPEKE
STATIC_KEY

HlsManifestCompression
When set to GZIP, compresses HLS playlist.

GZIP
NONE

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

FLOATING_POINT
INTEGER

HlsOfflineEncrypted
Enable this setting to insert the EXT-X-SESSION-KEY element into the master playlist. This allows for offline Apple HLS FairPlay content protection.

ENABLED
DISABLED

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

MANIFESTS_AND_SEGMENTS
SEGMENTS_ONLY

HlsProgramDateTime
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
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

Settings for HLS output groups

audioGroupId

Specifies the group to which the audio Rendition belongs.

- **Type**: string
- **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

audioTrackType

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

- **Type**: HlsAudioTrackType (p. 159)
- **Required**: False

iFrameOnlyManifest

When set to INCLUDE, writes I-Frame Only Manifest in addition to the HLS manifest

- **Type**: HlsIFrameOnlyManifest (p. 166)
- **Required**: False

segmentModifier

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

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

HlsStreamInfResolution

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

- INCLUDE
- EXCLUDE
**HlsTimedMetadataId3Frame**

Indicates ID3 frame that has the timecode.

- NONE
- PRIV
- TDRL

**Id3Insertion**

To insert ID3 tags in your output, specify two values. Use ID3 tag (Id3) to specify the base 64 encoded string and use Timecode (TimeCode) to specify the time when the tag should be inserted. To insert multiple ID3 tags in your output, create multiple instances of ID3 insertion (Id3Insertion).

**timecode**

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

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

**id3**

Use ID3 tag (Id3) to provide a tag value in base64-encode format.

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

**ImageInserter**

Enable the image inserter feature to include a graphic overlay on your video. Enable or disable this feature for each input or output individually. This setting is disabled by default.

**insertableImages**

Specify the images that you want to overlay on your video. The images must be PNG or TGA files.

- **Type:** Array of type InsertableImage (p. 173)
- **Required:** False

**InputClipping**

To transcode only portions of your input (clips), include one Input clipping (one instance of InputClipping in the JSON job file) for each input clip. All input clips you specify will be included in every output of the job.

**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. Use the format HH:MM:SS:FF or HH:MM:SS;FF, where HH is the hour, MM is the minute, SS is the second, and FF is the frame number. When choosing this value, take into account your setting for timecode source under input settings (InputTimecodeSource). For example, if you have embedded timecodes that start at 01:00:00:00 and you want your clip to end six minutes into the video, use 01:06:00:00.

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

**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. Use the format HH:MM:SS:FF or HH:MM:SS;FF, where HH is the hour, MM is the minute, SS is the second, and FF is the frame number. 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 01:05:00:00.

- **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**

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**

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**

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

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

InputRotate

Use Rotate (InputRotate) to specify how the service rotates your video. You can choose automatic rotation or specify a rotation. You can specify a clockwise rotation of 0, 90, 180, or 270 degrees. If your input video container is .mov or .mp4 and your input has rotation metadata, you can choose Automatic to have the service rotate your video according to the rotation specified in the metadata. The rotation must be within one degree of 90, 180, or 270 degrees. If the rotation metadata specifies any other rotation, the service will default to no rotation. By default, the service does no rotation, even if your input video has rotation metadata. The service doesn't pass through rotation metadata.

- DEGREE_0
- DEGREES_90
- DEGREES_180
- DEGREES_270
- AUTO

InputTemplate

Specified video input in a template.

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. 169)
- Required: False

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
**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**: 1
- **Maximum**: 2147483647

**videoSelector**

Selector for video.

- **Type**: VideoSelector (p. 235)
- **Required**: False

**filterEnable**

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.

- **Type**: InputFilterEnable (p. 170)
- **Required**: False

**psiControl**

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.

- **Type**: InputPsiControl (p. 171)
- **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

**deblockFilter**

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

- **Type**: InputDeblockFilter (p. 170)
Required: False

denoiseFilter

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

Type: InputDenoiseFilter (p. 170)
Required: False

timecodeSource

Timecode source under input settings (InputTimecodeSource) only affects the behavior of features that apply to a single input at a time, such as input clipping and synchronizing some captions formats. Use this setting to specify whether the service counts frames by timecodes embedded in the video (EMBEDDED) or by starting the first frame at zero (ZEROBASED). In both cases, the timecode format is HH:MM:SS:FF or HH:MM:SS;FF, where FF is the frame number. Only set this to EMBEDDED if your source video has embedded timecodes.

Type: InputTimecodeSource (p. 173)
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 mutliple captions selectors per input.

Type: object
Required: False

imageInserter

Enable the image inserter feature to include a graphic overlay on your video. Enable or disable this feature for each input individually. This setting is disabled by default.

Type: ImageInserter (p. 169)
Required: False

InputTimecodeSource

Timecode source under input settings (InputTimecodeSource) only affects the behavior of features that apply to a single input at a time, such as input clipping and synchronizing some captions formats. Use this setting to specify whether the service counts frames by timecodes embedded in the video (EMBEDDED) or by starting the first frame at zero (ZEROBASED). In both cases, the timecode format is HH:MM:SS:FF or HH:MM:SS;FF, where FF is the frame number. Only set this to EMBEDDED if your source video has embedded timecodes.

EMBEDDED
ZEROBASED
SPECIFIEDSTART

InsertableImage

Settings that specify how your still graphic overlay appears.
width

Specify the width of the inserted image in pixels. If you specify a value that's larger than the video resolution width, the service will crop your overlaid image to fit. To use the native width of the image, keep this setting blank.

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

height

Specify the height of the inserted image in pixels. If you specify a value that's larger than the video resolution height, the service will crop your overlaid image to fit. To use the native height of the image, keep this setting blank.

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

imageX

Specify the distance, in pixels, between the inserted image and the left edge of the video frame. Required for any image overlay that you specify.

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

imageY

Specify the distance, in pixels, between the overlaid image and the top edge of the video frame. Required for any image overlay that you specify.

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

duration

Specify the time, in milliseconds, for the image to remain on the output video. This duration includes fade-in time but not fade-out time.

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

fadeIn

Specify the length of time, in milliseconds, between the Start time that you specify for the image insertion and the time that the image appears at full opacity. Full opacity is the level that you specify for
the opacity setting. If you don't specify a value for Fade-in, the image will appear abruptly at the overlay start time.

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

Layer
Specify how overlapping inserted images appear. Images with higher values for Layer appear on top of images with lower values for Layer.

Type: integer
Required: False
Minimum: 0
Maximum: 99

ImageInserterInput
Specify the Amazon S3 location of the image that you want to overlay on the video. Use a PNG or TGA file.

Type: string
Required: False
Pattern: ^s3:\/\/(.*)\.(bmp|BMP|png|PNG|tga|TGA)$
MinLength: 14

StartTime
Specify the timecode of the frame that you want the overlay to first appear on. This must be in timecode (HH:MM:SS:FF or HH:MM:SS;FF) format. Remember to take into account your timecode source settings.

Type: string
Required: False
Pattern: ^(((\[0-1]\d)|(2\[0-3]))(:\[0-5]\d{2}(:;)[0-5]\d{2}))$

fadeOut
Specify the length of time, in milliseconds, between the end of the time that you have specified for the image overlay Duration and when the overlaid image has faded to total transparency. 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: 0
Maximum: 2147483647

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: False
Minimum: 0
Maximum: 100

JobTemplate
A job template is a pre-made set of encoding instructions that you can use to quickly create a job.

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

    Type: string
    Required: False

createdAt
The timestamp in epoch seconds for Job template creation.

    Type: string
    Required: False
    Format: date-time

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

    Type: string
    Required: False
    Format: date-time

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

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

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

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

**type**

A job template can be of two types: system or custom. System or built-in job templates can't be modified or deleted by the user.

- **Type**: Type (p. 230)
- **Required**: False

**settings**

JobTemplateSettings contains all the transcode settings saved in the template that will be applied to jobs created from it.

- **Type**: JobTemplateSettings (p. 177)
- **Required**: True

**accelerationSettings**

Accelerated transcoding is currently in private preview. Contact AWS for more information.

- **Type**: AccelerationSettings (p. 84)
- **Required**: False

**statusUpdateInterval**

Specify how often MediaConvert sends STATUS_UPDATE events to Amazon CloudWatch Events. Set the interval, in seconds, between status updates. MediaConvert sends an update at this interval from the time the service begins processing your job to the time it completes the transcode or encounters an error.

- **Type**: StatusUpdateInterval (p. 226)
- **Required**: False

**JobTemplateListBy**

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**

JobTemplateSettings contains all the transcode settings saved in the template that will be applied to jobs created from it.

**timecodeConfig**

Contains settings used to acquire and adjust timecode information from inputs.
**Type**: TimecodeConfig (p. 228)
**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
- CMAF_GROUP_SETTINGS, CmafGroupSettings

**Type**: Array of type OutputGroup (p. 215)
**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

**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. 94)
**Required**: False

**timedMetadataInsertion**

Enable Timed metadata insertion (TimedMetadataInsertion) to include ID3 tags in your job. To include timed metadata, you must enable it here, enable it in each output container, and specify tags and timecodes in ID3 insertion (Id3Insertion) objects.

**Type**: TimedMetadataInsertion (p. 229)
**Required**: False

**nielsenConfiguration**

Settings for Nielsen Configuration

**Type**: NielsenConfiguration (p. 211)
**Required**: False

**motionImageInserter**

Overlay motion graphics on top of your video. The motion graphics that you specify here appear on all outputs in all output groups.
**Properties**

- **Type**: MotionImageInserter (p. 196)
  
  **Required**: False

**esam**

Settings for Event Signaling And Messaging (ESAM).

- **Type**: EsamSettings (p. 130)
  
  **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. 171)
  
  **Required**: False

**LanguageCode**


- 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
<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<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>QUE</td>
<td></td>
</tr>
<tr>
<td>QAA</td>
<td></td>
</tr>
<tr>
<td>RON</td>
<td></td>
</tr>
<tr>
<td>ROH</td>
<td></td>
</tr>
<tr>
<td>RUN</td>
<td></td>
</tr>
<tr>
<td>SMO</td>
<td></td>
</tr>
<tr>
<td>SAG</td>
<td></td>
</tr>
<tr>
<td>SAN</td>
<td></td>
</tr>
<tr>
<td>SRD</td>
<td></td>
</tr>
<tr>
<td>SRB</td>
<td></td>
</tr>
<tr>
<td>SNA</td>
<td></td>
</tr>
<tr>
<td>III</td>
<td></td>
</tr>
<tr>
<td>SND</td>
<td></td>
</tr>
<tr>
<td>SIN</td>
<td></td>
</tr>
<tr>
<td>SLK</td>
<td></td>
</tr>
<tr>
<td>SLV</td>
<td></td>
</tr>
<tr>
<td>SOM</td>
<td></td>
</tr>
<tr>
<td>SOT</td>
<td></td>
</tr>
<tr>
<td>SUN</td>
<td></td>
</tr>
<tr>
<td>SWA</td>
<td></td>
</tr>
<tr>
<td>SWE</td>
<td></td>
</tr>
<tr>
<td>TGL</td>
<td></td>
</tr>
<tr>
<td>TAH</td>
<td></td>
</tr>
<tr>
<td>TGK</td>
<td></td>
</tr>
<tr>
<td>TAM</td>
<td></td>
</tr>
<tr>
<td>TAT</td>
<td></td>
</tr>
<tr>
<td>TEL</td>
<td></td>
</tr>
<tr>
<td>THA</td>
<td></td>
</tr>
<tr>
<td>BOD</td>
<td></td>
</tr>
<tr>
<td>TIR</td>
<td></td>
</tr>
<tr>
<td>TON</td>
<td></td>
</tr>
<tr>
<td>TSO</td>
<td></td>
</tr>
<tr>
<td>TSN</td>
<td></td>
</tr>
<tr>
<td>TUR</td>
<td></td>
</tr>
<tr>
<td>TUK</td>
<td></td>
</tr>
<tr>
<td>TWI</td>
<td></td>
</tr>
<tr>
<td>UIG</td>
<td></td>
</tr>
<tr>
<td>UKR</td>
<td></td>
</tr>
<tr>
<td>UZB</td>
<td></td>
</tr>
<tr>
<td>VEN</td>
<td></td>
</tr>
<tr>
<td>VOL</td>
<td></td>
</tr>
<tr>
<td>WLN</td>
<td></td>
</tr>
<tr>
<td>CYM</td>
<td></td>
</tr>
<tr>
<td>FRY</td>
<td></td>
</tr>
<tr>
<td>WOL</td>
<td></td>
</tr>
<tr>
<td>XHO</td>
<td></td>
</tr>
<tr>
<td>YID</td>
<td></td>
</tr>
</tbody>
</table>
You can send list job templates requests with an empty body. Optionally, you can filter the response by category by specifying it in your request body. You can also optionally specify the maximum number, up to twenty, of job templates to be returned.

**listBy**

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.

**category**

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

**order**

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

**nextToken**

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

**maxResults**

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

Successful list job templates requests return a JSON array of job templates. If you don't specify how they are ordered, you will receive them in alphabetical order by name.

jobTemplates

List of Job templates.

- **Type**: Array of type [JobTemplate](#)
- **Required**: False

nextToken

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

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

M2tsAudioBufferModel

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

- DVB
- ATSC

M2tsBufferModel

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

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

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

M2tsForceTsVideoEbpOrder
Keep the default value (DEFAULT) unless you know that your audio EBP markers are incorrectly appearing before your video EBP markers. To correct this problem, set this value to Force (FORCE).
   FORCE
   DEFAULT

M2tsNielsenId3
If INSERT, Nielsen inaudible tones for media tracking will be detected in the input audio and an equivalent ID3 tag will be inserted in the output.
   INSERT
   NONE

M2tsPcrControl
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
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

M2tsScte35Esam
Settings for SCTE-35 signals from ESAM. Include this in your job settings to put SCTE-35 markers in your HLS and transport stream outputs at the insertion points that you specify in an ESAM XML document. Provide the document in the setting SCC XML (sccXml).

scte35EsamPid
Packet Identifier (PID) of the SCTE-35 stream in the transport stream generated by ESAM.
**Type**: integer  
**Required**: False  
**Minimum**: 32  
**Maximum**: 8182

### M2tsScte35Source

Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output.

- PASSTHROUGH
- NONE

### M2tsSegmentationMarkers

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

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

MPEG-2 TS container settings. These apply to outputs in a File output group when the output's container (ContainerType) is MPEG-2 Transport Stream (M2TS). In these assets, data is organized by the program map table (PMT). Each transport stream program contains subsets of data, including audio, video, and metadata. Each of these subsets of data has a numerical label called a packet identifier (PID). Each transport stream program corresponds to one MediaConvert output. The PMT lists the types of data in a program along with their PID. Downstream systems and players use the program map table to look up the PID for each type of data it accesses and then uses the PIDs to locate specific data within the asset.
**audioBufferModel**

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

- **Type**: M2tsAudioBufferModel (p. 184)
- **Required**: False

**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

**esRateInPes**

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

- **Type**: M2tsEsRateInPes (p. 185)
- **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

**dvbNitSettings**

Inserts DVB Network Information Table (NIT) at the specified table repetition interval.

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

**dvbSdtSettings**

Inserts DVB Service Description Table (NIT) at the specified table repetition interval.

- **Type**: DvbSdtSettings (p. 118)
- **Required**: False

**scte35Source**

Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output.

- **Type**: M2tsScte35Source (p. 186)
- **Required**: False
scte35Pid

Specify the packet identifier (PID) of the SCTE-35 stream in the transport stream.

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

scte35Esam

Include this in your job settings to put SCTE-35 markers in your HLS and transport stream outputs at the insertion points that you specify in an ESAM XML document. Provide the document in the setting SCC XML (sccXml).

- **Type**: M2tsScte35Esam (p. 185)
- **Required**: False

videoPid

Specify the packet identifier (PID) of the elementary video stream in the transport stream.

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

dvbTdtSettings

Inserts DVB Time and Date Table (TDT) at the specified table repetition interval.

- **Type**: DvbTdtSettings (p. 123)
- **Required**: False

pmtInterval

Specify the number of milliseconds between instances of the program map table (PMT) in the output transport stream.

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

segmentationStyle

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.
Type: `M2tsSegmentationStyle (p. 186)`
Required: False

**segmentationTime**
Specify the length, in seconds, of each segment. Required unless markers is set to _none_.

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

**pmtPid**
Specify the packet identifier (PID) for the program map table (PMT) itself. Default is 480.

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

**bitrate**
Specify 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

**audioPids**
Specify the packet identifiers (PIDs) for any elementary audio streams you include in this output. Specify multiple PIDs as a JSON array. Default is the range 482-492.

Type: Array of type integer
Required: False
Minimum: 32
Maximum: 8182

**privateMetadataPid**
Specify the packet identifier (PID) of the private metadata stream. Default is 503.

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

**nielsenId3**
If INSERT, Nielsen inaudible tones for media tracking will be detected in the input audio and an equivalent ID3 tag will be inserted in the output.
**Properties**

- **timedMetadataPid**
  - Specify the packet identifier (PID) for timed metadata in this output. Default is 502.
  - **Type:** integer
  - **Required:** False
  - **Minimum:** 32
  - **Maximum:** 8182

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

- **transportStreamId**
  - Specify the ID for the transport stream itself in the program map table for this output. Transport stream IDs and program map tables are parts of MPEG-2 transport stream containers, used for organizing data.
  - **Type:** integer
  - **Required:** False
  - **Minimum:** 0
  - **Maximum:** 65535

- **dvbSubPids**
  - Specify the packet identifiers (PIDs) for DVB subtitle data included in this output. Specify multiple PIDs as a JSON array. Default is the range 460-479.
  - **Type:** Array of type integer
  - **Required:** False
  - **Minimum:** 32
  - **Maximum:** 8182

- **rateMode**
  - 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.
  - **Type:** M2tsRateMode (p. 185)
  - **Required:** False

- **audioFramesPerPes**
  - The number of audio frames to insert for each PES packet.
Properties

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

**pcrControl**

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.

**Type**: M2tsPcrControl (p. 185)  
**Required**: False

**segmentationMarkers**

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.

**Type**: M2tsSegmentationMarkers (p. 186)  
**Required**: False

**ebpAudioInterval**

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).

**Type**: M2tsEbpAudioInterval (p. 184)  
**Required**: False

**forceTsVideoEbpOrder**

Keep the default value (DEFAULT) unless you know that your audio EBP markers are incorrectly appearing before your video EBP markers. To correct this problem, set this value to Force (FORCE).

**Type**: M2tsForceTsVideoEbpOrder (p. 185)  
**Required**: False

**programNumber**

Use Program number (programNumber) to specify the program number used in the program map table (PMT) for this output. Default is 1. Program numbers and program map tables are parts of MPEG-2 transport stream containers, used for organizing data.

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

Specify the packet identifier (PID) for the program clock reference (PCR) in this output. If you do not specify a value, the service will use the value for Video PID (VideoPid).

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

**bufferModel**

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.

- **Type:** M2tsBufferModel (p. 184)
- **Required:** False

**dvbTeletextPid**

Specify the packet identifier (PID) for DVB teletext data you include in this output. Default is 499.

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

**fragmentTime**

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

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

**ebpPlacement**

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).

- **Type:** M2tsEbpPlacement (p. 184)
- **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

M3u8NielsenId3
If INSERT, Nielsen inaudible tones for media tracking will be detected in the input audio and an equivalent ID3 tag will be inserted in the output.

INSERT
NONE

M3u8PcrControl
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
Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output.

PASSTHROUGH
NONE

M3u8Settings
Settings for TS segments in HLS

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

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

pcrControl
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.

Type: M3u8PcrControl (p. 193)
Required: False

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.
**Properties**

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

**pmtPid**

Packet Identifier (PID) for the Program Map Table (PMT) in the transport stream.

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

**privateMetadataPid**

Packet Identifier (PID) of the private metadata stream in the transport stream.

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

**programNumber**

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

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

**patInterval**

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

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

**pmtInterval**

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

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

**scte35Source**

Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output.
**Type:** M3u8Scte35Source (p. 193)
**Required:** False

**scte35Pid**
Packet Identifier (PID) of the SCTE-35 stream in the transport stream.

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

**nielsenId3**
If INSERT, Nielsen inaudible tones for media tracking will be detected in the input audio and an equivalent ID3 tag will be inserted in the output.

**Type:** M3u8NielsenId3 (p. 193)
**Required:** False

**timedMetadata**
Applies only to HLS outputs. Use this setting to specify whether the service inserts the ID3 timed metadata from the input in this output.

**Type:** TimedMetadata (p. 229)
**Required:** False

**timedMetadataPid**
Packet Identifier (PID) of the timed metadata stream in the transport stream.

**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.

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

- **Type:** Array of type integer
- **Required:** False
- **Minimum:** 32
- **Maximum:** 8182

**MotionImageInserter**

Overlay motion graphics on top of your video at the time that you specify.

**insertionMode**

Choose the type of motion graphic asset that you are providing for your overlay. You can choose either a .mov file or a series of .png files.

- **Type:** MotionImageInsertionMode (p. 198)
- **Required:** False

**input**

Specify the .mov file or series of .png files that you want to overlay on your video. For .png files, provide the file name of the first file in the series. Make sure that the names of the .png files end with sequential numbers that specify the order that they are played in. For example, overlay_000.png, overlay_001.png, overlay_002.png, and so on. The sequence must start at zero, and each image file name must have the same number of digits. Pad your initial file names with enough zeros to complete the sequence. For example, if the first image is overlay_0.png, there can be only 10 images in the sequence, with the last image being overlay_9.png. But if the first image is overlay_00.png, there can be 100 images in the sequence.

- **Type:** string
- **Required:** False
- **Pattern:** ^s3://.*(\.mov|\[0-9]+\.png)$
- **MinLength:** 14
- **MaxLength:** 1285

**offset**

Use Offset to specify the placement of your motion graphic overlay on the video frame. Specify in pixels, from the upper-left corner of the frame. If you don’t specify an offset, the service scales your overlay to the full size of the frame. Otherwise, the service inserts the overlay at its native resolution and scales the size up or down with any video scaling.

- **Type:** MotionImageInsertionOffset (p. 198)
- **Required:** False

**startTime**

Specify when the motion overlay begins. Use timecode format (HH:MM:SS:FF or HH:MM:SS;FF). Make sure that the timecode you provide here takes into account how you have set up your timecode.
configuration under both job settings and input settings. The simplest way to do that is to set both to start at 0. If you need to set up your job to follow timecodes embedded in your source that don't start at zero, make sure that you specify a start time that is after the first embedded timecode. For more information, see https://docs.aws.amazon.com/mediaconvert/latest/ug/setting-up-timecode.html

Find job-wide and input timecode configuration settings in your JSON job settings specification at settings>timecodeConfig>source and settings>inputs>timecodeSource.

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

Specify whether your motion graphic overlay repeats on a loop or plays only once.

**Type:** MotionImagePlayback (p. 198)
**Required:** False

**framerate**

If your motion graphic asset is a .mov file, keep this setting unspecified. If your motion graphic asset is a series of .png files, specify the frame rate of the overlay in frames per second, as a fraction. For example, specify 24 fps as 24/1. Make sure that the number of images in your series matches the frame rate and your intended overlay duration. For example, if you want a 30-second overlay at 30 fps, you should have 900 .png images. This overlay frame rate doesn't need to match the frame rate of the underlying video.

**Type:** MotionImageInsertionFramerate (p. 197)
**Required:** False

**MotionImageInsertionFramerate**

For motion overlays that don't have a built-in frame rate, specify the frame rate of the overlay in frames per second, as a fraction. For example, specify 24 fps as 24/1. The overlay frame rate doesn't need to match the frame rate of the underlying video.

**framerateNumerator**

The top of the fraction that expresses your overlay frame rate. For example, if your frame rate is 24 fps, set this value to 24.

**Type:** integer
**Required:** False

**framerateDenominator**

The bottom of the fraction that expresses your overlay frame rate. For example, if your frame rate is 24 fps, set this value to 1.
**Maximum:** 17895697

**MotionImageInsertionMode**

Choose the type of motion graphic asset that you are providing for your overlay. You can choose either a .mov file or a series of .png files.

- MOV
- PNG

**MotionImageInsertionOffset**

Specify the offset between the upper-left corner of the video frame and the top left corner of the overlay.

**imageX**

Set the distance, in pixels, between the overlay and the left edge of the video frame.

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

**imageY**

Set the distance, in pixels, between the overlay and the top edge of the video frame.

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

**MotionImagePlayback**

Specify whether your motion graphic overlay repeats on a loop or plays only once.

- ONCE
- REPEAT

**MovClapAtom**

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

- INCLUDE
- EXCLUDE

**MovCslgAtom**

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.
**MovMpeg2FourCCControl**

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**

If set to OMNEON, inserts Omneon-compatible padding

- OMNEON
- NONE

**MovReference**

Always keep the default value (SELF_CONTAINED) for this setting.

- SELF_CONTAINED
- EXTERNAL

**MovSettings**

Settings for MOV Container.

**clapAtom**

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

*Type:* MovClapAtom (p. 198)

*Required:* False

**cslgAtom**

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.

*Type:* MovCslgAtom (p. 198)

*Required:* False

**paddingControl**

If set to OMNEON, inserts Omneon-compatible padding

*Type:* MovPaddingControl (p. 199)

*Required:* False
**reference**

Always keep the default value (SELF_CONTAINED) for this setting.

*Type: MovReference (p. 199)*
*Required: False*

**mpeg2FourCCControl**

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.

*Type: MovMpeg2FourCCControl (p. 199)*
*Required: False*

**Mp2Settings**

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value MP2.

**bitrate**

Average bitrate in bits/second.

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

**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*

**sampleRate**

Sample rate in hz.

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

**Mp4CslgAtom**

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.
**Properties**

**INCLUDE**

**EXCLUDE**

### Mp4FreeSpaceBox

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

**INCLUDE**

**EXCLUDE**

### Mp4MoovPlacement

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

Settings for MP4 Container

#### cslgAtom

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.

- **Type**: Mp4CslgAtom (p. 200)
- **Required**: False

#### freeSpaceBox

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

- **Type**: Mp4FreeSpaceBox (p. 201)
- **Required**: False

#### mp4MajorBrand

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

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

#### moovPlacement

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.

- **Type**: Mp4MoovPlacement (p. 201)
- **Required**: False
Mpeg2AdaptiveQuantization
Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

OFF
LOW
MEDIUM
HIGH

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

AUTO
LOW
MAIN
HIGH1440
HIGH

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

MAIN
PROFILE_422

Mpeg2DynamicSubGop
Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting B frames between reference frames (numberBFramesBetweenReferenceFrames).

ADAPTIVE
STATIC

Mpeg2FramerateControl
If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

INITIALIZE_FROM_SOURCE
SPECIFIED
**Mpeg2FramerateConversionAlgorithm**

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

- DUPLICATE_DROP
- INTERPOLATE

**Mpeg2GopSizeUnits**

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**

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**

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**

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**

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

- SINGLE_PASS
- MULTI_PASS

**Mpeg2RateControlMode**

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

- VBR
- CBR

**Mpeg2SceneChangeDetect**

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

- DISABLED
- ENABLED

**Mpeg2Settings**

Required when you set (Codec) under (VideoDescription)(CodecSettings) to the value MPEG2.

**interlaceMode**

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.

- **Type:** Mpeg2InterlaceMode (p. 203)
- **Required:** False

**parNumerator**

Pixel Aspect Ratio numerator.

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

**syntax**

Produces a Type D-10 compatible bitstream (SMPTE 356M-2001).
Type: Mpeg2Syntax (p. 209)
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

framerateDenominator
Frame rate denominator.
Type: integer
Required: False
Minimum: 1
Maximum: 1001

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

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

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

hrdBufferSize
Size of buffer (HRD buffer model) in bits. For example, enter five megabits as 5000000.
**maxBitrate**

Maximum bitrate in bits/second. For example, enter five megabits per second as 5000000.

**Type:** integer  
**Required:** False  
**Minimum:** 0  
**Maximum:** 47185920

**slowPal**

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

**Type:** Mpeg2SlowPal (p. 209)  
**Required:** False

**parDenominator**

Pixel Aspect Ratio denominator.

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

**spatialAdaptiveQuantization**

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

**Type:** Mpeg2SpatialAdaptiveQuantization (p. 209)  
**Required:** False

**temporalAdaptiveQuantization**

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

**Type:** Mpeg2TemporalAdaptiveQuantization (p. 210)  
**Required:** False

**bitrate**

Average bitrate in bits/second. Required for VBR and CBR. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.

**Type:** integer  
**Required:** False  
**Minimum:** 1000  
**Maximum:** 288000000
**intraDcPrecision**

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.

*Type:* Mpeg2IntraDcPrecision (p. 203)  
*Required:* False

**framerateControl**

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

*Type:* Mpeg2FramerateControl (p. 202)  
*Required:* False

**rateControlMode**

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

*Type:* Mpeg2RateControlMode (p. 204)  
*Required:* False

**codecProfile**

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

*Type:* Mpeg2CodecProfile (p. 202)  
*Required:* False

**telecine**

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.

*Type:* Mpeg2Telecine (p. 210)  
*Required:* False

**framerateNumerator**

Frame rate numerator - frame rate is a fraction, e.g. 24000 / 1001 = 23.976 fps.

*Type:* integer  
*Required:* False  
*Minimum:* 24
Maximum: 60000

**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

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

Type: Mpeg2AdaptiveQuantization (p. 202)
Required: False

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

Type: Mpeg2CodecLevel (p. 202)
Required: False

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

Type: Mpeg2SceneChangeDetect (p. 204)
Required: False

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

Type: Mpeg2QualityTuningLevel (p. 204)
Required: False

**framerateConversionAlgorithm**
When set to INTERPOLATE, produces smoother motion during frame rate conversion.

Type: Mpeg2FramerateConversionAlgorithm (p. 203)
Required: False

**gopSizeUnits**
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.
Type: Mpeg2GopSizeUnits (p. 203)
Required: False

**parControl**

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.

Type: Mpeg2ParControl (p. 203)
Required: False

**numberBFramesBetweenReferenceFrames**

Number of B-frames between reference frames.

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

**dynamicSubGop**

Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting B frames between reference frames (numberBFramesBetweenReferenceFrames).

Type: Mpeg2DynamicSubGop (p. 202)
Required: False

**Mpeg2SlowPal**

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

DISABLED
ENABLED

**Mpeg2SpatialAdaptiveQuantization**

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

DISABLED
ENABLED

**Mpeg2Syntax**

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

DEFAULT
D_10
**Mpeg2Telecine**

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**

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

- DISABLED
- ENABLED

**MsSmoothAudioDeduplication**

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

- COMBINE_DUPLICATE_STREAMS
- NONE

**MsSmoothEncryptionSettings**

If you are using DRM, set DRM System (MsSmoothEncryptionSettings) to specify the value SpekeKeyProvider.

**spekeKeyProvider**

Settings for use with a SPEKE key provider

- **Type:** SpekeKeyProvider (p. 224)
- **Required:** False

**MsSmoothGroupSettings**

Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to MS_SMOOTH_GROUP_SETTINGS.

**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:** False
- **Pattern:** ^s3:\/\/

210
destinationSettings

Settings associated with the destination. Will vary based on the type of destination

  Type: DestinationSettings (p. 117)
  Required: False

fragmentLength

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

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

cipher

If you are using DRM, set DRM System (MsSmoothEncryptionSettings) to specify the value SpekeKeyProvider.

  Type: MsSmoothEncryptionSettings (p. 210)
  Required: False

manifestEncoding

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

  Type: MsSmoothManifestEncoding (p. 211)
  Required: False

audioDeduplication

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

  Type: MsSmoothAudioDeduplication (p. 210)
  Required: False

MsSmoothManifestEncoding

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

  UTF8
  UTF16

NielsenConfiguration

Settings for Nielsen Configuration
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

distributorId

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

Type: string
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. When you enable Noise reducer (NoiseReducer), you must also select a value for Noise reducer filter (NoiseReducerFilter).

filter

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 a frequency-domain filter based on JND principles.

Type: NoiseReducerFilter (p. 212)
Required: False

filterSettings

Settings for a noise reducer filter

Type: NoiseReducerFilterSettings (p. 213)
Required: False

spatialFilterSettings

Noise reducer filter settings for spatial filter.

Type: NoiseReducerSpatialFilterSettings (p. 213)
Required: False

NoiseReducerFilter

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 a frequency-domain filter based on JND principles.

**BILATERAL**
**MEAN**
**GAUSSIAN**
**LANCZOS**
**SHARPEN**
**CONSERVE**
**SPATIAL**

### NoiseReducerFilterSettings
Settings for a noise reducer filter

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

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

### NoiseReducerSpatialFilterSettings
Noise reducer filter settings for spatial filter.

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

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

**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

### 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
Order

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

An output object describes the settings for a single output file or stream in an output group.

containerSettings

Container specific settings.

Type: ContainerSettings (p. 110)
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
MinLength: 0

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. 231)
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. 87)
Required: False

outputSettings

Specific settings for this type of output.

Type: OutputSettings (p. 217)
**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

**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 (NameModifier) 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

**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. 98)  
*Required*: False

**OutputChannelMapping**

OutputChannel mapping settings.

**inputChannels**

List of input channels

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

- *Minimum*: -60
- *Maximum*: 6

**OutputGroup**

Group of outputs

**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.
Properties

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

**name**
Name of the output group

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

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

**Type**: Array of type Output (p. 214)  
**Required**: False

**OutputGroupSettings**
Output Group settings, including type

**Type**: OutputGroupSettings (p. 216)  
**Required**: False

**OutputGroupSettings**
Output Group settings, including type

**type**
Type of output group (File group, Apple HLS, DASH ISO, Microsoft Smooth Streaming, CMAF)

**Type**: OutputGroupType (p. 217)  
**Required**: False

**hlsGroupSettings**
Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to HLS_GROUP_SETTINGS.

**Type**: HlsGroupSettings (p. 162)  
**Required**: False

**dashIsoGroupSettings**
Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to DASH_ISO_GROUP_SETTINGS.

**Type**: DashIsoGroupSettings (p. 113)  
**Required**: False

**fileGroupSettings**
Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to FILE_GROUP_SETTINGS.
**Properties**

<table>
<thead>
<tr>
<th>Type</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>FileGroupSettings (p. 132)</td>
<td>False</td>
</tr>
</tbody>
</table>

**msSmoothGroupSettings**

Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to MS_SMOOTH_GROUP_SETTINGS.

- **Type**: MsSmoothGroupSettings (p. 210)
- **Required**: False

**cmafGroupSettings**

Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to CMAF_GROUP_SETTINGS. Each output in a CMAF Output Group may only contain a single video, audio, or caption output.

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

**OutputGroupType**

Type of output group (File group, Apple HLS, DASH ISO, Microsoft Smooth Streaming, CMAF)

- HLS_GROUP_SETTINGS
- DASH_ISO_GROUP_SETTINGS
- FILE_GROUP_SETTINGS
- MS_SMOOTH_GROUP_SETTINGS
- CMAF_GROUP_SETTINGS

**OutputSdt**

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**

Specific settings for this type of output.

**hlsSettings**

Settings for HLS output groups

- **Type**: HlsSettings (p. 168)
- **Required**: False
ProresCodecProfile

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

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

INITIALIZE_FROM_SOURCE
SPECIFIED

ProresFramerateConversionAlgorithm

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

DUPLICATE_DROP
INTERPOLATE

ProresInterlaceMode

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

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

Required when you set (Codec) under (VideoDescription)->(CodecSettings) to the value PRORES.

interlaceMode

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.

Type: ProresInterlaceMode (p. 218)
Required: False

parNumerator

Pixel Aspect Ratio numerator.

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

framerateDenominator

Frame rate denominator.

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

codecProfile

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

Type: ProresCodecProfile (p. 218)
Required: False

slowPal

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

Type: ProresSlowPal (p. 221)
Required: False
parDenominator

Pixel Aspect Ratio denominator.

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

framerateControl

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

Type: ProresFramerateControl (p. 218)
Required: False

telecine

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.

Type: ProresTelecine (p. 221)
Required: False

framerateNumerator

When you use the API for transcoding jobs that use frame rate conversion, specify the frame rate 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

framerateConversionAlgorithm

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

Type: ProresFramerateConversionAlgorithm (p. 218)
Required: False

parControl

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).

**Type:** ProresParControl (p. 218)
**Required:** False

**ProresSlowPal**

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

DISABLED
ENABLED

**ProresTelecine**

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**

Use Rectangle to identify a specific area of the video frame.

**height**

Height of rectangle in pixels. Specify only even numbers.

**Type:** integer
**Required:** False
**Minimum:** 2
**Maximum:** 2147483647

**width**

Width of rectangle in pixels. Specify only even numbers.

**Type:** integer
**Required:** False
**Minimum:** 2
**Maximum:** 2147483647

**x**

The distance, in pixels, between the rectangle and the left edge of the video frame. Specify only even numbers.

**Type:** integer
**Required:** False
**Minimum:** 0
**Properties**

**Maximum**: 2147483647

**Y**
The distance, in pixels, between the rectangle and the top edge of the video frame. Specify only even numbers.

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

**RemixSettings**
Use Manual audio remixing (RemixSettings) to adjust audio levels for each audio channel in each output of your job. With audio remixing, you can output more or fewer audio channels than your input audio source provides.

**channelMapping**
Channel mapping (ChannelMapping) contains the group of fields that hold the remixing value for each channel. Units are in dB. Acceptable values are within the range from -60 (mute) through 6. A setting of 0 passes the input channel unchanged to the output channel (no attenuation or amplification).

- **Type**: ChannelMapping (p. 103)
- **Required**: False

**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**: False
- **Minimum**: 1
- **Maximum**: 16

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

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 8

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

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

**S3DestinationSettings**

Settings associated with S3 destination

**encryption**

Settings for how your job outputs are encrypted as they are uploaded to Amazon S3.

Type: S3EncryptionSettings (p. 223)
Required: False

**S3EncryptionSettings**

Settings for how your job outputs are encrypted as they are uploaded to Amazon S3.

**encryptionType**

Specify how you want your data keys managed. AWS uses data keys to encrypt your content. AWS also encrypts the data keys themselves, using a customer master key (CMK), and then stores the encrypted data keys alongside your encrypted content. Use this setting to specify which AWS service manages the CMK. For simplest set up, choose Amazon S3 (SERVER_SIDE_ENCRYPTION_S3). If you want your master key to be managed by AWS Key Management Service (KMS), choose AWS KMS (SERVER_SIDE_ENCRYPTION_KMS). By default, when you choose AWS KMS, KMS uses the AWS managed customer master key (CMK) associated with Amazon S3 to encrypt your data keys. You can optionally choose to specify a different, customer managed CMK. Do so by specifying the Amazon Resource Name (ARN) of the key for the setting KMS ARN (kmsKeyArn).

Type: S3ServerSideEncryptionType (p. 223)
Required: False

**kmsKeyArn**

Optionally, specify the customer master key (CMK) that you want to use to encrypt the data key that AWS uses to encrypt your output content. Enter the Amazon Resource Name (ARN) of the CMK. To use this setting, you must also set Server-side encryption (S3ServerSideEncryptionType) to AWS KMS (SERVER_SIDE_ENCRYPTION_KMS). If you set Server-side encryption to AWS KMS but don't specify a CMK here, AWS uses the AWS managed CMK associated with Amazon S3.

Type: string
Required: False
Pattern: ^arn:aws(-us-gov)?:kms:[a-z-]{2,6}-(east|west|central|((north|south)(east|west)?))-[1-9]{1,2}:[a-fA-F0-9]{12}:key/[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{12}$

**S3ServerSideEncryptionType**

Specify how you want your data keys managed. AWS uses data keys to encrypt your content. AWS also encrypts the data keys themselves, using a customer master key (CMK), and then stores the
encrypted data keys alongside your encrypted content. Use this setting to specify which AWS service manages the CMK. For simplest set up, choose Amazon S3 (SERVER_SIDE_ENCRYPTION_S3). If you want your master key to be managed by AWS Key Management Service (KMS), choose AWS KMS (SERVER_SIDE_ENCRYPTION_KMS). By default, when you choose AWS KMS, KMS uses the AWS managed customer master key (CMK) associated with Amazon S3 to encrypt your data keys. You can optionally choose to specify a different, customer managed CMK. Do so by specifying the Amazon Resource Name (ARN) of the key for the setting KMS ARN (kmsKeyArn).

SERVER_SIDE_ENCRYPTION_S3
SERVER_SIDE_ENCRYPTION_KMS

**ScalingBehavior**

Applies only if your input aspect ratio is different from your output aspect ratio. Choose "Stretch to output" to have the service stretch your video image to fit. Keep the setting "Default" 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**

Set Framerate (SccDestinationFramerate) to make sure that the captions and the video are synchronized in the output. Specify a frame rate that matches the frame rate of the associated video. If the video frame rate 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**

Settings for SCC caption output.

**framerate**

Set Framerate (SccDestinationFramerate) to make sure that the captions and the video are synchronized in the output. Specify a frame rate that matches the frame rate of the associated video. If the video frame rate 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).

Type: SccDestinationFramerate (p. 224)
Required: False

**SpekeKeyProvider**

Settings for use with a SPEKE key provider
**resourceId**

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

*Type: string*
*Required: False*

**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: False*
*Pattern: ^[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}$*

**url**

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

*Type: string*
*Required: False*
*Format: uri*
*Pattern: ^https:/$*

**certificateArn**

Optional AWS Certificate Manager ARN for a certificate to send to the keyprovider. The certificate holds a key used by the keyprovider to encrypt the keys in its response.

*Type: string*
*Required: False*
*Pattern: ^arn:aws(-us-gov)?:acm:*

**StaticKeyProvider**

Use these settings to set up encryption with a static key provider.

**staticKeyValue**

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

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

**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*
Properties

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+)*$)

url

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

Type: string
Required: False
Format: uri

StatusUpdateInterval

Specify how often MediaConvert sends STATUS_UPDATE events to Amazon CloudWatch Events. Set the interval, in seconds, between status updates. MediaConvert sends an update at this interval from the time the service begins processing your job to the time it completes the transcode or encounters an error.

 SECONDS_10
 SECONDS_12
 SECONDS_15
 SECONDS_20
 SECONDS_30
 SECONDS_60
 SECONDS_120
 SECONDS_180
 SECONDS_240
 SECONDS_300
 SECONDS_360
 SECONDS_420
 SECONDS_480
 SECONDS_540
 SECONDS_600

TeletextDestinationSettings

Settings for Teletext caption output

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.
TeletextSourceSettings

Settings specific to Teletext caption sources, including Page number.

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]$/  
MinLength: 3  
MaxLength: 3

TimecodeBurnin

Timecode burn-in (TimecodeBurnIn)--Burns the output timecode and specified prefix into the output.

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

Use Position (Position) under under Timecode burn-in (TimecodeBurnIn) to specify the location the burned-in timecode on output video.

Type: TimecodeBurninPosition (p. 228)  
Required: False

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: ^[^ ][ -~]+$/
**TimecodeBurninPosition**

Use Position (Position) under 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**

These settings control how the service handles timecodes throughout the job. These settings don't affect input clipping.

**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 frame rate conversion. System behavior for Anchor Timecode varies depending on your setting for Source (TimecodeSource).

* If 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 Source (TimecodeSource) is set to Start at 0 (ZEROBASED) the first frame is 00:00:00:00. * If 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}$`

**source**

Use Source (TimecodeSource) to set how timecodes are handled within this job. 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.

- **Type**: TimecodeSource (p. 229)
- **Required**: False

**start**

Only use when you set 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).
Properties

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

timestampOffset

Only applies to outputs that support program-date-time stamp. Use Timestamp offset (TimestampOffset) to overwrite the timecode date without affecting the time and frame number. Provide the new date as a string in the format "yyyy-mm-dd". To use Time stamp offset, you must also enable Insert program-date-time (InsertProgramDateTime) in the output settings. For example, if the date part of your timecodes is 2002-1-25 and you want to change it to one year later, set Timestamp offset (TimestampOffset) to 2003-1-25.

Type: string
Required: False
Pattern: ^([0-9]{4})-(0[1-9]|1[0-2])-(0[1-9]|1[2][0-9]|3[01])$

TimecodeSource

Use Source (TimecodeSource) to set how timecodes are handled within this job. 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

Applies only to HLS outputs. Use this setting to specify whether the service inserts the ID3 timed metadata from the input in this output.

PASSTHROUGH
NONE

TimedMetadataInsertion

Enable Timed metadata insertion (TimedMetadataInsertion) to include ID3 tags in your job. To include timed metadata, you must enable it here, enable it in each output container, and specify tags and timecodes in ID3 insertion (Id3Insertion) objects.

id3Insertions

Id3Insertions contains the array of Id3Insertion instances.

Type: Array of type Id3Insertion (p. 169)
Required: False
**TrackSourceSettings**

Settings specific to caption sources that are specified by track number. Sources include IMSC in IMF.

**trackNumber**

Use this setting to select a single captions track from a source. Track numbers correspond to the order in the captions source file. For IMF sources, track numbering is based on the order that the captions appear in the CPL. For example, use 1 to select the captions asset that is listed first in the CPL. To include more than one captions track in your job outputs, create multiple input captions selectors. Specify one track per selector.

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

**TtmlDestinationSettings**

Settings specific to TTML caption outputs, including Pass style information (TtmlStylePassthrough).

**stylePassthrough**

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.

- **Type**: TtmlStylePassthrough (p. 230)
- **Required**: False

**TtmlStylePassthrough**

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**: ENABLED, DISABLED

**Type**

- **SYSTEM**
- **CUSTOM**

**VideoCodec**

Type of video codec

- FRAME_CAPTURE
- H_264
- H_265
- MPEG2
- PRORES
Properties

VideoCodecSettings

Video codec settings, (CodecSettings) under (VideoDescription), contains the group of settings related to video encoding. The settings in this group vary depending on the value you choose for Video codec (Codec). For each codec enum you choose, define the corresponding settings object. The following lists the codec enum, settings object pairs. * H_264, H264Settings * H_265, H265Settings * MPEG2, Mpeg2Settings * PRORES, ProresSettings * FRAME_CAPTURE, FrameCaptureSettings

codec

Specifies the video codec. This must be equal to one of the enum values defined by the object VideoCodec.

  Type: VideoCodec (p. 230)
  Required: False

frameCaptureSettings

Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value FRAME_CAPTURE.

  Type: FrameCaptureSettings (p. 133)
  Required: False

h264Settings

Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value H_264.

  Type: H264Settings (p. 138)
  Required: False

h265Settings

Settings for H265 codec

  Type: H265Settings (p. 149)
  Required: False

mpeg2Settings

Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value MPEG2.

  Type: Mpeg2Settings (p. 204)
  Required: False

proresSettings

Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value PRORES.

  Type: ProresSettings (p. 219)
  Required: False

VideoDescription

Settings for video outputs
**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

**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

**scalingBehavior**

Applies only if your input aspect ratio is different from your output aspect ratio. Choose "Stretch to output" to have the service stretch your video image to fit. Keep the setting "Default" to allow the service to letterbox your video instead. This setting overrides any positioning value you specify elsewhere in the job.

- **Type**: ScalingBehavior (p. 224)
- **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. If you crop your input frame size to smaller than your output frame size, make sure to specify the behavior you want in your output setting "Scaling behavior".

- **Type**: Rectangle (p. 221)
- **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

**videoPreprocessors**

Find additional transcoding features under Preprocessors (VideoPreprocessors). Enable the features at each output individually. These features are disabled by default.
**Type:** VideoPreprocessor (p. 234)
**Required:** False

**timecodeInsertion**

Applies only to H.264, H.265, MPEG2, and ProRes outputs. Only enable Timecode insertion when the input frame rate is identical to the output frame rate. To include timecodes in this output, set Timecode insertion (VideoTimecodeInsertion) to PIC_TIMING_SEI. To leave them out, set it to DISABLED. Default is DISABLED. When the service inserts timecodes in an output, by default, it uses any embedded timecodes from the input. If none are present, the service will set the timecode for the first output frame to zero. To change this default behavior, adjust the settings under Timecode configuration (TimecodeConfig). In the console, these settings are located under Job > Job settings > Timecode configuration. Note - Timecode source under input settings (InputTimecodeSource) does not affect the timecodes that are inserted in the output. Source under Job settings > Timecode configuration (TimecodeSource) does.

**Type:** VideoTimecodeInsertion (p. 236)
**Required:** False

**antiAlias**

The anti-alias filter is automatically applied to all outputs. The service no longer accepts the value DISABLED for AntiAlias. If you specify that in your job, the service will ignore the setting.

**Type:** AntiAlias (p. 86)
**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. 221)
**Required:** False

**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. 0 is the softest setting, 100 the sharpest, and 50 recommended for most content.

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

**codecSettings**

Video codec settings, (CodecSettings) under (VideoDescription), contains the group of settings related to video encoding. The settings in this group vary depending on the value you choose for Video codec (Codec). For each codec enum you choose, define the corresponding settings object. The following lists the codec enum, settings object pairs. * H_264, H264Settings * H_265, H265Settings * MPEG2, Mpeg2Settings * PRORES, ProresSettings * FRAME_CAPTURE, FrameCaptureSettings

**Type:** VideoCodecSettings (p. 231)
Properties

Required: False

afdSignaling

This setting only applies to H.264, H.265, and MPEG2 outputs. Use Insert AFD signaling (AfdSignaling) to specify whether the service includes 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.

Type: AfdSignaling (p. 85)
Required: False

dropFrameTimecode

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.

Type: DropFrameTimecode (p. 117)
Required: False

respondToAfd

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.

Type: RespondToAfd (p. 222)
Required: False

colorMetadata

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

Type: ColorMetadata (p. 109)
Required: False

VideoPreprocessor

Find additional transcoding features under Preprocessors (VideoPreprocessors). Enable the features at each output individually. These features are disabled by default.

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. 108)
Required: False
deinterlacer

Use Deinterlacer (Deinterlacer) to produce smoother motion and a clearer picture.

Type: Deinterlacer (p. 116)
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. 169)
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. 212)
Required: False

timecodeBurnin

Timecode burn-in (TimecodeBurnIn)--Burns the output timecode and specified prefix into the output.

Type: TimecodeBurnin (p. 227)
Required: False

VideoSelector

Selector for video.

colorSpace

If your input video has accurate color space metadata, or if you don’t know about color space, leave this set to the default value FOLLOW. The service will automatically detect your input color space. If your input video has metadata indicating the wrong color space, or if your input video is missing color space metadata that should be there, specify the accurate color space here. If you choose HDR10, you can also correct inaccurate color space coefficients, using the HDR master display information controls. You must also set Color space usage (ColorSpaceUsage) to FORCE for the service to use these values.

Type: ColorSpace (p. 109)
Required: False

rotate

Use Rotate (InputRotate) to specify how the service rotates your video. You can choose automatic rotation or specify a rotation. You can specify a clockwise rotation of 0, 90, 180, or 270 degrees. If your input video container is .mov or .mp4 and your input has rotation metadata, you can choose Automatic to have the service rotate your video according to the rotation specified in the metadata. The rotation must be within one degree of 90, 180, or 270 degrees. If the rotation metadata specifies any other rotation, the service will default to no rotation. By default, the service does no rotation, even if your input video has rotation metadata. The service doesn’t pass through rotation metadata.
Properties

Type: InputRotate (p. 171)
Required: False

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

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

colorSpaceUsage

There are two sources for color metadata, the input file and the job configuration (in the Color space and HDR master display information settings). The Color space usage setting controls which takes precedence. FORCE: The 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: The 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.

Type: ColorSpaceUsage (p. 110)
Required: False

hdr10Metadata

Use the "HDR master display information" (Hdr10Metadata) settings to correct HDR metadata or to provide missing metadata. These values vary depending on the input video and must be provided by a color grader. Range is 0 to 50,000; each increment represents 0.00002 in CIE1931 color coordinate. Note that these settings are not color correction. Note that if you are creating HDR outputs inside of an HLS CMAF package, to comply with the Apple specification, you must use the following settings. Set "MP4 packaging type" (writeMp4PackagingType) to HVC1 (HVC1). Set "Profile" (H265Settings > codecProfile) to Main10/High (MAIN10_HIGH). Set "Level" (H265Settings > codecLevel) to 5 (LEVEL_5).

Type: Hdr10Metadata (p. 157)
Required: False

VideoTimecodeInsertion

Applies only to H.264, H.265, MPEG2, and ProRes outputs. Only enable Timecode insertion when the input frame rate is identical to the output frame rate. To include timecodes in this output, set Timecode insertion (VideoTimecodeInsertion) to PIC_TIMING_SEI. To leave them out, set it to DISABLED. Default is DISABLED. When the service inserts timecodes in an output, by default, it uses any embedded timecodes.
from the input. If none are present, the service will set the timecode for the first output frame to zero. To change this default behavior, adjust the settings under Timecode configuration (TimecodeConfig). In the console, these settings are located under Job > Job settings > Timecode configuration. Note - Timecode source under input settings (InputTimecodeSource) does not affect the timecodes that are inserted in the output. Source under Job settings > Timecode configuration (TimecodeSource) does.

   DISABLED
   PIC_TIMING_SEI

WavFormat

The service defaults to using RIFF for WAV outputs. If your output audio is likely to exceed 4 GB in file size, or if you otherwise need the extended support of the RF64 format, set your output WAV file format to RF64.

   RIFF
   RF64

WavSettings

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value WAV.

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

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

sampleRate

Sample rate in Hz.

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

format

The service defaults to using RIFF for WAV outputs. If your output audio is likely to exceed 4 GB in file size, or if you otherwise need the extended support of the RF64 format, set your output WAV file format to RF64.
Type: WavFormat (p. 237)
Required: False

See Also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

ListJobTemplates

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Go - Pilot
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

CreateJobTemplate

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Go - Pilot
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

JobTemplates name

URI

/2017-08-29/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. 253)</td>
<td>200 response</td>
</tr>
<tr>
<td>400</td>
<td>ExceptionBody (p. 277)</td>
<td>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>403</td>
<td>ExceptionBody (p. 277)</td>
<td>You don't have permissions for this action with the credentials you sent.</td>
</tr>
<tr>
<td>404</td>
<td>ExceptionBody (p. 277)</td>
<td>The resource you requested does not exist.</td>
</tr>
<tr>
<td>409</td>
<td>ExceptionBody (p. 277)</td>
<td>The service could not complete your request because there is a conflict with the current state of the resource.</td>
</tr>
<tr>
<td>429</td>
<td>ExceptionBody (p. 277)</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>
<tr>
<td>500</td>
<td>ExceptionBody (p. 277)</td>
<td>The service encountered an unexpected condition and cannot fulfill your request.</td>
</tr>
</tbody>
</table>

**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 (p. 265)</td>
<td>200 response</td>
</tr>
</tbody>
</table>
### 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>
<tbody>
<tr>
<td>202</td>
<td>DeleteJobTemplateResponse</td>
<td>202 response</td>
</tr>
<tr>
<td>400</td>
<td>ExceptionBody (p. 277)</td>
<td>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>403</td>
<td>ExceptionBody (p. 277)</td>
<td>You don’t have permissions for this action with the credentials you sent.</td>
</tr>
<tr>
<td>404</td>
<td>ExceptionBody (p. 277)</td>
<td>The resource you requested does not exist.</td>
</tr>
<tr>
<td>409</td>
<td>ExceptionBody (p. 277)</td>
<td>The service could not complete your request because there is a conflict with the current state of the resource.</td>
</tr>
<tr>
<td>429</td>
<td>ExceptionBody (p. 277)</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>
<tr>
<td>500</td>
<td>ExceptionBody (p. 277)</td>
<td>The service encountered an unexpected condition and cannot fulfill your request.</td>
</tr>
</tbody>
</table>
### Schemas

#### Request Bodies

**Example GET**

```json
{
    "name": "string"
}
```

**Example PUT**

```json
{
    "description": "string",
    "category": "string",
    "queue": "string",
    "name": "string",
    "settings": {
        "timecodeConfig": {
            "anchor": "string",
            "source": enum,
            "start": "string",
            "timestampOffset": "string"
        },
        "outputGroups": [
            {
                "customName": "string",
                "name": "string",
                "outputs": [
                    {
                        "containerSettings": {
                            "container": enum,
                            "m3u8Settings": {
                                "audioFramesPerPes": integer,
                                "pcrControl": enum,
                                "pcrPid": integer,
                                "pmtPid": integer,
                                "privateMetadataPid": integer,
                                "programNumber": integer,
                                "patInterval": integer,
                                "pmtInterval": integer,
                                "scte35Source": enum,
                            }
                        }
                    }
                ]
            }
        ]
    }
}
```
"scte35Pid": integer,
"nielsenId3": enum,
"timedMetadata": enum,
"timedMetadataPid": integer,
"transportStreamId": integer,
"videoPid": integer,
"audioPids": [ integer
 ],
"f4vSettings": {
  "moovPlacement": enum
 },
"m2tsSettings": {
  "audioBufferModel": enum,
  "minEbpInterval": integer,
  "esRateInPes": enum,
  "patInterval": integer,
  "dvbNitSettings": {
    "nitInterval": integer,
    "networkId": integer,
    "networkName": "string"
  },
  "dvbSdtSettings": {
    "outputSdt": enum,
    "sdInterval": integer,
    "serviceName": "string",
    "serviceProviderName": "string"
  },
  "scte35Source": enum,
  "scte35Pid": integer,
  "scte35Esam": {
    "scte35EsamPid": integer
  },
  "videoPid": integer,
  "dvbTdtSettings": {
    "tdtInterval": integer
  },
  "pmtInterval": integer,
  "segmentationStyle": enum,
  "segmentationTime": number,
  "pmtPid": integer,
  "bitrate": integer,
  "audioPids": [ integer
 ],
  "privateMetadataPid": integer,
  "nielsenId3": enum,
  "timedMetadataPid": integer,
  "maxPcrInterval": integer,
  "transportStreamId": integer,
  "dvbSubPids": [ integer
 ],
  "rateMode": enum,
  "audioFramesPerPes": integer,
  "pcrControl": enum,
  "segmentationMarkers": enum,
  "ebpAudioInterval": enum,
  "forceTsVideoEbpOrder": enum,
  "programNumber": integer,
  "pcrPid": integer,
  "bufferModel": enum,
  "dvbTeletextPid": integer,
  "fragmentTime": number,
  "ebpPlacement": enum,
"nullPacketBitrate": number
},
"movSettings": {
  "clapAtom": enum,
  "cslgAtom": enum,
  "paddingControl": enum,
  "reference": enum,
  "mpeg2FourCCControl": enum
},
"mp4Settings": {
  "cslgAtom": enum,
  "freeSpaceBox": enum,
  "mp4MajorBrand": "string",
  "moovPlacement": enum
},
"preset": "string",
"videoDescription": {
  "fixedAfd": integer,
  "width": integer,
  "scalingBehavior": enum,
  "crop": {
    "height": integer,
    "width": integer,
    "x": integer,
    "y": integer
  },
  "height": integer,
  "videoPreprocessors": {
    "colorCorrector": {
      "brightness": integer,
      "colorSpaceConversion": enum,
      "contrast": integer,
      "hue": integer,
      "saturation": integer,
      "hdr10Metadata": {
        "redPrimaryX": integer,
        "redPrimaryY": integer,
        "greenPrimaryX": integer,
        "greenPrimaryY": integer,
        "bluePrimaryX": integer,
        "bluePrimaryY": integer,
        "whitePointX": integer,
        "whitePointY": integer,
        "maxFrameAverageLightLevel": integer,
        "maxContentLightLevel": integer,
        "maxLuminance": integer,
        "minLuminance": integer
      }
    },
    "deinterlacer": {
      "algorithm": enum,
      "mode": enum,
      "control": enum
    },
    "imageInserter": {
      "insertableImages": [
        {
          "width": integer,
          "height": integer,
          "imageX": integer,
          "imageY": integer,
          "duration": integer,
          "fadeIn": integer,
          "layer": integer,
          "imageInserterInput": "string",
        }
      ]
    }
  }
}
"startTime": "string",
"fadeOut": integer,
"opacity": integer
]
},
"noiseReducer": {
"filter": enum,
"filterSettings": {
"strength": integer
},
"spatialFilterSettings": {
"strength": integer,
"speed": integer,
"postFilterSharpenStrength": integer
}
},
"timecodeBurnin": {
"fontSize": integer,
"position": enum,
"prefix": "string"
}
},
"timecodeInsertion": enum,
"antiAlias": enum,
"position": {
"height": integer,
"width": integer,
"x": integer,
"y": integer
},
"sharpness": integer,
"codecSettings": {
"codec": enum,
"frameCaptureSettings": {
"framerateNumerator": integer,
"framerateDenominator": integer,
"maxCaptures": integer,
"quality": integer
},
"h264Settings": {
"interlaceMode": enum,
"parNumerator": integer,
"numberReferenceFrames": integer,
"syntax": enum,
"softness": integer,
"framerateDenominator": integer,
"gopClosedCadence": integer,
"hrdBufferInitialFillPercentage": integer,
"gopSize": number,
"slices": integer,
"gopReference": enum,
"hrdBufferSize": integer,
"maxBitrate": integer,
"slowPal": enum,
"parDenominator": integer,
"spatialAdaptiveQuantization": enum,
"temporalAdaptiveQuantization": enum,
"flickerAdaptiveQuantization": enum,
"entropyEncoding": enum,
"bitrate": integer,
"framerateControl": enum,
"rateControlMode": enum,
"qvbrSettings": {
"qvbrQualityLevel": integer,
"maxAverageBitrate": integer
"h265Settings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "numberReferenceFrames": integer,
  "framerateDenominator": integer,
  "gopClosedCadence": integer,
  "alternateTransferFunctionSei": enum,
  "hrdBufferInitialFillPercentage": integer,
  "gopSize": number,
  "slices": integer,
  "gopBReference": enum,
  "hrdBufferSize": integer,
  "maxBitrate": integer,
  "slowPal": enum,
  "parDenominator": integer,
  "spatialAdaptiveQuantization": enum,
  "temporalAdaptiveQuantization": enum,
  "flickerAdaptiveQuantization": enum,
  "bitrate": integer,
  "framerateControl": enum,
  "rateControlMode": enum,
  "qvbrControlSettings": {
    "qvbrQualityLevel": integer,
    "maxAverageBitrate": integer
  },
  "codecProfile": enum,
  "tiles": enum,
  "telecine": enum,
  "framerateNumerator": integer,
  "minIInterval": integer,
  "adaptiveQuantization": enum,
  "codecLevel": enum,
  "sceneChangeDetect": enum,
  "qualityTuningLevel": enum,
  "framerateConversionAlgorithm": enum,
  "unregisteredSeiTimecode": enum,
  "gopSizeUnits": enum,
  "parControl": enum,
  "numberBFramesBetweenReferenceFrames": integer,
  "repeatPps": enum,
  "dynamicSubGop": enum
},
"mpeg2Settings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "numberReferenceFrames": integer,
  "framerateDenominator": integer,
  "gopClosedCadence": integer,
  "alternateTransferFunctionSei": enum,
  "hrdBufferInitialFillPercentage": integer,
  "gopSize": number,
  "slices": integer,
  "gopBReference": enum,
  "hrdBufferSize": integer,
  "maxBitrate": integer,
  "slowPal": enum,
  "parDenominator": integer,
  "spatialAdaptiveQuantization": enum,
  "temporalAdaptiveQuantization": enum,
  "flickerAdaptiveQuantization": enum,
  "bitrate": integer,
  "framerateControl": enum,
  "rateControlMode": enum,
  "qvbrControlSettings": {
    "qvbrQualityLevel": integer,
    "maxAverageBitrate": integer
  },
  "codecProfile": enum,
  "tiles": enum,
  "telecine": enum,
  "framerateNumerator": integer,
  "minIInterval": integer,
  "adaptiveQuantization": enum,
  "codecLevel": enum,
  "sceneChangeDetect": enum,
  "qualityTuningLevel": enum,
  "framerateConversionAlgorithm": enum,
  "unregisteredSeiTimecode": enum,
  "gopSizeUnits": enum,
  "parControl": enum,
  "numberBFramesBetweenReferenceFrames": integer,
  "temporalIds": enum,
  "sampleAdaptiveOffsetFilterMode": enum,
  "writeM4pPackagingType": enum,
  "dynamicSubGop": enum
},
"softness": integer,
"framerateDenominator": integer,
"gopClosedCadence": integer,
"hrdBufferInitialFillPercentage": integer,
"gopSize": number,
"hrdBufferSize": integer,
"maxBitrate": integer,
"slowPal": enum,
"parDenominator": integer,
"spatialAdaptiveQuantization": enum,
"temporalAdaptiveQuantization": enum,
"bitrate": integer,
"intraDcPrecision": enum,
"framerateControl": enum,
"rateControlMode": enum,
"codecProfile": enum,
"telecine": enum,
"framerateNumerator": integer,
"minIInterval": integer,
"adaptiveQuantization": enum,
"codecLevel": enum,
"sceneChangeDetect": enum,
"qualityTuningLevel": enum,
"framerateConversionAlgorithm": enum,
"gopSizeUnits": enum,
"parControl": enum,
"numberBFramesBetweenReferenceFrames": integer,
"dynamicSubGop": enum
},
"proresSettings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "framerateDenominator": integer,
  "codecProfile": enum,
  "slowPal": enum,
  "parDenominator": integer,
  "framerateControl": enum,
  "telecine": enum,
  "framerateNumerator": integer,
  "framerateConversionAlgorithm": enum,
  "parControl": enum
}
},
"afdSignaling": enum,
"dropFrameTimecode": enum,
"respondToAfd": enum,
"colorMetadata": enum
},
"audioDescriptions": [
  {
    "audioTypeControl": enum,
    "audioSourceName": "string",
    "audioNormalizationSettings": {
      "algorithm": enum,
      "algorithmControl": enum,
      "correctionGateLevel": integer,
      "loudnessLogging": enum,
      "targetLkfs": number,
      "peakCalculation": enum
    },
    "codecSettings": {
      "codec": enum,
      "aacSettings": {
        "audioDescriptionBroadcasterMix": enum,
        "vbrQuality": enum,
        "bitrate": integer,
    
  
  },
"rateControlMode": enum,
"codecProfile": enum,
"codingMode": enum,
"rawFormat": enum,
"sampleRate": integer,
"specification": enum
},
"ac3Settings": {
"bitrate": integer,
"bitstreamMode": enum,
"codingMode": enum,
"dialnorm": integer,
"dynamicRangeCompressionProfile": enum,
"metadataControl": enum,
"lfeFilter": enum,
"sampleRate": integer
},
"aiffSettings": {
"bitDepth": integer,
"channels": integer,
"sampleRate": integer
},
"eac3Settings": {
"metadataControl": enum,
"surroundEXMode": enum,
"loRoSurroundMixLevel": number,
"phaseControl": enum,
"dialnorm": integer,
"ltRtSurroundMixLevel": number,
"bitrate": integer,
"ltRtCenterMixLevel": number,
"passthroughControl": enum,
"lfeControl": enum,
"loRoCenterMixLevel": number,
"attenuationControl": enum,
"codingMode": enum,
"surroundMode": enum,
"bitstreamMode": enum,
"lfeFilter": enum,
"stereoDownmix": enum,
"dynamicRangeCompressionRf": enum,
"sampleRate": integer,
"dynamicRangeCompressionLine": enum,
"dcFilter": enum
},
"mp2Settings": {
"bitrate": integer,
"channels": integer,
"sampleRate": integer
},
"wavSettings": {
"bitDepth": integer,
"channels": integer,
"sampleRate": integer,
"format": enum
},
"remixSettings": {
"channelMapping": {
"outputChannels": [
{
"inputChannels": [
integer
]
}
]
`EncoderSettings`

- `channelsIn`: integer,
- `channelsOut`: integer
- `streamName`: string,
- `languageCodeControl`: enum,
- `audioType`: integer,
- `customLanguageCode`: string,
- `languageCode`: enum

`outputSettings`

- `hlsSettings`
  - `audioGroupId`: string,
  - `audioRenditionSets`: string,
  - `audioTrackType`: enum,
  - `iFrameOnlyManifest`: enum,
  - `segmentModifier`: string

`extension`: string,
`nameModifier`: string,
`captionDescriptions`: [
- `captionSelectorName`: string,
- `destinationSettings`
  - `destinationType`: enum,
  - `burninDestinationSettings`
    - `backgroundOpacity`: integer,
    - `shadowXOffset`: integer,
    - `teletextSpacing`: enum,
    - `alignment`: enum,
    - `outlineSize`: integer,
    - `yPosition`: integer,
    - `shadowColor`: enum,
    - `fontOpacity`: integer,
    - `fontSize`: integer,
    - `fontScript`: enum,
    - `fontColor`: enum,
    - `backgroundColor`: enum,
    - `fontResolution`: integer,
    - `outlineColor`: enum,
    - `shadowYOffset`: integer,
    - `xPosition`: integer,
    - `shadowOpacity`: integer
  - `dvbSubDestinationSettings`
    - `backgroundOpacity`: integer,
    - `shadowXOffset`: integer,
    - `teletextSpacing`: enum,
    - `alignment`: enum,
    - `outlineSize`: integer,
    - `yPosition`: integer,
    - `shadowColor`: enum,
    - `fontOpacity`: integer,
    - `fontSize`: integer,
    - `fontScript`: enum,
    - `fontColor`: enum,
    - `backgroundColor`: enum,
    - `fontResolution`: integer,
    - `outlineColor`: enum,
    - `shadowYOffset`: integer,
    - `xPosition`: integer,
    - `shadowOpacity`: integer
  - `sccDestinationSettings`
    - `backgroundOpacity`: integer,
    - `shadowXOffset`: integer,
    - `teletextSpacing`: enum,
    - `alignment`: enum,
    - `outlineSize`: integer,
    - `yPosition`: integer,
    - `shadowColor`: enum,
    - `fontOpacity`: integer,
    - `fontSize`: integer,
    - `fontScript`: enum,
    - `fontColor`: enum,
    - `backgroundColor`: enum,
    - `fontResolution`: integer,
    - `outlineColor`: enum,
    - `shadowYOffset`: integer,
    - `xPosition`: integer,
    - `shadowOpacity`: integer
]
"framerate": enum
},
"teletextDestinationSettings": {
"pageNumber": "string"
},
"ttmlDestinationSettings": {
"stylePassthrough": enum
},
"embeddedDestinationSettings": {
"destination608ChannelNumber": integer
},
"customLanguageCode": "string",
"languageCode": enum,
"languageDescription": "string"
}]
]
},
"outputGroupSettings": {
"type": enum,
"hlsGroupSettings": {
"manifestDurationFormat": enum,
"segmentLength": integer,
"timedMetadataId3Period": integer,
"captionLanguageSetting": enum,
"captionLanguageMappings": [
{
"captionChannel": integer,
"customLanguageCode": "string",
"languageCode": enum,
"languageDescription": "string"
}
],
"destination": "string",
"destinationSettings": {
"s3Settings": {
"encryption": {
"encryptionType": enum,
"kmsKeyArn": "string"
}
}
},
"encryption": {
"encryptionMethod": enum,
"constantInitializationVector": "string",
"initializationVectorInManifest": enum,
"offlineEncrypted": enum,
"spekeKeyProvider": {
"resourceId": "string",
"systemIds": [
"string"
],
"url": "string",
"certificateArn": "string"
},
"staticKeyProvider": {
"staticKeyValue": "string",
"keyFormat": "string",
"keyFormatVersions": "string",
"url": "string"
},
"type": enum
},
"timedMetadataId3Frame": enum,
"baseUrl": "string"
"codecSpecification": enum,
"outputSelection": enum,
"programDateTimePeriod": integer,
"segmentsPerSubdirectory": integer,
"minSegmentLength": integer,
"minFinalSegmentLength": number,
"directoryStructure": enum,
"programDateTime": enum,
"adMarkers": [
    enum
],
"segmentControl": enum,
"timestampDeltaMilliseconds": integer,
"manifestCompression": enum,
"clientCache": enum,
"streamInfResolution": enum
},
"dashIsoGroupSettings": {
    "segmentLength": integer,
    "destination": "string",
    "destinationSettings": {
        "s3Settings": {
            "encryption": {
                "encryptionType": enum,
                "kmsKeyArn": "string"
            }
        }
    },
    "encryption": {
        "spekeKeyProvider": {
            "resourceId": "string",
            "systemIds": [
                "string"
            ],
            "url": "string",
            "certificateArn": "string"
        }
    },
    "minBufferTime": integer,
    "fragmentLength": integer,
    "baseUrl": "string",
    "segmentControl": enum,
    "hbbtvCompliance": enum,
    "writeSegmentTimelineInRepresentation": enum
},
"fileGroupSettings": {
    "destination": "string",
    "destinationSettings": {
        "s3Settings": {
            "encryption": {
                "encryptionType": enum,
                "kmsKeyArn": "string"
            }
        }
    }
},
"msSmoothGroupSettings": {
    "destination": "string",
    "destinationSettings": {
        "s3Settings": {
            "encryption": {
                "encryptionType": enum,
                "kmsKeyArn": "string"
            }
        }
    }
}
"fragmentLength": integer,
"encryption": {
  "spekeKeyProvider": {
    "resourceId": "string",
    "systemIds": ["string"
  ],
  "url": "string",
  "certificateArn": "string"
}
},
"manifestEncoding": enum,
"audioDeduplication": enum
},
"cmafGroupSettings": {
  "writeHlsManifest": enum,
  "writeDashManifest": enum,
  "segmentLength": integer,
  "minFinalSegmentLength": number,
  "destination": "string",
  "destinationSettings": {
    "s3Settings": {
      "encryption": {
        "encryptionType": enum,
        "kmsKeyArn": "string"
      }
    }
  },
  "encryption": {
    "encryptionMethod": enum,
    "constantInitializationVector": "string",
    "initializationVectorInManifest": enum,
    "staticKeyProvider": {
      "staticKeyValue": "string",
      "KeyFormat": "string",
      "keyFormatVersions": "string",
      "url": "string"
    },
    "type": enum
  },
  "minBufferTime": integer,
  "fragmentLength": integer,
  "baseUrl": "string",
  "segmentControl": enum,
  "manifestDurationFormat": enum,
  "streamInfResolution": enum,
  "clientCache": enum,
  "manifestCompression": enum,
  "codecSpecification": enum
}
},
"adAvailOffset": integer,
"availBlanking": {
  "availBlankingImage": "string"
},
"timedMetadataInsertion": {
  "id3Insertions": [
    {
      "timecode": "string",
      "id3": "string"
    }
  ]
},
"nielsenConfiguration": {
"breakoutCode": integer,
"distributorId": "string"
},
"motionImageInserter": {
"insertionMode": enum,
"input": "string",
"offset": {
"imageX": integer,
"imageY": integer
},
"startTime": "string",
"playback": enum,
"framerate": {
"framerateNumerator": integer,
"framerateDenominator": integer
}
},
"esam": {
"signalProcessingNotification": {
"sccXml": "string"
},
"manifestConfirmConditionNotification": {
"mccXml": "string"
},
"responseSignalPreroll": integer
},
"inputs": [
{
"inputClippings": [
{
"endTimecode": "string",
"startEndTimecode": "string"
}
],
"audioSelectors": {
},
"audioSelectorGroups": {
},
"programNumber": integer,
"videoSelector": {
"colorSpace": enum,
"rotate": enum,
"pid": integer,
"programNumber": integer,
"colorSpaceUsage": enum,
"hdr10Metadata": {
"redPrimaryX": integer,
"redPrimaryY": integer,
"greenPrimaryX": integer,
"greenPrimaryY": integer,
"bluePrimaryX": integer,
"bluePrimaryY": integer,
"whitePointX": integer,
"whitePointY": integer,
"maxFrameAverageLightLevel": integer,
"maxContentLightLevel": integer,
"maxLuminance": integer,
"minLuminance": integer
}
},
"filterEnable": enum,
"psiControl": enum,
"filterStrength": integer,
"deblockFilter": enum,
"denoiseFilter": enum,
"timecodeSource": enum,
"captionSelectors": {
},
"imageInserter": {
    "insertableImages": [
    {
        "width": integer,
        "height": integer,
        "imageX": integer,
        "imageY": integer,
        "duration": integer,
        "fadeIn": integer,
        "layer": integer,
        "imageInserterInput": "string",
        "startTime": "string",
        "fadeOut": integer,
        "opacity": integer
    }
    ]
},
"accelerationSettings": {
    "mode": enum
},
"statusUpdateInterval": enum
}

Example DELETE

{
  "name": "string"
}

Response Bodies

Example GetJobTemplateResponse

{
  "jobTemplate": {
    "arn": "string",
    "createdAt": "string",
    "lastUpdated": "string",
    "description": "string",
    "category": "string",
    "queue": "string",
    "name": "string",
    "type": enum,
    "settings": {
      "timecodeConfig": {
        "anchor": "string",
        "source": enum,
        "start": "string",
        "timestampOffset": "string"
      },
      "outputGroups": [
      {
        "customName": "string",
        "name": "string",
        "outputs": [
        {
          "containerSettings": {


253
"container": enum,
"m3u8Settings": {
  "audioFramesPerPes": integer,
  "pcrControl": enum,
  "pcrPid": integer,
  "pmtPid": integer,
  "privateMetadataPid": integer,
  "programNumber": integer,
  "patInterval": integer,
  "pmtInterval": integer,
  "scte35Source": enum,
  "scte35Pid": integer,
  "nielsenId3": enum,
  " timedMetadata": enum,
  "timedMetadataPid": integer,
  "transportStreamId": integer,
  "videoPid": integer,
  "audioPids": [
    integer
  ]
},
"f4vSettings": {
  "moovPlacement": enum
},
"m2tsSettings": {
  "audioBufferModel": enum,
  "minEbpInterval": integer,
  "esRateInPes": enum,
  "patInterval": integer,
  "dvbNitSettings": {
    "nitInterval": integer,
    "networkId": integer,
    "networkName": "string"
  },
  "dvbSdtSettings": {
    "outputSdt": enum,
    "sdtInterval": integer,
    "serviceName": "string",
    "serviceProviderName": "string"
  },
  "scte35Source": enum,
  "scte35Pid": integer,
  "scte35Esam": {
    "scte35EsamPid": integer
  },
  "videoPid": integer,
  "dvbTdtSettings": {
    "tdtInterval": integer
  },
  "pmtInterval": integer,
  "segmentationStyle": enum,
  "segmentationTime": number,
  "pmtPid": integer,
  "bitrate": integer,
  "audioPids": [
    integer
  ],
  "privateMetadataPid": integer,
  "nielsenId3": enum,
  "timedMetadataPid": integer,
  "maxPcrInterval": integer,
  "transportStreamId": integer,
  "dvbSubPids": [
    integer
  ],
  "rateMode": enum,
"audioFramesPerPes": integer,
"pcrControl": enum,
"segmentationMarkers": enum,
"ebpAudioInterval": enum,
"forceTsVideoEbpOrder": enum,
"programNumber": integer,
"pcrPid": integer,
"bufferModel": enum,
"dubTeletextPid": integer,
"segmentationMarkers": enum,
"nullPacketBitrate": number
},
"movSettings": {
    "clapAtom": enum,
    "cslgAtom": enum,
    "paddingControl": enum,
    "reference": enum,
    "mpeg2FourCCControl": enum
},
"mp4Settings": {
    "cslgAtom": enum,
    "freeSpaceBox": enum,
    "mp4MajorBrand": "string",
    "moovPlacement": enum
},
"preset": "string",
"videoDescription": {
    "fixedAfd": integer,
    "width": integer,
    "scalingBehavior": enum,
    "crop": {
        "height": integer,
        "width": integer,
        "x": integer,
        "y": integer
    },
    "height": integer,
    "videoPreprocessors": {
        "colorCorrector": {
            "brightness": integer,
            "colorSpaceConversion": enum,
            "contrast": integer,
            "hue": integer,
            "saturation": integer,
            "hdr10Metadata": {
                "redPrimaryX": integer,
                "redPrimaryY": integer,
                "greenPrimaryX": integer,
                "greenPrimaryY": integer,
                "bluePrimaryX": integer,
                "bluePrimaryY": integer,
                "whitePointX": integer,
                "whitePointY": integer,
                "maxFrameAverageLightLevel": integer,
                "maxContentLightLevel": integer,
                "maxLuminance": integer,
                "minLuminance": integer
            }
        },
        "deinterlacer": {
            "algorithm": enum,
            "mode": enum,
            "control": enum
        }
    }
}
"imageInserter": {
  "insertableImages": [
    {
      "width": integer,
      "height": integer,
      "imageX": integer,
      "imageY": integer,
      "duration": integer,
      "fadeIn": integer,
      "layer": integer,
      "imageInserterInput": "string",
      "startTime": "string",
      "fadeOut": integer,
      "opacity": integer
    }
  ]
},

"noiseReducer": {
  "filter": enum,
  "filterSettings": {
    "strength": integer
  },
  "spatialFilterSettings": {
    "strength": integer,
    "speed": integer,
    "postFilterSharpenStrength": integer
  }
},

"timecodeBurnin": {
  "fontSize": integer,
  "position": enum,
  "prefix": "string"
},

"timecodeInsertion": enum,
"antiAlias": enum,
"position": {
  "height": integer,
  "width": integer,
  "x": integer,
  "y": integer
},

"sharpness": integer,
"codecSettings": {
  "codec": enum,
  "framerateSettings": {
    "framerateNumerator": integer,
    "framerateDenominator": integer,
    "maxCaptures": integer,
    "quality": integer
  },
  "h264Settings": {
    "interlaceMode": enum,
    "parNumerator": integer,
    "numberReferenceFrames": integer,
    "syntax": enum,
    "softness": integer,
    "framerateDenominator": integer,
    "gopClosedCadence": integer,
    "hrdBufferInitialFillPercentage": integer,
    "gopSize": number,
    "slices": integer,
    "gopBReference": enum,
    "hrdBufferSize": integer,
    "maxBitrate": integer,
    "slowPal": enum,
    "maxBitrate": integer
  }
}
"parDenominator": integer,
"spatialAdaptiveQuantization": enum,
"temporalAdaptiveQuantization": enum,
"flickerAdaptiveQuantization": enum,
"entropyEncoding": enum,
"bitrate": integer,
"framerateControl": enum,
"rateControlMode": enum,
"qvbrSettings": {
  "qvbrQualityLevel": integer,
  "maxAverageBitrate": integer
},
"codecProfile": enum,
"telecine": enum,
"framerateNumerator": integer,
"minIInterval": integer,
"adaptiveQuantization": enum,
"codecLevel": enum,
"fieldEncoding": enum,
"sceneChangeDetect": enum,
"qualityTuningLevel": enum,
"framerateConversionAlgorithm": enum,
"unregisteredSeiTimecode": enum,
"gopSizeUnits": enum,
"parControl": enum,
"numberBFramesBetweenReferenceFrames": integer,
"repeatPps": enum,
"dynamicSubGop": enum
},
"h265Settings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "numberReferenceFrames": integer,
  "framerateDenominator": integer,
  "gopClosedCadence": integer,
  "alternateTransferFunctionSei": enum,
  "hrdBufferInitialFillPercentage": integer,
  "gopSize": number,
  "slices": integer,
  "gopReference": enum,
  "hrdBufferSize": integer,
  "maxBitrate": integer,
  "slowPal": enum,
  "parDenominator": integer,
  "spatialAdaptiveQuantization": enum,
  "temporalAdaptiveQuantization": enum,
  "flickerAdaptiveQuantization": enum,
  "bitrate": integer,
  "framerateControl": enum,
  "rateControlMode": enum,
  "qvbrSettings": {
    "qvbrQualityLevel": integer,
    "maxAverageBitrate": integer
  },
  "codecProfile": enum,
  "tiles": enum,
  "telecine": enum,
  "framerateNumerator": integer,
  "minIInterval": integer,
  "adaptiveQuantization": enum,
  "codecLevel": enum,
  "sceneChangeDetect": enum,
  "qualityTuningLevel": enum,
  "framerateConversionAlgorithm": enum,
  "unregisteredSeiTimecode": enum,
  "gopSizeUnits": enum,
  "parDenominator": integer,
  "spatialAdaptiveQuantization": enum,
  "temporalAdaptiveQuantization": enum,
  "flickerAdaptiveQuantization": enum,
  "bitrate": integer,
  "framerateControl": enum,
  "rateControlMode": enum,
  "qvbrSettings": {
    "qvbrQualityLevel": integer,
    "maxAverageBitrate": integer
  },
  "codecProfile": enum,
  "tiles": enum,
  "telecine": enum,
  "framerateNumerator": integer,
  "minIInterval": integer,
  "adaptiveQuantization": enum,
  "codecLevel": enum,
  "sceneChangeDetect": enum,
  "qualityTuningLevel": enum,
  "framerateConversionAlgorithm": enum,
  "unregisteredSeiTimecode": enum,
  "gopSizeUnits": enum,
"parControl": enum,
"numberBFramesBetweenReferenceFrames": integer,
"temporalIds": enum,
"sampleAdaptiveOffsetFilterMode": enum,
"writeMp4PackagingType": enum,
"dynamicSubGop": enum
},
"mpeg2Settings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "syntax": enum,
  "softness": integer,
  "framerateDenominator": integer,
  "gopClosedCadence": integer,
  "hrdBufferInitialFillPercentage": integer,
  "gopSize": number,
  "hrdBufferSize": integer,
  "maxBitrate": integer,
  "slowPal": enum,
  "parDenominator": integer,
  "spatialAdaptiveQuantization": enum,
  "temporalAdaptiveQuantization": enum,
  "bitrate": integer,
  "intraDcPrecision": enum,
  "framerateControl": enum,
  "rateControlMode": enum,
  "codecProfile": enum,
  "telecine": enum,
  "framerateNumerator": integer,
  "minIInterval": integer,
  "adaptiveQuantization": enum,
  "codecLevel": enum,
  "sceneChangeDetect": enum,
  "qualityTuningLevel": enum,
  "framerateConversionAlgorithm": enum,
  "gopSizeUnits": enum,
  "parControl": enum,
  "numberBFramesBetweenReferenceFrames": integer,
  "dynamicSubGop": enum
},
"proresSettings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "framerateDenominator": integer,
  "codecProfile": enum,
  "slowPal": enum,
  "parDenominator": integer,
  "framerateControl": enum,
  "telecine": enum,
  "framerateNumerator": integer,
  "framerateConversionAlgorithm": enum,
  "parControl": enum
}
},
"afdSignaling": enum,
"dropFrameTimecode": enum,
"respondToAfd": enum,
"colorMetadata": enum
},
"audioDescriptions": [
{
  "audioTypeControl": enum,
  "audioSourceName": "string",
  "audioNormalizationSettings": {
    "algorithm": enum,
    "algorithmControl": enum,
  }
}
Schemas

"correctionGateLevel": integer,
"loudnessLogging": enum,
"targetLkfs": number,
"peakCalculation": enum
},
"codecSettings": {
  "codec": enum,
  "aacSettings": {
    "audioDescriptionBroadcasterMix": enum,
    "vbrQuality": enum,
    "bitrate": integer,
    "rateControlMode": enum,
    "codecProfile": enum,
    "codingMode": enum,
    "rawFormat": enum,
    "sampleRate": integer,
    "specification": enum
  },
  "ac3Settings": {
    "bitrate": integer,
    "bitstreamMode": enum,
    "codingMode": enum,
    "dialnorm": integer,
    "dynamicRangeCompressionProfile": enum,
    "metadataControl": enum,
    "lfeFilter": enum,
    "sampleRate": integer
  },
  "aiffSettings": {
    "bitDepth": integer,
    "channels": integer,
    "sampleRate": integer
  },
  "eac3Settings": {
    "metadataControl": enum,
    "surroundExMode": enum,
    "loRoSurroundMixLevel": number,
    "phaseControl": enum,
    "dialnorm": integer,
    "ltRtSurroundMixLevel": number,
    "bitrate": integer,
    "ltRtCenterMixLevel": number,
    "passthroughControl": enum,
    "lfeControl": enum,
    "loRoCenterMixLevel": number,
    "attenuationControl": enum,
    "codingMode": enum,
    "surroundMode": enum,
    "bitstreamMode": enum,
    "lfeFilter": enum,
    "stereoDownmix": enum,
    "dynamicRangeCompressionProfile": enum,
    "sampleRate": integer,
    "dynamicRangeCompressionProfile": enum,
    "dcFilter": enum
  },
  "mp2Settings": {
    "bitrate": integer,
    "channels": integer,
    "sampleRate": integer
  },
  "wavSettings": {
    "bitDepth": integer,
    "channels": integer,
    "sampleRate": integer,
    "format": enum
  }
}
"remixSettings": {
    "channelMapping": {
        "outputChannels": [
            {
                "inputChannels": [
                    integer
                ]
            }
        ]
    },
    "channelsIn": integer,
    "channelsOut": integer
},
"streamName": "string",
"languageCodeControl": enum,
"audioType": integer,
"customLanguageCode": "string",
"languageCode": enum
],
"outputSettings": {
    "hlsSettings": {
        "audioGroupId": "string",
        "audioRenditionSets": "string",
        "audioTrackType": enum,
        "iFrameOnlyManifest": enum,
        "segmentModifier": "string"
    }
},
"extension": "string",
"nameModifier": "string",
"captionDescriptions": [
    {
        "captionSelectorName": "string",
        "destinationSettings": {
            "destinationType": enum,
            "burninDestinationSettings": {
                "backgroundColor": integer,
                "shadowXOffset": integer,
                "teletextSpacing": enum,
                "alignment": enum,
                "outlineSize": integer,
                "yPosition": integer,
                "shadowColor": enum,
                "fontOpacity": integer,
                "fontSize": integer,
                "fontScript": enum,
                "fontColor": enum,
                "backgroundColor": enum,
                "fontResolution": integer,
                "outlineColor": enum,
                "shadowYOffset": integer,
                "xPosition": integer,
                "shadowOpacity": integer
            },
            "dvbSubDestinationSettings": {
                "backgroundColor": integer,
                "shadowXOffset": integer,
                "teletextSpacing": enum,
                "alignment": enum,
                "outlineSize": integer,
                "yPosition": integer,
                "shadowColor": enum,
                "fontOpacity": integer
            }
        }
    }
]
"fontSize": integer,
"fontScript": enum,
"fontColor": enum,
"backgroundColor": enum,
"fontResolution": integer,
"outlineColor": enum,
"shadowYOffset": integer,
"xPosition": integer,
"shadowOpacity": integer
},
"sccDestinationSettings": {
  "framerate": enum
},
"teletextDestinationSettings": {
  "pageNumber": "string"
},
"ttmlDestinationSettings": {
  "stylePassthrough": enum
},
"embeddedDestinationSettings": {
  "destination608ChannelNumber": integer
},
"customLanguageCode": "string",
"languageCode": enum,
"languageDescription": "string"
]}
],
"outputGroupSettings": {
  "type": enum,
  "hlsGroupSettings": {
    "manifestDurationFormat": enum,
    "segmentLength": integer,
    "timedMetadataId3Period": integer,
    "captionLanguageSetting": enum,
    "captionLanguageMappings": [
      {
        "captionChannel": integer,
        "customLanguageCode": "string",
        "languageCode": enum,
        "languageDescription": "string"
      }
    ],
    "destination": "string",
    "destinationSettings": {
      "s3Settings": {
        "encryption": {
          "encryptionType": enum,
          "kmsKeyArn": "string"
        }
      }
    }
  }
},
"encryption": {
  "encryptionMethod": enum,
  "constantInitializationVector": "string",
  "initializationVectorInManifest": enum,
  "offlineEncrypted": enum,
  "spekeKeyProvider": {
    "resourceId": "string",
    "systemIds": [
      "string"
    ],
    "url": "string",
    "certificateArn": "string"
Schemas

```
{
  "staticKeyProvider": {
    "staticKeyValue": "string",
    "keyFormat": "string",
    "keyFormatVersions": "string",
    "url": "string"
  },
  "type": enum
},
  "timedMetadataId3Frame": enum,
  "baseUrl": "string",
  "codecSpecification": enum,
  "outputSelection": enum,
  "programDateTimePeriod": integer,
  "segmentsPerSubdirectory": integer,
  "minSegmentLength": integer,
  "minFinalSegmentLength": number,
  "directoryStructure": enum,
  "programDateTime": enum,
  "adMarkers": [
    enum
  ],
  "segmentControl": enum,
  "timestampDeltaMilliseconds": integer,
  "manifestCompression": enum,
  "clientCache": enum,
  "streamInfResolution": enum
},
  "dashIsoGroupSettings": {
    "segmentLength": integer,
    "destination": "string",
    "destinationSettings": {
      "s3Settings": {
        "encryption": {
          "encryptionType": enum,
          "kmsKeyArn": "string"
        }
      }
    },
    "encryption": {
      "spekeKeyProvider": {
        "resourceId": "string",
        "systemIds": [
          "string"
        ],
        "url": "string",
        "certificateArn": "string"
      }
    },
    "minBufferTime": integer,
    "fragmentLength": integer,
    "baseUrl": "string",
    "segmentControl": enum,
    "hbbtvCompliance": enum,
    "writeSegmentTimelineInRepresentation": enum
  },
  "fileGroupSettings": {
    "destination": "string",
    "destinationSettings": {
      "s3Settings": {
        "encryption": {
          "encryptionType": enum,
          "kmsKeyArn": "string"
        }
      }
    }
  }
}```
"msSmoothGroupSettings": {
  "destination": "string",
  "destinationSettings": {
    "s3Settings": {
      "encryption": {
        "encryptionType": enum,
        "kmsKeyArn": "string"
      }
    }
  },
  "fragmentLength": integer,
  "encryption": {
    "spekeKeyProvider": {
      "resourceId": "string",
      "systemIds": ["string"
      ],
      "url": "string",
      "certificateArn": "string"
    }
  },
  "manifestEncoding": enum,
  "audioDeduplication": enum
},
"cmafGroupSettings": {
  "writeHlsManifest": enum,
  "writeDashManifest": enum,
  "segmentLength": integer,
  "minFinalSegmentLength": number,
  "destination": "string",
  "destinationSettings": {
    "s3Settings": {
      "encryption": {
        "encryptionType": enum,
        "kmsKeyArn": "string"
      }
    }
  },
  "encryption": {
    "encryptionMethod": enum,
    "constantInitializationVector": "string",
    "initializationVectorInManifest": enum,
    "staticKeyProvider": {
      "StaticKeyValue": "string",
      "keyFormat": "string",
      "keyFormatVersions": "string",
      "url": "string"
    },
    "type": enum
  },
  "minBufferTime": integer,
  "fragmentLength": integer,
  "baseUrl": "string",
  "segmentControl": enum,
  "manifestDurationFormat": enum,
  "streamInfResolution": enum,
  "clientCache": enum,
  "manifestCompression": enum,
  "codecSpecification": enum
}
}]
"adAvailOffset": integer,
"availBlanking": {
"availBlankingImage": "string",
"timedMetadataInsertion": {
  "id3Insertions": [
    {
      "timecode": "string",
      "id3": "string"
    }
  ]
},
"nielsenConfiguration": {
  "breakoutCode": integer,
  "distributorId": "string"
},
"motionImageInserter": {
  "insertionMode": enum,
  "input": "string",
  "offset": {
    "imageX": integer,
    "imageY": integer
  },
  "startTime": "string",
  "playback": enum,
  "framerate": {
    "framerateNumerator": integer,
    "framerateDenominator": integer
  }
},
"esam": {
  "signalProcessingNotification": {
    "sccXml": "string"
  },
  "manifestConfirmConditionNotification": {
    "mccXml": "string"
  },
  "responseSignalPreroll": integer
},
"inputs": [
  {
    "inputClippings": [
      {
        "endTimecode": "string",
        "startTimecode": "string"
      }
    ],
    "audioSelectors": {},
    "audioSelectorGroups": {}
  },
  "programNumber": integer,
  "videoSelector": {
    "colorSpace": enum,
    "rotate": enum,
    "pid": integer,
    "programNumber": integer,
    "colorSpaceUsage": enum,
    "hdr10Metadata": {
      "redPrimaryX": integer,
      "redPrimaryY": integer,
      "greenPrimaryX": integer,
      "greenPrimaryY": integer,
      "bluePrimaryX": integer,
      "bluePrimaryY": integer,
      "whitePointX": integer,
      "whitePointY": integer,
      "maxFrameAverageLightLevel": integer
    }
  }
]
"maxContentLightLevel": integer,
"maxLuminance": integer,
"minLuminance": integer
},
"filterEnable": enum,
"psiControl": enum,
"filterStrength": integer,
"deblockFilter": enum,
"denoiseFilter": enum,
"timecodeSource": enum,
"captionSelectors": {
},
"imageInserter": {
  "insertableImages": [
    {
      "width": integer,
      "height": integer,
      "imageX": integer,
      "imageY": integer,
      "duration": integer,
      "fadeIn": integer,
      "layer": integer,
      "imageInserterInput": "string",
      "startTime": "string",
      "fadeOut": integer,
      "opacity": integer
    }
  ]
},
"accelerationSettings": {
  "mode": enum
},
"statusUpdateInterval": enum
}

Example UpdateJobTemplateResponse

{  "jobTemplate": {
    "arn": "string",
    "createdAt": "string",
    "lastUpdated": "string",
    "description": "string",
    "category": "string",
    "queue": "string",
    "name": "string",
    "type": enum,
    "settings": {
      "timecodeConfig": {
        "anchor": "string",
        "source": enum,
        "start": "string",
        "timestampOffset": "string"
      },
      "outputGroups": [
        {
          "customName": "string",
          "name": "string",
          "outputs": [
            "string"
          ]
        }
      ]
    }
  }
}
{  
  "containerSettings": {  
    "container": enum,  
    "m3u8Settings": {  
      "audioFramesPerPes": integer,  
      "pcrControl": enum,  
      "pcrPid": integer,  
      "pmtPid": integer,  
      "privateMetadataPid": integer,  
      "programNumber": integer,  
      "patInterval": integer,  
      "pmtInterval": integer,  
      "scte35Source": enum,  
      "scte35Pid": integer,  
      "nielsenId3": enum,  
      "timedMetadata": enum,  
      "timedMetadataPid": integer,  
      "transportStreamId": integer,  
      "videoPid": integer,  
      "audioPids": [  
        integer  
      ]  
    },  
    "f4vSettings": {  
      "moovPlacement": enum  
    },  
    "m2tsSettings": {  
      "audioBufferModel": enum,  
      "minEbpInterval": integer,  
      "esRateInPes": enum,  
      "patInterval": integer,  
      "dvbNitSettings": {  
        "nitInterval": integer,  
        "networkId": integer,  
        "networkName": "string"  
      },  
      "dvbSdtSettings": {  
        "outputSdt": enum,  
        "sdtInterval": integer,  
        "serviceName": "string",  
        "serviceProviderName": "string"  
      },  
      "scte35Source": enum,  
      "scte35Pid": integer,  
      "scte35Esam": {  
        "scte35EsamPid": integer  
      },  
      "videoPid": integer,  
      "dvbTdtSettings": {  
        "tdtInterval": integer  
      },  
      "pmtInterval": integer,  
      "segmentationStyle": enum,  
      "segmentationTime": number,  
      "pmtPid": integer,  
      "bitrate": integer,  
      "audioPids": [  
        integer  
      ],  
      "privateMetadataPid": integer,  
      "nielsenId3": enum,  
      "timedMetadataPid": integer,  
      "maxPcrInterval": integer,  
      "transportStreamId": integer,  
      "dvbSubPids": [  
        integer  
      ]  
  }  
}
},
"rateMode": enum,
"audioFramesPerPES": integer,
"pcrControl": enum,
"segmentationMarkers": enum,
"ebpAudioInterval": enum,
"forcedTsVideoEbpOrder": enum,
"programNumber": integer,
"pcrPID": integer,
"bufferModel": enum,
"dvbTeletextPid": integer,
"fragmentTime": number,
"ebpPlacement": enum,
"nullPacketBitrate": number
},
"movSettings": {
  "clapAtom": enum,
  "celgAtom": enum,
  "paddingControl": enum,
  "reference": enum,
  "mpeg2FourCCControl": enum
},
"mp4Settings": {
  "celgAtom": enum,
  "freeSpaceBox": enum,
  "mp4MajorBrand": "string",
  "moovPlacement": enum
}
},
"preset": "string",
"videoDescription": {
  "fixedAfd": integer,
  "width": integer,
  "scalingBehavior": enum,
  "crop": {
    "height": integer,
    "width": integer,
    "x": integer,
    "y": integer
  },
  "height": integer,
  "videoPreprocessors": {
    "colorCorrector": {
      "brightness": integer,
      "colorSpaceConversion": enum,
      "contrast": integer,
      "hue": integer,
      "saturation": integer,
      "hdr10Metadata": {
        "redPrimaryX": integer,
        "redPrimaryY": integer,
        "greenPrimaryX": integer,
        "greenPrimaryY": integer,
        "bluePrimaryX": integer,
        "bluePrimaryY": integer,
        "whitePointX": integer,
        "whitePointY": integer,
        "maxFrameAverageLightLevel": integer,
        "maxContentLightLevel": integer,
        "maxLuminance": integer,
        "minLuminance": integer
      }
    },
    "deinterlacer": {
      "algorithm": enum,
      "mode": enum,
"control": enum
},
"imageInserters": {
  "insertableImages": [
    {
      "width": integer,
      "height": integer,
      "imageX": integer,
      "imageY": integer,
      "duration": integer,
      "fadeIn": integer,
      "layer": integer,
      "imageInsertersInput": "string",
      "startTime": "string",
      "fadeOut": integer,
      "opacity": integer
    }
  ]
},
"noiseReducer": {
  "filter": enum,
  "filterSettings": {
    "strength": integer
  },
  "spatialFilterSettings": {
    "strength": integer,
    "speed": integer,
    "postFilterSharpenStrength": integer
  }
},
"timecodeBurnin": {
  "fontSize": integer,
  "position": enum,
  "prefix": "string"
},
"timecodeInsertion": enum,
"antiAlias": enum,
"position": {
  "height": integer,
  "width": integer,
  "x": integer,
  "y": integer
},
"sharpness": integer,
"codecSettings": {
  "codec": enum,
  "frameCaptureSettings": {
    "frameRateNumerator": integer,
    "frameRateDenominator": integer,
    "maxCaptures": integer,
    "quality": integer
  },
  "h264Settings": {
    "interlaceMode": enum,
    "parNumerator": integer,
    "numberReferenceFrames": integer,
    "syntax": enum,
    "softness": integer,
    "frameRateDenominator": integer,
    "gopClosedCadence": integer,
    "hrdBufferInitialFillPercentage": integer,
    "gopSize": number,
    "slices": integer,
    "gobReference": enum,
    "hrdBufferSize": integer,
"maxBitrate": integer,
"slowPal": enum,
"parDenominator": integer,
"spatialAdaptiveQuantization": enum,
"temporalAdaptiveQuantization": enum,
"flickerAdaptiveQuantization": enum,
"entropyEncoding": enum,
"bitrate": integer,
"framerateControl": enum,
"rateControlMode": enum,
"qvbrSettings": {
  "qvbrQualityLevel": integer,
  "maxAverageBitrate": integer
},
"codecProfile": enum,
"telecine": enum,
"framerateNumerator": integer,
"minIInterval": integer,
"adaptiveQuantization": enum,
"codecLevel": enum,
"fieldEncoding": enum,
"sceneChangeDetect": enum,
"qualityTuningLevel": enum,
"framerateConversionAlgorithm": enum,
"unregisteredSeiTimecode": enum,
"gopSizeUnits": enum,
"parControl": enum,
"numberBFramesBetweenReferenceFrames": integer,
"repeatPps": enum,
"dynamicSubGop": enum
},
"h265Settings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "numberReferenceFrames": integer,
  "framerateDenominator": integer,
  "gopClosedCadence": integer,
  "alternateTransferFunctionSei": enum,
  "hrdBufferInitialFillPercentage": integer,
  "gopSize": number,
  "slices": integer,
  "gopBReference": enum,
  "hrdBufferSize": integer,
  "maxBitrate": integer,
  "slowPal": enum,
  "parDenominator": integer,
  "spatialAdaptiveQuantization": enum,
  "temporalAdaptiveQuantization": enum,
  "flickerAdaptiveQuantization": enum,
  "bitrate": integer,
  "framerateControl": enum,
  "rateControlMode": enum,
  "qvbrSettings": {
    "qvbrQualityLevel": integer,
    "maxAverageBitrate": integer
  },
  "codecProfile": enum,
  "tiles": enum,
  "telecine": enum,
  "framerateNumerator": integer,
  "minIInterval": integer,
  "adaptiveQuantization": enum,
  "codecLevel": enum,
  "sceneChangeDetect": enum,
  "qualityTuningLevel": enum,
  "framerateConversionAlgorithm": enum,
"unregisteredSeiTimecode": enum,
"gopSizeUnits": enum,
"parControl": enum,
"numberBFramesBetweenReferenceFrames": integer,
"temporalIds": enum,
"sampleAdaptiveOffsetFilterMode": enum,
"writeMp4PackagingType": enum,
"dynamicSubGop": enum
},
"mpeg2Settings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "syntax": enum,
  "softness": integer,
  "framerateDenominator": integer,
  "gopClosedCadence": integer,
  "hrdBufferInitialFillPercentage": integer,
  "gopSize": number,
  "hrdBufferSize": integer,
  "maxBitrate": integer,
  "slowPal": enum,
  "parDenominator": integer,
  "spatialAdaptiveQuantization": enum,
  "temporalAdaptiveQuantization": enum,
  "bitrate": integer,
  "intraDcPrecision": enum,
  "framerateControl": enum,
  "rateControlMode": enum,
  "codecProfile": enum,
  "telecine": enum,
  "framerateNumerator": integer,
  "minIInterval": integer,
  "adaptiveQuantization": enum,
  "codecLevel": enum,
  "sceneChangeDetect": enum,
  "qualityTuningLevel": enum,
  "framerateConversionAlgorithm": enum,
  "gopSizeUnits": enum,
  "parControl": enum,
  "numberBFramesBetweenReferenceFrames": integer,
  "dynamicSubGop": enum
},
"proresSettings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "framerateDenominator": integer,
  "codecProfile": enum,
  "slowPal": enum,
  "parDenominator": integer,
  "framerateControl": enum,
  "telecine": enum,
  "framerateNumerator": integer,
  "framerateConversionAlgorithm": enum,
  "parControl": enum
}
},
"afdsignaling": enum,
"dropFrameTimecode": enum,
"respondToAfds": enum,
"colormetadata": enum
},
"audiodescriptions": [
{
"audiotypecontrol": enum,
"audiosourcename": "string",
"audionormalizationsettings": {

"algorithm": enum,
"algorithmControl": enum,
"correctionGateLevel": integer,
"loudnessLogging": enum,
"targetLkfs": number,
"peakCalculation": enum
},
"codecSettings": {
  "codec": enum,
  "aacSettings": {
    "audioDescriptionBroadcasterMix": enum,
    "vbrQuality": enum,
    "bitrate": integer,
    "rateControlMode": enum,
    "codecProfile": enum,
    "codingMode": enum,
    "rawFormat": enum,
    "sampleRate": integer,
    "specification": enum
  },
  "ac3Settings": {
    "bitrate": integer,
    "bitstreamMode": enum,
    "codingMode": enum,
    "dialnorm": integer,
    "dynamicRangeCompressionProfile": enum,
    "metadataControl": enum,
    "lfeFilter": enum,
    "sampleRate": integer
  },
  "aiffSettings": {
    "bitDepth": integer,
    "channels": integer,
    "sampleRate": integer
  },
  "eac3Settings": {
    "metadataControl": enum,
    "surroundExMode": enum,
    "loRoSurroundMixLevel": number,
    "phaseControl": enum,
    "dialnorm": integer,
    "ltRtSurroundMixLevel": number,
    "bitrate": integer,
    "ltRtCenterMixLevel": number,
    "passthroughControl": enum,
    "lfeControl": enum,
    "loRoCenterMixLevel": number,
    "attenuationControl": enum,
    "codingMode": enum,
    "surroundMode": enum,
    "bitstreamMode": enum,
    "lfeFilter": enum,
    "stereoDownmix": enum,
    "dynamicRangeCompressionRf": enum,
    "sampleRate": integer,
    "dynamicRangeCompressionLine": enum,
    "dcFilter": enum
  },
  "mp2Settings": {
    "bitrate": integer,
    "channels": integer,
    "sampleRate": integer
  },
  "wavSettings": {
    "bitDepth": integer,
    "channels": integer,
"sampleRate": integer,
"format": enum
},
"remixSettings": {
"channelMapping": {
"outputChannels": [
{
"inputChannels": [
integer
]
]
},
"channelsIn": integer,
"channelsOut": integer
},
"streamName": "string",
"languageCodeControl": enum,
"audioType": integer,
"customLanguageCode": "string",
"languageCode": enum
}
],
"outputSettings": {
"hlsSettings": {
"audioGroupId": "string",
"audioRenditionSets": "string",
"audioTrackType": enum,
"iFrameOnlyManifest": enum,
"segmentModifier": "string"
}
},
"extension": "string",
"nameModifier": "string",
"captionDescriptions": [
{
"captionSelectorName": "string",
"destinationSettings": {
"burninDestinationSettings": {
"backgroundColor": enum,
"shadowXOffset": integer,
"teletextSpacing": enum,
"alignment": enum,
"outlineSize": integer,
"yPosition": integer,
"shadowColor": enum,
"fontOpacity": integer,
"fontSize": integer,
"fontScript": enum,
"fontColor": enum,
"backgroundColor": enum,
"fontResolution": integer,
"outlineColor": enum,
"shadowXOffset": integer,
"xPosition": integer,
"shadowOpacity": integer
},
"dvbSubDestinationSettings": {
"backgroundOpacity": integer,
"shadowXOffset": integer,
"teletextSpacing": enum,
"alignment": enum,
"outlineSize": integer,
"yPosition": integer,
"shadowColor": enum,
"shadowColor": enum,
"fontOpacity": integer,
"fontSize": integer,
"fontScript": enum,
"fontColor": enum,
"backgroundColor": enum,
"fontResolution": integer,
"outlineColor": enum,
"shadowXOffset": integer,
"shadowYOffset": integer,
"shadowOpacity": integer
},
"sccDestinationSettings": {
  "framerate": enum,
},
"teletextDestinationSettings": {
  "pageNumber": "string"
},
"ttmlDestinationSettings": {
  "stylePassthrough": enum
},
"embeddedDestinationSettings": {
  "destination608ChannelNumber": integer
},
"customLanguageCode": "string",
"languageCode": enum,
"languageDescription": "string"
]}
],
"outputGroupSettings": {
  "type": enum,
  "hisGroupSettings": {
    "manifestDurationFormat": enum,
    "segmentLength": integer,
    "timedMetadataId3Period": integer,
    "captionLanguageSetting": enum,
    "captionLanguageMappings": [
      {
        "captionChannel": integer,
        "customLanguageCode": "string",
        "languageCode": enum,
        "languageDescription": "string"
      }
    ],
    "destination": "string",
    "destinationSettings": {
      "s3Settings": {
        "encryption": {
          "encryptionType": enum,
          "kmsKeyArn": "string"
        }
      },
      "encryption": {
        "encryptionMethod": enum,
        "constantInitializationVector": "string",
        "initializationVectorInManifest": enum,
        "offlineEncrypted": enum,
        "spekeKeyProvider": {
          "resourceId": "string",
          "systemIds": ["string"
        ]
      }
    }
  }
}
"url": "string",
"certificateArn": "string"
},
"staticKeyProvider": {
"staticKeyValue": "string",
"keyFormat": "string",
"keyFormatVersions": "string",
"url": "string"
},
"type": enum,
"timedMetadataId3Frame": enum,
"baseUrl": "string",
"codecSpecification": enum,
"outputSelection": enum,
"programDateTimePeriod": integer,
"segmentsPerSubdirectory": integer,
"minSegmentLength": integer,
"minFinalSegmentLength": number,
"directoryStructure": enum,
"programDateTime": enum,
"adMarkers": [enum
],
"segmentControl": enum,
"timestampDeltaMilliseconds": integer,
"manifestCompression": enum,
"clientCache": enum,
"streamInfResolution": enum
},
"dashIsoGroupSettings": {
"segmentLength": integer,
"destination": "string",
"destinationSettings": {
"s3Settings": {
"encryption": {
"encryptionType": enum,
"kmsKeyArn": "string"
}
},
"encryption": {
"spekeKeyProvider": {
"resourceId": "string",
"systemIds": [string
],
"url": "string",
"certificateArn": "string"
}
},
"minBufferTime": integer,
"fragmentLength": integer,
"baseUrl": "string",
"segmentControl": enum,
"hbbtvCompliance": enum,
"writeSegmentTimelineInRepresentation": enum
},
"fileGroupSettings": {
"destination": "string",
"destinationSettings": {
"s3Settings": {
"encryption": {
"encryptionType": enum,
"kmsKeyArn": "string"
}
}
}
"msSmoothGroupSettings": {
  "destination": "string",
  "destinationSettings": {
    "s3Settings": {
      "encryption": {
        "encryptionType": enum,
        "kmsKeyArn": "string"
      }
    }
  },
  "fragmentLength": integer,
  "encryption": {
    "spekeKeyProvider": {
      "resourceId": "string",
      "systemIds": ["string"],
      "url": "string",
      "certificateArn": "string"
    }
  },
  "manifestEncoding": enum,
  "audioDeduplication": enum
},
"cmafGroupSettings": {
  "writeHlsManifest": enum,
  "writeDashManifest": enum,
  "segmentLength": integer,
  "minFinalSegmentLength": number,
  "destination": "string",
  "destinationSettings": {
    "s3Settings": {
      "encryption": {
        "encryptionType": enum,
        "kmsKeyArn": "string"
      }
    }
  },
  "encryption": {
    "encryptionMethod": enum,
    "constantInitializationVector": "string",
    "initializationVectorInManifest": enum,
    "staticKeyProvider": {
      "staticKeyValue": "string",
      "keyFormat": "string",
      "keyFormatVersions": "string",
      "url": "string"
    },
    "type": enum
  },
  "minBufferTime": integer,
  "fragmentLength": integer,
  "baseUrl": "string",
  "segmentControl": enum,
  "manifestDurationFormat": enum,
  "streamInfResolution": enum,
  "clientCache": enum,
  "manifestCompression": enum,
  "codecSpecification": enum
}
"adAvailOffset": integer,
"availBlanking": {
  "availBlankingImage": "string"
},
"timedMetadataInsertion": {
  "id3Insertions": [
    {
      "timecode": "string",
      "id3": "string"
    }
  ],
"nielsenConfiguration": {
  "breakoutCode": integer,
  "distributorId": "string"
},
"motionImageInserter": {
  "insertionMode": enum,
  "input": "string",
  "offset": {
    "imageX": integer,
    "imageY": integer
  },
  "startTime": "string",
  "playback": enum,
  "framerate": {
    "framerateNumerator": integer,
    "framerateDenominator": integer
  }
},
"esam": {
  "signalProcessingNotification": {
    "sccXml": "string"
  },
  "manifestConfirmConditionNotification": {
    "mccXml": "string"
  },
  "responseSignalPreroll": integer
},
"inputs": [
  {
    "inputClippings": [
      {
        "endTimecode": "string",
        "startTimecode": "string"
      }
    ],
    "audioSelectors": {
    },
    "audioSelectorGroups": {
    },
    "programNumber": integer,
    "videoSelector": {
      "colorSpace": enum,
      "rotate": enum,
      "pid": integer,
      "programNumber": integer,
      "colorSpaceUsage": enum,
      "hdr10Metadata": {
        "redPrimaryX": integer,
        "redPrimaryY": integer,
        "greenPrimaryX": integer,
        "greenPrimaryY": integer,
        "bluePrimaryX": integer,
        "bluePrimaryY": integer,
        "whitePointX": integer,
"whitePointY": integer,
"maxFrameAverageLightLevel": integer,
"maxContentLightLevel": integer,
"maxLuminance": integer,
"minLuminance": integer
},
"filterEnable": enum,
"psiControl": enum,
"filterStrength": integer,
"deblockFilter": enum,
"denoiseFilter": enum,
"timecodeSource": enum,
"captionSelectors": {
},
"imageInserter": {
"insertableImages": [
{
"width": integer,
"height": integer,
"imageX": integer,
"imageY": integer,
"duration": integer,
"fadeIn": integer,
"layer": integer,
"imageInserterInput": "string",
"startTime": "string",
"fadeOut": integer,
"opacity": integer
}
]
},
"accelerationSettings": {
"mode": enum
},
"statusUpdateInterval": enum
}
}

Example DeleteJobTemplateResponse
{
}

Example ExceptionBody
{
"message": "string"
}

Properties

**AacAudioDescriptionBroadcasterMix**

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
AAC Profile.

LC
HEV1
HEV2

AacCodingMode
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
Rate Control Mode.

CBR
VBR

AacRawFormat
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
Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value AAC. The service accepts one of two mutually exclusive groups of AAC settings--VBR and CBR. To select one of these modes, set the value of Bitrate control mode (rateControlMode) to "VBR" or "CBR". In VBR mode, you control the audio quality with the setting VBR quality (vbrQuality). In CBR mode, you use the setting Bitrate (bitrate). Defaults and valid values depend on the rate control mode.
**audioDescriptionBroadcasterMix**

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.

- **Type**: AacAudioDescriptionBroadcasterMix (p. 277)
- **Required**: False

**vbrQuality**

VBR Quality Level - Only used if rate_control_mode is VBR.

- **Type**: AacVbrQuality (p. 280)
- **Required**: False

**bitrate**

Average bitrate in bits/second. The set of valid values for this setting is: 6000, 8000, 10000, 12000, 14000, 16000, 20000, 24000, 28000, 32000, 40000, 48000, 56000, 64000, 80000, 96000, 112000, 128000, 160000, 192000, 224000, 256000, 288000, 320000, 384000, 448000, 512000, 576000, 640000, 768000, 896000, 1024000. The value you set is also constrained by the values you choose for Profile (codecProfile), Bitrate control mode (codingMode), and Sample rate (sampleRate). Default values depend on Bitrate control mode and Profile.

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

**rateControlMode**

Rate Control Mode.

- **Type**: AacRateControlMode (p. 278)
- **Required**: False

**codecProfile**

AAC Profile.

- **Type**: AacCodecProfile (p. 278)
- **Required**: False

**codingMode**

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.
Properties

**Type**: AacCodingMode (p. 278)
**Required**: False

**rawFormat**

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.

**Type**: AacRawFormat (p. 278)
**Required**: False

**sampleRate**

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

**Type**: integer
**Required**: False
**Minimum**: 8000
**Maximum**: 96000

**specification**

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

**Type**: AacSpecification (p. 280)
**Required**: False

**AacSpecification**

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

- MPEG2
- MPEG4

**AacVbrQuality**

VBR Quality Level - Only used if rate_control_mode is VBR.

- LOW
- MEDIUM_LOW
- MEDIUM_HIGH
- HIGH

**Ac3BitstreamMode**

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**

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**

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**

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

- ENABLED
- DISABLED

**Ac3MetadataControl**

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**

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value AC3.

**bitrate**

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

- **Type:** integer
- **Required:** False
- **Minimum:** 64000
- **Maximum:** 640000
Properties

**bitstreamMode**

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

- **Type:** Ac3BitstreamMode (p. 280)
- **Required:** False

**codingMode**

Dolby Digital coding mode. Determines number of channels.

- **Type:** Ac3CodingMode (p. 281)
- **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

**dynamicRangeCompressionProfile**

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

- **Type:** Ac3DynamicRangeCompressionProfile (p. 281)
- **Required:** False

**metadataControl**

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.

- **Type:** Ac3MetadataControl (p. 281)
- **Required:** False

**lfeFilter**

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

- **Type:** Ac3LfeFilter (p. 281)
- **Required:** False

**sampleRate**

Sample rate in hz. Sample rate is always 48000.

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

### AccelerationMode

Enable Acceleration (AccelerationMode) on any job that you want processed with accelerated transcoding.

- DISABLED
- ENABLED

### AccelerationSettings

Accelerated transcoding can significantly speed up jobs with long, visually complex content. Outputs that use this feature incur pro-tier pricing. For information about feature limitations, see the AWS Elemental MediaConvert User Guide.

**mode**

Acceleration configuration for the job.

- **Type**: AccelerationMode (p. 283)
- **Required**: True

### AfdSignaling

This setting only applies to H.264, H.265, and MPEG2 outputs. Use Insert AFD signaling (AfdSignaling) to specify whether the service includes 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

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value AIFF.

**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

**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

**sampleRate**

Sample rate in hz.

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

**AncillarySourceSettings**

Settings for ancillary captions source.

**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**

The anti-alias filter is automatically applied to all outputs. The service no longer accepts the value DISABLED for AntiAlias. If you specify that in your job, the service will ignore the setting.

DISABLED
ENABLED

**AudioCodec**

Type of Audio codec.

AAC
MP2
WAV
AIFF
AC3
EAC3
PASSTHROUGH

**AudioCodecSettings**

Audio codec settings (CodecSettings) under (AudioDescriptions) contains the group of settings related to audio encoding. The settings in this group vary depending on the value you choose for Audio codec (Codec). For each codec enum you choose, define the corresponding settings object. The following lists
the codec enum, settings object pairs. * AAC, AacSettings * MP2, Mp2Settings * WAV, WavSettings * AIFF, AiffSettings * AC3, Ac3Settings * EAC3, Eac3Settings

**codec**

Type of Audio codec.

* **Type**: AudioCodec (p. 284)
  * **Required**: False

**aacSettings**

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value AAC. The service accepts one of two mutually exclusive groups of AAC settings--VBR and CBR. To select one of these modes, set the value of Bitrate control mode (rateControlMode) to "VBR" or "CBR". In VBR mode, you control the audio quality with the setting VBR quality (vbrQuality). In CBR mode, you use the setting Bitrate (bitrate). Defaults and valid values depend on the rate control mode.

* **Type**: AacSettings (p. 278)
  * **Required**: False

**ac3Settings**

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value AC3.

* **Type**: Ac3Settings (p. 281)
  * **Required**: False

**aiffSettings**

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value AIFF.

* **Type**: AiffSettings (p. 283)
  * **Required**: False

**eac3Settings**

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value EAC3.

* **Type**: Eac3Settings (p. 322)
  * **Required**: False

**mp2Settings**

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value MP2.

* **Type**: Mp2Settings (p. 396)
  * **Required**: False

**wavSettings**

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value WAV.

* **Type**: WavSettings (p. 434)
  * **Required**: False
AudioDefaultSelection

Enable this setting on one audio selector to set it as the default for the job. The service uses this default for outputs where it can't find the specified input audio. If you don't set a default, those outputs have no audio.

- DEFAULT
- NOT_DEFAULT

AudioDescription

Description of audio output

audioTypeControl

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.

- Type: AudioTypeControl (p. 292)
- Required: False

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

audioNormalizationSettings

Advanced audio normalization settings.

- Type: AudioNormalizationSettings (p. 288)
- Required: False

codecSettings

Audio codec settings (CodecSettings) under (AudioDescriptions) contains the group of settings related to audio encoding. The settings in this group vary depending on the value you choose for Audio codec (Codec). For each codec enum you choose, define the corresponding settings object. The following lists the codec enum, settings object pairs. * AAC, AacSettings * MP2, Mp2Settings * WAV, WavSettings * AIFF, AiffSettings * AC3, Ac3Settings * EAC3, Eac3Settings

- Type: AudioCodecSettings (p. 284)
- Required: False
remixSettings
Advanced audio remixing settings.

  Type: RemixSettings (p. 417)
  Required: False

streamName
Used for MS Smooth and Apple HLS outputs. Indicates the name displayed by the player (e.g., English, or Director Commentary). Alphanumeric characters, spaces, and underscore are legal.

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

languageCodeControl
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.

  Type: AudioLanguageCodeControl (p. 288)
  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

customLanguageCode
Specify the language for this audio output track, using the ISO 639-2 or ISO 639-3 three-letter language code. The language specified 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: string
  Required: False
  Pattern: ^[A-Za-z]{3}$
  MinLength: 3
  MaxLength: 3

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. 376)
**Properties**

**Required:** False

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

LOG
DONT_LOG

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

TRUE_PEAK
NONE

**AudioNormalizationSettings**
Advanced audio normalization settings.

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

Type: AudioNormalizationAlgorithm (p. 288)
Required: False

algorithmControl

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

  Type: AudioNormalizationAlgorithmControl (p. 288)
  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

loudnessLogging

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

  Type: AudioNormalizationLoudnessLogging (p. 288)
  Required: False

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

peakCalculation

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

  Type: AudioNormalizationPeakCalculation (p. 288)
  Required: False

AudioSelector

Selector for Audio

tracks

Identify a track from the input audio to include in this selector by entering the track index number. To include several tracks in a single audio selector, specify multiple tracks as follows. Using the console,
enter a comma-separated list. For example, type "1,2,3" to include tracks 1 through 3. Specifying directly in your JSON job file, provide the track numbers in an array. For example, "tracks": [1,2,3].

**Type:** Array of type integer  
**Required:** False  
**Minimum:** 1  
**Maximum:** 2147483647

**offset**

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

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

**defaultSelection**

Enable this setting on one audio selector to set it as the default for the job. The service uses this default for outputs where it can't find the specified input audio. If you don't set a default, those outputs have no audio.

**Type:** AudioDefaultSelection (p. 286)  
**Required:** False

**selectorType**

Specifies the type of the audio selector.

**Type:** AudioSelectorType (p. 292)  
**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  
**Minimum:** 1  
**Maximum:** 2147483647

**externalAudioFileInput**

Specifies audio data from an external file source.

**Type:** string  
**Required:** False  
**Pattern:** ^s3:/[^
]+([^
]+)+([^
]+)+([[^
]+]+)([mM][pP][eE][gG]|[aA][vV][iI][eE][mM][pP][4][\ef]\[\[\vV]]|\[mM][pP][tT][\[mM][pP][\[gG]][mM][4][\vV][\[tT][\[rR][pP][\[fF][\[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][3][gG][pP][\[mM][xX][\fF][dD][\iI][vV][xX][xX][\vV][iI][dD][rR][aA][\wW][dD][\vV][gG][xX][\fF][mM][1][\vV][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]
programSelection

Use this setting for input streams that contain Dolby E, to have the service extract specific program data from the track. To select multiple programs, create multiple selectors with the same Track and different Program numbers. In the console, this setting is visible when you set Selector type to Track. Choose the program number from the dropdown list. If you are sending a JSON file, provide the program ID, which is part of the audio metadata. If your input file has incorrect metadata, you can choose All channels instead of a program number to have the service ignore the program IDs and include all the programs in the track.

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

customLanguageCode

Selects a specific language code from within an audio source, using the ISO 639-2 or ISO 639-3 three-letter language code

Type: string
Required: False
Pattern: ^[A-Za-z]{3}$
MinLength: 3
MaxLength: 3

languageCode

Selects a specific language code from within an audio source.

Type: LanguageCode (p. 376)
Required: False

remixSettings

Use these settings to reorder the audio channels of one input to match those of another input. This allows you to combine the two files into a single output, one after the other.

Type: RemixSettings (p. 417)
Required: False

AudioSelectorGroup

Group of Audio Selectors

audioSelectorNames

Name of an 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: False
MinLength: 1

**AudioSelectorType**

Specifies the type of the audio selector.

- **PID**
- **TRACK**
- **LANGUAGE_CODE**

**AudioTypeControl**

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**

Settings for Avail Blanking

**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)$`
- **MinLength**: 14

**BurninDestinationSettings**

Burn-In Destination Settings.

**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

**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.
teletextSpacing

Only applies to jobs with input captions in Teletext or STL formats. Specify whether the spacing between letters in your captions is set by the captions grid or varies depending on letter width. Choose fixed grid to conform to the spacing specified in the captions file more accurately. Choose proportional to make the text easier to read if the captions are closed caption.

Type: $\text{BurninSubtitleTeletextSpacing}$ (p. 296)
Required: False

alignment

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.

Type: $\text{BurninSubtitleAlignment}$ (p. 295)
Required: False

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: False
Minimum: 0
Maximum: 10

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

shadowColor

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

**Type**: BurninSubtitleShadowColor (p. 296)
**Required**: False

**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**: False  
**Minimum**: 0  
**Maximum**: 255

**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

**fontScript**

Provide the font script, using an ISO 15924 script code, if the LanguageCode is not sufficient for determining the script type. Where LanguageCode or CustomLanguageCode is sufficient, use "AUTOMATIC" or leave unset. This is used to help determine the appropriate font for rendering burn-in captions.

**Type**: FontScript (p. 330)  
**Required**: False

**fontColor**

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.

**Type**: BurninSubtitleFontColor (p. 296)  
**Required**: False

**backgroundColor**

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

**Type**: BurninSubtitleBackgroundColor (p. 296)  
**Required**: False

**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

**outlineColor**

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.

**Type**: `BurninSubtitleOutlineColor` (p. 296)  
**Required**: False

**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

**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

**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

**BurninSubtitleAlignment**

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**

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

- NONE
- BLACK
- WHITE

**BurninSubtitleFontColor**

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**

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**

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

- NONE
- BLACK
- WHITE

**BurninSubtitleTeletextSpacing**

Only applies to jobs with input captions in Teletext or STL formats. Specify whether the spacing between letters in your captions is set by the captions grid or varies depending on letter width. Choose fixed grid
to conform to the spacing specified in the captions file more accurately. Choose proportional to make the text easier to read if the captions are closed caption.

FIXED_GRID
PROPORTIONAL

**CaptionDescription**

Description of Caption output

**captionSelectorName**

Specifies which "Caption Selector":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: False
MinLength: 1

**destinationSettings**

Specific settings required by destination type. Note that burnin_destination_settings are not available if the source of the caption data is Embedded or Teletext.

Type: CaptionDestinationSettings (p. 298)
Required: False

**customLanguageCode**

Indicates the language of the caption output track, using the ISO 639-2 or ISO 639-3 three-letter language code. For most captions output formats, the encoder puts this language information in the output captions metadata. If your output captions format is DVB-Sub or Burn in, the encoder uses this language information to choose the font language for rendering the captions text.

Type: string
Required: False
Pattern: ^[A-Za-z]{3}$
MinLength: 3
MaxLength: 3

**languageCode**

Specify the language of this captions output track. For most captions output formats, the encoder puts this language information in the output captions metadata. If your output captions format is DVB-Sub or Burn in, the encoder uses this language information to choose the font language for rendering the captions text.

Type: LanguageCode (p. 376)
Required: False

**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

### CaptionDestinationSettings

Specific settings required by destination type. Note that burnin_destination_settings are not available if the source of the caption data is Embedded or Teletext.

**destinationType**

Specify the format for this set of captions on this output. The default format is embedded without SCTE-20. Other options are embedded with SCTE-20, burn-in, DVB-sub, SCC, SRT, teletext, TTML, and web-VTT. If you are using SCTE-20, choose SCTE-20 plus embedded (SCTE20_PLUS_EMBEDDED) to create an output that complies with the SCTE-43 spec. To create a non-compliant output where the embedded captions come first, choose Embedded plus SCTE-20 (EMBEDDED_PLUS_SCTE20).

**Type**: CaptionDestinationType (p. 299)  
**Required**: False

### burninDestinationSettings

Burn-In Destination Settings.

**Type**: BurninDestinationSettings (p. 292)  
**Required**: False

### dvbSubDestinationSettings

DVB-Sub Destination Settings

**Type**: DvbSubDestinationSettings (p. 315)  
**Required**: False

### sccDestinationSettings

Settings for SCC caption output.

**Type**: SccDestinationSettings (p. 420)  
**Required**: False

### teletextDestinationSettings

Settings for Teletext caption output.

**Type**: TeletextDestinationSettings (p. 422)  
**Required**: False

### ttmlDestinationSettings

Settings specific to TTML caption outputs, including Pass style information (TtmlStylePassthrough).

**Type**: TtmlDestinationSettings (p. 425)  
**Required**: False
**embeddedDestinationSettings**

Settings specific to embedded/ancillary caption outputs, including 608/708 Channel destination number.

- **Type:** EmbeddedDestinationSettings (p. 326)
- **Required:** False

**CaptionDestinationType**

Specify the format for this set of captions on this output. The default format is embedded without SCTE-20. Other options are embedded with SCTE-20, burn-in, DVB-sub, SCC, SRT, teletext, TTML, and web-VTT. If you are using SCTE-20, choose SCTE-20 plus embedded (SCTE20_PLUS_EMBEDDED) to create an output that complies with the SCTE-43 spec. To create a non-compliant output where the embedded captions come first, choose Embedded plus SCTE-20 (EMBEDDED_PLUS_SCTE20).

- BURN_IN
- DVB_SUB
- EMBEDDED
- EMBEDDED_PLUS_SCTE20
- SCTE20_PLUS_EMBEDDED
- SCC
- SRT
- SMI
- TELETEXT
- TTML
- WEBVTT

**CaptionSelector**

Set up captions in your outputs by first selecting them from your input here.

**customLanguageCode**

The specific language to extract from source, using the ISO 639-2 or ISO 639-3 three-letter language code. 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:** string
- **Required:** False
- **Pattern:** ^[A-Za-z]{3}$
- **MinLength:** 3
- **MaxLength:** 3

**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. 376)
sourceSettings

Source settings (SourceSettings) contains the group of settings for captions in the input.

Type: CaptionSourceSettings (p. 300)
Required: False

CaptionSourceSettings

Source settings (SourceSettings) contains the group of settings for captions in the input.

sourceType

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

Type: CaptionSourceType (p. 301)
Required: False

ancillarySourceSettings

Settings for ancillary captions source.

Type: AncillarySourceSettings (p. 284)
Required: False

dvbSubSourceSettings

DVB Sub Source Settings

Type: DvbSubSourceSettings (p. 319)
Required: False

embeddedSourceSettings

Settings for embedded captions Source

Type: EmbeddedSourceSettings (p. 327)
Required: False

fileSourceSettings

Settings for File-based Captions in Source

Type: FileSourceSettings (p. 329)
Required: False

teletextSourceSettings

Settings specific to Teletext caption sources, including Page number.

Type: TeletextSourceSettings (p. 422)
**Required**: False

**trackSourceSettings**

Settings specific to caption sources that are specified by track number. Sources include IMSC in IMF.

*Type*: TrackSourceSettings (p. 425)

*Required*: False

**CaptionSourceType**

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

- ANCILLARY
- DVB_SUB
- EMBEDDED
- SCTE20
- SCC
- TTML
- STL
- SRT
- SMI
- TELETEXT
- NULL_SOURCE
- IMSC

**ChannelMapping**

Channel mapping (ChannelMapping) contains the group of fields that hold the remixing value for each channel. Units are in dB. Acceptable values are within the range from -60 (mute) through 6. A setting of 0 passes the input channel unchanged to the output channel (no attenuation or amplification).

**outputChannels**

List of output channels

*Type*: Array of type OutputChannelMapping (p. 411)

*Required*: False

**CmafClientCache**

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

- DISABLED
- ENABLED

**CmafCodecSpecification**

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

- RFC_6381
**CmafEncryptionSettings**

Settings for CMAF encryption

**encryptionMethod**

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

- **Type:** CmafEncryptionType (p. 302)
- **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}$
- **MinLength:** 32
- **MaxLength:** 32

**initializationVectorInManifest**

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.

- **Type:** CmafInitializationVectorInManifest (p. 305)
- **Required:** False

**staticKeyProvider**

Use these settings to set up encryption with a static key provider.

- **Type:** StaticKeyProvider (p. 421)
- **Required:** False

**type**

Indicates which type of key provider is used for encryption.

- **Type:** CmafKeyProviderType (p. 305)
- **Required:** False

**CmafEncryptionType**

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

SAMPLE_AES
CmafGroupSettings

Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to CMAF_GROUP_SETTINGS. Each output in a CMAF Output Group may only contain a single video, audio, or caption output.

writeHlsManifest

When set to ENABLED, an Apple HLS manifest will be generated for this output.

    Type: CmafWriteHLSManifest (p. 306)
    Required: False

writeDashManifest

When set to ENABLED, a DASH MPD manifest will be generated for this output.

    Type: CmafWriteDASHManifest (p. 306)
    Required: False

segmentLength

Use this setting to specify the length, in seconds, of each individual CMAF segment. This value applies to the whole package; that is, to every output in the output group. Note that segments end on the first keyframe after this number of seconds, so the actual segment length might be slightly longer. If you set Segment control (CmafSegmentControl) to single file, the service puts the content of each output in a single file that has metadata that marks these segments. If you set it to segmented files, the service creates multiple files for each output, each with the content of one segment.

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

minFinalSegmentLength

Keep this setting at the default value of 0, unless you are troubleshooting a problem with how devices play back the end of your video asset. If you know that player devices are hanging on the final segment of your video because the length of your final segment is too short, use this setting to specify a minimum final segment length, in seconds. Choose a value that is greater than or equal to 1 and less than your segment length. When you specify a value for this setting, the encoder will combine any final segment that is shorter than the length that you specify with the previous segment. For example, your segment length is 3 seconds and your final segment is .5 seconds without a minimum final segment length; when you set the minimum final segment length to 1, your final segment is 3.5 seconds.

    Type: number
    Required: False
    Format: float
    Minimum: 0.0
    Maximum: 2147483647

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:** False
- **Pattern:** ^s3:/\/

### destinationSettings

Settings associated with the destination. Will vary based on the type of destination.

- **Type:** DestinationSettings (p. 314)
- **Required:** False

### encryption

DRM settings.

- **Type:** CmafEncryptionSettings (p. 302)
- **Required:** False

### minBufferTime

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

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

### 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:** False
- **Minimum:** 1
- **Maximum:** 2147483647

### baseUrl

A partial URI prefix that will be put in the manifest 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

### segmentControl

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.
**Properties**

**Type:** CmafSegmentControl (p. 306)
**Required:** False

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

**Type:** CmafManifestDurationFormat (p. 306)
**Required:** False

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

**Type:** CmafStreamInfResolution (p. 306)
**Required:** False

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

**Type:** CmafClientCache (p. 301)
**Required:** False

**manifestCompression**
When set to GZIP, compresses HLS playlist.

**Type:** CmafManifestCompression (p. 306)
**Required:** False

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

**Type:** CmafCodecSpecification (p. 301)
**Required:** False

**CmafInitializationVectorInManifest**
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**

**CmafKeyProviderType**
Indicates which type of key provider is used for encryption.

**STATIC_KEY**
CmafManifestCompression
When set to GZIP, compresses HLS playlist.
  GZIP
  NONE

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

CmafSegmentControl
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

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

CmafWriteDASHManifest
When set to ENABLED, a DASH MPD manifest will be generated for this output.
  DISABLED
  ENABLED

CmafWriteHLSManifest
When set to ENABLED, an Apple HLS manifest will be generated for this output.
  DISABLED
  ENABLED

ColorCorrector
Settings for color correction.

  brightness
  Brightness level.
Properties

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

**colorSpaceConversion**

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.

**Type**: ColorSpaceConversion (p. 308)
**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

**saturation**

Saturation level.

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

**hdr10Metadata**

Use the HDR master display (Hdr10Metadata) settings to correct HDR metadata or to provide missing metadata. Note that these settings are not color correction.

**Type**: Hdr10Metadata (p. 354)
**Required**: False

**ColorMetadata**

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

If your input video has accurate color space metadata, or if you don't know about color space, leave this set to the default value FOLLOW. The service will automatically detect your input color space. If your input video has metadata indicating the wrong color space, or if your input video is missing color space metadata that should be there, specify the accurate color space here. If you choose HDR10, you can also correct inaccurate color space coefficients, using the HDR master display information controls. You must also set Color space usage (ColorSpaceUsage) to FORCE for the service to use these values.

FOLLOW
REC_601
REC_709
HDR10
HLG_2020

ColorSpaceConversion

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 colorspace. 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

There are two sources for color metadata, the input file and the job configuration (in the Color space and HDR master display informaiton settings). The Color space usage setting controls which takes precedence. FORCE: The 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: The 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 specific settings.

container

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

*Type: ContainerType (p. 309)*
*Required: False*
m3u8Settings
Settings for TS segments in HLS

Type: M3u8Settings (p. 389)
Required: False

f4vSettings
Settings for F4v container

Type: F4vSettings (p. 329)
Required: False

m2tsSettings
MPEG-2 TS container settings. These apply to outputs in a File output group when the output's container
(ContainerType) is MPEG-2 Transport Stream (M2TS). In these assets, data is organized by the program
map table (PMT). Each transport stream program contains subsets of data, including audio, video, and
metadata. Each of these subsets of data has a numerical label called a packet identifier (PID). Each
transport stream program corresponds to one MediaConvert output. The PMT lists the types of data in
a program along with their PID. Downstream systems and players use the program map table to look up
the PID for each type of data it accesses and then uses the PIDs to locate specific data within the asset.

Type: M2tsSettings (p. 382)
Required: False

movSettings
Settings for MOV Container.

Type: MovSettings (p. 395)
Required: False

mp4Settings
Settings for MP4 Container

Type: Mp4Settings (p. 397)
Required: False

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

F4V
ISMV
M2TS
M3U8
CMFC
MOV
MP4
MPD
MXF
DashIsoEncryptionSettings

Specifies DRM settings for DASH outputs.

**spekeKeyProvider**

Settings for use with a SPEKE key provider

- **Type:** SpekeKeyProvider (p. 420)
- **Required:** False

DashIsoGroupSettings

Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to DASH_ISO_GROUP_SETTINGS.

**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:** False
- **Minimum:** 1
- **Maximum:** 2147483647

**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:** False
- **Pattern:** ^s3:\/\/

**destinationSettings**

Settings associated with the destination. Will vary based on the type of destination

- **Type:** DestinationSettings (p. 314)
- **Required:** False

**encryption**

DRM settings.

- **Type:** DashIsoEncryptionSettings (p. 310)
- **Required:** False
**minBufferTime**

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

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

**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:** False
- **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

**segmentControl**

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.

- **Type:** DashIsoSegmentControl (p. 312)
- **Required:** False

**hbbtvCompliance**

Supports HbbTV specification as indicated

- **Type:** DashIsoHbbtvCompliance (p. 312)
- **Required:** False

**writeSegmentTimelineInRepresentation**

When you enable Precise segment duration in manifests (writeSegmentTimelineInRepresentation), your DASH manifest shows precise segment durations. The segment duration information appears inside the SegmentTimeline element, inside SegmentTemplate at the Representation level. When this feature isn't enabled, the segment durations in your DASH manifest are approximate. The segment duration information appears in the duration attribute of the SegmentTemplate element.

- **Type:** DashIsoWriteSegmentTimelineInRepresentation (p. 312)
- **Required:** False
DashIsoHbbtvCompliance

Supports HbbTV specification as indicated

- HBBTV_1_5
- NONE

DashIsoSegmentControl

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

DashIsoWriteSegmentTimelineInRepresentation

When you enable Precise segment duration in manifests (writeSegmentTimelineInRepresentation), your DASH manifest shows precise segment durations. The segment duration information appears inside the SegmentTimeline element, inside SegmentTemplate at the Representation level. When this feature isn't enabled, the segment durations in your DASH manifest are approximate. The segment duration information appears in the duration attribute of the SegmentTemplate element.

- ENABLED
- DISABLED

DeinterlaceAlgorithm

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

Settings for deinterlacer

**algorithm**

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.

- **Type:** DeinterlaceAlgorithm (p. 312)
- **Required:** False
mode

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.

Type: DeinterlaceMode (p. 313)
Required: False

control

- 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.

Type: DeinterlacerControl (p. 313)
Required: False

DeinterlacerControl

- 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

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

Delete a job template by sending a request with the job template name

name

The name of the job template to be deleted.

Type: string
Required: False
DeleteJobTemplateResponse
Delete job template requests will return an OK message or error message with an empty body.

DestinationSettings
Settings associated with the destination. Will vary based on the type of destination

s3Settings
Settings associated with S3 destination

  Type: S3DestinationSettings (p. 418)
  Required: False

DropFrameTimecode
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
Inserts DVB Network Information Table (NIT) at the specified table repetition interval.

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

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

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

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

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

    Type: string
    Required: False
    MinLength: 1
    MaxLength: 256
DvbSdtSettings

Inserts DVB Service Description Table (NIT) at the specified table repetition interval.

outputSdt

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.

Type: OutputSdt (p. 413)
Required: False

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
MinLength: 1
MaxLength: 256

serviceProviderName

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

Type: string
Required: False
MinLength: 1
MaxLength: 256

DvbSubDestinationSettings

DVB-Sub Destination Settings

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

**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

**teletextSpacing**

Only applies to jobs with input captions in Teletext or STL formats. Specify whether the spacing between letters in your captions is set by the captions grid or varies depending on letter width. Choose fixed grid to conform to the spacing specified in the captions file more accurately. Choose proportional to make the text easier to read if the captions are closed caption.

- **Type:** DvbSubtitleTeletextSpacing (p. 320)
- **Required:** False

**alignment**

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.

- **Type:** DvbSubtitleAlignment (p. 319)
- **Required:** False

**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:** False
- **Minimum:** 0
- **Maximum:** 10

**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
**shadowColor**

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

**Type:** DvbSubtitleShadowColor (p. 320)
**Required:** False

**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:** False
**Minimum:** 0
**Maximum:** 255

**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

**fontScript**

Provide the font script, using an ISO 15924 script code, if the LanguageCode is not sufficient for determining the script type. Where LanguageCode or CustomLanguageCode is sufficient, use "AUTOMATIC" or leave unset. This is used to help determine the appropriate font for rendering DVB-Sub captions.

**Type:** FontScript (p. 330)
**Required:** False

**fontColor**

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.

**Type:** DvbSubtitleFontColor (p. 319)
**Required:** False

**backgroundColor**

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

**Type**: DvbSubtitleBackgroundColor (p. 319)
**Required**: False

**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

**outlineColor**

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.

**Type**: DvbSubtitleOutlineColor (p. 319)
**Required**: False

**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

**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

**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
DvbSubSourceSettings

DVB Sub Source Settings

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

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

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

- NONE
- BLACK
- WHITE

DvbSubtitleFontColor

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

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.
DvbSubtitleShadowColor

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

NONE
BLACK
WHITE

DvbSubtitleTeletextSpacing

Only applies to jobs with input captions in Teletext or STL formats. Specify whether the spacing between letters in your captions is set by the captions grid or varies depending on letter width. Choose fixed grid to conform to the spacing specified in the captions file more accurately. Choose proportional to make the text easier to read if the captions are closed caption.

FIXED_GRID
PROPORTIONAL

DvbTdtSettings

Inserts DVB Time and Date Table (TDT) at the specified table repetition interval.

tdtInterval

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

Type: integer
Required: False
Minimum: 1000
Maximum: 30000

Eac3AttenuationControl

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

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

COMPLETE_MAIN
COMMENTARY
EMERGENCY
HEARING_IMPAIRED
VISUALLY_IMPAIRED

Eac3CodingMode

Dolby Digital Plus coding mode. Determines number of channels.

- CODING_MODE_1_0
- CODING_MODE_2_0
- CODING_MODE_3_2

Eac3DcFilter

Activates a DC highpass filter for all input channels.

- ENABLED
- DISABLED

Eac3DynamicRangeCompressionLine

Enables Dynamic Range Compression that restricts the absolute peak level for a signal.

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

Eac3DynamicRangeCompressionRf

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

When encoding 3/2 audio, controls whether the LFE channel is enabled

- LFE
- NO_LFE
**Eac3LfeFilter**

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

- ENABLED
- DISABLED

**Eac3MetadataControl**

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**

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**

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

- SHIFT_90_DEGREES
- NO_SHIFT

**Eac3Settings**

Required when you set (Codec) under (AudioDescriptions)->(CodecSettings) to the value EAC3.

- **metadataControl**
  
  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.

  - **Type:** Eac3MetadataControl (p. 322)
  - **Required:** False

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

  - **Type:** Eac3SurroundExMode (p. 326)
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

**phaseControl**

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

Type: Eac3PhaseControl (p. 322)
Required: False

**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

**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

**bitrate**

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

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

**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
### Properties

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

#### passthroughControl

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.

**Type**: Eac3PassthroughControl (p. 322)  
**Required**: False

#### lfeControl

When encoding 3/2 audio, controls whether the LFE channel is enabled.

**Type**: Eac3LfeControl (p. 321)  
**Required**: False

#### 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

#### attenuationControl

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

**Type**: Eac3AttenuationControl (p. 320)  
**Required**: False

#### codingMode

Dolby Digital Plus coding mode. Determines number of channels.

**Type**: Eac3CodingMode (p. 321)  
**Required**: False

#### surroundMode

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

**Type**: Eac3SurroundMode (p. 326)
Required: False

bitstreamMode

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

Type: Eac3BitstreamMode (p. 320)
Required: False

lfeFilter

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

Type: Eac3LfeFilter (p. 322)
Required: False

stereoDownmix

Stereo downmix preference. Only used for 3/2 coding mode.

Type: Eac3StereoDownmix (p. 326)
Required: False

dynamicRangeCompressionRf

Enables Heavy Dynamic Range Compression, ensures that the instantaneous signal peaks do not exceed specified levels.

Type: Eac3DynamicRangeCompressionRf (p. 321)
Required: False

sampleRate

Sample rate in hz. Sample rate is always 48000.

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

dynamicRangeCompressionLine

Enables Dynamic Range Compression that restricts the absolute peak level for a signal.

Type: Eac3DynamicRangeCompressionLine (p. 321)
Required: False

dcFilter

Activates a DC highpass filter for all input channels.

Type: Eac3DcFilter (p. 321)
Required: False
**Eac3StereoDownmix**

Stereo downmix preference. Only used for 3/2 coding mode.

- NOT_INDICATED
- LO_RO
- LT_RT
- DPL2

**Eac3SurroundExMode**

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**

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

- NOT_INDICATED
- ENABLED
- DISABLED

**EmbeddedConvert608To708**

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

**EmbeddedDestinationSettings**

Settings specific to embedded/ancillary caption outputs, including 608/708 Channel destination number.

**destination608ChannelNumber**

Ignore this setting unless your input captions are SCC format and your output container is MXF. With this combination of input captions format and output container, you can optionally use this setting to replace the input channel number with the track number that you specify. Specify a different number for each output captions track. If you don't specify an output track number, the system uses the input channel number for the output channel number. This setting applies to each output individually. You can optionally combine two captions channels in your output. The two output channel numbers can be one of the following pairs: 1,3; 2,4; 1,4; or 2,3.

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

Settings for embedded captions Source

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

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

convert608To708

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.

Type: EmbeddedConvert608To708 (p. 326)
Required: False

EsamManifestConfirmConditionNotification

ESAM ManifestConfirmConditionNotification defined by OC-SP-ESAM-API-I03-131025.

mccXml

Provide your ESAM ManifestConfirmConditionNotification XML document inside your JSON job settings. Form the XML document as per OC-SP-ESAM-API-I03-131025. The transcoder will use the Manifest Conditioning instructions in the message that you supply.

Type: string
Required: False
Pattern: ^\s*<(.|\n)*ManifestConfirmConditionNotification(.|\n)*>\s*$

EsamSettings

Settings for Event Signaling And Messaging (ESAM). If you don’t do ad insertion, you can ignore these settings.

signalProcessingNotification

Specifies an ESAM SignalProcessingNotification XML as per OC-SP-ESAM-API-I03-131025. The transcoder uses the signal processing instructions that you provide in the setting SCC XML (sccXml).
**manifestConfirmConditionNotification**

Specifies an ESAM ManifestConfirmConditionNotification XML as per OC-SP-ESAM-API-I03-131025. The transcoder uses the manifest conditioning instructions that you provide in the setting MCC XML (mccXml).

- **Type**: EsamManifestConfirmConditionNotification (p. 327)
- **Required**: False

**responseSignalPreroll**

Specifies the stream distance, in milliseconds, between the SCTE 35 messages that the transcoder places and the splice points that they refer to. If the time between the start of the asset and the SCTE-35 message is less than this value, then the transcoder places the SCTE-35 marker at the beginning of the stream.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 30000

**EsamSignalProcessingNotification**

ESAM SignalProcessingNotification data defined by OC-SP-ESAM-API-I03-131025.

**sccXml**

Provide your ESAM SignalProcessingNotification XML document inside your JSON job settings. Form the XML document as per OC-SP-ESAM-API-I03-131025. The transcoder will use the signal processing instructions in the message that you supply. Provide your ESAM SignalProcessingNotification XML document inside your JSON job settings. If you want the service to place SCTE-35 markers at the insertion points you specify in the XML document, you must also enable SCTE-35 ESAM (scte35Esam). Note that you can either specify an ESAM XML document or enable SCTE-35 passthrough. You can't do both.

- **Type**: string
- **Required**: False
- **Pattern**: `^\s*<(.|
)*SignalProcessingNotification(.|
)*>$`

**ExceptionBody**

**message**

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

**F4vMoovPlacement**

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
Settings for F4v container

moovPlacement
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.

Type: F4vMoovPlacement (p. 328)
Required: False

FileGroupSettings
Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to FILE_GROUP_SETTINGS.

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: False
Pattern: ^s3:\/\/\/

destinationSettings
Settings associated with the destination. Will vary based on the type of destination

Type: DestinationSettings (p. 314)
Required: False

FileSourceConvert608To708
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
Settings for File-based Captions in Source

sourceFile
External caption file used for loading captions. Accepted file extensions are 'scc', 'ttml', 'dfxp', 'stl', 'srt', and 'smi'.
Properties

**Type**
string

**Required**
False

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

**MinLength**
14

**timeDelta**

Specifies a time delta in seconds to offset the captions from the source file.

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

**convert608To708**

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.

- **Type**: FileSourceConvert608To708 (p. 329)
- **Required**: False

**FontScript**

Provide the font script, using an ISO 15924 script code, if the LanguageCode is not sufficient for determining the script type. Where LanguageCode or CustomLanguageCode is sufficient, use “AUTOMATIC” or leave unset.

- AUTOMATIC
- HANS
- HANT

**FrameCaptureSettings**

Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value FRAME_CAPTURE.

**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

**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

quality

JPEG Quality - a higher value equals higher quality.

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

GetJobTemplateRequest

Query a job template by sending a request with the job template name.

name

The name of the job template.

Type: string
Required: False

GetJobTemplateResponse

Successful get job template requests will return an OK message and the job template JSON.

jobTemplate

A job template is a pre-made set of encoding instructions that you can use to quickly create a job.

Type: JobTemplate (p. 373)
Required: False

H264AdaptiveQuantization

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

OFF
LOW
MEDIUM
HIGH
HIGHER
MAX

**H264CodecLevel**

Specify an H.264 level that is consistent with your output video settings. If you aren't sure what level to specify, choose Auto (AUTO).

- 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**

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

**H264DynamicSubGop**

Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting B frames between reference frames (numberBFramesBetweenReferenceFrames).

- ADAPTIVE
- STATIC

**H264EntropyEncoding**

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

CABAC
CAVLC

H264FieldEncoding
Choosing FORCE_FIELD disables PAFF encoding for interlaced outputs.

PAFF
FORCE_FIELD

H264FlickerAdaptiveQuantization
Adjust quantization within each frame to reduce flicker or 'pop' on I-frames.

DISABLED
ENABLED

H264FramerateControl
If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

INITIALIZE_FROM_SOURCE
SPECIFIED

H264FramerateConversionAlgorithm
When set to INTERPOLATE, produces smoother motion during frame rate conversion.

DUPLICATE_DROP
INTERPOLATE

H264GopBReference
If enable, use reference B frames for GOP structures that have B frames > 1.

DISABLED
ENABLED

H264GopSizeUnits
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
H264InterlaceMode

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, as follows. - 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

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

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

H264QvbrSettings

Settings for quality-defined variable bitrate encoding with the H.264 codec. Required when you set Rate control mode to QVBR. Not valid when you set Rate control mode to a value other than QVBR, or when you don't define Rate control mode.

qvbrQualityLevel

Required when you use QVBR rate control mode. That is, when you specify qvbrSettings within h264Settings. Specify the target quality level for this output, from 1 to 10. Use higher numbers for greater quality. Level 10 results in nearly lossless compression. The quality level for most broadcast-quality transcodes is between 6 and 9.

Type: integer
  Required: False
  Minimum: 1
Maximum: 10

**maxAverageBitrate**

Use this setting only when Rate control mode is QVBR and Quality tuning level is Multi-pass HQ. For Max average bitrate values suited to the complexity of your input video, the service limits the average bitrate of the video part of this output to the value you choose. That is, the total size of the video element is less than or equal to the value you set multiplied by the number of seconds of encoded output.

- **Type**: integer
- **Required**: False
- **Minimum**: 1000
- **Maximum**: 1152000000

**H264RateControlMode**

Use this setting to specify whether this output has a variable bitrate (VBR), constant bitrate (CBR) or quality-defined variable bitrate (QVBR).

- VBR
- CBR
- QVBR

**H264RepeatPps**

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

- DISABLED
- ENABLED

**H264SceneChangeDetect**

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

- DISABLED
- ENABLED

**H264Settings**

Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value H_264.

**interlaceMode**

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, as follows. - 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.

- **Type**: H264InterlaceMode (p. 334)
Required: False

parNumerator

Pixel Aspect Ratio numerator.

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

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

syntax

Produces a bitstream compliant with SMPTE RP-2027.

Type: H264Syntax (p. 342)
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

framerateDenominator

When you use the API for transcode jobs that use frame rate conversion, specify the frame rate as a fraction. For example, \( \frac{24000}{1001} = 23.976 \text{ fps} \). 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 frame rate conversion, provide the value as a decimal number for Framerate. In this example, specify 23.976.

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

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

**hrdBufferInitialFillPercentage**

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

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

**gopSize**

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

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

**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

**gopBReference**

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

**Type**: `H264GopBReference` (p. 333)  
**Required**: False

**hrdBufferSize**

Size of buffer (HRD buffer model) in bits. For example, enter five megabits as 5000000.

**Type**: integer  
**Required**: False  
**Minimum**: 0  
**Maximum**: 1152000000

**maxBitrate**

Maximum bitrate in bits/second. For example, enter five megabits per second as 5000000. Required when Rate control mode is QVBR.
Type: integer
Required: False
Minimum: 1000
Maximum: 1152000000

slowPal

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

Type: H264SlowPal
Required: False

parDenominator

Pixel Aspect Ratio denominator.

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

spatialAdaptiveQuantization

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

Type: H264SpatialAdaptiveQuantization
Required: False

temporalAdaptiveQuantization

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

Type: H264TemporalAdaptiveQuantization
Required: False

flickerAdaptiveQuantization

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

Type: H264FlickerAdaptiveQuantization
Required: False

entropyEncoding

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

Type: H264EntropyEncoding
Required: False

bitrate

Average bitrate in bits/second. Required for VBR and CBR. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.
**Type**: integer  
**Required**: False  
**Minimum**: 1000  
**Maximum**: 1152000000

### framerateControl

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

**Type**: H264FramerateControl (p. 333)  
**Required**: False

### rateControlMode

Use this setting to specify whether this output has a variable bitrate (VBR), constant bitrate (CBR) or quality-defined variable bitrate (QVBR).

**Type**: H264RateControlMode (p. 335)  
**Required**: False

### qvbrSettings

Settings for quality-defined variable bitrate encoding with the H.264 codec. Required when you set Rate control mode to QVBR. Not valid when you set Rate control mode to a value other than QVBR, or when you don't define Rate control mode.

**Type**: H264QvbrSettings (p. 334)  
**Required**: False

### codecProfile

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

**Type**: H264CodecProfile (p. 332)  
**Required**: False

### telecine

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.

**Type**: H264Telecine (p. 342)  
**Required**: False
framerateNumerator

Frame rate numerator - frame rate is a fraction, e.g. 24000 / 1001 = 23.976 fps.

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

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

adaptiveQuantization

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

- **Type**: H264AdaptiveQuantization (p. 331)
- **Required**: False

codecLevel

Specify an H.264 level that is consistent with your output video settings. If you aren't sure what level to specify, choose Auto (AUTO).

- **Type**: H264CodecLevel (p. 332)
- **Required**: False

fieldEncoding

Choosing FORCE_FIELD disables PAFF encoding for interlaced outputs.

- **Type**: H264FieldEncoding (p. 333)
- **Required**: False

sceneChangeDetect

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

- **Type**: H264SceneChangeDetect (p. 335)
- **Required**: False

qualityTuningLevel

Use Quality tuning level (H264QualityTuningLevel) to specify whether to use fast single-pass, high-quality singlepass, or high-quality multipass video encoding.
Type: H264QualityTuningLevel (p. 334)  
Required: False

framerateConversionAlgorithm
When set to INTERPOLATE, produces smoother motion during frame rate conversion.

Type: H264FramerateConversionAlgorithm (p. 333)  
Required: False

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

Type: H264UnregisteredSeiT imecode (p. 342)  
Required: False

gopSizeUnits
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.

Type: H264GopSizeUnits (p. 333)  
Required: False

parControl
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.

Type: H264ParControl (p. 334)  
Required: False

numberBFramesBetweenReferenceFrames
Number of B-frames between reference frames.

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

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

Type: H264RepeatPps (p. 335)  
Required: False

dynamicSubGop
Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-
frames is limited by the value you provide for the setting B frames between reference frames (numberBFramesBetweenReferenceFrames).

**Type:** H264DynamicSubGop (p. 332)
**Required:** False

**H264SlowPal**
Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

DISABLED
ENABLED

**H264SpatialAdaptiveQuantization**
Adjust quantization within each frame based on spatial variation of content complexity.

DISABLED
ENABLED

**H264Syntax**
Produces a bitstream compliant with SMPTE RP-2027.

DEFAULT
RP2027

**H264Telecine**
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**
Adjust quantization within each frame based on temporal variation of content complexity.

DISABLED
ENABLED

**H264UnregisteredSeiTimecode**
Inserts timecode for each frame as 4 bytes of an unregistered SEI message.

DISABLED
**ENABLED**

**H265AdaptiveQuantization**
Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

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

**H265AlternateTransferFunctionSei**
Enables Alternate Transfer Function SEI message for outputs using Hybrid Log Gamma (HLG) Electro-Optical Transfer Function (EOTF).

- DISABLED
- ENABLED

**H265CodecLevel**
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**
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
H265DynamicSubGop

Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting B frames between reference frames (numberBFramesBetweenReferenceFrames).

ADAPTIVE
STATIC

H265FlickerAdaptiveQuantization

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

DISABLED
ENABLED

H265FramerateControl

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specication as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

INITIALIZE_FROM_SOURCE
SPECIFIED

H265FramerateConversionAlgorithm

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

DUPLICATE_DROP
INTERPOLATE

H265GopBReference

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

DISABLED
ENABLED

H265GopSizeUnits

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**

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**

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**

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

**H265QvbrSettings**

Settings for quality-defined variable bitrate encoding with the H.265 codec. Required when you set Rate control mode to QVBR. Not valid when you set Rate control mode to a value other than QVBR, or when you don’t define Rate control mode.

- **qvbrQualityLevel**
  - Type: integer
  - Required: False
  - Minimum: 1

345
**maxAverageBitrate**

Use this setting only when Rate control mode is QVBR and Quality tuning level is Multi-pass HQ. For Max average bitrate values suited to the complexity of your input video, the service limits the average bitrate of the video part of this output to the value you choose. That is, the total size of the video element is less than or equal to the value you set multiplied by the number of seconds of encoded output.

- **Type:** integer
- **Required:** False
- **Minimum:** 1000
- **Maximum:** 1466400000

**H265RateControlMode**

Use this setting to specify whether this output has a variable bitrate (VBR), constant bitrate (CBR) or quality-defined variable bitrate (QVBR).

- VBR
- CBR
- QVBR

**H265SampleAdaptiveOffsetFilterMode**

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

- DEFAULT
- ADAPTIVE
- OFF

**H265SceneChangeDetect**

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

- DISABLED
- ENABLED

**H265Settings**

Settings for H265 codec

**interlaceMode**

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.
**Properties**

- **Type**: `H265InterlaceMode (p. 345)`
  - **Required**: False

**parNumerator**

Pixel Aspect Ratio numerator.

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

**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

**framerateDenominator**

Frame rate denominator.

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

**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

**alternateTransferFunctionSei**

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

- **Type**: `H265AlternateTransferFunctionSei (p. 343)`
  - **Required**: False

**hrdBufferInitialFillPercentage**

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

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

gopSize

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

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

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

gopBReference

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

Type: H265GopBReference (p. 344)
Required: False

hrdBufferSize

Size of buffer (HRD buffer model) in bits. For example, enter five megabits as 5000000.

Type: integer
Required: False
Minimum: 0
Maximum: 1466400000

maxBitrate

Maximum bitrate in bits/second. For example, enter five megabits per second as 5000000. Required when Rate control mode is QVBR.

Type: integer
Required: False
Minimum: 1000
Maximum: 1466400000

slowPal

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.
**Type:** H265SlowPal (p. 353)
**Required:** False

**parDenominator**

Pixel Aspect Ratio denominator.

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

**spatialAdaptiveQuantization**

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

**Type:** H265SpatialAdaptiveQuantization (p. 353)
**Required:** False

**temporalAdaptiveQuantization**

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

**Type:** H265TemporalAdaptiveQuantization (p. 353)
**Required:** False

**flickerAdaptiveQuantization**

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

**Type:** H265FlickerAdaptiveQuantization (p. 344)
**Required:** False

**bitrate**

Average bitrate in bits/second. Required for VBR and CBR. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.

**Type:** integer
**Required:** False
**Minimum:** 1000
**Maximum:** 1466400000

**framerateControl**

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.
Properties

rateControlMode

Use this setting to specify whether this output has a variable bitrate (VBR), constant bitrate (CBR) or quality-defined variable bitrate (QVBR).

Type: H265RateControlMode (p. 346)
Required: False

qvbrSettings

Settings for quality-defined variable bitrate encoding with the H.265 codec. Required when you set Rate control mode to QVBR. Not valid when you set Rate control mode to a value other than QVBR, or when you don’t define Rate control mode.

Type: H265QvbrSettings (p. 345)
Required: False

codecProfile

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.

Type: H265CodecProfile (p. 343)
Required: False

tiles

Enable use of tiles, allowing horizontal as well as vertical subdivision of the encoded pictures.

Type: H265Tiles (p. 354)
Required: False

telecine

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.

Type: H265Telecine (p. 353)
Required: False

framerateNumerator

Frame rate numerator - frame rate is a fraction, e.g. 24000 / 1001 = 23.976 fps.

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

adaptiveQuantization

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

- **Type**: H265AdaptiveQuantization (p. 343)
- **Required**: False

codecLevel

H.265 Level.

- **Type**: H265CodecLevel (p. 343)
- **Required**: False

sceneChangeDetect

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

- **Type**: H265SceneChangeDetect (p. 346)
- **Required**: False

qualityTuningLevel

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

- **Type**: H265QualityTuningLevel (p. 345)
- **Required**: False

framerateConversionAlgorithm

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

- **Type**: H265FramerateConversionAlgorithm (p. 344)
- **Required**: False

unregisteredSeiTimecode

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

- **Type**: H265UnregisteredSeiTimecode (p. 354)
- **Required**: False
gopSizeUnits
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.

Type: H265GopSizeUnits (p. 344)
Required: False

parControl
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.

Type: H265ParControl (p. 345)
Required: False

numberBFramesBetweenReferenceFrames
Number of B-frames between reference frames.

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

temporalIds
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.

Type: H265TemporalIds (p. 353)
Required: False

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

Type: H265SampleAdaptiveOffsetFilterMode (p. 346)
Required: False

writeMp4PackagingType
Use this setting only for outputs encoded with H.265 that are in CMAF or DASH output groups. If you
include writeMp4PackagingType in your JSON job specification for other outputs, your video might not
work properly with downstream systems and video players. If the location of parameter set NAL units
don’t matter in your workflow, ignore this setting. The service defaults to marking your output as HEV1.
Choose HVC1 to mark your output as HVC1. This makes your output compliant with this specification:
ISO IEC/JTC1 SC29 N13798 Text ISO/IEC FDIS 14496-15 3rd Edition. For these outputs, the service stores
parameter set NAL units in the sample headers but not in the samples directly. Keep the default HEV1 to
mark your output as HEV1. For these outputs, the service writes parameter set NAL units directly into the
samples.
**Properties**

Type: H265WriteMp4PackagingType (p. 354)
Required: False

**dynamicSubGop**

Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting B frames between reference frames (numberBFramesBetweenReferenceFrames).

Type: H265DynamicSubGop (p. 344)
Required: False

**H265SlowPal**

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

DISABLED
ENABLED

**H265SpatialAdaptiveQuantization**

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

DISABLED
ENABLED

**H265Telecine**

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**

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

DISABLED
ENABLED

**H265TemporalIds**

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**

Enable use of tiles, allowing horizontal as well as vertical subdivision of the encoded pictures.

```
DISABLED
ENABLED
```

**H265UnregisteredSeiTimecode**

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

```
DISABLED
ENABLED
```

**H265WriteMp4PackagingType**

Use this setting only for outputs encoded with H.265 that are in CMAF or DASH output groups. If you include writeMp4PackagingType in your JSON job specification for other outputs, your video might not work properly with downstream systems and video players. If the location of parameter set NAL units don't matter in your workflow, ignore this setting. The service defaults to marking your output as HEV1. Choose HVC1 to mark your output as HVC1. This makes your output compliant with this specification: ISO IEC/JTC1 SC29 N13798 Text ISO/IEC FDIS 14496-15 3rd Edition. For these outputs, the service stores parameter set NAL units in the sample headers but not in the samples directly. Keep the default HEV1 to mark your output as HEV1. For these outputs, the service writes parameter set NAL units directly into the samples.

```
HVC1
HEV1
```

**Hdr10Metadata**

Use the "HDR master display information" (Hdr10Metadata) settings to correct HDR metadata or to provide missing metadata. These values vary depending on the input video and must be provided by a color grader. Range is 0 to 50,000; each increment represents 0.00002 in CIE1931 color coordinate. Note that these settings are not color correction. Note that if you are creating HDR outputs inside of an HLS CMAF package, to comply with the Apple specification, you must use the following settings. Set "MP4 packaging type" (writeMp4PackagingType) to HVC1 (HVC1). Set "Profile" (H265Settings > codecProfile) to Main10/High (MAIN10_HIGH). Set "Level" (H265Settings > codecLevel) to 5 (LEVEL_5).

```
redPrimaryX
```

**Hdr10Metadata**

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

```
Type: integer
```

354
Properties

**redPrimaryY**

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

*Type:* integer

*Required:* False

*Minimum:* 0

*Maximum:* 50000

**greenPrimaryX**

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

*Type:* integer

*Required:* False

*Minimum:* 0

*Maximum:* 50000

**greenPrimaryY**

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

*Type:* integer

*Required:* False

*Minimum:* 0

*Maximum:* 50000

**bluePrimaryX**

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

*Type:* integer

*Required:* False

*Minimum:* 0

*Maximum:* 50000

**bluePrimaryY**

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

*Type:* integer
whitePointX

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

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

whitePointY

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

- **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**: False
- **Minimum**: 0
- **Maximum**: 65535

maxContentLightLevel

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

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

maxLuminance

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

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 65535
**Minimum**: 2147483647

**minLuminance**

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

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

**HlsAdMarkers**

- ELEMENTAL
- ELEMENTAL_SCTE35

**HlsAudioTrackType**

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**

Caption Language Mapping

**captionChannel**

Caption channel.

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

**customLanguageCode**

Specify the language for this caption channel, using the ISO 639-2 or ISO 639-3 three-letter language code

- **Type**: string
- **Required**: False
**Language Code**


*Type:* LanguageCode (p. 376)

**Language Description**

Caption language description.

*Type:* string

**HlsCaptionLanguageSetting**

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.

*Options:*

- INSERT
- OMIT
- NONE

**HlsClientCache**

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

*Options:*

- DISABLED
- ENABLED

**HlsCodecSpecification**

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

*Options:*

- RFC_6381
- RFC_4281

**HlsDirectoryStructure**

Indicates whether segments should be placed in subdirectories.

*Options:*

- SINGLE_DIRECTORY
SUBDIRECTORY_PER_STREAM

**HlsEncryptionSettings**

Settings for HLS encryption

**encryptionMethod**

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

*Type: HlsEncryptionType (p. 360)*
*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}$**

**initializationVectorInManifest**

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.

*Type: HlsInitializationVectorInManifest (p. 364)*
*Required: False*

**offlineEncrypted**

Enable this setting to insert the EXT-X-SESSION-KEY element into the master playlist. This allows for offline Apple HLS FairPlay content protection.

*Type: HlsOfflineEncrypted (p. 364)*
*Required: False*

**spekeKeyProvider**

Settings for use with a SPEKE key provider

*Type: SpekeKeyProvider (p. 420)*
*Required: False*

**staticKeyProvider**

Use these settings to set up encryption with a static key provider.

*Type: StaticKeyProvider (p. 421)*
*Required: False*
**type**

Indicates which type of key provider is used for encryption.

- **Type:** HlsKeyProviderType (p. 364)
- **Required:** False

**HlsEncryptionType**

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**

Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to HLS_GROUP_SETTINGS.

**manifestDurationFormat**

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

- **Type:** HlsManifestDurationFormat (p. 364)
- **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:** False
- **Minimum:** 1
- **Maximum:** 2147483647

**timedMetadataId3Period**

Timed Metadata interval in seconds.

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

**captionLanguageSetting**

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.
**Properties**

**Type**: HlsCaptionLanguageSetting (p. 358)
**Required**: False

**captionLanguageMappings**
Language to be used on Caption outputs

**Type**: Array of type HlsCaptionLanguageMapping (p. 357)
**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**: False
**Pattern**: ^s3: \/

**destinationSettings**
Settings associated with the destination. Will vary based on the type of destination

**Type**: DestinationSettings (p. 314)
**Required**: False

**encryption**
DRM settings.

**Type**: HlsEncryptionSettings (p. 359)
**Required**: False

**timedMetadataId3Frame**
Indicates ID3 frame that has the timecode.

**Type**: HlsTimedMetadataId3Frame (p. 366)
**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

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

**Type**: HlsCodecSpecification (p. 358)
### Required: False

**outputSelection**

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

**Type:** HlsOutputSelection (p. 365)

**Required:** False

**programDateTimePeriod**

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

**Type:** integer

**Required:** False

**Minimum:** 0

**Maximum:** 3600

**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

**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:** False

**Minimum:** 0

**Maximum:** 2147483647

**minFinalSegmentLength**

Keep this setting at the default value of 0, unless you are troubleshooting a problem with how devices play back the end of your video asset. If you know that player devices are hanging on the final segment of your video because the length of your final segment is too short, use this setting to specify a minimum final segment length, in seconds. Choose a value that is greater than or equal to 1 and less than your segment length. When you specify a value for this setting, the encoder will combine any final segment that is shorter than the length that you specify with the previous segment. For example, your segment length is 3 seconds and your final segment is .5 seconds without a minimum final segment length; when you set the minimum final segment length to 1, your final segment is 3.5 seconds.

**Type:** number

**Required:** False

**Format:** float

**Minimum:** 0.0

**Maximum:** 2147483647
**directoryStructure**

Indicates whether segments should be placed in subdirectories.

  *Type*: HlsDirectoryStructure (p. 358)
  *Required*: False

**programDateTime**

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.

  *Type*: HlsProgramDateTime (p. 365)
  *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. 357)
  *Required*: False

**segmentControl**

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

  *Type*: HlsSegmentControl (p. 365)
  *Required*: False

**timestampDeltaMilliseconds**

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

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

**manifestCompression**

When set to GZIP, compresses HLS playlist.

  *Type*: HlsManifestCompression (p. 364)
  *Required*: False

**clientCache**

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

  *Type*: HlsClientCache (p. 358)
  *Required*: False
streamInfResolution

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

Type: HlsStreamInfResolution (p. 366)
Required: False

HlsIFrameOnlyManifest

When set to INCLUDE, writes I-Frame Only Manifest in addition to the HLS manifest

INCLUDE
EXCLUDE

HlsInitializationVectorInManifest

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

Indicates which type of key provider is used for encryption.

SPEKE
STATIC_KEY

HlsManifestCompression

When set to GZIP, compresses HLS playlist.

GZIP
NONE

HlsManifestDurationFormat

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

FLOATING_POINT
INTEGER

HlsOfflineEncrypted

Enable this setting to insert the EXT-X-SESSION-KEY element into the master playlist. This allows for offline Apple HLS FairPlay content protection.

ENABLED
DISABLED
**HlsOutputSelection**

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

- MANIFESTS_AND_SEGMENTS
- SEGMENTS_ONLY

**HlsProgramDateTime**

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**

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**

Settings for HLS output groups

**audioGroupId**

Specifies the group to which the audio Rendition belongs.

- **Type**: string
- **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

**audioTrackType**

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
Type: HlsAudioTrackType (p. 357)
Required: False

**iFrameOnlyManifest**
When set to INCLUDE, writes I-Frame Only Manifest in addition to the HLS manifest

Type: HlsIFrameOnlyManifest (p. 364)
Required: False

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

Type: string
Required: False

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

INCLUDE
EXCLUDE

**HlsTimedMetadataId3Frame**
Indicates ID3 frame that has the timecode.

NONE
PRIV
TDRL

**Id3Insertion**
To insert ID3 tags in your output, specify two values. Use ID3 tag (Id3) to specify the base 64 encoded string and use Timecode (TimeCode) to specify the time when the tag should be inserted. To insert multiple ID3 tags in your output, create multiple instances of ID3 insertion (Id3Insertion).

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

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

**id3**
Use ID3 tag (Id3) to provide a tag value in base64-encode format.

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

**ImageInserter**

Enable the image inserter feature to include a graphic overlay on your video. Enable or disable this feature for each input or output individually. This setting is disabled by default.

**insertableImages**

Specify the images that you want to overlay on your video. The images must be PNG or TGA files.

- **Type:** Array of type InsertableImage (p. 371)
- **Required:** False

**InputClipping**

To transcode only portions of your input (clips), include one Input clipping (one instance of InputClipping in the JSON job file) for each input clip. All input clips you specify will be included in every output of the job.

**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. Use the format HH:MM:SS:FF or HH:MM:SS;FF, where HH is the hour, MM is the minute, SS is the second, and FF is the frame number. When choosing this value, take into account your setting for timecode source under input settings (InputTimecodeSource). For example, if you have embedded timecodes that start at 01:00:00:00 and you want your clip to end six minutes into the video, use 01:06:00:00.

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

**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. Use the format HH:MM:SS:FF or HH:MM:SS;FF, where HH is the hour, MM is the minute, SS is the second, and FF is the frame number. 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 01:05:00:00.

- **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**

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

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**

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**

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**

**InputRotate**

Use Rotate (InputRotate) to specify how the service rotates your video. You can choose automatic rotation or specify a rotation. You can specify a clockwise rotation of 0, 90, 180, or 270 degrees. If your input video container is .mov or .mp4 and your input has rotation metadata, you can choose Automatic to have the service rotate your video according to the rotation specified in the metadata. The rotation must be within one degree of 90, 180, or 270 degrees. If the rotation metadata specifies any other rotation, the service will default to no rotation. By default, the service does no rotation, even if your input video has rotation metadata. The service doesn't pass through rotation metadata.

- **DEGREE_0**
- **DEGREES_90**
- **DEGREES_180**
- **DEGREES_270**
- **AUTO**

**InputTemplate**

Specified video input in a template.
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](#)
  **Required:** False

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

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:** 1
  **Maximum:** 2147483647

videoSelector

Selector for video.

  **Type:** [VideoSelector](#)
  **Required:** False

filterEnable

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.

  **Type:** [InputFilterEnable](#)
Required: False

psiControl
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.

Type: InputPsiControl (p. 368)
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

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

Type: InputDeblockFilter (p. 367)
Required: False

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

Type: InputDenoiseFilter (p. 368)
Required: False

timecodeSource
Timecode source under input settings (InputTimecodeSource) only affects the behavior of features that apply to a single input at a time, such as input clipping and synchronizing some captions formats. Use this setting to specify whether the service counts frames by timecodes embedded in the video (EMBEDDED) or by starting the first frame at zero (ZEROBASED). In both cases, the timecode format is HH:MM:SS:FF or HH:MM:SS;FF, where FF is the frame number. Only set this to EMBEDDED if your source video has embedded timecodes.

Type: InputTimecodeSource (p. 371)
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
Properties

**Required**: False

**imageInserter**

Enable the image inserter feature to include a graphic overlay on your video. Enable or disable this feature for each input individually. This setting is disabled by default.

*Type*: ImageInserter (p. 367)
*Required*: False

**InputTimecodeSource**

Timecode source under input settings (InputTimecodeSource) only affects the behavior of features that apply to a single input at a time, such as input clipping and synchronizing some captions formats. Use this setting to specify whether the service counts frames by timecodes embedded in the video (EMBEDDED) or by starting the first frame at zero (ZEROBASED). In both cases, the timecode format is HH:MM:SS:FF or HH:MM:SS;FF, where FF is the frame number. Only set this to EMBEDDED if your source video has embedded timecodes.

- EMBEDDED
- ZEROBASED
- SPECIFIED

**InsertableImage**

Settings that specify how your still graphic overlay appears.

**width**

Specify the width of the inserted image in pixels. If you specify a value that's larger than the video resolution width, the service will crop your overlaid image to fit. To use the native width of the image, keep this setting blank.

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

**height**

Specify the height of the inserted image in pixels. If you specify a value that's larger than the video resolution height, the service will crop your overlaid image to fit. To use the native height of the image, keep this setting blank.

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

**imageX**

Specify the distance, in pixels, between the inserted image and the left edge of the video frame. Required for any image overlay that you specify.
**Properties**

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

**imageY**
Specify the distance, in pixels, between the overlaid image and the top edge of the video frame. Required for any image overlay that you specify.
- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 2147483647

**duration**
Specify the time, in milliseconds, for the image to remain on the output video. This duration includes fade-in time but not fade-out time.
- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 2147483647

**fadeIn**
Specify the length of time, in milliseconds, between the Start time that you specify for the image insertion and the time that the image appears at full opacity. Full opacity is the level that you specify for the opacity setting. If you don’t specify a value for Fade-in, the image will appear abruptly at the overlay start time.
- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 2147483647

**layer**
Specify how overlapping inserted images appear. Images with higher values for Layer appear on top of images with lower values for Layer.
- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 99

**imageInserterInput**
Specify the Amazon S3 location of the image that you want to overlay on the video. Use a PNG or TGA file.
- **Type:** string
- **Required:** False
- **Pattern:** ^s3://.*\.(bmp|BMP|png|PNG|tga|TGA)$
**MinLength**: 14

**startTime**

Specify the timecode of the frame that you want the overlay to first appear on. This must be in timecode (HH:MM:SS:FF or HH:MM:SS;FF) format. Remember to take into account your timecode source settings.

- **Type**: string
- **Required**: False
- **Pattern**: ^(((\[0-1]\d)|(2\[0-3]\d))(\[0-5]\d)(2)(\[;\][0-5]\d))$ 

**fadeOut**

Specify the length of time, in milliseconds, between the end of the time that you have specified for the image overlay Duration and when the overlaid image has faded to total transparency. 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**: 0
- **Maximum**: 2147483647

**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**: False
- **Minimum**: 0
- **Maximum**: 100

**JobTemplate**

A job template is a pre-made set of encoding instructions that you can use to quickly create a job.

**arn**

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

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

**createdAt**

The timestamp in epoch seconds for Job template creation.

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

**lastUpdated**

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

Type: string
Required: False
Format: date-time

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

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

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

Type: string
Required: True

type
A job template can be of two types: system or custom. System or built-in job templates can't be modified or deleted by the user.

Type: Type (p. 426)
Required: False

settings
JobTemplateSettings contains all the transcode settings saved in the template that will be applied to jobs created from it.

Type: JobTemplateSettings (p. 375)
Required: True

accelerationSettings
Accelerated transcoding is currently in private preview. Contact AWS for more information.

Type: AccelerationSettings (p. 283)
**statusUpdateInterval**

Specify how often MediaConvert sends STATUS_UPDATE events to Amazon CloudWatch Events. Set the interval, in seconds, between status updates. MediaConvert sends an update at this interval from the time the service begins processing your job to the time it completes the transcode or encounters an error.

*Type:* StatusUpdateInterval (p. 421)
*Required:* False

**JobTemplateSettings**

JobTemplateSettings contains all the transcode settings saved in the template that will be applied to jobs created from it.

**timecodeConfig**

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

*Type:* TimecodeConfig (p. 423)
*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 * CMAF_GROUP_SETTINGS, CmafGroupSettings

*Type:* Array of type OutputGroup (p. 411)
*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

**availBlanking**

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

*Type:* AvailBlanking (p. 292)
*Required:* False
**timedMetadataInsertion**

Enable Timed metadata insertion (TimedMetadataInsertion) to include ID3 tags in your job. To include timed metadata, you must enable it here, enable it in each output container, and specify tags and timecodes in ID3 insertion (Id3Insertion) objects.

- **Type:** TimedMetadataInsertion (p. 425)
- **Required:** False

**nielsenConfiguration**

Settings for Nielsen Configuration

- **Type:** NielsenConfiguration (p. 407)
- **Required:** False

**motionImageInserter**

Overlay motion graphics on top of your video. The motion graphics that you specify here appear on all outputs in all output groups.

- **Type:** MotionImageInserter (p. 392)
- **Required:** False

**esam**

Settings for Event Signaling And Messaging (ESAM).

- **Type:** EsamSettings (p. 327)
- **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. 368)
- **Required:** False

**LanguageCode**


- ENG
- SPA
- FRA
- DEU
- GER
- ZHO
- ARA
- HIN
- JPN
- RUS
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
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
**M2tsAudioBufferModel**

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

- DVB
- ATSC

**M2tsBufferModel**

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**

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**

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**

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

- INCLUDE
- EXCLUDE

**M2tsForceTsVideoEbpOrder**

Keep the default value (DEFAULT) unless you know that your audio EBP markers are incorrectly appearing before your video EBP markers. To correct this problem, set this value to Force (FORCE).

- FORCE
- DEFAULT

**M2tsNielsenId3**

If INSERT, Nielsen inaudible tones for media tracking will be detected in the input audio and an equivalent ID3 tag will be inserted in the output.

- INSERT
- NONE

**M2tsPcrControl**

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**

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

**M2tsScte35Esam**

Settings for SCTE-35 signals from ESAM. Include this in your job settings to put SCTE-35 markers in your HLS and transport stream outputs at the insertion points that you specify in an ESAM XML document. Provide the document in the setting SCC XML (sccXml).

**scte35EsamPid**

Packet Identifier (PID) of the SCTE-35 stream in the transport stream generated by ESAM.
**Properties**

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

**M2tsScte35Source**

Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output.

- PASSTHROUGH
- NONE

**M2tsSegmentationMarkers**

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**

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**

MPEG-2 TS container settings. These apply to outputs in a File output group when the output's container (ContainerType) is MPEG-2 Transport Stream (M2TS). In these assets, data is organized by the program map table (PMT). Each transport stream program contains subsets of data, including audio, video, and metadata. Each of these subsets of data has a numerical label called a packet identifier (PID). Each transport stream program corresponds to one MediaConvert output. The PMT lists the types of data in a program along with their PID. Downstream systems and players use the program map table to look up the PID for each type of data it accesses and then uses the PIDs to locate specific data within the asset.
AudioBufferModel
Selects between the DVB and ATSC buffer models for Dolby Digital audio.

**Type:** M2tsAudioBufferModel (p. 380)
**Required:** False

MinimumEbpInterval
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

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

**Type:** M2tsEsRateInPes (p. 381)
**Required:** False

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

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

dvbNitSettings
Inserts DVB Network Information Table (NIT) at the specified table repetition interval.

**Type:** DvbNitSettings (p. 314)
**Required:** False

dvbSdtSettings
Inserts DVB Service Description Table (NIT) at the specified table repetition interval.

**Type:** DvbSdtSettings (p. 315)
**Required:** False

Scte35Source
Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output.

**Type:** M2tsScte35Source (p. 382)
**Required:** False
**scte35Pid**

Specify the packet identifier (PID) of the SCTE-35 stream in the transport stream.

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

**scte35Esam**

Include this in your job settings to put SCTE-35 markers in your HLS and transport stream outputs at the insertion points that you specify in an ESAM XML document. Provide the document in the setting SCC XML (sccXml).

- **Type**: M2tsScte35Esam (p. 381)
- **Required**: False

**videoPid**

Specify the packet identifier (PID) of the elementary video stream in the transport stream.

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

**dvbTdtSettings**

Inserts DVB Time and Date Table (TDT) at the specified table repetition interval.

- **Type**: DvbTdtSettings (p. 320)
- **Required**: False

**pmtInterval**

Specify the number of milliseconds between instances of the program map table (PMT) in the output transport stream.

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

**segmentationStyle**

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.
**Properties**

**segmentationTime**
Specify the length, in seconds, of each segment. Required unless markers is set to `none`.

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

**pmtPid**
Specify the packet identifier (PID) for the program map table (PMT) itself. Default is 480.

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

**bitrate**
Specify 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

**audioPids**
Specify the packet identifiers (PIDs) for any elementary audio streams you include in this output. Specify multiple PIDs as a JSON array. Default is the range 482-492.

*Type:* Array of type integer  
*Required:* False  
*Minimum:* 32  
*Maximum:* 8182

**privateMetadataPid**
Specify the packet identifier (PID) of the private metadata stream. Default is 503.

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

**nielsenId3**
If INSERT, Nielsen inaudible tones for media tracking will be detected in the input audio and an equivalent ID3 tag will be inserted in the output.
**Properties**

**timedMetadataPid**
Specify the packet identifier (PID) for timed metadata in this output. Default is 502.

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

**maxPcrInterval**
Specify the maximum time, in milliseconds, between Program Clock References (PCRs) inserted into the transport stream.

Type: integer  
Required: False
Minimum: 0  
Maximum: 500

**transportStreamId**
Specify the ID for the transport stream itself in the program map table for this output. Transport stream IDs and program map tables are parts of MPEG-2 transport stream containers, used for organizing data.

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

**dvbSubPids**
Specify the packet identifiers (PIDs) for DVB subtitle data included in this output. Specify multiple PIDs as a JSON array. Default is the range 460-479.

Type: Array of type integer  
Required: False
Minimum: 32  
Maximum: 8182

**rateMode**
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.

Type: M2tsRateMode (p. 381)  
Required: False

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

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

pcrControl

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.

Type: M2tsPcrControl (p. 381)
Required: False

segmentationMarkers

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.

Type: M2tsSegmentationMarkers (p. 382)
Required: False

ebpAudioInterval

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).

Type: M2tsEbpAudioInterval (p. 380)
Required: False

forceTsVideoEbpOrder

Keep the default value (DEFAULT) unless you know that your audio EBP markers are incorrectly appearing before your video EBP markers. To correct this problem, set this value to Force (FORCE).

Type: M2tsForceTsVideoEbpOrder (p. 381)
Required: False

programNumber

Use Program number (programNumber) to specify the program number used in the program map table (PMT) for this output. Default is 1. Program numbers and program map tables are parts of MPEG-2 transport stream containers, used for organizing data.

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

Specify the packet identifier (PID) for the program clock reference (PCR) in this output. If you do not specify a value, the service will use the value for Video PID (VideoPid).

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

**bufferModel**

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.

- **Type:** M2tsBufferModel (p. 380)
- **Required:** False

**dvbTeletextPid**

Specify the packet identifier (PID) for DVB teletext data you include in this output. Default is 499.

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

**fragmentTime**

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

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

**ebpPlacement**

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).

- **Type:** M2tsEbpPlacement (p. 380)
- **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

M3u8NielsenId3
If INSERT, Nielsen inaudible tones for media tracking will be detected in the input audio and an equivalent ID3 tag will be inserted in the output.

  INSERT
  NONE

M3u8PcrControl
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
Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output.

  PASSTHROUGH
  NONE

M3u8Settings
Settings for TS segments in HLS

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

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

  pcrControl
  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.

  Type: M3u8PcrControl (p. 389)
  Required: False

  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.
Properties

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

pmtPid
Packet Identifier (PID) for the Program Map Table (PMT) in the transport stream.

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

privateMetadataPid
Packet Identifier (PID) of the private metadata stream in the transport stream.

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

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

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

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

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

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

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

scte35Source
Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output.
Type: `M3u8Scte35Source` (p. 389)
Required: False

`scte35Pid`
Packet Identifier (PID) of the SCTE-35 stream in the transport stream.

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

`nielsenId3`
If INSERT, Nielsen inaudible tones for media tracking will be detected in the input audio and an equivalent ID3 tag will be inserted in the output.

- Type: `M3u8NielsenId3` (p. 389)
- Required: False

`timedMetadata`
Applies only to HLS outputs. Use this setting to specify whether the service inserts the ID3 timed metadata from the input in this output.

- Type: `TimedMetadata` (p. 425)
- Required: False

`timedMetadataPid`
Packet Identifier (PID) of the timed metadata stream in the transport stream.

- 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.

- 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.

- **Type:** Array of type integer
- **Required:** False
- **Minimum:** 32
- **Maximum:** 8182

MotionImageInserter
Overlay motion graphics on top of your video at the time that you specify.

**insertionMode**
Choose the type of motion graphic asset that you are providing for your overlay. You can choose either a .mov file or a series of .png files.

- **Type:** MotionImageInsertionMode
- **Required:** False

**input**
Specify the .mov file or series of .png files that you want to overlay on your video. For .png files, provide the file name of the first file in the series. Make sure that the names of the .png files end with sequential numbers that specify the order that they are played in. For example, overlay_000.png, overlay_001.png, overlay_002.png, and so on. The sequence must start at zero, and each image file name must have the same number of digits. Pad your initial file names with enough zeros to complete the sequence. For example, if the first image is overlay_0.png, there can be only 10 images in the sequence, with the last image being overlay_9.png. But if the first image is overlay_00.png, there can be 100 images in the sequence.

- **Type:** string
- **Required:** False
- **Pattern:** ^s3://\/*\.[\(0-9\]+\.png)$
- **MinLength:** 14
- **MaxLength:** 1285

**offset**
Use Offset to specify the placement of your motion graphic overlay on the video frame. Specify in pixels, from the upper-left corner of the frame. If you don't specify an offset, the service scales your overlay to the full size of the frame. Otherwise, the service inserts the overlay at its native resolution and scales the size up or down with any video scaling.

- **Type:** MotionImageInsertionOffset
- **Required:** False

**startTime**
Specify when the motion overlay begins. Use timecode format (HH:MM:SS:FF or HH:MM:SS;FF). Make sure that the timecode you provide here takes into account how you have set up your timecode configuration under both job settings and input settings. The simplest way to do that is to set both to start at 0. If you need to set up your job to follow timecodes embedded in your source that don't start
at zero, make sure that you specify a start time that is after the first embedded timecode. For more information, see https://docs.aws.amazon.com/mediaconvert/latest/ug/setting-up-timecode.html Find job-wide and input timecode configuration settings in your JSON job settings specification at settings>timecodeConfig>source and settings>inputs>timecodeSource.

**Properties**

Type: string  
Required: False  
MinLength: 11  
MaxLength: 11

**playback**

Specify whether your motion graphic overlay repeats on a loop or plays only once.

Type: MotionImagePlayback (p. 394)  
Required: False

**framerate**

If your motion graphic asset is a .mov file, keep this setting unspecified. If your motion graphic asset is a series of .png files, specify the frame rate of the overlay in frames per second, as a fraction. For example, specify 24 fps as 24/1. Make sure that the number of images in your series matches the frame rate and your intended overlay duration. For example, if you want a 30-second overlay at 30 fps, you should have 900 .png images. This overlay frame rate doesn't need to match the frame rate of the underlying video.

Type: MotionImageInsertionFramerate (p. 393)  
Required: False

**MotionImageInsertionFramerate**

For motion overlays that don't have a built-in frame rate, specify the frame rate of the overlay in frames per second, as a fraction. For example, specify 24 fps as 24/1. The overlay frame rate doesn't need to match the frame rate of the underlying video.

**framerateNumerator**

The top of the fraction that expresses your overlay frame rate. For example, if your frame rate is 24 fps, set this value to 24.

Type: integer  
Required: False  
Minimum: 1  
Maximum: 2147483640

**framerateDenominator**

The bottom of the fraction that expresses your overlay frame rate. For example, if your frame rate is 24 fps, set this value to 1.

Type: integer  
Required: False  
Minimum: 1  
Maximum: 17895697
**MotionImageInsertionMode**

Choose the type of motion graphic asset that you are providing for your overlay. You can choose either a .mov file or a series of .png files.

- MOV
- PNG

**MotionImageInsertionOffset**

Specify the offset between the upper-left corner of the video frame and the top left corner of the overlay.

---

**imageX**

Set the distance, in pixels, between the overlay and the left edge of the video frame.

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

---

**imageY**

Set the distance, in pixels, between the overlay and the top edge of the video frame.

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

**MotionImagePlayback**

Specify whether your motion graphic overlay repeats on a loop or plays only once.

- ONCE
- REPEAT

**MovClapAtom**

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

- INCLUDE
- EXCLUDE

**MovCslgAtom**

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**

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**

If set to OMNEON, inserts Omneon-compatible padding

- OMNEON
- NONE

**MovReference**

Always keep the default value (SELF_CONTAINED) for this setting.

- SELF_CONTAINED
- EXTERNAL

**MovSettings**

Settings for MOV Container.

**clapAtom**

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

- Type: MovClapAtom (p. 394)
- Required: False

**cslgAtom**

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.

- Type: MovCslgAtom (p. 394)
- Required: False

**paddingControl**

If set to OMNEON, inserts Omneon-compatible padding

- Type: MovPaddingControl (p. 395)
- Required: False

**reference**

Always keep the default value (SELF_CONTAINED) for this setting.
**Properties**

**Type**: MovReference (p. 395)  
**Required**: False

**mpeg2FourCCControl**

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.

**Type**: MovMpeg2FourCCControl (p. 395)  
**Required**: False

**Mp2Settings**

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value MP2.

**bitrate**

Average bitrate in bits/second.

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

**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

**sampleRate**

Sample rate in hz.

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

**Mp4CslgAtom**

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**

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

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

**Mp4MoovPlacement**

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**

Settings for MP4 Container

**cslgAtom**

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.

- **Type:** Mp4CslgAtom (p. 396)
- **Required:** False

**freeSpaceBox**

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

- **Type:** Mp4FreeSpaceBox (p. 397)
- **Required:** False

**mp4MajorBrand**

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

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

**moovPlacement**

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.

- **Type:** Mp4MoovPlacement (p. 397)
- **Required:** False

**Mpeg2AdaptiveQuantization**

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

Mpeg2CodecLevel

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

- AUTO
- LOW
- MAIN
- HIGH
- MAIN 1440
- HIGH

Mpeg2CodecProfile

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

- MAIN
- PROFILE_422

Mpeg2DynamicSubGop

Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting B frames between reference frames
(numberBFramesBetweenReferenceFrames).

- ADAPTIVE
- STATIC

Mpeg2FramerateControl

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job sepecification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

- INITIALIZE_FROM_SOURCE
- SPECIFIED

Mpeg2FramerateConversionAlgorithm

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

- DUPLICATE_DROP
**INTERPOLATE**

**Mpeg2GopSizeUnits**
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**
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**
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**
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**
Use Quality tuning level (Mpeg2QualityTuningLevel) to specify whether to use single-pass or multipass video encoding.

- SINGLE_PASS
MULTI_PASS

**Mpeg2RateControlMode**

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

- VBR
- CBR

**Mpeg2SceneChangeDetect**

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

- DISABLED
- ENABLED

**Mpeg2Settings**

Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value MPEG2.

**interlaceMode**

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.

- **Type:** Mpeg2InterlaceMode (p. 399)
- **Required:** False

**parNumerator**

Pixel Aspect Ratio numerator.

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

**syntax**

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

- **Type:** Mpeg2Syntax (p. 405)
- **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

framerateDenominator

Frame rate denominator.
Type: integer
Required: False
Minimum: 1
Maximum: 1001

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

hrdBufferInitialFillPercentage

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

gopSize

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

hrdBufferSize

Size of buffer (HRD buffer model) in bits. For example, enter five megabits as 5000000.
Type: integer
Required: False
Minimum: 0
Maximum: 47185920

maxBitrate

Maximum bitrate in bits/second. For example, enter five megabits per second as 5000000.
## Properties

**Type**: integer  
**Required**: False  
**Minimum**: 1000  
**Maximum**: 300000000

### slowPal

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

**Type**: Mpeg2SlowPal (p. 405)  
**Required**: False

### parDenominator

Pixel Aspect Ratio denominator.

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

### spatialAdaptiveQuantization

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

**Type**: Mpeg2SpatialAdaptiveQuantization (p. 405)  
**Required**: False

### temporalAdaptiveQuantization

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

**Type**: Mpeg2TemporalAdaptiveQuantization (p. 406)  
**Required**: False

### bitrate

Average bitrate in bits/second. Required for VBR and CBR. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.

**Type**: integer  
**Required**: False  
**Minimum**: 1000  
**Maximum**: 288000000

### intraDcPrecision

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.

**Type**: Mpeg2IntraDcPrecision (p. 399)  
**Required**: False
framerateControl

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job sepecification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

Type: Mpeg2FramerateControl (p. 398)
Required: False

rateControlMode

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

Type: Mpeg2RateControlMode (p. 400)
Required: False

codecProfile

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

Type: Mpeg2CodecProfile (p. 398)
Required: False

telecine

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.

Type: Mpeg2Telecine (p. 405)
Required: False

framerateNumerator

Frame rate numerator - frame rate is a fraction, e.g. 24000 / 1001 = 23.976 fps.

Type: integer
Required: False
Minimum: 24
Maximum: 60000

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
**Properties**

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

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

**Type**: Mpeg2AdaptiveQuantization  
**Required**: False

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

**Type**: Mpeg2CodecLevel  
**Required**: False

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

**Type**: Mpeg2SceneChangeDetect  
**Required**: False

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

**Type**: Mpeg2QualityTuningLevel  
**Required**: False

**framerateConversionAlgorithm**  
When set to INTERPOLATE, produces smoother motion during frame rate conversion.

**Type**: Mpeg2FramerateConversionAlgorithm  
**Required**: False

**gopSizeUnits**  
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.

**Type**: Mpeg2GopSizeUnits  
**Required**: False

**parControl**  
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.
**Properties**

**Type:** Mpeg2ParControl (p. 399)  
**Required:** False

### numberBFramesBetweenReferenceFrames

Number of B-frames between reference frames.

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

### dynamicSubGop

Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting B frames between reference frames (numberBFramesBetweenReferenceFrames).

**Type:** Mpeg2DynamicSubGop (p. 398)  
**Required:** False

### Mpeg2SlowPal

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

**DISABLED**  
**ENABLED**

### Mpeg2SpatialAdaptiveQuantization

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

**DISABLED**  
**ENABLED**

### Mpeg2Syntax

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

**DEFAULT**  
**D_10**

### Mpeg2Telecine

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**
Mpeg2TemporalAdaptiveQuantization

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

- DISABLED
- ENABLED

MsSmoothAudioDeduplication

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

- COMBINE_DUPLICATE_STREAMS
- NONE

MsSmoothEncryptionSettings

If you are using DRM, set DRM System (MsSmoothEncryptionSettings) to specify the value SpekeKeyProvider.

spekeKeyProvider

Settings for use with a SPEKE key provider

- Type: SpekeKeyProvider (p. 420)
- Required: False

MsSmoothGroupSettings

Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to MS_SMOOTH_GROUP_SETTINGS.

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: False
- Pattern: ^s3:\/\/

destinationSettings

Settings associated with the destination. Will vary based on the type of destination

- Type: DestinationSettings (p. 314)
- Required: False
**fragmentLength**

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

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

**encryption**

If you are using DRM, set DRM System (MsSmoothEncryptionSettings) to specify the value SpekeKeyProvider.

- **Type:** MsSmoothEncryptionSettings (p. 406)
- **Required:** False

**manifestEncoding**

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

- **Type:** MsSmoothManifestEncoding (p. 407)
- **Required:** False

**audioDeduplication**

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

- **Type:** MsSmoothAudioDeduplication (p. 406)
- **Required:** False

**MsSmoothManifestEncoding**

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

- UTF8
- UTF16

**NielsenConfiguration**

Settings for Nielsen Configuration

**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
distributorId

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

Type: string
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. When you enable Noise reducer (NoiseReducer), you must also select a value for Noise reducer filter (NoiseReducerFilter).

filter

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 a frequency-domain filter based on JND principles.

Type: NoiseReducerFilter (p. 408)
Required: False

filterSettings

Settings for a noise reducer filter

Type: NoiseReducerFilterSettings (p. 409)
Required: False

spatialFilterSettings

Noise reducer filter settings for spatial filter.

Type: NoiseReducerSpatialFilterSettings (p. 409)
Required: False

NoiseReducerFilter

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 a frequency-domain filter based on JND principles.

BILATERAL
MEAN
GAUSSIAN
LANCZOS
SHARPEN
CONSERVE
SPATIAL
## NoiseReducerFilterSettings

Settings for a noise reducer filter

**strength**

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

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

## NoiseReducerSpatialFilterSettings

Noise reducer filter settings for spatial filter.

**strength**

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

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

**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

## 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

## Output

An output object describes the settings for a single output file or stream in an output group.

**containerSettings**

Container specific settings.

- **Type:** ContainerSettings (p. 308)
- **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
- **MinLength**: 0

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. 428)
- **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. 286)
- **Required**: False

outputSettings

Specific settings for this type of output.

- **Type**: OutputSettings (p. 413)
- **Required**: False

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

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 (NameModifier) 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
- **MinLength**: 1
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. 297)
  Required: False

OutputChannelMapping

OutputChannel mapping settings.

inputChannels

List of input channels

  Type: Array of type integer
  Required: False
  Minimum: -60
  Maximum: 6

OutputGroup

Group of outputs

  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

name

Name of the output group

  Type: string
  Required: False

outputs

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

  Type: Array of type Output (p. 409)
  Required: False

outputGroupSettings

Output Group settings, including type

  Type: OutputGroupSettings (p. 412)
  Required: False
OutputGroupSettings

Output Group settings, including type

type

Type of output group (File group, Apple HLS, DASH ISO, Microsoft Smooth Streaming, CMAF)

Type: OutputGroupType (p. 412)
Required: False

hlsGroupSettings

Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to HLS_GROUP_SETTINGS.

Type: HlsGroupSettings (p. 360)
Required: False

dashIsoGroupSettings

Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to DASH_ISO_GROUP_SETTINGS.

Type: DashIsoGroupSettings (p. 310)
Required: False

fileGroupSettings

Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to FILE_GROUP_SETTINGS.

Type: FileGroupSettings (p. 329)
Required: False

msSmoothGroupSettings

Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to MS_SMOOTH_GROUP_SETTINGS.

Type: MsSmoothGroupSettings (p. 406)
Required: False

cmafGroupSettings

Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to CMAF_GROUP_SETTINGS. Each output in a CMAF Output Group may only contain a single video, audio, or caption output.

Type: CmafGroupSettings (p. 303)
Required: False

OutputGroupType

Type of output group (File group, Apple HLS, DASH ISO, Microsoft Smooth Streaming, CMAF)
HLS_GROUP_SETTINGS
DASH_ISO_GROUP_SETTINGS
FILE_GROUP_SETTINGS
MS_SMOOTH_GROUP_SETTINGS
CMAF_GROUP_SETTINGS

OutputSdt

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

Specific settings for this type of output.

hlsSettings

Settings for HLS output groups

Type: HlsSettings (p. 365)
Required: False

ProresCodecProfile

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

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

- INITIALIZE_FROM_SOURCE
**ProresFramerateConversionAlgorithm**

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

- **DUPLICATE_DROP**
- **INTERPOLATE**

**ProresInterlaceMode**

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**

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**

Required when you set (Codec) under (VideoDescription)->(CodecSettings) to the value PRORES.

**interlaceMode**

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.

- **Type**: ProresInterlaceMode (p. 414)
- **Required**: False
**parNumerator**

Pixel Aspect Ratio numerator.

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

**framerateDenominator**

Frame rate denominator.

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

**codecProfile**

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

- **Type**: ProresCodecProfile (p. 413)
- **Required**: False

**slowPal**

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

- **Type**: ProresSlowPal (p. 416)
- **Required**: False

**parDenominator**

Pixel Aspect Ratio denominator.

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

**framerateControl**

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

- **Type**: ProresFramerateControl (p. 413)
- **Required**: False
telecine

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.

Type: ProresTelecine (p. 416)  
Required: False

framerateNumerator

When you use the API for transcode jobs that use frame rate conversion, specify the frame rate 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

framerateConversionAlgorithm

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

Type: ProresFramerateConversionAlgorithm (p. 414)  
Required: False

parControl

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).

Type: ProresParControl (p. 414)  
Required: False

ProresSlowPal

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

DISABLED  
ENABLED

ProresTelecine

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

Use Rectangle to identify a specific area of the video frame.

height

Height of rectangle in pixels. Specify only even numbers.

Type: integer  
Required: False  
Minimum: 2  
Maximum: 2147483647

width

Width of rectangle in pixels. Specify only even numbers.

Type: integer  
Required: False  
Minimum: 2  
Maximum: 2147483647

x

The distance, in pixels, between the rectangle and the left edge of the video frame. Specify only even numbers.

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

y

The distance, in pixels, between the rectangle and the top edge of the video frame. Specify only even numbers.

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

RemixSettings

Use Manual audio remixing (RemixSettings) to adjust audio levels for each audio channel in each output of your job. With audio remixing, you can output more or fewer audio channels than your input audio source provides.

channelMapping

Channel mapping (ChannelMapping) contains the group of fields that hold the remixing value for each channel. Units are in dB. Acceptable values are within the range from -60 (mute) through 6. A setting of 0 passes the input channel unchanged to the output channel (no attenuation or amplification).

Type: ChannelMapping (p. 301)  
Required: False
**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:** False
- **Minimum:** 1
- **Maximum:** 16

**channelsOut**

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

- **Type:** integer
- **Required:** False
- **Minimum:** 1
- **Maximum:** 8

**RespondToAfd**

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**

**S3DestinationSettings**

Settings associated with S3 destination encryption

**encryption**

Settings for how your job outputs are encrypted as they are uploaded to Amazon S3.

- **Type:** S3EncryptionSettings (p. 418)
- **Required:** False

**S3EncryptionSettings**

Settings for how your job outputs are encrypted as they are uploaded to Amazon S3.

**encryptionType**

Specify how you want your data keys managed. AWS uses data keys to encrypt your content. AWS also encrypts the data keys themselves, using a customer master key (CMK), and then stores the encrypted data keys alongside your encrypted content. Use this setting to specify which AWS service manages the CMK. For simplest set up, choose Amazon S3 (SERVER_SIDE_ENCRYPTION_S3). If you want your master key to be managed by AWS Key Management Service (KMS), choose AWS KMS.
By default, when you choose AWS KMS, KMS uses the AWS managed customer master key (CMK) associated with Amazon S3 to encrypt your data keys. You can optionally choose to specify a different, customer managed CMK. Do so by specifying the Amazon Resource Name (ARN) of the key for the setting KMS ARN (kmsKeyArn).

**Type:** S3ServerSideEncryptionType (p. 419)

**Required:** False

### kmsKeyArn

Optionally, specify the customer master key (CMK) that you want to use to encrypt the data key that AWS uses to encrypt your output content. Enter the Amazon Resource Name (ARN) of the CMK. To use this setting, you must also set Server-side encryption (S3ServerSideEncryptionType) to AWS KMS (SERVER_SIDE_ENCRYPTION_KMS). If you set Server-side encryption to AWS KMS but don't specify a CMK here, AWS uses the AWS managed CMK associated with Amazon S3.

**Type:** string

**Required:** False

**Pattern:** ^arn:aws(-us-gov)?:kms:[a-z-]{2,6}-(east|west|central|((north|south) (east|west)?))-[1-9]{1,2}:d(12):key/[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{12}$

### S3ServerSideEncryptionType

Specify how you want your data keys managed. AWS uses data keys to encrypt your content. AWS also encrypts the data keys themselves, using a customer master key (CMK), and then stores the encrypted data keys alongside your encrypted content. Use this setting to specify which AWS service manages the CMK. For simplest setup, choose Amazon S3 (SERVER_SIDE_ENCRYPTION_S3). If you want your master key to be managed by AWS Key Management Service (KMS), choose AWS KMS (SERVER_SIDE_ENCRYPTION_KMS). By default, when you choose AWS KMS, KMS uses the AWS managed customer master key (CMK) associated with Amazon S3 to encrypt your data keys. You can optionally choose to specify a different, customer managed CMK. Do so by specifying the Amazon Resource Name (ARN) of the key for the setting KMS ARN (kmsKeyArn).

- SERVER_SIDE_ENCRYPTION_S3
- SERVER_SIDE_ENCRYPTION_KMS

### ScalingBehavior

Applies only if your input aspect ratio is different from your output aspect ratio. Choose "Stretch to output" to have the service stretch your video image to fit. Keep the setting "Default" 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

Set Framerate (SccDestinationFramerate) to make sure that the captions and the video are synchronized in the output. Specify a frame rate that matches the frame rate of the associated video. If the video frame rate 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**

Settings for SCC caption output.

**framerate**

Set framerate (SccDestinationFramerate) to make sure that the captions and the video are synchronized in the output. Specify a frame rate that matches the frame rate of the associated video. If the video frame rate 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).

*Type:* SccDestinationFramerate (p. 419)

*Required:* False

**SpekeKeyProvider**

Settings for use with a SPEKE key provider

**resourceId**

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

*Type:* string

*Required:* False

**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:* False

*Pattern:* ^[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}$

**url**

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

*Type:* string

*Required:* False

*Format:* uri

*Pattern:* ^https:\/\/

**certificateArn**

Optional AWS Certificate Manager ARN for a certificate to send to the keyprovider. The certificate holds a key used by the keyprovider to encrypt the keys in its response.
Properties

Type: string  
Required: False  
Pattern: ^arn:aws(-us-gov)?:acm:

StaticKeyProvider

Use these settings to set up encryption with a static key provider.

staticKeyValue

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

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

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}+)?)$

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+)*)$

url

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

Type: string  
Required: False  
Format: uri

StatusUpdateInterval

Specify how often MediaConvert sends STATUS_UPDATE events to Amazon CloudWatch Events. Set the interval, in seconds, between status updates. MediaConvert sends an update at this interval from the time the service begins processing your job to the time it completes the transcode or encounters an error.

SECONDS_10  
SECONDS_12  
SECONDS_15  
SECONDS_20
SECONDS_30
SECONDS_60
SECONDS_120
SECONDS_180
SECONDS_240
SECONDS_300
SECONDS_360
SECONDS_420
SECONDS_480
SECONDS_540
SECONDS_600

**TeletextDestinationSettings**

Settings for Teletext caption output

**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]$`  
MinLength: 3  
MaxLength: 3

**TeletextSourceSettings**

Settings specific to Teletext caption sources, including Page number.

**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]$`  
MinLength: 3  
MaxLength: 3

**TimecodeBurnin**

Timecode burn-in (TimecodeBurnIn)--Burns the output timecode and specified prefix into the output.

**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**

Use Position (Position) under under Timecode burn-in (TimecodeBurnIn) to specify the location the burned-in timecode on output video.

*Type: TimecodeBurninPosition (p. 423)*

*Required: False*

**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: ^[ -~]+$*

**TimecodeBurninPosition**

Use Position (Position) under 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**

These settings control how the service handles timecodes throughout the job. These settings don’t affect input clipping.

**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 frame rate conversion. System behavior for Anchor Timecode varies depending on your setting for Source (TimecodeSource).

* If 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 Source (TimecodeSource) is set to Start at 0 (ZEROBASED) the first frame is 00:00:00:00. * If 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}$(

**Properties**

**source**

Use **Source** (TimecodeSource) to set how timecodes are handled within this job. 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.

**Type:** TimecodeSource (p. 424)

**Required:** False

**start**

Only use when you set **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}$(

**timestampOffset**

Only applies to outputs that support program-date-time stamp. Use **Timestamp offset (TimestampOffset)** to overwrite the timecode date without affecting the time and frame number. Provide the new date as a string in the format "yyyy-mm-dd". To use Time stamp offset, you must also enable Insert program-date-time (InsertProgramDateTime) in the output settings. For example, if the date part of your timecodes is 2002-1-25 and you want to change it to one year later, set Timestamp offset (TimestampOffset) to 2003-1-25.

**Type:** string

**Required:** False

**Pattern:** ^([0-9]{4})-(0[1-9]|1[0-2])-(0[1-9]|12)[0-9]|3[01])$

**TimecodeSource**

Use **Source** (TimecodeSource) to set how timecodes are handled within this job. 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**

Applies only to HLS outputs. Use this setting to specify whether the service inserts the ID3 timed metadata from the input in this output.

- PASSTHROUGH
- NONE

**TimedMetadataInsertion**

Enable Timed metadata insertion (TimedMetadataInsertion) to include ID3 tags in your job. To include timed metadata, you must enable it here, enable it in each output container, and specify tags and timecodes in ID3 insertion (Id3Insertion) objects.

**id3Insertions**

Id3Insertions contains the array of Id3Insertion instances.

- **Type**: Array of type Id3Insertion (p. 366)
- **Required**: False

**TrackSourceSettings**

Settings specific to caption sources that are specified by track number. Sources include IMSC in IMF.

**trackNumber**

Use this setting to select a single captions track from a source. Track numbers correspond to the order in the captions source file. For IMF sources, track numbering is based on the order that the captions appear in the CPL. For example, use 1 to select the captions asset that is listed first in the CPL. To include more than one captions track in your job outputs, create multiple input captions selectors. Specify one track per selector.

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

**TtmlDestinationSettings**

Settings specific to TTML caption outputs, including Pass style information (TtmlStylePassthrough).

**stylePassthrough**

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.

- **Type**: TtmlStylePassthrough (p. 425)
- **Required**: False

**TtmlStylePassthrough**

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.
UpdateJobTemplateRequest

Modify a job template by sending a request with the job template name and any of the following that you wish to change: description, category, and queue.

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

name

The name of the job template you are modifying

Type: string
Required: False

settings

JobTemplateSettings contains all the transcode settings saved in the template that will be applied to jobs created from it.

Type: JobTemplateSettings (p. 375)
Required: False

accelerationSettings

Accelerated transcoding can significantly speed up jobs with long, visually complex content. Outputs that use this feature incur pro-tier pricing. For information about feature limitations, see the AWS Elemental MediaConvert User Guide.
**Properties**

**Type**: AccelerationSettings (p. 283)

**Required**: False

**statusUpdateInterval**

Specify how often MediaConvert sends STATUS_UPDATE events to Amazon CloudWatch Events. Set the interval, in seconds, between status updates. MediaConvert sends an update at this interval from the time the service begins processing your job to the time it completes the transcode or encounters an error.

**Type**: StatusUpdateInterval (p. 421)

**Required**: False

**UpdateJobTemplateResponse**

Successful update job template requests will return the new job template JSON.

**jobTemplate**

A job template is a pre-made set of encoding instructions that you can use to quickly create a job.

**Type**: JobTemplate (p. 373)

**Required**: False

**VideoCodec**

Type of video codec

- FRAME_CAPTURE
- H_264
- H_265
- MPEG2
- PRORES

**VideoCodecSettings**

Video codec settings, (CodecSettings) under (VideoDescription), contains the group of settings related to video encoding. The settings in this group vary depending on the value you choose for Video codec (Codec). For each codec enum you choose, define the corresponding settings object. The following lists the codec enum, settings object pairs. * H_264, H264Settings * H_265, H265Settings * MPEG2, Mpeg2Settings * PRORES, ProresSettings * FRAME_CAPTURE, FrameCaptureSettings

**codec**

Specifies the video codec. This must be equal to one of the enum values defined by the object VideoCodec.

**Type**: VideoCodec (p. 427)

**Required**: False

**frameCaptureSettings**

Required when you set (Codec) under (VideoDescription)> (CodecSettings) to the value FRAME_CAPTURE.
Properties

**Type**: FrameCaptureSettings (p. 330)
**Required**: False

**h264Settings**
Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value H_264.
**Type**: H264Settings (p. 335)
**Required**: False

**h265Settings**
Settings for H265 codec
**Type**: H265Settings (p. 346)
**Required**: False

**mpeg2Settings**
Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value MPEG2.
**Type**: Mpeg2Settings (p. 400)
**Required**: False

**proresSettings**
Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value PRORES.
**Type**: ProresSettings (p. 414)
**Required**: False

**VideoDescription**
Settings for video outputs

**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

**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
Properties

Maximum: 4096

**scalingBehavior**

Applies only if your input aspect ratio is different from your output aspect ratio. Choose "Stretch to output" to have the service stretch your video image to fit. Keep the setting "Default" to allow the service to letterbox your video instead. This setting overrides any positioning value you specify elsewhere in the job.

*Type: ScalingBehavior (p. 419)*
*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. If you crop your input frame size to smaller than your output frame size, make sure to specify the behavior you want in your output setting "Scaling behavior".

*Type: Rectangle (p. 417)*
*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*

**videoPreprocessors**

Find additional transcoding features under Preprocessors (VideoPreprocessors). Enable the features at each output individually. These features are disabled by default.

*Type: VideoPreprocessor (p. 431)*
*Required: False*

**timecodeInsertion**

Applies only to H.264, H.265, MPEG2, and ProRes outputs. Only enable Timecode insertion when the input frame rate is identical to the output frame rate. To include timecodes in this output, set Timecode insertion (VideoTimecodeInsertion) to PIC_TIMING_SEI. To leave them out, set it to DISABLED. Default is DISABLED. When the service inserts timecodes in an output, by default, it uses any embedded timecodes from the input. If none are present, the service will set the timecode for the first output frame to zero. To change this default behavior, adjust the settings under Timecode configuration (TimecodeConfig). In the console, these settings are located under Job > Job settings > Timecode configuration. Note - Timecode source under input settings (InputTimecodeSource) does not affect the timecodes that are inserted in the output. Source under Job settings > Timecode configuration (TimecodeSource) does.

*Type: VideoTimecodeInsertion (p. 433)*
*Required: False*
antiAlias

The anti-alias filter is automatically applied to all outputs. The service no longer accepts the value DISABLED for AntiAlias. If you specify that in your job, the service will ignore the setting.

  Type: AntiAlias (p. 284)
  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. 417)
  Required: False

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. 0 is the softest setting, 100 the sharpest, and 50 recommended for most content.

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

codecSettings

Video codec settings, (CodecSettings) under (VideoDescription), contains the group of settings related to video encoding. The settings in this group vary depending on the value you choose for Video codec (Codec). For each codec enum you choose, define the corresponding settings object. The following lists the codec enum, settings object pairs.

* H_264, H264Settings
* H_265, H265Settings
* MPEG2, Mpeg2Settings
* PRORES, ProresSettings
* FRAME_CAPTURE, FrameCaptureSettings

  Type: VideoCodecSettings (p. 427)
  Required: False

afdSignaling

This setting only applies to H.264, H.265, and MPEG2 outputs. Use Insert AFD signaling (AfdSignaling) to specify whether the service includes 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.

  Type: AfdSignaling (p. 283)
  Required: False

dropFrameTimecode

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.
**Properties**

**Type**: DropFrameTimecode (p. 314)  
**Required**: False

**respondToAfd**

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.

**Type**: RespondToAfd (p. 418)  
**Required**: False

**colorMetadata**

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

**Type**: ColorMetadata (p. 307)  
**Required**: False

**VideoPreprocessor**

Find additional transcoding features under Preprocessors (VideoPreprocessors). Enable the features at each output individually. These features are disabled by default.

**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. 306)  
**Required**: False

**deinterlacer**

Use Deinterlacer (Deinterlacer) to produce smoother motion and a clearer picture.

**Type**: Deinterlacer (p. 312)  
**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. 367)  
**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.
Properties

Type: NoiseReducer (p. 408)
Required: False

timecodeBurnin

Timecode burn-in (TimecodeBurnIn)--Burns the output timecode and specified prefix into the output.

Type: TimecodeBurnin (p. 422)
Required: False

VideoSelector

Selector for video.

colorSpace

If your input video has accurate color space metadata, or if you don't know about color space, leave this set to the default value FOLLOW. The service will automatically detect your input color space. If your input video has metadata indicating the wrong color space, or if your input video is missing color space metadata that should be there, specify the accurate color space here. If you choose HDR10, you can also correct inaccurate color space coefficients, using the HDR master display information controls. You must also set Color space usage (ColorSpaceUsage) to FORCE for the service to use these values.

Type: ColorSpace (p. 308)
Required: False

rotate

Use Rotate (InputRotate) to specify how the service rotates your video. You can choose automatic rotation or specify a rotation. You can specify a clockwise rotation of 0, 90, 180, or 270 degrees. If your input video container is .mov or .mp4 and your input has rotation metadata, you can choose Automatic to have the service rotate your video according to the rotation specified in the metadata. The rotation must be within one degree of 90, 180, or 270 degrees. If the rotation metadata specifies any other rotation, the service will default to no rotation. By default, the service does no rotation, even if your input video has rotation metadata. The service doesn't pass through rotation metadata.

Type: InputRotate (p. 368)
Required: False

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

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

colorSpaceUsage

There are two sources for color metadata, the input file and the job configuration (in the Color space and HDR master display informaiton settings). The Color space usage setting controls which takes precedence. *FORCE:* The 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:* The 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.

Type: ColorSpaceUsage (p. 308)
Required: False

hdr10Metadata

Use the "HDR master display information" (Hdr10Metadata) settings to correct HDR metadata or to provide missing metadata. These values vary depending on the input video and must be provided by a color grader. Range is 0 to 50,000; each increment represents 0.00002 in CIE1931 color coordinate. Note that these settings are not color correction. Note that if you are creating HDR outputs inside of an HLS CMAF package, to comply with the Apple specification, you must use the following settings. Set "MP4 packaging type" (writeMp4PackagingType) to HVC1 (HVC1). Set "Profile" (H265Settings > codecProfile) to Main10/High (MAIN10_HIGH). Set "Level" (H265Settings > codecLevel) to 5 (LEVEL_5).

Type: Hdr10Metadata (p. 354)
Required: False

VideoTimecodeInsertion

Applies only to H.264, H.265, MPEG2, and ProRes outputs. Only enable Timecode insertion when the input frame rate is identical to the output frame rate. To include timecodes in this output, set Timecode insertion (VideoTimecodeInsertion) to PIC_TIMING_SEI. To leave them out, set it to DISABLED. Default is DISABLED. When the service inserts timecodes in an output, by default, it uses any embedded timecodes from the input. If none are present, the service will set the timecode for the first output frame to zero. To change this default behavior, adjust the settings under Timecode configuration (TimecodeConfig). In the console, these settings are located under Job > Job settings > Timecode configuration. Note - Timecode source under input settings (InputTimecodeSource) does not affect the timecodes that are inserted in the output. Source under Job settings > Timecode configuration (TimecodeSource) does.

DISABLED
PIC_TIMING_SEI

WavFormat

The service defaults to using RIFF for WAV outputs. If your output audio is likely to exceed 4 GB in file size, or if you otherwise need the extended support of the RF64 format, set your output WAV file format to RF64.

RIFF
RF64
**WavSettings**

Required when you set (Codec) under (AudioDescriptions)>({CodecSettings}) to the value WAV.

**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

**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

**sampleRate**

Sample rate in Hz.

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

**format**

The service defaults to using RIFF for WAV outputs. If your output audio is likely to exceed 4 GB in file size, or if you otherwise need the extended support of the RF64 format, set your output WAV file format to RF64.

- **Type**: WavFormat (p. 433)
- **Required**: False

**See Also**

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

**GetJobTemplate**

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Go - Pilot
- AWS SDK for Java
UpdateJobTemplate

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Go - Pilot
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

DeleteJobTemplate

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Go - Pilot
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

Jobs

URI

/2017-08-29/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. 449)</td>
<td>200 response</td>
</tr>
<tr>
<td>400</td>
<td>ExceptionBody (p. 475)</td>
<td>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>403</td>
<td>ExceptionBody (p. 475)</td>
<td>You don't have permissions for this action with the credentials you sent.</td>
</tr>
<tr>
<td>404</td>
<td>ExceptionBody (p. 475)</td>
<td>The resource you requested does not exist.</td>
</tr>
<tr>
<td>409</td>
<td>ExceptionBody (p. 475)</td>
<td>The service could not complete your request because there is a conflict with the current state of the resource.</td>
</tr>
<tr>
<td>429</td>
<td>ExceptionBody (p. 475)</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>
<tr>
<td>500</td>
<td>ExceptionBody (p. 475)</td>
<td>The service encountered an unexpected condition and cannot fulfill your request.</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. 462)</td>
<td>201 response</td>
</tr>
</tbody>
</table>
### Status Code | Response Model | Description
--- | --- | ---
400 | ExceptionBody (p. 475) | The service can't process your request because of a problem in the request. Please check your request form and syntax.
403 | ExceptionBody (p. 475) | You don't have permissions for this action with the credentials you sent.
404 | ExceptionBody (p. 475) | The resource you requested does not exist.
409 | ExceptionBody (p. 475) | The service could not complete your request because there is a conflict with the current state of the resource.
429 | ExceptionBody (p. 475) | Too many requests have been sent in too short of a time. The service limits the rate at which it will accept requests.
500 | ExceptionBody (p. 475) | The service encountered an unexpected condition and cannot fulfill your request.

### Schemas

#### Request Bodies

**Example GET**

```json
{
    "queue": "string",
    "status": enum,
    "order": enum,
    "nextToken": "string",
    "maxResults": integer
}
```

**Example POST**

```json
{
    "clientRequestToken": "string",
    "jobTemplate": "string",
    "queue": "string",
    "role": "string",
    "settings": {
        "timecodeConfig": {
            "anchor": "string",
            "source": enum,
            "start": "string",
            "timestampOffset": "string"
        },
        "outputGroups": [
```

---

437
"customName": "string",
"name": "string",
"outputs": [
  {
    "containerSettings": {
      "container": enum,
      "m3u8Settings": {
        "audioFramesPerPes": integer,
        "pcrControl": enum,
        "pcrPid": integer,
        "privateMetadataPid": integer,
        "programNumber": integer,
        "patInterval": integer,
        "pmtInterval": integer,
        "scte35Source": enum,
        "scte35Pid": integer,
        "nielsenId3": enum,
        "timedMetadata": enum,
        "timedMetadataPid": integer,
        "transportStreamId": integer,
        "videoPid": integer,
        "audioPids": [ integer
      ]
    },
    "f4vSettings": {
      "moovPlacement": enum
    },
    "m2tsSettings": {
      "audioBufferModel": enum,
      "minEbpInterval": integer,
      "esRateInPes": enum,
      "patInterval": integer,
      "dvbNitSettings": {
        "nitInterval": integer,
        "networkId": integer,
        "networkName": "string"
      },
      "dvbSdtSettings": {
        "outputSdt": enum,
        "sdtInterval": integer,
        "serviceName": "string",
        "serviceProviderName": "string"
      },
      "scte35Source": enum,
      "scte35Pid": integer,
      "scte35Esam": {
        "scte35EsamPid": integer
      },
      "videoPid": integer,
      "dvbTdtSettings": {
        "tdtInterval": integer
      },
      "pmtInterval": integer,
      "segmentationStyle": enum,
      "segmentationTime": number,
      "pmtPid": integer,
      "bitrate": integer,
      "audioPids": [ integer
    },
    "privateMetadataPid": integer,
    "nielsenId3": enum,
    "timedMetadataPid": integer,
    "maxPcrInterval": integer,
  },
"transportStreamId": integer,
"dvbSubPids": [ integer
],
"rateMode": enum,
"audioFramesPerPes": integer,
"pcrControl": enum,
"segmentationMarkers": enum,
"ebpAudioInterval": enum,
"forceTsVideoEbpOrder": enum,
"programNumber": integer,
"pcrPid": integer,
"bufferModel": enum,
"dvbTeletextPid": integer,
"fragmentTime": number,
"ebpPlacement": enum,
"nullPacketBitrate": number
],
"movSettings": { "clapAtom": enum,
"cslgAtom": enum,
"paddingControl": enum,
"reference": enum,
"mpeg2FourCCControl": enum
},
"mp4Settings": { "cslgAtom": enum,
"freeSpaceBox": enum,
"mp4MajorBrand": "string",
"moovPlacement": enum
},
"preset": "string",
"videoDescription": { "fixedAfd": integer,
"width": integer,
"scalingBehavior": enum,
"crop": { "height": integer,
"width": integer,
"x": integer,
"y": integer
},
"height": integer,
"videoPreprocessors": { "colorCorrector": { "brightness": integer,
"colorSpaceConversion": enum,
"contrast": integer,
"hue": integer,
"saturation": integer,
"hdr10Metadata": { "redPrimaryX": integer,
"redPrimaryY": integer,
"greenPrimaryX": integer,
"greenPrimaryY": integer,
"bluePrimaryX": integer,
"bluePrimaryY": integer,
"whitePointX": integer,
"whitePointY": integer,
"maxFrameAverageLightLevel": integer,
"maxContentLightLevel": integer,
"maxLuminance": integer,
"minLuminance": integer
} }},
"deinterlacer": {
   "algorithm": enum,
   "mode": enum,
   "control": enum
},
"imageInserter": {
   "insertableImages": [
      {
         "width": integer,
         "height": integer,
         "imageX": integer,
         "imageY": integer,
         "duration": integer,
         "fadeIn": integer,
         "layer": integer,
         "imageInserterInput": "string",
         "startTime": "string",
         "fadeOut": integer,
         "opacity": integer
      }
   ]
},
"noiseReducer": {
   "filter": enum,
   "filterSettings": {
      "strength": integer
   },
   "spatialFilterSettings": {
      "strength": integer,
      "speed": integer,
      "postFilterSharpenStrength": integer
   }
},
"timecodeBurnin": {
   "fontSize": integer,
   "position": enum,
   "prefix": "string"
},
"timecodeInsertion": enum,
"antiAlias": enum,
"position": {
   "height": integer,
   "width": integer,
   "x": integer,
   "y": integer
},
"sharpness": integer,
"codecSettings": {
   "codec": enum,
   "frameCaptureSettings": {
      "framerateNumerator": integer,
      "framerateDenominator": integer,
      "maxCaptures": integer,
      "quality": integer
   },
   "h264Settings": {
      "interlaceMode": enum,
      "parNumerator": integer,
      "numberReferenceFrames": integer,
      "syntax": enum,
      "softness": integer,
      "framerateDenominator": integer,
      "gopClosedCadence": integer,
      "hrdBufferInitialFillPercentage": integer,
      "gopSize": number,
"slices": integer,
"gopBReference": enum,
"hrdBufferSize": integer,
"maxBitrate": integer,
"slowPal": enum,
"parDenominator": integer,
"spatialAdaptiveQuantization": enum,
"temporalAdaptiveQuantization": enum,
"flickerAdaptiveQuantization": enum,
"entropyEncoding": enum,
"bitrate": integer,
"framerateControl": enum,
"rateControlMode": enum,
"qvbrSettings": {
  "qvbrQualityLevel": integer,
  "maxAverageBitrate": integer
},
"codecProfile": enum,
"telecine": enum,
"framerateNumerator": integer,
"minIInterval": integer,
"adaptiveQuantization": enum,
"codecLevel": enum,
"fieldEncoding": enum,
"sceneChangeDetect": enum,
"qualityTuningLevel": enum,
"framerateConversionAlgorithm": enum,
"unregisteredSeiTimecode": enum,
"hrdBufferSize": integer,
"maxBitrate": integer,
"framerateControl": enum,
"rateControlMode": enum,
"qvbrSettings": {
  "qvbrQualityLevel": integer,
  "maxAverageBitrate": integer
},

"h265Settings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "numberReferenceFrames": integer,
  "framerateDenominator": integer,
  "gopClosedCadence": integer,
  "alternateTransferFunctionSei": enum,
  "hrdBufferInitialFillPercentage": integer,
  "gopSize": number,
  "slices": integer,
  "gopBReference": enum,
  "hrdBufferSize": integer,
  "maxBitrate": integer,
  "slowPal": enum,
  "parDenominator": integer,
  "spatialAdaptiveQuantization": enum,
  "temporalAdaptiveQuantization": enum,
  "flickerAdaptiveQuantization": enum,
  "bitrate": integer,
  "framerateControl": enum,
  "rateControlMode": enum,
  "qvbrSettings": {
    "qvbrQualityLevel": integer,
    "maxAverageBitrate": integer
  }
},

"codecProfile": enum,
"tiles": enum,
"telecine": enum,
"framerateNumerator": integer,
"minIInterval": integer,
"adaptiveQuantization": enum,
"codecLevel": enum,
"sceneChangeDetect": enum,
"qualityTuningLevel": enum,
"framerateConversionAlgorithm": enum,
"unregisteredSeiTimecode": enum,
"gopSizeUnits": enum,
"parControl": enum,
"numberBFramesBetweenReferenceFrames": integer,
"temporalIds": enum,
"sampleAdaptiveOffsetFilterMode": enum,
"writeMp4PackagingType": enum,
"dynamicSubGop": enum
},
"mpeg2Settings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "syntax": enum,
  "softness": integer,
  "framerateDenominator": integer,
  "gopClosedCadence": integer,
  "hrdBufferInitialFillPercentage": integer,
  "gopSize": number,
  "hrdBufferSize": integer,
  "maxBitrate": integer,
  "slowPal": enum,
  "parDenominator": integer,
  "spatialAdaptiveQuantization": enum,
  "temporalAdaptiveQuantization": enum,
  "bitrate": integer,
  "intraDcPrecision": enum,
  "framerateControl": enum,
  "rateControlMode": enum,
  "codecProfile": enum,
  "telecine": enum,
  "framerateNumerator": integer,
  "minIInterval": integer,
  "adaptiveQuantization": enum,
  "codecLevel": enum,
  "sceneChangeDetect": enum,
  "qualityTuningLevel": enum,
  "framerateConversionAlgorithm": enum,
  "gopSizeUnits": enum,
  "parControl": enum,
  "numberBFramesBetweenReferenceFrames": integer,
  "dynamicSubGop": enum
},
"proresSettings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "parClosedCadence": integer,
  "codecProfile": enum,
  "slowPal": enum,
  "parDenominator": integer,
  "framerateControl": enum,
  "telecine": enum,
  "framerateNumerator": integer,
  "framerateConversionAlgorithm": enum,
  "parControl": enum
}
},
"afdsSignaling": enum,
"dropFrameTimecode": enum,
"respondToAfds": enum,
"colorMetadata": enum
},
"audioDescriptions": [
}
"audioTypeControl": enum,
"audioSourceName": "string",
"audioNormalizationSettings": {
  "algorithm": enum,
  "algorithmControl": enum,
  "correctionGateLevel": integer,
  "loudnessLogging": enum,
  "targetLkfs": number,
  "peakCalculation": enum
},
"codecSettings": {
  "codec": enum,
  "aacSettings": {
    "audioDescriptionBroadcasterMix": enum,
    "vbrQuality": enum,
    "bitrate": integer,
    "rateControlMode": enum,
    "codecProfile": enum,
    "codingMode": enum,
    "rawFormat": enum,
    "sampleRate": integer,
    "specification": enum
  },
  "ac3Settings": {
    "bitrate": integer,
    "bitstreamMode": enum,
    "codingMode": enum,
    "dialnorm": integer,
    "dynamicRangeCompressionProfile": enum,
    "metadataControl": enum,
    "lfeFilter": enum,
    "sampleRate": integer
  },
  "aiffSettings": {
    "bitDepth": integer,
    "channels": integer,
    "sampleRate": integer
  },
  "eac3Settings": {
    "metadataControl": enum,
    "surroundExMode": enum,
    "loRoSurroundMixLevel": number,
    "phaseControl": enum,
    "dialnorm": integer,
    "ltRtSurroundMixLevel": number,
    "bitrate": integer,
    "ltRtCenterMixLevel": number,
    "passthroughControl": enum,
    "lfeControl": enum,
    "loRoCenterMixLevel": number,
    "attenuationControl": enum,
    "codingMode": enum,
    "surroundMode": enum,
    "bitstreamMode": enum,
    "lfeFilter": enum,
    "stereoDownmix": enum,
    "dynamicRangeCompressionRf": enum,
    "sampleRate": integer,
    "dynamicRangeCompressionLine": enum,
    "dcFilter": enum
  },
  "mp2Settings": {
    "bitrate": integer,
    "channels": integer,
    "sampleRate": integer
  }
},
"wavSettings": {
  "bitDepth": integer,
  "channels": integer,
  "sampleRate": integer,
  "format": enum
},

"remixSettings": {
  "channelMapping": {
    "outputChannels": [
      {
        "inputChannels": [integer
      ]
    },
    "channelsIn": integer,
    "channelsOut": integer
  },
  "streamName": "string",
  "languageCodeControl": enum,
  "audioType": integer,
  "customLanguageCode": "string",
  "languageCode": enum
},

"outputSettings": {
  "hlsSettings": {
    "audioGroupId": "string",
    "audioRenditionSets": "string",
    "audioTrackType": enum,
    "iFrameOnlyManifest": enum,
    "segmentModifier": "string"
  }
},

"extension": "string",

"nameModifier": "string",

"captionDescriptions": [
  {
    "captionSelectorName": "string",
    "destinationSettings": {
      "destinationType": enum,
      "burninDestinationSettings": {
        "backgroundOpacity": integer,
        "shadowXOffset": integer,
        "teletextSpacing": enum,
        "alignment": enum,
        "outlineSize": integer,
        "yPosition": integer,
        "shadowColor": enum,
        "fontOpacity": integer,
        "fontSize": integer,
        "fontScript": enum,
        "fontColor": enum,
        "backgroundColor": enum,
        "fontResolution": integer,
        "outlineColor": enum,
        "shadowYOffset": integer,
        "xPosition": integer,
        "shadowOpacity": integer
      },
      "dvbSubDestinationSettings": {
        "backgroundOpacity": integer,
        "shadowXOffset": integer,
        "teletextSpacing": enum
      }
    }
  }
]
"alignment": enum,
"outlineSize": integer,
"yPosition": integer,
"shadowColor": enum,
"fontOpacity": integer,
"fontSize": integer,
"fontScript": enum,
"foregroundColor": enum,
"backgroundColor": enum,
"fontResolution": integer,
"outlineColor": enum,
"shadowYOffset": integer,
"xPosition": integer,
"shadowOpacity": integer
},
"sccDestinationSettings": {
  "framerate": enum
},
"teletextDestinationSettings": {
  "pageNumber": "string"
},
"ttmlDestinationSettings": {
  "stylePassthrough": enum
},
"embeddedDestinationSettings": {
  "destination608ChannelNumber": integer
},
"customLanguageCode": "string",
"languageCode": enum,
"languageDescription": "string"
]
],
"outputGroupSettings": {
  "type": enum,
  "hlsGroupSettings": {
    "manifestDurationFormat": enum,
    "segmentLength": integer,
    "timedMetadata1d3Period": integer,
    "captionLanguageSetting": enum,
    "captionLanguageMappings": [
      {
        "captionChannel": integer,
        "customLanguageCode": "string",
        "languageCode": enum,
        "languageDescription": "string"
      }
    ],
    "destination": "string",
    "destinationSettings": {
      "s3Settings": {
        "encryption": {
          "encryptionType": enum,
          "kmsKeyArn": "string"
        }
      }
    }
  }
},
"encryption": {
  "encryptionMethod": enum,
  "constantInitializationVector": "string",
  "initializationVectorInManifest": enum,
  "offlineEncrypted": enum,
  "spekeKeyProvider": {
    "resourceId": "string",
  }
}
AWS Elemental MediaConvert API Reference API Reference
Schemas
"systemIds": [
"string"
],
"url": "string",
"certificateArn": "string"

},
"staticKeyProvider": {
"staticKeyValue": "string",
"keyFormat": "string",
"keyFormatVersions": "string",
"url": "string"
},
"type": enum

},
"timedMetadataId3Frame": enum,
"baseUrl": "string",
"codecSpecification": enum,
"outputSelection": enum,
"programDateTimePeriod": integer,
"segmentsPerSubdirectory": integer,
"minSegmentLength": integer,
"minFinalSegmentLength": number,
"directoryStructure": enum,
"programDateTime": enum,
"adMarkers": [
enum
],
"segmentControl": enum,
"timestampDeltaMilliseconds": integer,
"manifestCompression": enum,
"clientCache": enum,
"streamInfResolution": enum

},
"dashIsoGroupSettings": {
"segmentLength": integer,
"destination": "string",
"destinationSettings": {
"s3Settings": {
"encryption": {
"encryptionType": enum,
"kmsKeyArn": "string"
}
}
},
"encryption": {
"spekeKeyProvider": {
"resourceId": "string",
"systemIds": [
"string"
],
"url": "string",
"certificateArn": "string"
}
},
"minBufferTime": integer,
"fragmentLength": integer,
"baseUrl": "string",
"segmentControl": enum,
"hbbtvCompliance": enum,
"writeSegmentTimelineInRepresentation": enum
},
"fileGroupSettings": {
"destination": "string",
"destinationSettings": {
"s3Settings": {
"encryption": {

446


"encryptionType": enum,
"kmsKeyArn": "string"
}
}
},
"msSmoothGroupSettings": {
"destination": "string",
"destinationSettings": {
"s3Settings": {
"encryption": {
"encryptionType": enum,
"kmsKeyArn": "string"
}
}
},
"fragmentLength": integer,
"encryption": {
"spekeKeyProvider": {
"resourceId": "string",
"systemIds": [
"string"
],
"url": "string",
"certificateArn": "string"
}
}
},
"manifestEncoding": enum,
"audioDeduplication": enum
},
"cmafGroupSettings": {
"writeHlsManifest": enum,
"writeDashManifest": enum,
"segmentLength": integer,
"minFinalSegmentLength": number,
"destination": "string",
"destinationSettings": {
"s3Settings": {
"encryption": {
"encryptionType": enum,
"kmsKeyArn": "string"
}
}
},
"encryption": {
"encryptionMethod": enum,
"constantInitializationVector": "string",
"initializationVectorInManifest": enum,
"staticKeyProvider": {
"staticKeyValue": "string",
"keyFormat": "string",
"keyFormatVersions": "string",
"url": "string"
}
},
"type": enum
},
"minBufferTime": integer,
"fragmentLength": integer,
"baseUrl": "string",
"segmentControl": enum,
"manifestDurationFormat": enum,
"streamInfResolution": enum,
"clientCache": enum,
"manifestCompression": enum,
"codecSpecification": enum
}
"adAvailOffset": integer,
"availBlanking": {
  "availBlankingImage": "string"
},
"timedMetadataInsertion": {
  "id3Insertions": [
    {
      "timecode": "string",
      "id3": "string"
    }
  ],
  "nielsenConfiguration": {
    "breakoutCode": integer,
    "distributorId": "string"
  },
  "motionImageInserter": {
    "insertionMode": enum,
    "input": "string",
    "offset": {
      "imageX": integer,
      "imageY": integer
    },
    "startTime": "string",
    "playback": enum,
    "framerate": {
      "framerateNumerator": integer,
      "framerateDenominator": integer
    }
  },
  "esam": {
    "signalProcessingNotification": {
      "sccXml": "string"
    },
    "manifestConfirmConditionNotification": {
      "mccXml": "string"
    },
    "responseSignalPreroll": integer
  },
  "inputs": [
  {
    "inputClippings": [
      {
        "endTimecode": "string",
        "startTimecode": "string"
      }
    ],
    "audioSelectors": {
    },
    "audioSelectorGroups": {
    },
    "programNumber": integer,
    "videoSelector": {
      "colorSpace": enum,
      "rotate": enum,
      "pid": integer,
      "programNumber": integer,
      "colorSpaceUsage": enum,
      "hdr10Metadata": {
        "redPrimaryX": integer,
        "redPrimaryY": integer,
        "greenPrimaryX": integer,
        "greenPrimaryY": integer,
"bluePrimaryX": integer,
"bluePrimaryY": integer,
"whitePointX": integer,
"whitePointY": integer,
"maxFrameAverageLightLevel": integer,
"maxContentLightLevel": integer,
"maxLuminance": integer,
"minLuminance": integer
},
"filterEnable": enum,
"psiControl": enum,
"deblockStrength": integer,
"denoiseFilter": enum,
"timecodeSource": enum,
"captionSelectors": {}},
"imageInserter": {
  "insertableImages": [  
    {
      "width": integer,
      "height": integer,
      "imageX": integer,
      "imageY": integer,
      "duration": integer,
      "fadeIn": integer,
      "layer": integer,
      "imageInserterInput": "string",
      "startTime": "string",
      "fadeOut": integer,
      "opacity": integer
      
      },
      "fileInput": "string",
      "decryptionSettings": {
        "decryptionMode": enum,
        "encryptedDecryptionKey": "string",
        "initializationVector": "string",
        "kmsKeyRegion": "string"
      }
    },
    "supplementalImps": [  "string"
  ]
]},
"userMetadata": {}},
"billingTagsSource": enum,
"accelerationSettings": {
  "mode": enum
  
},
"statusUpdateInterval": enum
]}

Response Bodies

Example ListJobsResponse

{
  "jobs": [
  
  ]
}
"arn": "string",
"id": "string",
"createdAt": "string",
"jobTemplate": "string",
"queue": "string",
"userMetadata": {
},
"role": "string",
"settings": {
  "timecodeConfig": {
    "anchor": "string",
    "source": enum,
    "start": "string",
    "timestampOffset": "string"
  },
  "outputGroups": [
    {
      "customName": "string",
      "name": "string",
      "outputs": [
        {
          "containerSettings": {
            "container": enum,
            "m3u8Settings": {
              "audioFramesPerPes": integer,
              "pcrControl": enum,
              "pcrPid": integer,
              "pmtpid": integer,
              "privateMetadataPid": integer,
              "programNumber": integer,
              "patInterval": integer,
              "pmtInterval": integer,
              "scte35Source": enum,
              "scte35Pid": integer,
              "nielsenId3": enum,
              "timedMetadata": enum,
              "timedMetadataPid": integer,
              "transportStreamId": integer,
              "videoPid": integer,
              "audioPids": [
                integer
              ]
            },
            "f4vSettings": {
              "moovPlacement": enum
            },
            "m2tsSettings": {
              "audioBufferModel": enum,
              "minEbpInterval": integer,
              "esRateInPes": enum,
              "patInterval": integer,
              "dvbNitSettings": {
                "nitInterval": integer,
                "networkId": integer,
                "networkName": "string"
              },
              "dvbSdtSettings": {
                "outputSdt": enum,
                "sdtInterval": integer,
                "serviceName": "string",
                "serviceProviderName": "string"
              },
              "scte35Source": enum,
              "scte35Pid": integer,
              "scte35Esam": {
                "scte35EsamPid": integer
              }
            }
          }
        }
      ]
    }
  ]
}
},
"videoPid": integer,
"dvbTdtSettings": {
    "tdtInterval": integer
},
"pmtInterval": integer,
"segmentationStyle": enum,
"segmentationTime": number,
"pmtPid": integer,
"bitrate": integer,
"audioPids": [
    integer
],
"privateMetadataPid": integer,
"nielsenId3": enum,
"timedMetadataPid": integer,
"maxPcrInterval": integer,
"transportStreamId": integer,
"dvbSubPids": [
    integer
],
"rateMode": enum,
"audioFramesPerPes": integer,
"pcrControl": enum,
"segmentationMarkers": enum,
"ebpAudioInterval": enum,
"forceTuVideoEbpOrder": enum,
"programNumber": integer,
"pcrPid": integer,
"bufferModel": enum,
"dvbTeletextPid": integer,
"fragmentTime": number,
"ebpPlacement": enum,
"nullPacketBitrate": number
},
"movSettings": {
    "clapAtom": enum,
    "cslgAtom": enum,
    "paddingControl": enum,
    "reference": enum,
    "mpeg2FourCCControl": enum
},
"mp4Settings": {
    "cslgAtom": enum,
    "freeSpaceBox": enum,
    "mp4MajorBrand": "string",
    "moovPlacement": enum
}
},
"preset": "string",
"videoDescription": {
    "fixedAfd": integer,
    "width": integer,
    "scalingBehavior": enum,
    "crop": {
        "height": integer,
        "width": integer,
        "x": integer,
        "y": integer
    },
    "height": integer,
    "videoPreprocessors": {
        "colorCorrector": {
            "brightness": integer,
            "colorSpaceConversion": enum,
            "contrast": integer,
"hue": integer,
"saturation": integer,
"hdr10Metadata": {
    "redPrimaryX": integer,
    "redPrimaryY": integer,
    "greenPrimaryX": integer,
    "greenPrimaryY": integer,
    "bluePrimaryX": integer,
    "bluePrimaryY": integer,
    "whitePointX": integer,
    "whitePointY": integer,
    "maxFrameAverageLightLevel": integer,
    "maxContentLightLevel": integer,
    "maxLuminance": integer,
    "minLuminance": integer
},
"deinterlacer": {
    "algorithm": enum,
    "mode": enum,
    "control": enum
},
"imageInserter": {
    "insertableImages": [
    {
        "width": integer,
        "height": integer,
        "imageX": integer,
        "imageY": integer,
        "duration": integer,
        "fadeIn": integer,
        "layer": integer,
        "imageInserterInput": "string",
        "startTime": "string",
        "fadeOut": integer,
        "opacity": integer
    }
    ]
},
"noiseReducer": {
    "filter": enum,
    "filterSettings": {
        "strength": integer
    },
    "spatialFilterSettings": {
        "strength": integer,
        "speed": integer,
        "postFilterSharpenStrength": integer
    },
},
"timecodeBurnin": {
    "fontSize": integer,
    "position": enum,
    "prefix": "string"
},
"timecodeInsertion": enum,
"antiAlias": enum,
"position": {
    "height": integer,
    "width": integer,
    "x": integer,
    "y": integer
},
"sharpness": integer,
"codecSettings": {
"codec": enum,
"frameCaptureSettings": {
  "framerateNumerator": integer,
  "framerateDenominator": integer,
  "maxCaptures": integer,
  "quality": integer
},
"h264Settings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "numberReferenceFrames": integer,
  "syntax": enum,
  "softness": integer,
  "framerateDenominator": integer,
  "gopClosedCadence": integer,
  "hrdBufferInitialFillPercentage": integer,
  "gopSize": number,
  "slices": integer,
  "gopBReference": enum,
  "hrdBufferSize": integer,
  "maxBitrate": integer,
  "slowPal": enum,
  "parDenominator": integer,
  "spatialAdaptiveQuantization": enum,
  "temporalAdaptiveQuantization": enum,
  "entropyEncoding": enum,
  "bitrate": integer,
  "framerateControl": enum,
  "rateControlMode": enum,
  "qvbrSettings": {
    "qvbrQualityLevel": integer,
    "maxAverageBitrate": integer
  },
  "codecProfile": enum,
  "telecine": enum,
  "framerateNumerator": integer,
  "minIInterval": integer,
  "adaptiveQuantization": enum,
  "codecLevel": enum,
  "fieldEncoding": enum,
  "sceneChangeDetect": enum,
  "qualityTuningLevel": enum,
  "framerateConversionAlgorithm": enum,
  "unregisteredSeiTimecode": enum,
  "gopSizeUnits": enum,
  "parControl": enum,
  "numberBFramesBetweenReferenceFrames": integer,
  "repeatPps": enum,
  "dynamicSubGop": enum
},
"h265Settings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "framerateNumerator": integer,
  "framerateDenominator": integer,
  "gopClosedCadence": integer,
  "alternateTransferFunctionSei": enum,
  "hrdBufferInitialFillPercentage": integer,
  "gopSize": number,
  "slices": integer,
  "gopBReference": enum,
  "hrdBufferSize": integer,
  "maxBitrate": integer,
  "slowPal": enum,
  "parDenominator": integer,
"spatialAdaptiveQuantization": enum,
"temporalAdaptiveQuantization": enum,
"flickerAdaptiveQuantization": enum,
"bitrate": integer,
"framerateControl": enum,
"rateControlMode": enum,
"qvbrSettings": {
  "qvbrQualityLevel": integer,
  "maxAverageBitrate": integer
},
"codecProfile": enum,
"tiles": enum,
"telecine": enum,
"framerateNumerator": integer,
"minIInterval": integer,
"adaptiveQuantization": enum,
"codecLevel": enum,
"sceneChangeDetect": enum,
"qualityTuningLevel": enum,
"framerateConversionAlgorithm": enum,
"unregisteredSeiTimecode": enum,
"gopSizeUnits": enum,
"parControl": enum,
"numberBFramesBetweenReferenceFrames": integer,
"temporalIds": enum,
"sampleAdaptiveOffsetFilterMode": enum,
"writeMp4PackagingType": enum,
"dynamicSubGop": enum
},
"mpeg2Settings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "syntax": enum,
  "softness": integer,
  "framerateDenominator": integer,
  "gopClosedCadence": integer,
  "hrdBufferInitialFillPercentage": integer,
  "gopSize": number,
  "hrdBufferSize": integer,
  "maxBitrate": integer,
  "slowPal": enum,
  "parDenominator": integer,
  "spatialAdaptiveQuantization": enum,
  "temporalAdaptiveQuantization": enum,
  "bitrate": integer,
  "intraDcPrecision": enum,
  "framerateControl": enum,
  "rateControlMode": enum,
  "codecProfile": enum,
  "telecine": enum,
  "framerateNumerator": integer,
  "minIInterval": integer,
  "adaptiveQuantization": enum,
  "codecLevel": enum,
  "sceneChangeDetect": enum,
  "qualityTuningLevel": enum,
  "framerateConversionAlgorithm": enum,
  "gopSizeUnits": enum,
  "parControl": enum,
  "numberBFramesBetweenReferenceFrames": integer,
  "dynamicSubGop": enum
},
"proresSettings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "framerateDenominator": integer,
"codecProfile": enum,
"slowPal": enum,
"parDenominator": integer,
"framerateControl": enum,
"telecine": enum,
"framerateNumerator": integer,
"framerateConversionAlgorithm": enum,
"parControl": enum
},
"afdSignaling": enum,
"dropFrameTimecode": enum,
"respondToAfd": enum,
"colorMetadata": enum
},
"audioDescriptions": [
{
"audioTypeControl": enum,
"audioSourceName": "string",
"audioNormalizationSettings": {
"algorithm": enum,
"algorithmControl": enum,
"correctionGateLevel": integer,
"loudnessLogging": enum,
"targetLkfs": number,
"peakCalculation": enum
},
"codecSettings": {
"codec": enum,
"aacSettings": {
"audioDescriptionBroadcasterMix": enum,
"vbrQuality": enum,
"bitrate": integer,
"rateControlMode": enum,
"codecProfile": enum,
"codingMode": enum,
"rawFormat": enum,
"sampleRate": integer,
"specification": enum
},
"ac3Settings": {
"bitrate": integer,
"bitstreamMode": enum,
"codingMode": enum,
"dialnorm": integer,
"dynamicRangeCompressionProfile": enum,
"metadataControl": enum,
"lfeFilter": enum,
"sampleRate": integer
},
"aiffSettings": {
"bitDepth": integer,
"channels": integer,
"sampleRate": integer
},
"eac3Settings": {
"metadataControl": enum,
"surroundExMode": enum,
"loRoSurroundMixLevel": number,
"phaseControl": enum,
"dialnorm": integer,
"ltRtSurroundMixLevel": number,
"bitrate": integer,
"ltRtCenterMixLevel": number,
"passthroughControl": enum,
"lfeControl": enum,
"loRoCenterMixLevel": number,
"attenuationControl": enum,
"codingMode": enum,
"surroundMode": enum,
"bitstreamMode": enum,
"lfeFilter": enum,
"stereoDownmix": enum,
"dynamicRangeCompressionRf": enum,
"sampleRate": integer,
"dynamicRangeCompressionLine": enum,
"dcFilter": enum
},
"mp2Settings": {
"bitrate": integer,
"channels": integer,
"sampleRate": integer
},
"wavSettings": {
"bitDepth": integer,
"channels": integer,
"sampleRate": integer,
"format": enum
},
"remixSettings": {
"channelMapping": {
"outputChannels": [
  "inputChannels": [integer
   ]
  ]
},
"channelsIn": integer,
"channelsOut": integer
},
"streamName": "string",
"languageCodeControl": enum,
"audioType": integer,
"customLanguageCode": "string",
"languageCode": enum
[
"outputSettings": {
  "hlsSettings": {
    "audioGroupId": "string",
    "audioRenditionSets": "string",
    "audioTrackType": enum,
    "iFrameOnlyManifest": enum,
    "segmentModifier": "string"
  }
],
"extension": "string",
"nameModifier": "string",
"captionDescriptions": [
  {
    "captionSelectorName": "string",
    "destinationSettings": {
      "destinationType": enum,
      "burninDestinationSettings": {
        "backgroundOpacity": integer,
        "shadowXOffset": integer,
        "teletextSpacing": enum,
        "alignment": enum,
        "outlineSize": integer,
      }
    }
  }]}
"yPosition": integer,
"shadowColor": enum,
"fontOpacity": integer,
"fontSize": integer,
"fontScript": enum,
"fontColor": enum,
"backgroundColor": enum,
"fontResolution": integer,
"outlineColor": enum,
"shadowYOffset": integer,
"xPosition": integer,
"shadowOpacity": integer
},
"dvbSubDestinationSettings": {
"backgroundOpacity": integer,
"shadowXOffset": integer,
"teletextSpacing": enum,
"alignment": enum,
"outlineSize": integer,
"yPosition": integer,
"shadowColor": enum,
"fontOpacity": integer,
"fontSize": integer,
"fontScript": enum,
"fontColor": enum,
"backgroundColor": enum,
"fontResolution": integer,
"outlineColor": enum,
"shadowYOffset": integer,
"xPosition": integer,
"shadowOpacity": integer
},
"sccDestinationSettings": {
"framerate": enum
},
"teletextDestinationSettings": {
"pageNumber": "string"
},
"ttmlDestinationSettings": {
"stylePassthrough": enum
},
"embeddedDestinationSettings": {
"destination608ChannelNumber": integer
},
"customLanguageCode": "string",
"languageCode": enum,
"languageDescription": "string"
]}
],
"outputGroupSettings": {
"type": enum,
"hlsGroupSettings": {
"manifestDurationFormat": enum,
"segmentLength": integer,
"timedMetadataId3Period": integer,
"captionLanguageSetting": enum,
"captionLanguageMappings": [

"captionChannel": integer,
"customLanguageCode": "string",
"languageCode": enum,
"languageDescription": "string"
]}
}
"destination": "string",
"destinationSettings": {
  "s3Settings": {
    "encryption": {
      "encryptionType": enum,
      "kmsKeyArn": "string"
    }
  }
},
"encryption": {
  "encryptionMethod": enum,
  "constantInitializationVector": "string",
  "initializationVectorInManifest": enum,
  "offlineEncrypted": enum,
  "spekeKeyProvider": {
    "resourceId": "string",
    "systemIds": ["string"]
  },
  "url": "string",
  "certificateArn": "string"
},
"staticKeyProvider": {
  "staticKeyValue": "string",
  "keyFormat": "string",
  "keyFormatVersions": "string",
  "url": "string"
},
"type": enum
},
"timedMetadataId3Frame": enum,
"baseUrl": "string",
"codecSpecification": enum,
"outputSelection": enum,
"programDateTimePeriod": integer,
"segmentsPerSubdirectory": integer,
"minSegmentLength": integer,
"minFinalSegmentLength": number,
"directoryStructure": enum,
"programDateTime": enum,
"adMarkers": [
  enum
],
"segmentControl": enum,
"timestampDeltaMilliseconds": integer,
"manifestCompression": enum,
"clientCache": enum,
"streamInfResolution": enum
},
"dashIsoGroupSettings": {
  "segmentLength": integer,
  "destination": "string",
  "destinationSettings": {
    "s3Settings": {
      "encryption": {
        "encryptionType": enum,
        "kmsKeyArn": "string"
      }
    }
  },
  "encryption": {
    "spekeKeyProvider": {
      "resourceId": "string",
      "systemIds": ["string"
},
  "url": "string",
  "certificateArn": "string"
}
},
"minBufferTime": integer,
"fragmentLength": integer,
"baseUrl": "string",
"segmentControl": enum,
"hbbtvCompliance": enum,
"writeSegmentTimelineInRepresentation": enum
},
"fileGroupSettings": {
  "destination": "string",
  "destinationSettings": {
    "s3Settings": {
      "encryption": {
        "encryptionType": enum,
        "kmsKeyArn": "string"
      }
    }
  }
},
"msSmoothGroupSettings": {
  "destination": "string",
  "destinationSettings": {
    "s3Settings": {
      "encryption": {
        "encryptionType": enum,
        "kmsKeyArn": "string"
      }
    }
  }
},
"manifestEncoding": enum,
"audioDeduplication": enum
},
"cmafGroupSettings": {
  "writeHlsManifest": enum,
  "writeDashManifest": enum,
  "segmentLength": integer,
  "minFinalSegmentLength": number,
  "destination": "string",
  "destinationSettings": {
    "s3Settings": {
      "encryption": {
        "encryptionType": enum,
        "kmsKeyArn": "string"
      }
    }
  }
},
"encryption": {
  "encryptionMethod": enum,
  "constantInitializationVector": "string",
  "initializationVectorInManifest": enum,
  "staticKeyProvider": {
    "resourceId": "string",
    "systemIds": [
      "string"
    ],
    "url": "string",
    "certificateArn": "string"
  }
},
"manifestEncoding": enum,
"staticKeyValue": "string",
"keyFormat": "string",
"keyFormatVersions": "string",
"url": "string"
},
"type": enum,
"minBufferTime": integer,
"fragmentLength": integer,
"baseUrl": "string",
"segmentControl": enum,
"manifestDurationFormat": enum,
"streamInfResolution": enum,
"clientCache": enum,
"manifestCompression": enum,
"codecSpecification": enum
]}
}
,"adAvailOffset": integer,
"availBlanking": {
  "availBlankingImage": "string"
},
"timedMetadataInsertion": {
  "id3Insertions": [
    {
      "timecode": "string",
      "id3": "string"
    }
  ]
},
"nielsenConfiguration": {
  "breakoutCode": integer,
  "distributorId": "string"
},
"motionImageInserter": {
  "insertionMode": enum,
  "input": "string",
  "offset": {
    "imageX": integer,
    "imageY": integer
  },
  "startTime": "string",
  "playback": enum,
  "framerate": {
    "framerateNumerator": integer,
    "framerateDenominator": integer
  }
},
"esam": {
  "signalProcessingNotification": {
    "sccXml": "string"
  },
  "manifestConfirmConditionNotification": {
    "mccXml": "string"
  },
  "responseSignalPreroll": integer
},
"inputs": [
  {
    "inputClippings": [
      {
        "endTimecode": "string",
        "startTimecode": "string"
      }
    ]
  }
]
"audioSelectors": {},
"audioSelectorGroups": {},
"programNumber": integer,
"videoSelector": {
  "colorSpace": enum,
  "rotate": enum,
  "pid": integer,
  "programNumber": integer,
  "colorSpaceUsage": enum,
  "hdr10Metadata": {
    "redPrimaryX": integer,
    "redPrimaryY": integer,
    "greenPrimaryX": integer,
    "greenPrimaryY": integer,
    "bluePrimaryX": integer,
    "bluePrimaryY": integer,
    "whitePointX": integer,
    "whitePointY": integer,
    "maxFrameAverageLightLevel": integer,
    "maxContentLightLevel": integer,
    "maxLuminance": integer,
    "minLuminance": integer
  }
},
"filterEnable": enum,
"psiControl": enum,
"filterStrength": integer,
"deblockFilter": enum,
"denoiseFilter": enum,
"timecodeSource": enum,
"captionSelectors": {},
"imageInserter": {
  "insertableImages": [
    {
      "width": integer,
      "height": integer,
      "imageX": integer,
      "imageY": integer,
      "duration": integer,
      "fadeIn": integer,
      "layer": integer,
      "imageInserterInput": "string",
      "startTime": "string",
      "fadeOut": integer,
      "opacity": integer
    }
  ]
},
"fileInput": "string",
"decryptionSettings": {
  "decryptionMode": enum,
  "encryptedDecryptionKey": "string",
  "initializationVector": "string",
  "kmsKeyRegion": "string"
},
"supplementalImps": [
  "string"
]}
"status": enum,
"errorCode": integer,
"errorMessage": "string",
"timing": {
  "submitButton": "string",
  "startTime": "string",
  "finishTime": "string"
},
"outputGroupDetails": [
  {
    "outputDetails": [
      {
        "durationInMs": integer,
        "videoDetails": {
          "widthInPx": integer,
          "heightInPx": integer
        }
      }
    ]
  }
],
"billingTagsSource": enum,
"accelerationSettings": {
  "mode": enum
},
"statusUpdateInterval": enum,
"jobPercentComplete": integer,
"currentPhase": enum,
"retryCount": integer
],
"nextToken": "string"

Example CreateJobResponse

{
  "job": {
    "arn": "string",
    "id": "string",
    "createdAt": "string",
    "jobTemplate": "string",
    "queue": "string",
    "userMetadata": {
    },
    "role": "string",
    "settings": {
      "timecodeConfig": {
        "anchor": "string",
        "source": enum,
        "start": "string",
        "timestampOffset": "string"
      },
      "outputGroups": [
        {
          "customName": "string",
          "name": "string",
          "outputs": [
            {
              "containerSettings": {
                "container": enum,
                "m3u8Settings": {
                  "audioFramesPerPes": integer,
                  "pcrControl": enum,
                  "pcrPid": integer,
                }
              },
            }
          ]
        }
      ]
    }
  }
}
"pmtPid": integer,
"privateMetadataPid": integer,
"programNumber": integer,
"patInterval": integer,
"pmtInterval": integer,
"scte35Source": enum,
"scte35Pid": integer,
"nielsenId3": enum,
"timedMetadata": enum,
"timedMetadataPid": integer,
"transportStreamId": integer,
"videoPid": integer,
"audioPids": [integer
  ]
},
"f4vSettings": {
  "moovPlacement": enum
},
"m2tsSettings": {
  "audioBufferModel": enum,
  "minEbpInterval": integer,
  "esRateInPes": enum,
  "patInterval": integer,
  "dvbNitSettings": {
    "nitInterval": integer,
    "networkId": integer,
    "networkName": "string"
  },
  "dvbSdtSettings": {
    "outputSdt": enum,
    "sdtInterval": integer,
    "serviceName": "string",
    "serviceProviderName": "string"
  },
  "scte35Source": enum,
  "scte35Pid": integer,
  "scte35Esam": {
    "scte35EsamPid": integer
  },
  "videoPid": integer,
  "dvbTdtSettings": {
    "tdtInterval": integer
  },
  "pmtInterval": integer,
  "segmentationStyle": enum,
  "segmentationTime": number,
  "pmtPid": integer,
  "bitrate": integer,
  "audioPids": [integer
    ]
},
"privateMetadataPid": integer,
"nielsenId3": enum,
"timedMetadataPid": integer,
"maxPcrInterval": integer,
"transportStreamId": integer,
"dvbSubPids": [integer
  ],
"rateMode": enum,
"audioFramesPerPes": integer,
"pcrControl": enum,
"segmentationMarkers": enum,
"ebpAudioInterval": enum,
"forceTsVideoEbpoOrder": enum,
<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;programNumber&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;pcrPid&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;bufferModel&quot;</td>
<td>enum</td>
</tr>
<tr>
<td>&quot;dvbTeletextPid&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;fragmentTime&quot;</td>
<td>number</td>
</tr>
<tr>
<td>&quot;ebpPlacement&quot;</td>
<td>enum</td>
</tr>
<tr>
<td>&quot;nullPacketBitrate&quot;</td>
<td>number</td>
</tr>
<tr>
<td>&quot;movSettings&quot;</td>
<td></td>
</tr>
<tr>
<td>&quot;clapAtom&quot;</td>
<td>enum</td>
</tr>
<tr>
<td>&quot;cslgAtom&quot;</td>
<td>enum</td>
</tr>
<tr>
<td>&quot;paddingControl&quot;</td>
<td>enum</td>
</tr>
<tr>
<td>&quot;reference&quot;</td>
<td>enum</td>
</tr>
<tr>
<td>&quot;mpeg2FourCCControl&quot;</td>
<td></td>
</tr>
<tr>
<td>&quot;mp4Settings&quot;</td>
<td></td>
</tr>
<tr>
<td>&quot;cslgAtom&quot;</td>
<td>enum</td>
</tr>
<tr>
<td>&quot;freeSpaceBox&quot;</td>
<td>enum</td>
</tr>
<tr>
<td>&quot;mp4MajorBrand&quot;</td>
<td>string</td>
</tr>
<tr>
<td>&quot;moovPlacement&quot;</td>
<td>enum</td>
</tr>
<tr>
<td>&quot;preset&quot;</td>
<td>string</td>
</tr>
<tr>
<td>&quot;videoDescription&quot;</td>
<td></td>
</tr>
<tr>
<td>&quot;fixedAfd&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;width&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;scalingBehavior&quot;</td>
<td>enum</td>
</tr>
<tr>
<td>&quot;crop&quot;</td>
<td></td>
</tr>
<tr>
<td>&quot;height&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;width&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;x&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;y&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;height&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;videoPreprocessors&quot;</td>
<td></td>
</tr>
<tr>
<td>&quot;colorCorrector&quot;</td>
<td></td>
</tr>
<tr>
<td>&quot;brightness&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;colorSpaceConversion&quot;</td>
<td>enum</td>
</tr>
<tr>
<td>&quot;contrast&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;hue&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;saturation&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;hdr10Metadata&quot;</td>
<td></td>
</tr>
<tr>
<td>&quot;redPrimaryX&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;redPrimaryY&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;greenPrimaryX&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;greenPrimaryY&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;bluePrimaryX&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;bluePrimaryY&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;whitePointX&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;whitePointY&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;maxFrameAverageLightLevel&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;maxContentLightLevel&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;maxLuminance&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;minLuminance&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;deinterlacer&quot;</td>
<td></td>
</tr>
<tr>
<td>&quot;algorithm&quot;</td>
<td>enum</td>
</tr>
<tr>
<td>&quot;mode&quot;</td>
<td>enum</td>
</tr>
<tr>
<td>&quot;control&quot;</td>
<td>enum</td>
</tr>
<tr>
<td>&quot;imageInserter&quot;</td>
<td></td>
</tr>
<tr>
<td>&quot;insertableImages&quot;</td>
<td></td>
</tr>
<tr>
<td>&quot;width&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;height&quot;</td>
<td>integer</td>
</tr>
</tbody>
</table>
"imageX": integer,
"imageY": integer,
"duration": integer,
"fadeIn": integer,
"layer": integer,
"imageInserterInput": "string",
"startTime": "string",
"fadeOut": integer,
"opacity": integer
}
]
},
"noiseReducer": {
  "filter": enum,
  "filterSettings": {
    "strength": integer
  },
  "spatialFilterSettings": {
    "strength": integer,
    "speed": integer,
    "postFilterSharpenStrength": integer
  }
},
"timecodeBurnin": {
  "fontSize": integer,
  "position": enum,
  "prefix": "string"
}
},
"timecodeInsertion": enum,
"antiAlias": enum,
"position": {
  "height": integer,
  "width": integer,
  "x": integer,
  "y": integer
},
"sharpness": integer,
"codecSettings": {
  "codec": enum,
  "frameCaptureSettings": {
    "framerateNumerator": integer,
    "framerateDenominator": integer,
    "maxCaptures": integer,
    "quality": integer
  },
  "h264Settings": {
    "interlaceMode": enum,
    "parNumerator": integer,
    "numberReferenceFrames": integer,
    "syntax": enum,
    "softness": integer,
    "framerateDenominator": integer,
    "gopClosedCadence": integer,
    "hrdBufferInitialFillPercentage": integer,
    "gopSize": number,
    "slices": integer,
    "gopReference": enum,
    "hrdBufferSize": integer,
    "maxBitrate": integer,
    "slowPal": enum,
    "parDenominator": integer,
    "spatialAdaptiveQuantization": enum,
    "temporalAdaptiveQuantization": enum,
    "flickerAdaptiveQuantization": enum,
    "entropyEncoding": enum,
"bitrate": integer,
"framerateControl": enum,
"rateControlMode": enum,
"qvbrSettings": {
    "qvbrQualityLevel": integer,
    "maxAverageBitrate": integer
},
"codecProfile": enum,
"telecine": enum,
"framerateNumerator": integer,
"minIInterval": integer,
"adaptiveQuantization": enum,
"codecLevel": enum,
"fieldEncoding": enum,
"sceneChangeDetect": enum,
"qualityTuningLevel": enum,
"framerateConversionAlgorithm": enum,
"unregisteredSeiTimecode": enum,
"gopSizeUnits": enum,
"parControl": enum,
"numberBFramesBetweenReferenceFrames": integer,
"repeatPps": enum,
"dynamicSubGop": enum
},
"h265Settings": {
    "interlaceMode": enum,
    "parNumerator": integer,
    "numberReferenceFrames": integer,
    "framerateDenominator": integer,
    "gopClosedCadence": integer,
    "alternateTransferFunctionSei": enum,
    "hrdBufferInitialFillPercentage": integer,
    "gopSize": number,
    "slices": integer,
    "gopBReference": enum,
    "hrdBufferSize": integer,
    "maxBitrate": integer,
    "temporalIds": enum,
    "sampleAdaptiveOffsetFilterMode": enum,
    "writeMp4PackagingType": enum,
"dynamicSubGop": enum,
  "mpeg2Settings": {
    "interlaceMode": enum,
    "parNumerator": integer,
    "syntax": enum,
    "softness": integer,
    "framerateDenominator": integer,
    "gopClosedCadence": integer,
    "hrdBuffeInitialFillPercentage": integer,
    "gopSize": number,
    "hrdBuffeSize": integer,
    "maxBitrate": integer,
    "slowPal": enum,
    "parDenominator": integer,
    "spatialAdaptiveQuantization": enum,
    "temporalAdaptiveQuantization": enum,
    "bitrate": integer,
    "intraDcPrecision": enum,
    "framerateControl": enum,
    "rateControlMode": enum,
    "codecProfile": enum,
    "telecine": enum,
    "framerateNumerator": integer,
    "minIInterval": integer,
    "adaptiveQuantization": enum,
    "codecLevel": enum,
    "sceneChangeDetect": enum,
    "qualityTuningLevel": enum,
    "framerateConversionAlgorithm": enum,
    "gopSizeUnits": enum,
    "parControl": enum,
    "numberBFramesBetweenReferenceFrames": integer,
    "dynamicSubGop": enum
  },
  "proresSettings": {
    "interlaceMode": enum,
    "parNumerator": integer,
    "framerateDenominator": integer,
    "codecProfile": enum,
    "slowPal": enum,
    "parDenominator": integer,
    "framerateControl": enum,
    "telecine": enum,
    "framerateNumerator": integer,
    "framerateConversionAlgorithm": enum,
    "parControl": enum
  },
  "afdSignaling": enum,
  "dropFrameTimecode": enum,
  "respondToAfd": enum,
  "colorMetadata": enum
},
"audioDescriptions": [
  {
    "audioTypeControl": enum,
    "audioSourceName": "string",
    "audioNormalizationSettings": {
      "algorithm": enum,
      "algorithmControl": enum,
      "correctionGateLevel": integer,
      "loudnessLogging": enum,
      "targetLkfs": number,
      "peakCalculation": enum
    }
  }
]
"codecSettings": {
  "codec": enum,
  "aacSettings": {
    "audioDescriptionBroadcasterMix": enum,
    "vbrQuality": enum,
    "bitrate": integer,
    "rateControlMode": enum,
    "codecProfile": enum,
    "codingMode": enum,
    "rawFormat": enum,
    "sampleRate": integer,
    "specification": enum
  },
  "ac3Settings": {
    "bitrate": integer,
    "bitstreamMode": enum,
    "codingMode": enum,
    "dialnorm": integer,
    "dynamicRangeCompressionProfile": enum,
    "metadataControl": enum,
    "lfeFilter": enum,
    "sampleRate": integer
  },
  "aiffSettings": {
    "bitDepth": integer,
    "channels": integer,
    "sampleRate": integer
  },
  "eac3Settings": {
    "metadataControl": enum,
    "surroundExMode": enum,
    "loRoSurroundMixLevel": number,
    "phaseControl": enum,
    "dialnorm": integer,
    "ltRtSurroundMixLevel": number,
    "bitrate": integer,
    "ltRtCenterMixLevel": number,
    "passthroughControl": enum,
    "lfeControl": enum,
    "loRoCenterMixLevel": number,
    "attenuationControl": enum,
    "codingMode": enum,
    "surroundMode": enum,
    "bitstreamMode": enum,
    "lfeFilter": enum,
    "stereoDownmix": enum,
    "dynamicRangeCompressionRf": enum,
    "sampleRate": integer,
    "dynamicRangeCompressionLine": enum,
    "dcFilter": enum
  },
  "mp2Settings": {
    "bitrate": integer,
    "channels": integer,
    "sampleRate": integer
  },
  "wavSettings": {
    "bitDepth": integer,
    "channels": integer,
    "sampleRate": integer,
    "format": enum
  }
},
"remixSettings": {
  "channelMapping": {
    "outputChannels": [
{  
  "inputChannels": [
    integer
  ]
},
"channelsIn": integer,
"channelsOut": integer,
"streamName": "string",
"languageCodeControl": enum,
"audioType": integer,
"customLanguageCode": "string",
"languageCode": enum
]
},
"outputSettings": {
  "hlsSettings": {
    "audioGroupId": "string",
    "audioRenditionSets": "string",
    "audioTrackType": enum,
    "iFrameOnlyManifest": enum,
    "segmentModifier": "string"
  }
},
"extension": "string",
"nameModifier": "string",
"captionDescriptions": [
  {
    "captionSelectorName": "string",
    "destinationSettings": {
      "destinationType": enum,
      "burninDestinationSettings": {
        "backgroundOpacity": integer,
        "shadowXOffset": integer,
        "teletextSpacing": enum,
        "alignment": enum,
        "outlineSize": integer,
        "yPosition": integer,
        "shadowColor": enum,
        "fontOpacity": integer,
        "fontSize": integer,
        "fontScript": enum,
        "fontColor": enum,
        "backgroundColor": enum,
        "fontResolution": integer,
        "outlineColor": enum,
        "shadowYOffset": integer,
        "xPosition": integer,
        "shadowOpacity": integer
      },
      "dvbSubDestinationSettings": {
        "backgroundOpacity": integer,
        "shadowXOffset": integer,
        "teletextSpacing": enum,
        "alignment": enum,
        "outlineSize": integer,
        "yPosition": integer,
        "shadowColor": enum,
        "fontOpacity": integer,
        "fontSize": integer,
        "fontScript": enum,
        "fontColor": enum,
        "backgroundColor": enum,
        "fontResolution": integer,
      }
    }
  }
}
"outlineColor": enum,
"shadowYOffset": integer,
"xPosition": integer,
"shadowOpacity": integer
},
"sccDestinationSettings": {
"framerate": enum
},
"teletextDestinationSettings": {
"pageNumber": "string"
},
"ttmlDestinationSettings": {
"stylePassthrough": enum
},
"embeddedDestinationSettings": {
"destination608ChannelNumber": integer
},
"customLanguageCode": "string",
"languageCode": enum,
"languageDescription": "string"
]}
],
"outputGroupSettings": {
"type": enum,
"hlsGroupSettings": {
"manifestDurationFormat": enum,
"segmentLength": integer,
"timedMetadataId3Period": integer,
"captionLanguageSetting": enum,
"captionLanguageMappings": [
{
"captionChannel": integer,
"customLanguageCode": "string",
"languageCode": enum,
"languageDescription": "string"
}
],
"destination": "string",
"destinationSettings": {
"s3Settings": {
"encryption": {
"encryptionType": enum,
"kmsKeyArn": "string"
}
}
},
"encryption": {
"encryptionMethod": enum,
"constantInitializationVector": "string",
"initializationVectorInManifest": enum,
"offlineEncrypted": enum,
"spekeKeyProvider": {
"resourceId": "string",
"systemIds": [
"string"
],
"url": "string",
"certificateArn": "string"
},
"staticKeyProvider": {
"staticKeyValue": "string",
"keyFormat": "string",
"keyFormatVersions": "string",
"470"
"url": "string",
},
"type": enum,
"timedMetadataId3Frame": enum,
"baseUrl": "string",
"codecSpecification": enum,
"outputSelection": enum,
"programDateTimePeriod": integer,
"segmentsPerSubdirectory": integer,
"minSegmentLength": integer,
"minFinalSegmentLength": number,
"directoryStructure": enum,
"programDateTime": enum,
"adMarkers": [
  enum
],
"segmentControl": enum,
"timestampDeltaMilliseconds": integer,
"manifestCompression": enum,
"clientCache": enum,
"streamInfResolution": enum,
"dashIsoGroupSettings": {
  "segmentLength": integer,
  "destination": "string",
  "destinationSettings": {
    "s3Settings": {
      "encryption": {
        "encryptionType": enum,
        "kmsKeyArn": "string"
      }
    }
  },
  "encryption": {
    "spekeKeyProvider": {
      "resourceId": "string",
      "systemIds": [
        "string"
      ],
      "url": "string",
      "certificateArn": "string"
    }
  },
  "minBufferTime": integer,
  "fragmentLength": integer,
  "baseUrl": "string",
  "segmentControl": enum,
  "hbbtvCompliance": enum,
  "writeSegmentTimelineInRepresentation": enum
},
"fileGroupSettings": {
  "destination": "string",
  "destinationSettings": {
    "s3Settings": {
      "encryption": {
        "encryptionType": enum,
        "kmsKeyArn": "string"
      }
    }
  },
},
"msSmoothGroupSettings": {
  "destination": "string",
  "destinationSettings": {
    "s3Settings": {
    }
"encryption": {
  "encryptionType": enum,
  "kmsKeyArn": "string"
}
},
"fragmentLength": integer,
"encryption": {
  "spekeKeyProvider": {
    "resourceId": "string",
    "systemIds": [
      "string"
    ],
    "url": "string",
    "certificateArn": "string"
  }
},
"manifestEncoding": enum,
"audioDeduplication": enum
},
"cmafGroupSettings": {
  "writeHlsManifest": enum,
  "writeDashManifest": enum,
  "segmentLength": integer,
  "minFinalSegmentLength": number,
  "destination": "string",
  "destinationSettings": {
    "s3Settings": {
      "encryption": {
        "encryptionType": enum,
        "kmsKeyArn": "string"
      }
    }
  },
  "encryption": {
    "encryptionMethod": enum,
    "constantInitializationVector": "string",
    "initializationVectorInManifest": enum,
    "staticKeyProvider": {
      "staticKeyValue": "string",
      "keyFormat": "string",
      "keyFormatVersions": "string",
      "url": "string"
    },
    "type": enum
  },
  "minBufferTime": integer,
  "fragmentLength": integer,
  "baseUrl": "string",
  "segmentControl": enum,
  "manifestDurationFormat": enum,
  "streamInfResolution": enum,
  "clientCache": enum,
  "manifestCompression": enum,
  "codecSpecification": enum
}
},
"adAvailOffset": integer,
"availBlanking": {
  "availBlankingImage": "string"
},
"timedMetadataInsertion": {
  "id3Insertions": [
    
  ]
}
"timecode": "string",
"id3": "string"
}]
},
"nielsenConfiguration": {
"breakoutCode": integer,
"distributorId": "string"
},
"motionImageInserter": {
"insertionMode": enum,
"input": "string",
"offset": {
"imageX": integer,
"imageY": integer
},
"startTime": "string",
"playback": enum,
"framerate": {
"framerateNumerator": integer,
"framerateDenominator": integer
}
},
"esam": {
"signalProcessingNotification": {
"sccXml": "string"
},
"manifestConfirmConditionNotification": {
"mccXml": "string"
},
"responseSignalPreroll": integer
},
"inputs": [
{
"inputClippings": [
{
"endTimecode": "string",
"startTimecode": "string"
}
],
"audioSelectors": {
},
"audioSelectorGroups": {
},
"programNumber": integer,
"videoSelector": {
"colorSpace": enum,
"rotate": enum,
"pid": integer,
"programNumber": integer,
"colorSpaceUsage": enum,
"hdr10Metadata": {
"redPrimaryX": integer,
"redPrimaryY": integer,
"greenPrimaryX": integer,
"greenPrimaryY": integer,
"bluePrimaryX": integer,
"bluePrimaryY": integer,
"whitePointX": integer,
"whitePointY": integer,
"maxFrameAverageLightLevel": integer,
"maxContentLightLevel": integer,
"maxLuminance": integer,
"minLuminance": integer
}
}]}
"filterEnable": enum,
"psiControl": enum,
"filterStrength": integer,
"deblockFilter": enum,
"denoiseFilter": enum,
"timecodeSource": enum,
"captionSelectors": {
},
"imageInserter": {
  "insertableImages": [
    {
      "width": integer,
      "height": integer,
      "imageX": integer,
      "imageY": integer,
      "duration": integer,
      "fadeIn": integer,
      "layer": integer,
      "imageInserterInput": "string",
      "startTime": "string",
      "fadeOut": integer,
      "opacity": integer
    }
  ],
  "fileInput": "string",
  "decryptionSettings": {
    "decryptionMode": enum,
    "encryptedDecryptionKey": "string",
    "initializationVector": "string",
    "kmsKeyRegion": "string"
  },
  "supplementalImps": ["string"
}],
"status": enum,
"errorCode": integer,
"errorMessage": "string",
"timing": {
  "submitTime": "string",
  "startTime": "string",
  "finishTime": "string"
},
"outputGroupDetails": [
  {
    "outputDetails": [
      {
        "durationInMs": integer,
        "videoDetails": {
          "widthInPx": integer,
          "heightInPx": integer
        }
      }
    ]
  }
],
"billingTagsSource": enum,
"accelerationSettings": {
  "mode": enum
},
"statusUpdateInterval": enum,
"jobPercentComplete": integer,
"currentPhase": enum,
Properties

AacAudioDescriptionBroadcasterMix

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

AAC Profile.

|||
| LC |
| HEV1 |
| HEV2 |

AacCodingMode

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

Rate Control Mode.

| CBR |
| VBR |
AacRawFormat

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

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value AAC. The service accepts one of two mutually exclusive groups of AAC settings--VBR and CBR. To select one of these modes, set the value of Bitrate control mode (rateControlMode) to "VBR" or "CBR". In VBR mode, you control the audio quality with the setting VBR quality (vbrQuality). In CBR mode, you use the setting Bitrate (bitrate). Defaults and valid values depend on the rate control mode.

audioDescriptionBroadcasterMix

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.

- **Type**: AacAudioDescriptionBroadcasterMix (p. 475)
- **Required**: False

vbrQuality

VBR Quality Level - Only used if rate_control_mode is VBR.

- **Type**: AacVbrQuality (p. 477)
- **Required**: False

bitrate

Average bitrate in bits/second. The set of valid values for this setting is: 6000, 8000, 10000, 12000, 14000, 16000, 20000, 24000, 28000, 32000, 40000, 48000, 56000, 64000, 80000, 96000, 112000, 128000, 160000, 192000, 224000, 256000, 288000, 320000, 384000, 448000, 512000, 576000, 640000, 768000, 896000, 1024000. The value you set is also constrained by the values you choose for Profile (codecProfile), Bitrate control mode (codingMode), and Sample rate (sampleRate). Default values depend on Bitrate control mode and Profile.

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

rateControlMode

Rate Control Mode.
Properties

**Type:** AacRateControlMode (p. 475)
**Required:** False

**codecProfile**

AAC Profile.

**Type:** AacCodecProfile (p. 475)
**Required:** False

**codingMode**

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.

**Type:** AacCodingMode (p. 475)
**Required:** False

**rawFormat**

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.

**Type:** AacRawFormat (p. 476)
**Required:** False

**sampleRate**

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

**Type:** integer
**Required:** False
**Minimum:** 8000
**Maximum:** 96000

**specification**

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

**Type:** AacSpecification (p. 477)
**Required:** False

**AacSpecification**

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

MPEG2
MPEG4

**AacVbrQuality**

VBR Quality Level - Only used if rate_control_mode is VBR.
LOW
MEDIUM_LOW
MEDIUM_HIGH
HIGH

**Ac3BitstreamMode**

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**

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**

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**

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

- ENABLED
- DISABLED

**Ac3MetadataControl**

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**

Required when you set (Codec) under (AudioDescriptions)->(CodecSettings) to the value AC3.

**bitrate**

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

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

**bitstreamMode**

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

*Type:* Ac3BitstreamMode (p. 478)  
*Required:* False

**codingMode**

Dolby Digital coding mode. Determines number of channels.

*Type:* Ac3CodingMode (p. 478)  
*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

**dynamicRangeCompressionProfile**

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

*Type:* Ac3DynamicRangeCompressionProfile (p. 478)  
*Required:* False

**metadataControl**

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.
**Ac3MetadataControl**

*Type:* `Ac3MetadataControl` (p. 478)

*Required:* False

**IfFilter**

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

*Type:* `Ac3LfeFilter` (p. 478)

*Required:* False

**SampleRate**

Sample rate in hz. Sample rate is always 48000.

*Type:* integer

*Required:* False

*Minimum:* 48000

*Maximum:* 48000

**AccelerationMode**

Enable Acceleration (AccelerationMode) on any job that you want processed with accelerated transcoding.

*DISABLED*

*ENABLED*

**AccelerationSettings**

Accelerated transcoding can significantly speed up jobs with long, visually complex content. Outputs that use this feature incur pro-tier pricing. For information about feature limitations, see the AWS Elemental MediaConvert User Guide.

**Mode**

Acceleration configuration for the job.

*Type:* `AccelerationMode` (p. 480)

*Required:* True

**AfdSignaling**

This setting only applies to H.264, H.265, and MPEG2 outputs. Use Insert AFD signaling (AfdSignaling) to specify whether the service includes 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

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value AIFF.

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

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

sampleRate

Sample rate in hz.

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

AncillarySourceSettings

Settings for ancillary captions source.

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

The anti-alias filter is automatically applied to all outputs. The service no longer accepts the value DISABLED for AntiAlias. If you specify that in your job, the service will ignore the setting.

- DISABLED
- ENABLED
Properties

**AudioCodec**

Type of Audio codec.

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

**AudioCodecSettings**

Audio codec settings (CodecSettings) under (AudioDescriptions) contains the group of settings related to audio encoding. The settings in this group vary depending on the value you choose for Audio codec (Codec). For each codec enum you choose, define the corresponding settings object. The following lists the codec enum, settings object pairs: * AAC, AacSettings * MP2, Mp2Settings * WAV, WavSettings * AIFF, AiffSettings * AC3, Ac3Settings * EAC3, Eac3Settings

**codec**

Type of Audio codec.

- **Type**: AudioCodec (p. 482)
- **Required**: False

**aacSettings**

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value AAC. The service accepts one of two mutually exclusive groups of AAC settings--VBR and CBR. To select one of these modes, set the value of Bitrate control mode (rateControlMode) to "VBR" or "CBR". In VBR mode, you control the audio quality with the setting VBR quality (vbrQuality). In CBR mode, you use the setting Bitrate (bitrate). Defaults and valid values depend on the rate control mode.

- **Type**: AacSettings (p. 476)
- **Required**: False

**ac3Settings**

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value AC3.

- **Type**: Ac3Settings (p. 479)
- **Required**: False

**aiffSettings**

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value AIFF.

- **Type**: AiffSettings (p. 481)
- **Required**: False

**eac3Settings**

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value EAC3.
**Type**: `Eac3Settings` (p. 521)
**Required**: False

**mp2Settings**
Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value MP2.

**Type**: `Mp2Settings` (p. 599)
**Required**: False

**wavSettings**
Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value WAV.

**Type**: `WavSettings` (p. 637)
**Required**: False

**AudioDefaultSelection**
Enable this setting on one audio selector to set it as the default for the job. The service uses this default for outputs where it can't find the specified input audio. If you don't set a default, those outputs have no audio.

DEFAULT
NOT_DEFAULT

**AudioDescription**
Description of audio output

**audioTypeControl**
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.

**Type**: `AudioTypeControl` (p. 489)
**Required**: False

**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
**audioNormalizationSettings**

Advanced audio normalization settings.

*Type:* AudioNormalizationSettings (p. 486)

*Required:* False

**codecSettings**

Audio codec settings (CodecSettings) under (AudioDescriptions) contains the group of settings related to audio encoding. The settings in this group vary depending on the value you choose for Audio codec (Codec). For each codec enum you choose, define the corresponding settings object. The following lists the codec enum, settings object pairs. * AAC, AacSettings * MP2, Mp2Settings * WAV, WavSettings * AIFF, AiffSettings * AC3, Ac3Settings * EAC3, Eac3Settings

*Type:* AudioCodecSettings (p. 482)

*Required:* False

**remixSettings**

Advanced audio remixing settings.

*Type:* RemixSettings (p. 621)

*Required:* False

**streamName**

Used for MS Smooth 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]*$`

**languageCodeControl**

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.

*Type:* AudioLanguageCodeControl (p. 485)

*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
**customLanguageCode**

Specify the language for this audio output track, using the ISO 639-2 or ISO 639-3 three-letter language code. The language specified 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:* string  
*Required:* False  
*Pattern:* `^[A-Za-z]{3}$`  
*MinLength:* 3  
*MaxLength:* 3

**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. 578)  
*Required:* False

**AudioLanguageCodeControl**

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**

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**

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**

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

LOG  
DONT_LOG
**AudioNormalizationPeakCalculation**

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

- TRUE_PEAK
- NONE

**AudioNormalizationSettings**

Advanced audio normalization settings.

**algorithm**

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

- Type: AudioNormalizationAlgorithm (p. 485)
- Required: False

**algorithmControl**

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

- Type: AudioNormalizationAlgorithmControl (p. 485)
- 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

**loudnessLogging**

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

- Type: AudioNormalizationLoudnessLogging (p. 485)
- Required: False

**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
Properties

peakCalculation

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

Type: AudioNormalizationPeakCalculation (p. 486)
Required: False

AudioSelector

Selector for Audio

tracks

Identify a track from the input audio to include in this selector by entering the track index number. To include several tracks in a single audio selector, specify multiple tracks as follows. Using the console, enter a comma-separated list. For example, type "1,2,3" to include tracks 1 through 3. Specifying directly in your JSON job file, provide the track numbers in an array. For example, "tracks": [1,2,3].

Type: Array of type integer
Required: False
Minimum: 1
Maximum: 2147483647

offset

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

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

defaultSelection

Enable this setting on one audio selector to set it as the default for the job. The service uses this default for outputs where it can't find the specified input audio. If you don't set a default, those outputs have no audio.

Type: AudioDefaultSelection (p. 483)
Required: False

selectorType

Specifies the type of the audio selector.

Type: AudioSelectorType (p. 489)
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
Minimum: 1
Maximum: 2147483647
externalAudioFileInput

Specifies audio data from an external file source.

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

**Pattern:** ^s3://[^/]+/[^/.]+([Mm]2[Vv])?([Mm][Pp][Ee][Gg])?([Mm][Pp][Ee][Gg][Pp][44])?([Mm][Xx][Ff][Ff])?([Mm][Xx][Ff][Hh])?([Ww][Aa][Vv])?([Yy]4[44])?([Aa][Vv][Ii][Ee][Gg])?([Mm][Pp][Tt][Tt])?([Mm][Oo][Vv])?([Mm][Kk][Mm][Vv])?([Mm][Tt][Ss][Ss])?([Mm][Ss][Zz])?([Mm][Pp][22])?([Aa][Mm][Aa][Ff][Ff])?([Mm][Oo][Vv])?([Mm][Tt][Ss])?([Mm][Pp][44])?([Ww][Cc][Hh])?([Xx][Ee][Pp][44])?([Mm][Xx][Ff][Ff][Hh])?([Ww][Aa][Vv])?([Yy]4[44])?([Aa][Aa][Cc][Cc][Cc][Cc])?([Dd][Dd][Tt][Ss])?([Ee][Ee][Ee])?([Mm][Pp][Ee][Gg][Pp][44])?([Mm][Xx][Ff][Ff][Hh])?([Ww][Aa][Vv])?([Yy]4[44])?([Aa][Aa][Cc][Cc][Cc][Cc])?([Dd][Dd][Tt][Ss])?([Ee][Ee][Ee])?

programSelection

Use this setting for input streams that contain Dolby E, to have the service extract specific program data from the track. To select multiple programs, create multiple selectors with the same Track and different Program numbers. In the console, this setting is visible when you set Selector type to Track. Choose the program number from the dropdown list. If you are sending a JSON file, provide the program ID, which is part of the audio metadata. If your input file has incorrect metadata, you can choose All channels instead of a program number to have the service ignore the program IDs and include all the programs in the track.

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

customLanguageCode

Selects a specific language code from within an audio source, using the ISO 639-2 or ISO 639-3 three-letter language code.

**Type:** string  
**Required:** False
**Pattern:** ^[A-Za-z]{3}$  
**MinLength:** 3  
**MaxLength:** 3

languageCode

Selects a specific language code from within an audio source.

**Type:** LanguageCode (p. 578)  
**Required:** False

remixSettings

Use these settings to reorder the audio channels of one input to match those of another input. This allows you to combine the two files into a single output, one after the other.

**Type:** RemixSettings (p. 621)  
**Required:** False
AudioSelectorGroup

Group of Audio Selectors

audioSelectorNames

Name of an 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: False
MinLength: 1

AudioSelectorType

Specifies the type of the audio selector.

PID
TRACK
LANGUAGE_CODE

AudioTypeControl

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

Settings for Avail Blanking

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)$
MinLength: 14

BillingTagsSource

Optional. Choose a tag type that AWS Billing and Cost Management will use to sort your AWS Elemental MediaConvert costs on any billing report that you set up. Any transcoding outputs that don’t have an associated tag will appear in your billing report unsorted. If you don’t choose a valid value for this field, your job outputs will appear on the billing report unsorted.

QUEUE
**BurninDestinationSettings**

Burn-In Destination Settings.

**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

**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

**teletextSpacing**

Only applies to jobs with input captions in Teletext or STL formats. Specify whether the spacing between letters in your captions is set by the captions grid or varies depending on letter width. Choose fixed grid to conform to the spacing specified in the captions file more accurately. Choose proportional to make the text easier to read if the captions are closed caption.

- **Type**: BurninSubtitleTeletextSpacing (p. 494)
- **Required**: False

**alignment**

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.

- **Type**: BurninSubtitleAlignment (p. 493)
- **Required**: False

**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

**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:** 10

**shadowColor**

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

- **Type:** BurninSubtitleShadowColor (p. 494)
- **Required:** False

**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:** False
- **Minimum:** 0
- **Maximum:** 255

**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

**fontScript**

Provide the font script, using an ISO 15924 script code, if the LanguageCode is not sufficient for determining the script type. Where LanguageCode or CustomLanguageCode is sufficient, use “AUTOMATIC” or leave unset. This is used to help determine the appropriate font for rendering burn-in captions.

- **Type:** FontScript (p. 529)
- **Required:** False
**fontColor**

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.

*Type: BurninSubtitleFontColor (p. 493)*

*Required: False*

**backgroundColor**

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

*Type: BurninSubtitleBackgroundColor (p. 493)*

*Required: False*

**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*

**outlineColor**

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.

*Type: BurninSubtitleOutlineColor (p. 493)*

*Required: False*

**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*

**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*
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

**BurninSubtitleAlignment**

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**

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

- **NONE**
- **BLACK**
- **WHITE**

**BurninSubtitleFontColor**

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**

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.
Properties

BLACK
WHITE
YELLOW
RED
GREEN
BLUE

**BurninSubtitleShadowColor**

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

- NONE
- BLACK
- WHITE

**BurninSubtitleTeletextSpacing**

Only applies to jobs with input captions in Teletext or STL formats. Specify whether the spacing between letters in your captions is set by the captions grid or varies depending on letter width. Choose fixed grid to conform to the spacing specified in the captions file more accurately. Choose proportional to make the text easier to read if the captions are closed caption.

- FIXED_GRID
- PROPORTIONAL

**CaptionDescription**

Description of Caption output

**captionSelectorName**

Specifies which "Caption Selector":#inputs-capion_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**: False
- **MinLength**: 1

**destinationSettings**

Specific settings required by destination type. Note that burnin_destination_settings are not available if the source of the caption data is Embedded or Teletext.

- **Type**: CaptionDestinationSettings (p. 495)
- **Required**: False

**customLanguageCode**

Indicates the language of the caption output track, using the ISO 639-2 or ISO 639-3 three-letter language code. For most captions output formats, the encoder puts this language information in the
output captions metadata. If your output captions format is DVB-Sub or Burn in, the encoder uses this language information to choose the font language for rendering the captions text.

**Properties**

**languageCode**

Specify the language of this captions output track. For most captions output formats, the encoder puts this language information in the output captions metadata. If your output captions format is DVB-Sub or Burn in, the encoder uses this language information to choose the font language for rendering the captions text.

- **Type:** LanguageCode (p. 578)
- **Required:** False

**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

**CaptionDestinationSettings**

Specific settings required by destination type. Note that burnin_destination_settings are not available if the source of the caption data is Embedded or Teletext.

**destinationType**

Specify the format for this set of captions on this output. The default format is embedded without SCTE-20. Other options are embedded with SCTE-20, burn-in, DVB-sub, SCC, SRT, teletext, TTML, and web-VTT. If you are using SCTE-20, choose SCTE-20 plus embedded (SCTE20_PLUS_EMBEDDED) to create an output that complies with the SCTE-43 spec. To create a non-compliant output where the embedded captions come first, choose Embedded plus SCTE-20 (EMBEDDED_PLUS_SCTE20).

- **Type:** CaptionDestinationType (p. 496)
- **Required:** False

**burninDestinationSettings**

Burn-In Destination Settings.

- **Type:** BurninDestinationSettings (p. 490)
- **Required:** False

**dvbSubDestinationSettings**

DVB-Sub Destination Settings

- **Type:** DvbSubDestinationSettings (p. 514)
**Required**: False

**sccDestinationSettings**
Settings for SCC caption output.

- **Type**: SccDestinationSettings (p. 623)
- **Required**: False

**teletextDestinationSettings**
Settings for Teletext caption output

- **Type**: TeletextDestinationSettings (p. 626)
- **Required**: False

**ttmlDestinationSettings**
Settings specific to TTML caption outputs, including Pass style information (TtmlStylePassthrough).

- **Type**: TtmlDestinationSettings (p. 630)
- **Required**: False

**embeddedDestinationSettings**
Settings specific to embedded/ancillary caption outputs, including 608/708 Channel destination number.

- **Type**: EmbeddedDestinationSettings (p. 525)
- **Required**: False

**CaptionDestinationType**
Specify the format for this set of captions on this output. The default format is embedded without SCTE-20. Other options are embedded with SCTE-20, burn-in, DVB-sub, SCC, SRT, teletext, TTML, and web-VTT. If you are using SCTE-20, choose SCTE-20 plus embedded (SCTE20_PLUS_EMBEDDED) to create an output that complies with the SCTE-43 spec. To create a non-compliant output where the embedded captions come first, choose Embedded plus SCTE-20 (EMBEDDED_PLUS_SCTE20).

- BURN_IN
- DVB_SUB
- EMBEDDED
- EMBEDDED_PLUS_SCTE20
- SCTE20_PLUS_EMBEDDED
- SCC
- SRT
- SMI
- TELETEXT
- TTML
- WEBVTT

**CaptionSelector**
Set up captions in your outputs by first selecting them from your input here.
customLanguageCode

The specific language to extract from source, using the ISO 639-2 or ISO 639-3 three-letter language code. 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: string
Required: False
Pattern: ^[A-Za-z]{3}$
MinLength: 3
MaxLength: 3

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. 578)
Required: False

sourceSettings

Source settings (SourceSettings) contains the group of settings for captions in the input.

Type: CaptionSourceSettings (p. 497)
Required: False

CaptionSourceSettings

Source settings (SourceSettings) contains the group of settings for captions in the input.

sourceType

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

Type: CaptionSourceType (p. 498)
Required: False

ancillarySourceSettings

Settings for ancillary captions source.

Type: AncillarySourceSettings (p. 481)
Required: False

dvbSubSourceSettings

DVB Sub Source Settings
**Type**: DvbSubSourceSettings (p. 518)
**Required**: False

**embeddedSourceSettings**
Settings for embedded captions Source

**Type**: EmbeddedSourceSettings (p. 526)
**Required**: False

**fileSourceSettings**
Settings for File-based Captions in Source

**Type**: FileSourceSettings (p. 528)
**Required**: False

**teletextSourceSettings**
Settings specific to Teletext caption sources, including Page number.

**Type**: TeletextSourceSettings (p. 626)
**Required**: False

**trackSourceSettings**
Settings specific to caption sources that are specified by track number. Sources include IMSC in IMF.

**Type**: TrackSourceSettings (p. 629)
**Required**: False

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

ANCILLARY
DVB_SUB
EMBEDDED
SCTE20
SCC
TTML
STL
SRT
SMI
TELETEXT
NULL_SOURCE
IMSC

**ChannelMapping**
Channel mapping (ChannelMapping) contains the group of fields that hold the remixing value for each channel. Units are in dB. Acceptable values are within the range from -60 (mute) through 6. A setting of 0 passes the input channel unchanged to the output channel (no attenuation or amplification).
Properties

outputChannels

List of output channels

  Type: Array of type OutputChannelMapping (p. 614)
  Required: False

CmafClientCache

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

  DISABLED
  ENABLED

CmafCodecSpecification

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

  RFC_6381
  RFC_4281

CmafEncryptionSettings

Settings for CMAF encryption

encryptionMethod

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

  Type: CmafEncryptionType (p. 500)
  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}$
  MinLength: 32
  MaxLength: 32

initializationVectorInManifest

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.

  Type: CmafInitializationVectorInManifest (p. 503)
  Required: False
staticKeyProvider

Use these settings to set up encryption with a static key provider.

Type: StaticKeyProvider (p. 624)
Required: False

type

Indicates which type of key provider is used for encryption.

Type: CmafKeyProviderType (p. 503)
Required: False

CmafEncryptionType

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

SAMPLE_AES

CmafGroupSettings

Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to CMAF_GROUP_SETTINGS. Each output in a CMAF Output Group may only contain a single video, audio, or caption output.

writeHlsManifest

When set to ENABLED, an Apple HLS manifest will be generated for this output.

Type: CmafWriteHLSManifest (p. 504)
Required: False

writeDashManifest

When set to ENABLED, a DASH MPD manifest will be generated for this output.

Type: CmafWriteDASHManifest (p. 504)
Required: False

segmentLength

Use this setting to specify the length, in seconds, of each individual CMAF segment. This value applies to the whole package; that is, to every output in the output group. Note that segments end on the first keyframe after this number of seconds, so the actual segment length might be slightly longer. If you set Segment control (CmafSegmentControl) to single file, the service puts the content of each output in a single file that has metadata that marks these segments. If you set it to segmented files, the service creates multiple files for each output, each with the content of one segment.

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

Keep this setting at the default value of 0, unless you are troubleshooting a problem with how devices play back the end of your video asset. If you know that player devices are hanging on the final segment of your video because the length of your final segment is too short, use this setting to specify a minimum final segment length, in seconds. Choose a value that is greater than or equal to 1 and less than your segment length. When you specify a value for this setting, the encoder will combine any final segment that is shorter than the length that you specify with the previous segment. For example, your segment length is 3 seconds and your final segment is .5 seconds without a minimum final segment length; when you set the minimum final segment length to 1, your final segment is 3.5 seconds.

- **Type:** number
- **Required:** False
- **Format:** float
- **Minimum:** 0.0
- **Maximum:** 2147483647

**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:** False
- **Pattern:** ^s3:////

**destinationSettings**

Settings associated with the destination. Will vary based on the type of destination

- **Type:** DestinationSettings (p. 513)
- **Required:** False

**encryption**

DRM settings.

- **Type:** CmafEncryptionSettings (p. 499)
- **Required:** False

**minBufferTime**

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

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

**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**: False  
  **Minimum**: 1  
  **Maximum**: 2147483647

**baseUrl**

A partial URI prefix that will be put in the manifest 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

**segmentControl**

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.

  **Type**: CmafSegmentControl (p. 503)  
  **Required**: False

**manifestDurationFormat**

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

  **Type**: CmafManifestDurationFormat (p. 503)  
  **Required**: False

**streamInfResolution**

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

  **Type**: CmafStreamInfResolution (p. 503)  
  **Required**: False

**clientCache**

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

  **Type**: CmafClientCache (p. 499)  
  **Required**: False

**manifestCompression**

When set to GZIP, compresses HLS playlist.

  **Type**: CmafManifestCompression (p. 503)  
  **Required**: False
**codecSpecification**

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

*Type:* CmafCodecSpecification (p. 499)

*Required:* False

**CmafInitializationVectorInManifest**

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*

**CmafKeyProviderType**

Indicates which type of key provider is used for encryption.

*STATIC_KEY*

**CmafManifestCompression**

When set to GZIP, compresses HLS playlist.

*GZIP*

*NONE*

**CmafManifestDurationFormat**

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

*FLOATING_POINT*

*INTEGER*

**CmafSegmentControl**

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*

**CmafStreamInfResolution**

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

*INCLUDE*

*EXCLUDE*
CmafWriteDASHManifest
When set to ENABLED, a DASH MPD manifest will be generated for this output.

- DISABLED
- ENABLED

CmafWriteHLSManifest
When set to ENABLED, an Apple HLS manifest will be generated for this output.

- DISABLED
- ENABLED

ColorCorrector
Settings for color correction.

brightness
Brightness level.

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

colorSpaceConversion
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.

- **Type**: ColorSpaceConversion (p. 505)
- **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

**saturation**

Saturation level.

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

**hdr10Metadata**

Use the HDR master display (Hdr10Metadata) settings to correct HDR metadata or to provide missing metadata. Note that these settings are not color correction.

- **Type**: Hdr10Metadata (p. 553)
- **Required**: False

**ColorMetadata**

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

- **IGNORE**
- **INSERT**

**ColorSpace**

If your input video has accurate color space metadata, or if you don't know about color space, leave this set to the default value FOLLOW. The service will automatically detect your input color space. If your input video has metadata indicating the wrong color space, or if your input video is missing color space metadata that should be there, specify the accurate color space here. If you choose HDR10, you can also correct inaccurate color space coefficients, using the HDR master display information controls. You must also set Color space usage (ColorSpaceUsage) to FORCE for the service to use these values.

- **FOLLOW**
- **REC_601**
- **REC_709**
- **HDR10**
- **HLG_2020**

**ColorSpaceConversion**

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**

There are two sources for color metadata, the input file and the job configuration (in the Color space and HDR master display information settings). The Color space usage setting controls which takes precedence. FORCE: The 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: The 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 specific settings.

**container**

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

- **Type:** ContainerType (p. 507)
- **Required:** False

**m3u8Settings**

Settings for TS segments in HLS

- **Type:** M3u8Settings (p. 592)
- **Required:** False

**f4vSettings**

Settings for F4v container

- **Type:** F4vSettings (p. 528)
- **Required:** False

**m2tsSettings**

MPEG-2 TS container settings. These apply to outputs in a File output group when the output's container (ContainerType) is MPEG-2 Transport Stream (M2TS). In these assets, data is organized by the program map table (PMT). Each transport stream program contains subsets of data, including audio, video, and metadata. Each of these subsets of data has a numerical label called a packet identifier (PID). Each transport stream program corresponds to one MediaConvert output. The PMT lists the types of data in a program along with their PID. Downstream systems and players use the program map table to look up the PID for each type of data it accesses and then uses the PIDs to locate specific data within the asset.

- **Type:** M2tsSettings (p. 585)
- **Required:** False

**movSettings**

Settings for MOV Container.
**MoveSettings**

Type: MovSettings (p. 598)
Required: False

**mp4Settings**

Settings for MP4 Container

Type: Mp4Settings (p. 600)
Required: False

**ContainerType**

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

- F4V
- ISMV
- M2TS
- M3U8
- CMFC
- MOV
- MP4
- MPD
- MPD
- RAW

**CreateJobRequest**

Send your create job request with your job settings and IAM role. Optionally, include user metadata and the ARN for the queue.

**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

**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
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

settings

JobSettings contains all the transcode settings for a job.

Type: JobSettings (p. 576)
Required: True

userMetadata

User-defined metadata that you want to associate with an MediaConvert job. You specify metadata in key/value pairs.

Type: object
Required: False

billingTagsSource

Optional. Choose a tag type that AWS Billing and Cost Management will use to sort your AWS Elemental MediaConvert costs on any billing report that you set up. Any transcoding outputs that don't have an associated tag will appear in your billing report unsorted. If you don't choose a valid value for this field, your job outputs will appear on the billing report unsorted.

Type: BillingTagsSource (p. 489)
Required: False

accelerationSettings

Accelerated transcoding can significantly speed up jobs with long, visually complex content. Outputs that use this feature incur pro-tier pricing. For information about feature limitations, see the AWS Elemental MediaConvert User Guide.

Type: AccelerationSettings (p. 480)
Required: False

statusUpdateInterval

Specify how often MediaConvert sends STATUS_UPDATE events to Amazon CloudWatch Events. Set the interval, in seconds, between status updates. MediaConvert sends an update at this interval from the time the service begins processing your job to the time it completes the transcode or encounters an error.

Type: StatusUpdateInterval (p. 625)
Required: False

CreateJobResponse

Successful create job requests will return the job JSON.
job

Each job converts an input file into an output file or files. For more information, see the User Guide at http://docs.aws.amazon.com/mediaconvert/latest/ug/what-is.html

    Type: Job (p. 573)
    Required: False

DashIsoEncryptionSettings

Specifies DRM settings for DASH outputs.

spekeKeyProvider

Settings for use with a SPEKE key provider

    Type: SpekeKeyProvider (p. 624)
    Required: False

DashIsoGroupSettings

Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to DASH_ISO_GROUP_SETTINGS.

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: False
    Minimum: 1
    Maximum: 2147483647

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: False
    Pattern: ^s3:/\/

destinationSettings

Settings associated with the destination. Will vary based on the type of destination

    Type: DestinationSettings (p. 513)
    Required: False
encryption

DRM settings.

Type: DashIsoEncryptionSettings (p. 509)
Required: False

minBufferTime

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

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

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: False
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

segmentControl

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.

Type: DashIsoSegmentControl (p. 511)
Required: False

hbbtvCompliance

Supports HbbTV specification as indicated

Type: DashIsoHbbtvCompliance (p. 511)
Required: False

writeSegmentTimelineInRepresentation

When you enable Precise segment duration in manifests (writeSegmentTimelineInRepresentation), your DASH manifest shows precise segment durations. The segment duration information appears inside
the SegmentTimeline element, inside SegmentTemplate at the Representation level. When this feature
isn't enabled, the segment durations in your DASH manifest are approximate. The segment duration
information appears in the duration attribute of the SegmentTemplate element.

**Type:** DashIsoWriteSegmentTimelineInRepresentation (p. 511)
**Required:** False

**DashIsoHbbtvCompliance**

Supports HbbTV specification as indicated

- HBBTV_1_5
- NONE

**DashIsoSegmentControl**

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

**DashIsoWriteSegmentTimelineInRepresentation**

When you enable Precise segment duration in manifests (writeSegmentTimelineInRepresentation), your
DASH manifest shows precise segment durations. The segment duration information appears inside
the SegmentTimeline element, inside SegmentTemplate at the Representation level. When this feature
isn't enabled, the segment durations in your DASH manifest are approximate. The segment duration
information appears in the duration attribute of the SegmentTemplate element.

- ENABLED
- DISABLED

**DecryptionMode**

Specify the encryption mode that you used to encrypt your input files.

- AES_CTR
- AES_CBC
- AES_GCM

**DeinterlaceAlgorithm**

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

Settings for deinterlacer

algorithm

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.

  Type: DeinterlaceAlgorithm (p. 511)
  Required: False

mode

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.

  Type: DeinterlacerMode (p. 512)
  Required: False

control

- 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.

  Type: DeinterlacerControl (p. 512)
  Required: False

DeinterlacerControl

- 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

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

**DestinationSettings**
Settings associated with the destination. Will vary based on the type of destination

**s3Settings**
Settings associated with S3 destination

*Type: S3DestinationSettings (p. 622)*
*Required: False*

**DropFrameTimecode**
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**
Inserts DVB Network Information Table (NIT) at the specified table repetition interval.

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

*Type: integer*
*Required: False*
*Minimum: 25*
*Maximum: 10000*

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

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

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

*Type: string*
*Required: False*
*MinLength: 1*
*MaxLength: 256*
**DvbSdtSettings**

Inserts DVB Service Description Table (NIT) at the specified table repetition interval.

**outputSdt**

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.

Type: OutputSdt (p. 616)
Required: False

**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
MinLength: 1
MaxLength: 256

**serviceProviderName**

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

Type: string
Required: False
MinLength: 1
MaxLength: 256

**DvbSubDestinationSettings**

DVB-Sub Destination Settings

**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

**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

**teletextSpacing**

Only applies to jobs with input captions in Teletext or STL formats. Specify whether the spacing between letters in your captions is set by the captions grid or varies depending on letter width. Choose fixed grid to conform to the spacing specified in the captions file more accurately. Choose proportional to make the text easier to read if the captions are closed caption.

- **Type:** DvbSubtitleTeletextSpacing (p. 519)  
- **Required:** False

**alignment**

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.

- **Type:** DvbSubtitleAlignment (p. 518)  
- **Required:** False

**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:** False  
- **Minimum:** 0  
- **Maximum:** 10

**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
shadowColor

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

**Type:** DvbSubtitleShadowColor (p. 519)
**Required:** False

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:** False
**Minimum:** 0
**Maximum:** 255

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

fontScript

Provide the font script, using an ISO 15924 script code, if the LanguageCode is not sufficient for determining the script type. Where LanguageCode or CustomLanguageCode is sufficient, use "AUTOMATIC" or leave unset. This is used to help determine the appropriate font for rendering DVB-Sub captions.

**Type:** FontScript (p. 529)
**Required:** False

fontColor

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.

**Type:** DvbSubtitleFontColor (p. 518)
**Required:** False

backgroundColor

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

**Type**: DvbSubtitleBackgroundColor (p. 518)
**Required**: False

**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

**outlineColor**

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.

**Type**: DvbSubtitleOutlineColor (p. 518)
**Required**: False

**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

**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

**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
**DvbSubSourceSettings**

DVB Sub Source Settings

**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**

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**

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

*NONE*

*BLACK*

*WHITE*

**DvbSubtitleFontColor**

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**

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.
DvbSubtitleShadowColor

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

- NONE
- BLACK
- WHITE

DvbSubtitleTeletextSpacing

Only applies to jobs with input captions in Teletext or STL formats. Specify whether the spacing between letters in your captions is set by the captions grid or varies depending on letter width. Choose fixed grid to conform to the spacing specified in the captions file more accurately. Choose proportional to make the text easier to read if the captions are closed caption.

- FIXED_GRID
- PROPORTIONAL

DvbTdtSettings

Inserts DVB Time and Date Table (TDT) at the specified table repetition interval.

tdtInterval

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

- Type: integer
- Required: False
- Minimum: 1000
- Maximum: 30000

Eac3AttenuationControl

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

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

Dolby Digital Plus coding mode. Determines number of channels.

CODING_MODE_1_0
CODING_MODE_2_0
CODING_MODE_3_2

Eac3DcFilter

Activates a DC highpass filter for all input channels.

ENABLED
DISABLED

Eac3DynamicRangeCompressionLine

Enables Dynamic Range Compression that restricts the absolute peak level for a signal.

NONE
FILM_STANDARD
FILM_LIGHT
MUSIC_STANDARD
MUSIC_LIGHT
SPEECH

Eac3DynamicRangeCompressionRf

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

When encoding 3/2 audio, controls whether the LFE channel is enabled

LFE
NO_LFE
Eac3LfeFilter

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

- ENABLED
- DISABLED

Eac3MetadataControl

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

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

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

- SHIFT_90_DEGREES
- NO_SHIFT

Eac3Settings

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value EAC3.

metadataControl

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.

- Type: Eac3MetadataControl (p. 521)
- Required: False

surroundExMode

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

- Type: Eac3SurroundExMode (p. 525)
**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

**phaseControl**

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

- **Type**: Eac3PhaseControl (p. 521)
- **Required**: False

**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

**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

**bitrate**

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

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

**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
Properties

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

**passthroughControl**

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.

**Type**: Eac3PassthroughControl (p. 521)  
**Required**: False

**lfeControl**

When encoding 3/2 audio, controls whether the LFE channel is enabled.

**Type**: Eac3LfeControl (p. 520)  
**Required**: False

**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

**attenuationControl**

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

**Type**: Eac3AttenuationControl (p. 519)  
**Required**: False

**codingMode**

Dolby Digital Plus coding mode. Determines number of channels.

**Type**: Eac3CodingMode (p. 520)  
**Required**: False

**surroundMode**

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

**Type**: Eac3SurroundMode (p. 525)
**Properties**

**Required**: False

### bitstreamMode

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

- **Type**: Eac3BitstreamMode (p. 519)
- **Required**: False

### lfeFilter

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

- **Type**: Eac3LfeFilter (p. 521)
- **Required**: False

### stereoDownmix

Stereo downmix preference. Only used for 3/2 coding mode.

- **Type**: Eac3StereoDownmix (p. 525)
- **Required**: False

### dynamicRangeCompressionRf

Enables Heavy Dynamic Range Compression, ensures that the instantaneous signal peaks do not exceed specified levels.

- **Type**: Eac3DynamicRangeCompressionRf (p. 520)
- **Required**: False

### sampleRate

Sample rate in hz. Sample rate is always 48000.

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

### dynamicRangeCompressionLine

Enables Dynamic Range Compression that restricts the absolute peak level for a signal.

- **Type**: Eac3DynamicRangeCompressionLine (p. 520)
- **Required**: False

### dcFilter

Activates a DC highpass filter for all input channels.

- **Type**: Eac3DcFilter (p. 520)
- **Required**: False
**Eac3StereoDownmix**

Stereo downmix preference. Only used for 3/2 coding mode.

- NOT_INDICATED
- LO_RO
- LT_RT
- DPL2

**Eac3SurroundExMode**

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**

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

- NOT_INDICATED
- ENABLED
- DISABLED

**EmbeddedConvert608To708**

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

**EmbeddedDestinationSettings**

Settings specific to embedded/ancillary caption outputs, including 608/708 Channel destination number.

**destination608ChannelNumber**

Ignore this setting unless your input captions are SCC format and your output container is MXF. With this combination of input captions format and output container, you can optionally use this setting to replace the input channel number with the track number that you specify. Specify a different number for each output captions track. If you don't specify an output track number, the system uses the input channel number for the output channel number. This setting applies to each output individually. You can optionally combine two captions channels in your output. The two output channel numbers can be one of the following pairs: 1,3; 2,4; 1,4; or 2,3.

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

Settings for embedded captions Source

**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

**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

**convert608To708**

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.

*Type*: EmbeddedConvert608To708 (p. 525)  
*Required*: False

**EsamManifestConfirmConditionNotification**

ESAM ManifestConfirmConditionNotification defined by OC-SP-ESAM-API-I03-131025.

**mccXml**

Provide your ESAM ManifestConfirmConditionNotification XML document inside your JSON job settings. Form the XML document as per OC-SP-ESAM-API-I03-131025. The transcoder will use the Manifest Conditioning instructions in the message that you supply.

*Type*: string  
*Required*: False  
*Pattern*: `^\s*<(.|\n)*ManifestConfirmConditionNotification(.|\n)*>\s*$`

**EsamSettings**

Settings for Event Signaling And Messaging (ESAM). If you don’t do ad insertion, you can ignore these settings.

**signalProcessingNotification**

Specifies an ESAM SignalProcessingNotification XML as per OC-SP-ESAM-API-I03-131025. The transcoder uses the signal processing instructions that you provide in the setting SCC XML (sccXml).
**manifestConfirmConditionNotification**

Specifies an ESAM ManifestConfirmConditionNotification XML as per OC-SP-ESAM-API-I03-131025. The transcoder uses the manifest conditioning instructions that you provide in the setting MCC XML (mccXml).

*Type:* EsamManifestConfirmConditionNotification (p. 526)  
*Required:* False

**responseSignalPreroll**

Specifies the stream distance, in milliseconds, between the SCTE 35 messages that the transcoder places and the splice points that they refer to. If the time between the start of the asset and the SCTE-35 message is less than this value, then the transcoder places the SCTE-35 marker at the beginning of the stream.

*Type:* integer  
*Required:* False  
*Minimum:* 0  
*Maximum:* 30000

**EsamSignalProcessingNotification**

ESAM SignalProcessingNotification data defined by OC-SP-ESAM-API-I03-131025.

**sccXml**

Provide your ESAM SignalProcessingNotification XML document inside your JSON job settings. Form the XML document as per OC-SP-ESAM-API-I03-131025. The transcoder will use the signal processing instructions in the message that you supply. Provide your ESAM SignalProcessingNotification XML document inside your JSON job settings. If you want the service to place SCTE-35 markers at the insertion points you specify in the XML document, you must also enable SCTE-35 ESAM (scte35Esam). Note that you can either specify an ESAM XML document or enable SCTE-35 passthrough. You can't do both.

*Type:* string  
*Required:* False  
*Pattern:* ^\s*<(.|
)*SignalProcessingNotification(.|
)*>\s*$

**ExceptionBody**

**message**

*Type:* string  
*Required:* False

**F4vMoovPlacement**

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**

Settings for F4v container

**moovPlacement**

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.

*Type: F4vMoovPlacement (p. 527)*

*Required: False*

**FileGroupSettings**

Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to FILE_GROUPSETTINGS.

**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: False*

*Pattern: ^s3:// \\

**destinationSettings**

Settings associated with the destination. Will vary based on the type of destination

*Type: DestinationSettings (p. 513)*

*Required: False*

**FileSourceConvert608To708**

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**

Settings for File-based Captions in Source

**sourceFile**

External caption file used for loading captions. Accepted file extensions are 'scc', 'ttml', 'dfxp', 'stl', 'srt', and 'smi'.
Type: string
Required: False
Pattern: ^(s3://)(.*?)\.(scc|SCC|ttml|TTML|dfxp|DFXP|stl|STL|srt|SRT|smi|SMI)\$
MinLength: 14

timeDelta
Specifies a time delta in seconds to offset the captions from the source file.

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

convert608To708
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.

Type: FileSourceConvert608To708 (p. 528)
Required: False

FontScript
Provide the font script, using an ISO 15924 script code, if the LanguageCode is not sufficient for determining the script type. Where LanguageCode or CustomLanguageCode is sufficient, use "AUTOMATIC" or leave unset.

AUTOMATIC
HANS
HANT

FrameCaptureSettings
Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value FRAME_CAPTURE.

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

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

### quality

JPEG Quality - a higher value equals higher quality.

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

### H264AdaptiveQuantization

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

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

### H264CodecLevel

Specify an H.264 level that is consistent with your output video settings. If you aren't sure what level to
specify, choose Auto (AUTO).

- 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

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

H264DynamicSubGop

Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting B frames between reference frames (numberBFramesBetweenReferenceFrames).

ADAPTIVE
STATIC

H264EntropyEncoding

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

CABAC
CAVLC

H264FieldEncoding

Choosing FORCE_FIELD disables PAFF encoding for interlaced outputs.

PAFF
FORCE_FIELD

H264FlickerAdaptiveQuantization

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

DISABLED
ENABLED
H264FramerateControl

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

INITIALIZE_FROM_SOURCE
SPECIFIED

H264FramerateConversionAlgorithm

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

DUPLICATE_DROP
INTERPOLATE

H264GopBReference

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

DISABLED
ENABLED

H264GopSizeUnits

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

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, as follows. - 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

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

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

H264QvbrSettings

Settings for quality-defined variable bitrate encoding with the H.264 codec. Required when you set Rate control mode to QVBR. Not valid when you set Rate control mode to a value other than QVBR, or when you don’t define Rate control mode.

qvbrQualityLevel

Required when you use QVBR rate control mode. That is, when you specify qvbrSettings within h264Settings. Specify the target quality level for this output, from 1 to 10. Use higher numbers for greater quality. Level 10 results in nearly lossless compression. The quality level for most broadcast-quality transcodes is between 6 and 9.

Type: integer
Required: False
Minimum: 1
Maximum: 10

maxAverageBitrate

Use this setting only when Rate control mode is QVBR and Quality tuning level is Multi-pass HQ. For Max average bitrate values suited to the complexity of your input video, the service limits the average bitrate of the video part of this output to the value you choose. That is, the total size of the video element is less than or equal to the value you set multiplied by the number of seconds of encoded output.

Type: integer
Required: False
Minimum: 1000
Maximum: 1152000000

H264RateControlMode

Use this setting to specify whether this output has a variable bitrate (VBR), constant bitrate (CBR) or quality-defined variable bitrate (QVBR).

VBR
CBR
QVBR

**H264RepeatPps**

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

**DISABLED**
**ENABLED**

**H264SceneChangeDetect**

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

**DISABLED**
**ENABLED**

**H264Settings**

Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value H_264.

**interlaceMode**

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, as follows. - 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.

**Type:** H264InterlaceMode (p. 532)
**Required:** False

**parNumerator**

Pixel Aspect Ratio numerator.

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

**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
syntax

Produces a bitstream compliant with SMPTE RP-2027.

  Type: H264Syntax (p. 541)
  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

framerateDenominator

When you use the API for transcode jobs that use frame rate conversion, specify the frame rate as a fraction. For example, \( \frac{24000}{1001} = 23.976 \) 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 frame rate conversion, provide the value as a decimal number for Framerate. In this example, specify 23.976.

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

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

hrdBufferInitialFillPercentage

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

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

gopSize

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

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

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

gopBReference

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

Type: H264GopBReference (p. 532)
Required: False

hrdBufferSize

Size of buffer (HRD buffer model) in bits. For example, enter five megabits as 5000000.

Type: integer
Required: False
Minimum: 0
Maximum: 1152000000

maxBitrate

Maximum bitrate in bits/second. For example, enter five megabits per second as 5000000. Required when Rate control mode is QVBR.

Type: integer
Required: False
Minimum: 1000
Maximum: 1152000000

slowPal

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

Type: H264SlowPal (p. 540)
Required: False

parDenominator

Pixel Aspect Ratio denominator.

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

spatialAdaptiveQuantization
Adjust quantization within each frame based on spatial variation of content complexity.
Type: H264SpatialAdaptiveQuantization (p. 540)
Required: False

temporalAdaptiveQuantization
Adjust quantization within each frame based on temporal variation of content complexity.
Type: H264TemporalAdaptiveQuantization (p. 541)
Required: False

flickerAdaptiveQuantization
Adjust quantization within each frame to reduce flicker or ‘pop’ on I-frames.
Type: H264FlickerAdaptiveQuantization (p. 531)
Required: False

entropyEncoding
Entropy encoding mode. Use CABAC (must be in Main or High profile) or CAVLC.
Type: H264EntropyEncoding (p. 531)
Required: False

bitrate
Average bitrate in bits/second. Required for VBR and CBR. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.
Type: integer
Required: False
Minimum: 1000
Maximum: 1152000000

framerateControl
If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do framerate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.
Type: H264FramerateControl (p. 532)
Required: False

rateControlMode

Use this setting to specify whether this output has a variable bitrate (VBR), constant bitrate (CBR) or quality-defined variable bitrate (QVBR).

Type: H264RateControlMode (p. 533)
Required: False

qvbrSettings

Settings for quality-defined variable bitrate encoding with the H.264 codec. Required when you set Rate control mode to QVBR. Not valid when you set Rate control mode to a value other than QVBR, or when you don’t define Rate control mode.

Type: H264QvbrSettings (p. 533)
Required: False

codecProfile

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

Type: H264CodecProfile (p. 531)
Required: False

telecine

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.

Type: H264Telecine (p. 541)
Required: False

framerateNumerator

Frame rate numerator - frame rate is a fraction, e.g. 24000 / 1001 = 23.976 fps.

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

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

538
Properties

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

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

- **Type**: H264AdaptiveQuantization (p. 530)
- **Required**: False

**codecLevel**
Specify an H.264 level that is consistent with your output video settings. If you aren't sure what level to specify, choose Auto (AUTO).

- **Type**: H264CodecLevel (p. 530)
- **Required**: False

**fieldEncoding**
Choosing FORCE_FIELD disables PAFF encoding for interlaced outputs.

- **Type**: H264FieldEncoding (p. 531)
- **Required**: False

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

- **Type**: H264SceneChangeDetect (p. 534)
- **Required**: False

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

- **Type**: H264QualityTuningLevel (p. 533)
- **Required**: False

**framerateConversionAlgorithm**
When set to INTERPOLATE, produces smoother motion during frame rate conversion.

- **Type**: H264FramerateConversionAlgorithm (p. 532)
- **Required**: False

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

- **Type**: H264UnregisteredSeiTimecode (p. 541)
- **Required**: False
**Properties**

**gopSizeUnits**
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.

*Type:* H264GopSizeUnits (p. 532)
*Required:* False

**parControl**
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.

*Type:* H264ParControl (p. 533)
*Required:* False

**numberBFramesBetweenReferenceFrames**
Number of B-frames between reference frames.

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

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

*Type:* H264RepeatPps (p. 534)
*Required:* False

**dynamicSubGop**
Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting B frames between reference frames (numberBFramesBetweenReferenceFrames).

*Type:* H264DynamicSubGop (p. 531)
*Required:* False

**H264SlowPal**
Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

DISABLED
ENABLED

**H264SpatialAdaptiveQuantization**
Adjust quantization within each frame based on spatial variation of content complexity.
DISABLED
ENABLED

**H264Syntax**

Produces a bitstream compliant with SMPTE RP-2027.

DEFAULT
RP2027

**H264Telecine**

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**

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

DISABLED
ENABLED

**H264UnregisteredSeiTimecode**

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

DISABLED
ENABLED

**H265AdaptiveQuantization**

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

OFF
LOW
MEDIUM
HIGH
HIGHER
MAX

**H265AlternateTransferFunctionSei**

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

H265CodecLevel

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

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

H265DynamicSubGop

Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting B frames between reference frames (numberBFramesBetweenReferenceFrames).

ADAPTIVE
STATIC

H265FlickerAdaptiveQuantization

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

DISABLED
ENABLED

542
H265FramerateControl
If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

INITIALIZE_FROM_SOURCE
SPECIFIED

H265FramerateConversionAlgorithm
When set to INTERPOLATE, produces smoother motion during frame rate conversion.

DUPLICATE_DROP
INTERPOLATE

H265GopBReference
If enable, use reference B frames for GOP structures that have B frames > 1.

DISABLED
ENABLED

H265GopSizeUnits
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
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**

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**

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

**H265QvbrSettings**

Settings for quality-defined variable bitrate encoding with the H.265 codec. Required when you set Rate control mode to QVBR. Not valid when you set Rate control mode to a value other than QVBR, or when you don’t define Rate control mode.

**qvbrQualityLevel**

Required when you use QVBR rate control mode. That is, when you specify qvbrSettings within h265Settings. Specify the target quality level for this output, from 1 to 10. Use higher numbers for greater quality. Level 10 results in nearly lossless compression. The quality level for most broadcast-quality transcodes is between 6 and 9.

- **Type:** integer
- **Required:** False
- **Minimum:** 1
- **Maximum:** 10

**maxAverageBitrate**

Use this setting only when Rate control mode is QVBR and Quality tuning level is Multi-pass HQ. For Max average bitrate values suited to the complexity of your input video, the service limits the average bitrate of the video part of this output to the value you choose. That is, the total size of the video element is less than or equal to the value you set multiplied by the number of seconds of encoded output.

- **Type:** integer
- **Required:** False
- **Minimum:** 1000
- **Maximum:** 1466400000

**H265RateControlMode**

Use this setting to specify whether this output has a variable bitrate (VBR), constant bitrate (CBR) or quality-defined variable bitrate (QVBR).
CBR
QVBR

H265SampleAdaptiveOffsetFilterMode

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

- DEFAULT
- ADAPTIVE
- OFF

H265SceneChangeDetect

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

- DISABLED
- ENABLED

H265Settings

Settings for H265 codec

interlaceMode

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.

- **Type**: H265InterlaceMode (p. 543)
- **Required**: False

parNumerator

Pixel Aspect Ratio numerator.

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

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

framerateDenominator
Frame rate denominator.

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

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

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

Type: H265AlternateTransferFunctionSei (p. 541)
Required: False

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

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

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

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

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.
Properties

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

gopBReference

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

Type: H265GopBReference (p. 543)
Required: False

hrdBufferSize

Size of buffer (HRD buffer model) in bits. For example, enter five megabits as 5000000.

Type: integer
Required: False
Minimum: 0
Maximum: 1466400000

maxBitrate

Maximum bitrate in bits/second. For example, enter five megabits per second as 5000000. Required when Rate control mode is QVBR.

Type: integer
Required: False
Minimum: 1000
Maximum: 1466400000

slowPal

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

Type: H265SlowPal (p. 551)
Required: False

parDenominator

Pixel Aspect Ratio denominator.

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

spatialAdaptiveQuantization

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

Type: H265SpatialAdaptiveQuantization (p. 552)
Required: False
temporalAdaptiveQuantization

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

Type: H265TemporalAdaptiveQuantization (p. 552)
Required: False

flickerAdaptiveQuantization

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

Type: H265FlickerAdaptiveQuantization (p. 542)
Required: False

bitrate

Average bitrate in bits/second. Required for VBR and CBR. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.

Type: integer
Required: False
Minimum: 1000
Maximum: 1466400000

framerateControl

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

Type: H265FramerateControl (p. 543)
Required: False

rateControlMode

Use this setting to specify whether this output has a variable bitrate (VBR), constant bitrate (CBR) or quality-defined variable bitrate (QVBR).

Type: H265RateControlMode (p. 544)
Required: False

qvbrSettings

Settings for quality-defined variable bitrate encoding with the H.265 codec. Required when you set Rate control mode to QVBR. Not valid when you set Rate control mode to a value other than QVBR, or when you don't define Rate control mode.

Type: H265QvbrSettings (p. 544)
Required: False
codecProfile

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.

Type: H265CodecProfile (p. 542)
Required: False

tiles

Enable use of tiles, allowing horizontal as well as vertical subdivision of the encoded pictures.

Type: H265Tiles (p. 552)
Required: False

telecine

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.

Type: H265Telecine (p. 552)
Required: False

framerateNumerator

Frame rate numerator - frame rate is a fraction, e.g. 24000 / 1001 = 23.976 fps.

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

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

adaptiveQuantization

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

Type: H265AdaptiveQuantization (p. 541)
Required: False
**CodecLevel**

H.265 Level.

- **Type**: `H265CodecLevel (p. 542)`
- **Required**: False

**SceneChangeDetect**

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

- **Type**: `H265SceneChangeDetect (p. 545)`
- **Required**: False

**QualityTuningLevel**

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

- **Type**: `H265QualityTuningLevel (p. 544)`
- **Required**: False

**FramerateConversionAlgorithm**

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

- **Type**: `H265FramerateConversionAlgorithm (p. 543)`
- **Required**: False

**UnregisteredSeiTimecode**

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

- **Type**: `H265UnregisteredSeiTimecode (p. 552)`
- **Required**: False

**GopSizeUnits**

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.

- **Type**: `H265GopSizeUnits (p. 543)`
- **Required**: False

**ParControl**

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.

- **Type**: `H265ParControl (p. 544)`
- **Required**: False

**NumberBFramesBetweenReferenceFrames**

Number of B-frames between reference frames.
**temporalIds**

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.

- **Type**: H265TemporalIds (p. 552)
- **Required**: False

**sampleAdaptiveOffsetFilterMode**

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

- **Type**: H265SampleAdaptiveOffsetFilterMode (p. 545)
- **Required**: False

**writeMp4PackagingType**

Use this setting only for outputs encoded with H.265 that are in CMAF or DASH output groups. If you include writeMp4PackagingType in your JSON job specification for other outputs, your video might not work properly with downstream systems and video players. If the location of parameter set NAL units don't matter in your workflow, ignore this setting. The service defaults to marking your output as HEV1. Choose HVC1 to mark your output as HVC1. This makes your output compliant with this specification: ISO IECJTC1 SC29 N13798 Text ISO/IEC FDIS 14496-15 3rd Edition. For these outputs, the service stores parameter set NAL units in the sample headers but not in the samples directly. Keep the default HEV1 to mark your output as HEV1. For these outputs, the service writes parameter set NAL units directly into the samples.

- **Type**: H265WriteMp4PackagingType (p. 553)
- **Required**: False

**dynamicSubGop**

Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting B frames between reference frames (numberBFramesBetweenReferenceFrames).

- **Type**: H265DynamicSubGop (p. 542)
- **Required**: False

**H265SlowPal**

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.
H265SpatialAdaptiveQuantization

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

DISABLED
ENABLED

H265Telecine

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

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

DISABLED
ENABLED

H265TemporalIds

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

Enable use of tiles, allowing horizontal as well as vertical subdivision of the encoded pictures.

DISABLED
ENABLED

H265UnregisteredSeiTimecode

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

**DISABLED**
**ENABLED**

**H265WriteMp4PackagingType**

Use this setting only for outputs encoded with H.265 that are in CMAF or DASH output groups. If you include writeMp4PackagingType in your JSON job specification for other outputs, your video might not work properly with downstream systems and video players. If the location of parameter set NAL units don’t matter in your workflow, ignore this setting. The service defaults to marking your output as HEV1. Choose HVC1 to mark your output as HVC1. This makes your output compliant with this specification: ISO IECJTC1 SC29 N13798 Text ISO/IEC FDIS 14496-15 3rd Edition. For these outputs, the service stores parameter set NAL units in the sample headers but not in the samples directly. Keep the default HEV1 to mark your output as HEV1. For these outputs, the service writes parameter set NAL units directly into the samples.

- HVC1
- HEV1

**Hdr10Metadata**

Use the "HDR master display information" (Hdr10Metadata) settings to correct HDR metadata or to provide missing metadata. These values vary depending on the input video and must be provided by a color grader. Range is 0 to 50,000; each increment represents 0.00002 in CIE1931 color coordinate. Note that these settings are not color correction. Note that if you are creating HDR outputs inside of an HLS CMAF package, to comply with the Apple specification, you must use the following settings. Set "MP4 packaging type" (writeMp4PackagingType) to HVC1 (HVC1). Set "Profile" (H265Settings > codecProfile) to Main10/High (MAIN10_HIGH). Set "Level" (H265Settings > codecLevel) to 5 (LEVEL_5).

**redPrimaryX**

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

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

**redPrimaryY**

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

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

**greenPrimaryX**

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.
Properties

**greenPrimaryY**

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

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

**bluePrimaryX**

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

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

**bluePrimaryY**

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

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

**whitePointX**

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

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

**whitePointY**

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.
maxFrameAverageLightLevel
Maximum average light level of any frame in the coded video sequence, in units of candelas per square meter.

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: False
Minimum: 0
Maximum: 65535

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

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

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

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

HlsAdMarkers

- ELEMENTAL
- ELEMENTAL_SCTE35

HlsAudioTrackType
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=NO 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

Caption Language Mapping

captionChannel

Caption channel.

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

customLanguageCode

Specify the language for this caption channel, using the ISO 639-2 or ISO 639-3 three-letter language
code

Type: string
Required: False
Pattern: ^[A-Za-z]{3}$
MinLength: 3
MaxLength: 3

languageCode

Specify the language, using the ISO 639-2 three-letter code listed at https://www.loc.gov/standards/

Type: LanguageCode (p. 578)
Required: False

languageDescription

Caption language description.

Type: string
Required: False

HlsCaptionLanguageSetting

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**

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

| DISABLED | ENABLED |

**HlsCodecSpecification**

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

| RFC_6381 | RFC_4281 |

**HlsDirectoryStructure**

Indicates whether segments should be placed in subdirectories.

| SINGLE_DIRECTORY | SUBDIRECTORY_PER_STREAM |

**HlsEncryptionSettings**

Settings for HLS encryption

**encryptionMethod**

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

*Type:* HlsEncryptionType (p. 558)

*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}$

*MinLength:* 32

*MaxLength:* 32
**InitializationVectorInManifest**

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.

- **Type**: HlsInitializationVectorInManifest (p. 562)
- **Required**: False

**offlineEncrypted**

Enable this setting to insert the EXT-X-SESSION-KEY element into the master playlist. This allows for offline Apple HLS FairPlay content protection.

- **Type**: HlsOfflineEncrypted (p. 563)
- **Required**: False

**spekeKeyProvider**

Settings for use with a SPEKE key provider

- **Type**: SpekeKeyProvider (p. 624)
- **Required**: False

**staticKeyProvider**

Use these settings to set up encryption with a static key provider.

- **Type**: StaticKeyProvider (p. 624)
- **Required**: False

**type**

Indicates which type of key provider is used for encryption.

- **Type**: HlsKeyProviderType (p. 563)
- **Required**: False

**HlsEncryptionType**

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**

Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to HLS_GROUP_SETTINGS.

**manifestDurationFormat**

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

- **Type**: HlsManifestDurationFormat (p. 563)
Properties

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: False
Minimum: 1
Maximum: 2147483647

**timedMetadataId3Period**

Timed Metadata interval in seconds.

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

**captionLanguageSetting**

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.

Type: HlsCaptionLanguageSetting (p. 556)
Required: False

**captionLanguageMappings**

Language to be used on Caption outputs

Type: Array of type HlsCaptionLanguageMapping (p. 556)
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: False
Pattern: ^s3:\/\/

**destinationSettings**

Settings associated with the destination. Will vary based on the type of destination
Type: DestinationSettings (p. 513)
Required: False

encryption
DRM settings.
Type: HlsEncryptionSettings (p. 557)
Required: False

timedMetadataId3Frame
Indicates ID3 frame that has the timecode.
Type: HlsTimedMetadataId3Frame (p. 565)
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

codecSpecification
Specification to use (RFC-6381 or the default RFC-4281) during m3u8 playlist generation.
Type: HlsCodecSpecification (p. 557)
Required: False

outputSelection
Indicates whether the .m3u8 manifest file should be generated for this HLS output group.
Type: HlsOutputSelection (p. 563)
Required: False

programDateTimePeriod
Period of insertion of EXT-X-PROGRAM-DATE-TIME entry, in seconds.
Type: integer
Required: False
Minimum: 0
Maximum: 3600

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
Properties

Required: False
Minimum: 1
Maximum: 2147483647

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: False
Minimum: 0
Maximum: 2147483647

minFinalSegmentLength

Keep this setting at the default value of 0, unless you are troubleshooting a problem with how devices play back the end of your video asset. If you know that player devices are hanging on the final segment of your video because the length of your final segment is too short, use this setting to specify a minimum final segment length, in seconds. Choose a value that is greater than or equal to 1 and less than your segment length. When you specify a value for this setting, the encoder will combine any final segment that is shorter than the length that you specify with the previous segment. For example, your segment length is 3 seconds and your final segment is .5 seconds without a minimum final segment length; when you set the minimum final segment length to 1, your final segment is 3.5 seconds.

Type: number
Required: False
Format: float
Minimum: 0.0
Maximum: 2147483647

directoryStructure

Indicates whether segments should be placed in subdirectories.

Type: HlsDirectoryStructure (p. 557)
Required: False

programDateTime

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.

Type: HlsProgramDateTime (p. 563)
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. 555)
Required: False
segmentControl

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

Type: HlsSegmentControl (p. 563)
Required: False

timestampDeltaMilliseconds

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

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

manifestCompression

When set to GZIP, compresses HLS playlist.

Type: HlsManifestCompression (p. 563)
Required: False

clientCache

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

Type: HlsClientCache (p. 557)
Required: False

streamInfResolution

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

Type: HlsStreamInfResolution (p. 564)
Required: False

HlsIFrameOnlyManifest

When set to INCLUDE, writes I-Frame Only Manifest in addition to the HLS manifest

INCLUDE
EXCLUDE

HlsInitializationVectorInManifest

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**

Indicates which type of key provider is used for encryption.

- SPEKE
- STATIC_KEY

**HlsManifestCompression**

When set to GZIP, compresses HLS playlist.

- GZIP
- NONE

**HlsManifestDurationFormat**

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

- FLOATING_POINT
- INTEGER

**HlsOfflineEncrypted**

Enable this setting to insert the EXT-X-SESSION-KEY element into the master playlist. This allows for offline Apple HLS FairPlay content protection.

- ENABLED
- DISABLED

**HlsOutputSelection**

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

- MANIFESTS_AND_SEGMENTS
- SEGMENTS_ONLY

**HlsProgramDateTime**

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**

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

Settings for HLS output groups

**audioGroupId**

Specifies the group to which the audio Rendition belongs.

- **Type:** string
- **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

**audioTrackType**

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=NO 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

- **Type:** HlsAudioTrackType (p. 555)
- **Required:** False

**iFrameOnlyManifest**

When set to INCLUDE, writes I-Frame Only Manifest in addition to the HLS manifest

- **Type:** HlsIFrameOnlyManifest (p. 562)
- **Required:** False

**segmentModifier**

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

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

HlsStreamInfResolution

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

- **INCLUDE**
- **EXCLUDE**
HlsTimedMetadataId3Frame

Indicates ID3 frame that has the timecode.

- NONE
- PRIV
- TDRL

Id3Insertion

To insert ID3 tags in your output, specify two values. Use ID3 tag (Id3) to specify the base 64 encoded string and use Timecode (TimeCode) to specify the time when the tag should be inserted. To insert multiple ID3 tags in your output, create multiple instances of ID3 insertion (Id3Insertion).

**timecode**

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

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

**id3**

Use ID3 tag (Id3) to provide a tag value in base64-encode format.

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

ImageInserter

Enable the image inserter feature to include a graphic overlay on your video. Enable or disable this feature for each input or output individually. This setting is disabled by default.

**insertableImages**

Specify the images that you want to overlay on your video. The images must be PNG or TGA files.

- **Type:** Array of type InsertableImage (p. 571)
- **Required:** False

Input

Specifies media input

**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.
Properties

**Type**: Array of type `InputClipping (p. 568)`
**Required**: False

**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

**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**: 1
**Maximum**: 2147483647

**videoSelector**

Selector for video.

**Type**: `VideoSelector (p. 635)`
**Required**: False

**filterEnable**

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.

**Type**: `InputFilterEnable (p. 570)`
**Required**: False

**psiControl**

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.

**Type**: `InputPsiControl (p. 570)`
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

**deblockFilter**

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

Type: InputDeblockFilter (p. 569)  
Required: False

**denoiseFilter**

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

Type: InputDenoiseFilter (p. 570)  
Required: False

**timecodeSource**

Timecode source under input settings (InputTimecodeSource) only affects the behavior of features that apply to a single input at a time, such as input clipping and synchronizing some captions formats. Use this setting to specify whether the service counts frames by timecodes embedded in the video (EMBEDDED) or by starting the first frame at zero (ZEROBASED). In both cases, the timecode format is HH:MM:SS:FF or HH:MM:SS;FF, where FF is the frame number. Only set this to EMBEDDED if your source video has embedded timecodes.

Type: InputTimecodeSource (p. 571)  
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

**imageInserter**

Enable the image inserter feature to include a graphic overlay on your video. Enable or disable this feature for each input individually. This setting is disabled by default.

Type: ImageInserter (p. 565)
**Required:** False

**fileInput**

Specify the source file for your transcoding job. You can use multiple inputs in a single job. The service concatenates these inputs, in the order that you specify them in the job, to create the outputs. If your input format is IMF, specify your input by providing the path to your CPL. For example, "s3://bucket/vf/cpl.xml". If the CPL is in an incomplete IMP, make sure to use "Supplemental IMPs" (SupplementalImps) to specify any supplemental IMPs that contain assets referenced by the CPL.

**Type:** string

**decryptionSettings**

Settings for decrypting any input files that you encrypt before you upload them to Amazon S3. MediaConvert can decrypt files only when you use AWS Key Management Service (KMS) to encrypt the data key that you use to encrypt your content.

**Type:** InputDecryptionSettings (p. 569)

**supplementalImps**

Provide a list of any necessary supplemental IMPs. You need supplemental IMPs if the CPL that you're using for your input is in an incomplete IMP. Specify either the supplemental IMP directories with a trailing slash or the ASSETMAP.xml files. For example, "s3://bucket/ov/", "s3://bucket/vf2/ASSETMAP.xml". You don't need to specify the IMP that contains your input CPL, because the service automatically detects it.

**Type:** Array of type string

**InputClipping**

To transcode only portions of your input (clips), include one Input clipping (one instance of InputClipping in the JSON job file) for each input clip. All input clips you specify will be included in every output of the job.

**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. Use the format HH:MM:SS:FF or HH:MM:SS;FF, where HH is the hour, MM is the minute, SS is the second, and FF is the frame number. When choosing this value, take into account your setting for timecode source under input settings (InputTimecodeSource). For example, if you have
embedded timecodes that start at 01:00:00:00 and you want your clip to end six minutes into the video, use 01:06:00:00.

**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. Use the format HH:MM:SS:FF or HH:MM:SS;FF, where HH is the hour, MM is the minute, SS is the second, and FF is the frame number. 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 01:05:00:00.

**InputDeblockFilter**

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

**InputDecryptionSettings**

Settings for decrypting any input files that you encrypt before you upload them to Amazon S3. MediaConvert can decrypt files only when you use AWS Key Management Service (KMS) to encrypt the data key that you use to encrypt your content.

**decryptionMode**

Specify the encryption mode that you used to encrypt your input files.

**encryptedDecryptionKey**

Warning! Don't provide your encryption key in plaintext. Your job settings could be intercepted, making your encrypted content vulnerable. Specify the encrypted version of the data key that you used to encrypt your content. The data key must be encrypted by AWS Key Management Service (KMS). The key can be 128, 192, or 256 bits.
**MaxLength**: 512

**initializationVector**

Specify the initialization vector that you used when you encrypted your content before uploading it to Amazon S3. You can use a 16-byte initialization vector with any encryption mode. Or, you can use a 12-byte initialization vector with GCM or CTR. MediaConvert accepts only initialization vectors that are base64-encoded.

**Type**: string
**Required**: False
**Pattern**: `^[A-Za-z0-9+/]{22}==$|^[A-Za-z0-9+/]{16}$`
**MinLength**: 16
**MaxLength**: 24

**kmsKeyRegion**

Specify the AWS Region for AWS Key Management Service (KMS) that you used to encrypt your data key, if that Region is different from the one you are using for AWS Elemental MediaConvert.

**Type**: string
**Required**: False
**Pattern**: `^[a-z-]{2,6}-(east|west|central|((north|south)(east|west)?)-(1-9}\{1,2}\$`
**MinLength**: 9
**MaxLength**: 19

**InputDenoiseFilter**

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**

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**

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

**InputRotate**

Use Rotate (InputRotate) to specify how the service rotates your video. You can choose automatic rotation or specify a rotation. You can specify a clockwise rotation of 0, 90, 180, or 270 degrees. If your input video container is .mov or .mp4 and your input has rotation metadata, you can choose Automatic to have the service rotate your video according to the rotation specified in the metadata. The rotation must be within one degree of 90, 180, or 270 degrees. If the rotation metadata specifies any other rotation, the service will default to no rotation. By default, the service does no rotation, even if your input video has rotation metadata. The service doesn't pass through rotation metadata.

- DEGREE_0
- DEGREES_90
- DEGREES_180
- DEGREES_270
- AUTO

**InputTimecodeSource**

Timecode source under input settings (InputTimecodeSource) only affects the behavior of features that apply to a single input at a time, such as input clipping and synchronizing some captions formats. Use this setting to specify whether the service counts frames by timecodes embedded in the video (EMBEDDED) or by starting the first frame at zero (ZEROBASED). In both cases, the timecode format is HH:MM:SS:FF or HH:MM:SS;FF, where FF is the frame number. Only set this to EMBEDDED if your source video has embedded timecodes.

- EMBEDDED
- ZEROBASED
- SPECIFIEDSTART

**InsertableImage**

Settings that specify how your still graphic overlay appears.

**width**

Specify the width of the inserted image in pixels. If you specify a value that's larger than the video resolution width, the service will crop your overlaid image to fit. To use the native width of the image, keep this setting blank.

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

**height**

Specify the height of the inserted image in pixels. If you specify a value that's larger than the video resolution height, the service will crop your overlaid image to fit. To use the native height of the image, keep this setting blank.

- **Type**: integer
Properties

Required: False
Minimum: 0
Maximum: 2147483647

imageX

Specify the distance, in pixels, between the inserted image and the left edge of the video frame. Required for any image overlay that you specify.

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

imageY

Specify the distance, in pixels, between the overlaid image and the top edge of the video frame. Required for any image overlay that you specify.

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

duration

Specify the time, in milliseconds, for the image to remain on the output video. This duration includes fade-in time but not fade-out time.

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

fadeIn

Specify the length of time, in milliseconds, between the Start time that you specify for the image insertion and the time that the image appears at full opacity. Full opacity is the level that you specify for the opacity setting. If you don't specify a value for Fade-in, the image will appear abruptly at the overlay start time.

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

layer

Specify how overlapping inserted images appear. Images with higher values for Layer appear on top of images with lower values for Layer.

Type: integer
Required: False
<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Required</th>
<th>Minimum</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>minimum</td>
<td>integer</td>
<td>False</td>
<td>0</td>
<td>99</td>
</tr>
<tr>
<td>maximum</td>
<td>integer</td>
<td>False</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**imageInserterInput**

Specify the Amazon S3 location of the image that you want to overlay on the video. Use a PNG or TGA file.

- **Type**: string
- **Required**: False
- **Pattern**: `^(s3://)(.*)(.bmp|BMP|png|PNG|tga|TGA)$`
- **MinLength**: 14

**startTime**

Specify the timecode of the frame that you want the overlay to first appear on. This must be in timecode (HH:MM:SS:FF or HH:MM:SS;FF) format. Remember to take into account your timecode source settings.

- **Type**: string
- **Required**: False
- **Pattern**: `^(((\[0-1]\d)|(2\[0-3]))(:\[0-5]\d){2}(:;\[0-5]\d))$`

**fadeOut**

Specify the length of time, in milliseconds, between the end of the time that you have specified for the image overlay Duration and when the overlaid image has faded to total transparency. 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**: 0
- **Maximum**: 2147483647

**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**: False
- **Minimum**: 0
- **Maximum**: 100

**Job**

Each job converts an input file into an output file or files. For more information, see the User Guide at http://docs.aws.amazon.com/mediaconvert/latest/ug/what-is.html

**arn**

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

- **Type**: string
Properties

**Required:** False

**id**
A portion of the job's ARN, unique within your AWS Elemental MediaConvert resources

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

**createdAt**
The time, in Unix epoch format in seconds, when the job got created.

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

**jobTemplate**
The job template that the job is created from, if it is created from a job template.

**Type:** string
**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

**userMetadata**
User-defined metadata that you want to associate with an MediaConvert job. You specify metadata in key/value pairs.

**Type:** object
**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

**settings**
JobSettings contains all the transcode settings for a job.

**Type:** JobSettings (p. 576)
Required: True

status
A job's status can be SUBMITTED, PROGRESSING, COMPLETE, CANCELED, or ERROR.

Type: JobStatus (p. 578)
Required: False

code
Error code for the job

Type: integer
Required: False
Format: int32

errorMessage
Error message of Job

Type: string
Required: False

timing
Information about when jobs are submitted, started, and finished is specified in Unix epoch format in seconds.

Type: Timing (p. 629)
Required: False

outputGroupDetails
List of output group details

Type: Array of type OutputGroupDetail (p. 615)
Required: False

billingTagsSource
Optional. Choose a tag type that AWS Billing and Cost Management will use to sort your AWS Elemental MediaConvert costs on any billing report that you set up. Any transcoding outputs that don't have an associated tag will appear in your billing report unsorted. If you don't choose a valid value for this field, your job outputs will appear on the billing report unsorted.

Type: BillingTagsSource (p. 489)
Required: False

accelerationSettings
Accelerated transcoding can significantly speed up jobs with long, visually complex content.

Type: AccelerationSettings (p. 480)
Required: False
Properties

**statusUpdateInterval**

Specify how often MediaConvert sends STATUS_UPDATE events to Amazon CloudWatch Events. Set the interval, in seconds, between status updates. MediaConvert sends an update at this interval from the time the service begins processing your job to the time it completes the transcode or encounters an error.

*Type:* StatusUpdateInterval (p. 625)  
*Required:* False

**jobPercentComplete**

An estimate of how far your job has progressed. This estimate is shown as a percentage of the total time from when your job leaves its queue to when your output files appear in your output Amazon S3 bucket. AWS Elemental MediaConvert provides jobPercentComplete in CloudWatch STATUS_UPDATE events and in the response to GetJob and ListJobs requests. The jobPercentComplete estimate is reliable for the following input containers: Quicktime, Transport Stream, MP4, and MXF. For some jobs, including audio-only jobs and jobs that use input clipping, the service can't provide information about job progress. In those cases, jobPercentComplete returns a null value.

*Type:* integer  
*Required:* False

**currentPhase**

A job's phase can be PROBING, TRANSCODING OR UPLOADING

*Type:* JobPhase (p. 576)  
*Required:* False

**retryCount**

The number of times that the service automatically attempted to process your job after encountering an error.

*Type:* integer  
*Required:* False

**JobPhase**

A job's phase can be PROBING, TRANSCODING OR UPLOADING

- PROBING
- TRANSCODING
- UPLOADING

**JobSettings**

JobSettings contains all the transcode settings for a job.

**timecodeConfig**

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

*Type:* TimecodeConfig (p. 627)
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 * CMAF_GROUP_SETTINGS, CmafGroupSettings

  Type: Array of type OutputGroup (p. 614)
  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

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. 489)
  Required: False

timedMetadataInsertion

Enable Timed metadata insertion (TimedMetadataInsertion) to include ID3 tags in your job. To include timed metadata, you must enable it here, enable it in each output container, and specify tags and timecodes in ID3 insertion (Id3Insertion) objects.

  Type: TimedMetadataInsertion (p. 629)
  Required: False

nielsenConfiguration

Settings for Nielsen Configuration

  Type: NielsenConfiguration (p. 610)
  Required: False

motionImageInserter

Overlay motion graphics on top of your video. The motion graphics that you specify here appear on all outputs in all output groups.

  Type: MotionImageInserter (p. 595)
**Required:** False

**esam**

Settings for Event Signaling And Messaging (ESAM).

**Type:** EsamSettings (p. 526)

**Required:** False

**inputs**

Use Inputs (inputs) to define source file used in the transcode job. There can be multiple inputs add in a job. These inputs will be concatenated together to create the output.

**Type:** Array of type Input (p. 565)

**Required:** False

**JobStatus**

A job's status can be SUBMITTED, PROGRESSING, COMPLETE, CANCELED, or ERROR.

SUBMITTED

PROGRESSING

COMPLETE

CANCELED

ERROR

**LanguageCode**


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
<table>
<thead>
<tr>
<th>Code</th>
<th>Language</th>
</tr>
</thead>
<tbody>
<tr>
<td>HEB</td>
<td>Hebrew</td>
</tr>
<tr>
<td>HER</td>
<td>Herero</td>
</tr>
<tr>
<td>HMO</td>
<td>Mende</td>
</tr>
<tr>
<td>HUN</td>
<td>Hungarian</td>
</tr>
<tr>
<td>ISL</td>
<td>Icelandic</td>
</tr>
<tr>
<td>IDO</td>
<td>Maa</td>
</tr>
<tr>
<td>IBO</td>
<td>Afrikaans</td>
</tr>
<tr>
<td>IND</td>
<td>Indonesian</td>
</tr>
<tr>
<td>INA</td>
<td>Indonesian</td>
</tr>
<tr>
<td>ILE</td>
<td>Ilonggo</td>
</tr>
<tr>
<td>IKU</td>
<td>Ameereka</td>
</tr>
<tr>
<td>IPK</td>
<td>Mbaaka</td>
</tr>
<tr>
<td>GLE</td>
<td>Alemanni</td>
</tr>
<tr>
<td>JAV</td>
<td>Javanese</td>
</tr>
<tr>
<td>KAL</td>
<td>Kalema</td>
</tr>
<tr>
<td>KAN</td>
<td>Kannada</td>
</tr>
<tr>
<td>KAU</td>
<td>Kauri</td>
</tr>
<tr>
<td>KAS</td>
<td>Kazakh</td>
</tr>
<tr>
<td>KAZ</td>
<td>Kazakh</td>
</tr>
<tr>
<td>KIK</td>
<td>Kikuyu</td>
</tr>
<tr>
<td>KIN</td>
<td>Kinande</td>
</tr>
<tr>
<td>KIR</td>
<td>Kirundi</td>
</tr>
<tr>
<td>KOM</td>
<td>Komorense</td>
</tr>
<tr>
<td>KON</td>
<td>Kubu</td>
</tr>
<tr>
<td>KUA</td>
<td>Kwa</td>
</tr>
<tr>
<td>KUR</td>
<td>Kurundi</td>
</tr>
<tr>
<td>LAO</td>
<td>Lao</td>
</tr>
<tr>
<td>LAT</td>
<td>Latvian</td>
</tr>
<tr>
<td>LAV</td>
<td>Lao</td>
</tr>
<tr>
<td>LIM</td>
<td>Limbas</td>
</tr>
<tr>
<td>LIN</td>
<td>Limbese</td>
</tr>
<tr>
<td>LIT</td>
<td>Lithuanian</td>
</tr>
<tr>
<td>LUB</td>
<td>Mbuta</td>
</tr>
<tr>
<td>LTZ</td>
<td>Venda</td>
</tr>
<tr>
<td>MKD</td>
<td>Macedonian</td>
</tr>
<tr>
<td>MLG</td>
<td>Malagasy</td>
</tr>
<tr>
<td>MSA</td>
<td>Arabic</td>
</tr>
<tr>
<td>MAL</td>
<td>Māori</td>
</tr>
<tr>
<td>MLT</td>
<td>Maltese</td>
</tr>
<tr>
<td>GLV</td>
<td>Volapük</td>
</tr>
<tr>
<td>MRI</td>
<td>Mirandese</td>
</tr>
<tr>
<td>MAR</td>
<td>Marathi</td>
</tr>
<tr>
<td>MAH</td>
<td>Māori</td>
</tr>
<tr>
<td>MON</td>
<td>Mongolian</td>
</tr>
<tr>
<td>NAU</td>
<td>Nauruan</td>
</tr>
<tr>
<td>NAV</td>
<td>Navajo</td>
</tr>
<tr>
<td>NDE</td>
<td>Ndebele</td>
</tr>
<tr>
<td>NBL</td>
<td>Ndebele</td>
</tr>
<tr>
<td>NDO</td>
<td>Wolof</td>
</tr>
<tr>
<td>NEP</td>
<td>Nepali</td>
</tr>
<tr>
<td>SME</td>
<td>Samoan</td>
</tr>
<tr>
<td>NOR</td>
<td>Norwegian</td>
</tr>
<tr>
<td>NOB</td>
<td>Norwegian</td>
</tr>
<tr>
<td>NNO</td>
<td>Norfuk</td>
</tr>
<tr>
<td>Code</td>
<td>Description</td>
</tr>
<tr>
<td>------</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>QUE</td>
<td></td>
</tr>
<tr>
<td>QAA</td>
<td></td>
</tr>
<tr>
<td>RON</td>
<td></td>
</tr>
<tr>
<td>ROH</td>
<td></td>
</tr>
<tr>
<td>RUN</td>
<td></td>
</tr>
<tr>
<td>SMO</td>
<td></td>
</tr>
<tr>
<td>SAG</td>
<td></td>
</tr>
<tr>
<td>SAN</td>
<td></td>
</tr>
<tr>
<td>SRD</td>
<td></td>
</tr>
<tr>
<td>SRB</td>
<td></td>
</tr>
<tr>
<td>SNA</td>
<td></td>
</tr>
<tr>
<td>III</td>
<td></td>
</tr>
<tr>
<td>SND</td>
<td></td>
</tr>
<tr>
<td>SIN</td>
<td></td>
</tr>
<tr>
<td>SLK</td>
<td></td>
</tr>
<tr>
<td>SLV</td>
<td></td>
</tr>
<tr>
<td>SOM</td>
<td></td>
</tr>
<tr>
<td>SOT</td>
<td></td>
</tr>
<tr>
<td>SUN</td>
<td></td>
</tr>
<tr>
<td>SWA</td>
<td></td>
</tr>
<tr>
<td>SSW</td>
<td></td>
</tr>
<tr>
<td>SWE</td>
<td></td>
</tr>
<tr>
<td>TGL</td>
<td></td>
</tr>
<tr>
<td>TAH</td>
<td></td>
</tr>
<tr>
<td>TGK</td>
<td></td>
</tr>
<tr>
<td>TAM</td>
<td></td>
</tr>
<tr>
<td>TAT</td>
<td></td>
</tr>
<tr>
<td>TEL</td>
<td></td>
</tr>
<tr>
<td>THA</td>
<td></td>
</tr>
<tr>
<td>BOD</td>
<td></td>
</tr>
<tr>
<td>TIR</td>
<td></td>
</tr>
<tr>
<td>TON</td>
<td></td>
</tr>
<tr>
<td>TSO</td>
<td></td>
</tr>
<tr>
<td>TSN</td>
<td></td>
</tr>
<tr>
<td>TUR</td>
<td></td>
</tr>
<tr>
<td>TUK</td>
<td></td>
</tr>
<tr>
<td>TWI</td>
<td></td>
</tr>
<tr>
<td>UIG</td>
<td></td>
</tr>
<tr>
<td>UKR</td>
<td></td>
</tr>
<tr>
<td>UZB</td>
<td></td>
</tr>
<tr>
<td>VEN</td>
<td></td>
</tr>
<tr>
<td>VOL</td>
<td></td>
</tr>
<tr>
<td>WLN</td>
<td></td>
</tr>
<tr>
<td>CYM</td>
<td></td>
</tr>
<tr>
<td>FRY</td>
<td></td>
</tr>
</tbody>
</table>
ListJobsRequest

You can send list jobs requests with an empty body. Optionally, you can filter the response by queue and/or job status by specifying them in your request body. You can also optionally specify the maximum number, up to twenty, of jobs to be returned.

queue

Provide a queue name to get back only jobs from that queue.

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

status

A job's status can be SUBMITTED, PROGRESSING, COMPLETE, CANCELED, or ERROR.

- **Type:** JobStatus (p. 578)
- **Required:** False

order

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

- **Type:** Order (p. 612)
- **Required:** False

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
- **Minimum:** 1
- **Maximum:** 20
ListJobsResponse

Successful list jobs requests return a JSON array of jobs. If you don't specify how they are ordered, you will receive the most recently created first.

jobs

List of jobs

**Type:** Array of type Job (p. 573)
**Required:** False

nextToken

Use this string to request the next batch of jobs.

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

M2tsAudioBufferModel

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

- **DVB**
- **ATSC**

M2tsBufferModel

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

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 selected (segmentationMarkers is EBP or EBP_LEGACY).

- **VIDEO_AND_FIXED_INTERVALS**
- **VIDEO_INTERVAL**

M2tsEbpPlacement

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 selected (segmentationMarkers is EBP or EBP_LEGACY).

- **VIDEO_AND_AUDIO_PIDS**
VIDEO_PID

M2tsEsRateInPes
Controls whether to include the ES Rate field in the PES header.

  INCLUDE
  EXCLUDE

M2tsForceTsVideoEbpOrder
Keep the default value (DEFAULT) unless you know that your audio EBP markers are incorrectly appearing before your video EBP markers. To correct this problem, set this value to Force (FORCE).

  FORCE
  DEFAULT

M2tsNielsenId3
If INSERT, Nielsen inaudible tones for media tracking will be detected in the input audio and an equivalent ID3 tag will be inserted in the output.

  INSERT
  NONE

M2tsPcrControl
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
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

M2tsScte35Esam
Settings for SCTE-35 signals from ESAM. Include this in your job settings to put SCTE-35 markers in your HLS and transport stream outputs at the insertion points that you specify in an ESAM XML document. Provide the document in the setting SCC XML (sccXml).

  scte35EsamPid
  Packet Identifier (PID) of the SCTE-35 stream in the transport stream generated by ESAM.
Properties

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

M2tsScte35Source

Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output.

PASSTHROUGH  
NONE

M2tsSegmentationMarkers

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

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

MPEG-2 TS container settings. These apply to outputs in a File output group when the output's container (ContainerType) is MPEG-2 Transport Stream (M2TS). In these assets, data is organized by the program map table (PMT). Each transport stream program contains subsets of data, including audio, video, and metadata. Each of these subsets of data has a numerical label called a packet identifier (PID). Each transport stream program corresponds to one MediaConvert output. The PMT lists the types of data in a program along with their PID. Downstream systems and players use the program map table to look up the PID for each type of data it accesses and then uses the PIDs to locate specific data within the asset.
audioBufferModel

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

Type: M2tsAudioBufferModel (p. 583)
Required: False

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: 1000

esRateInPes

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

Type: M2tsEsRateInPes (p. 584)
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

dvbNitSettings

Inserts DVB Network Information Table (NIT) at the specified table repetition interval.

Type: DvbNitSettings (p. 513)
Required: False

dvbSdtSettings

Inserts DVB Service Description Table (NIT) at the specified table repetition interval.

Type: DvbSdtSettings (p. 514)
Required: False

scte35Source

Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output.

Type: M2tsScte35Source (p. 585)
Required: False
scte35Pid

Specify the packet identifier (PID) of the SCTE-35 stream in the transport stream.

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

scte35Esam

Include this in your job settings to put SCTE-35 markers in your HLS and transport stream outputs at the insertion points that you specify in an ESAM XML document. Provide the document in the setting SCC XML (sccXml).

Type: M2tsScte35Esam (p. 584)
Required: False

videoPid

Specify the packet identifier (PID) of the elementary video stream in the transport stream.

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

dvbTdtSettings

Inserts DVB Time and Date Table (TDT) at the specified table repetition interval.

Type: DvbTdtSettings (p. 519)
Required: False

pmtInterval

Specify the number of milliseconds between instances of the program map table (PMT) in the output transport stream.

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

segmentationStyle

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.
**Properties**

**Type**: `M2tsSegmentationStyle (p. 585)`
**Required**: False

**segmentationTime**

Specify the length, in seconds, of each segment. Required unless markers is set to `_none_`.

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

**pmtPid**

Specify the packet identifier (PID) for the program map table (PMT) itself. Default is 480.

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

**bitrate**

Specify 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

**audioPids**

Specify the packet identifiers (PIDs) for any elementary audio streams you include in this output. Specify multiple PIDs as a JSON array. Default is the range 482-492.

**Type**: Array of type integer
**Required**: False
**Minimum**: 32
**Maximum**: 8182

**privateMetadataPid**

Specify the packet identifier (PID) of the private metadata stream. Default is 503.

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

**nielsenId3**

If INSERT, Nielsen inaudible tones for media tracking will be detected in the input audio and an equivalent ID3 tag will be inserted in the output.
Properties

Type: M2tsNielsenId3 (p. 584)
Required: False

timedMetadataPid

Specify the packet identifier (PID) for timed metadata in this output. Default is 502.

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

maxPcrInterval

Specify the maximum time, in milliseconds, between Program Clock References (PCRs) inserted into the transport stream.

Type: integer
Required: False
Minimum: 0
Maximum: 500

transportStreamId

Specify the ID for the transport stream itself in the program map table for this output. Transport stream IDs and program map tables are parts of MPEG-2 transport stream containers, used for organizing data.

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

dvbSubPids

Specify the packet identifiers (PIDs) for DVB subtitle data included in this output. Specify multiple PIDs as a JSON array. Default is the range 460-479.

Type: Array of type integer
Required: False
Minimum: 32
Maximum: 8182

rateMode

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.

Type: M2tsRateMode (p. 584)
Required: False

audioFramesPerPes

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

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

**pcrControl**

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.

- **Type**: M2tsPcrControl (p. 584)  
- **Required**: False

**segmentationMarkers**

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.

- **Type**: M2tsSegmentationMarkers (p. 585)  
- **Required**: False

**ebpAudioInterval**

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).

- **Type**: M2tsEbpAudioInterval (p. 583)  
- **Required**: False

**forceTsVideoEbpOrder**

Keep the default value (DEFAULT) unless you know that your audio EBP markers are incorrectly appearing before your video EBP markers. To correct this problem, set this value to Force (FORCE).

- **Type**: M2tsForceTsVideoEbpOrder (p. 584)  
- **Required**: False

**programNumber**

Use Program number (programNumber) to specify the program number used in the program map table (PMT) for this output. Default is 1. Program numbers and program map tables are parts of MPEG-2 transport stream containers, used for organizing data.

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

Specify the packet identifier (PID) for the program clock reference (PCR) in this output. If you do not specify a value, the service will use the value for Video PID (VideoPid).

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

**bufferModel**

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.

- **Type:** M2tsBufferModel (p. 583)
- **Required:** False

**dvbTeletextPid**

Specify the packet identifier (PID) for DVB teletext data you include in this output. Default is 499.

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

**fragmentTime**

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

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

**ebpPlacement**

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).

- **Type:** M2tsEbpPlacement (p. 583)
- **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

**M3u8NielsenId3**

If INSERT, Nielsen inaudible tones for media tracking will be detected in the input audio and an equivalent ID3 tag will be inserted in the output.

- INSERT
- NONE

**M3u8PcrControl**

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**

Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output.

- PASSTHROUGH
- NONE

**M3u8Settings**

Settings for TS segments in HLS

**audioFramesPerPes**

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

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

**pcrControl**

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.

- **Type:** M3u8PcrControl (p. 592)
- **Required:** False

**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.
Properties

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

**pmtpid**

Packet Identifier (PID) for the Program Map Table (PMT) in the transport stream.

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

**privateMetadataPid**

Packet Identifier (PID) of the private metadata stream in the transport stream.

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

**programNumber**

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

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

**patInterval**

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

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

**pmtInterval**

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

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

**scte35Source**

Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output.
**Type**: `M3u8Scte35Source (p. 592)`  
**Required**: False

**scte35Pid**
Packet Identifier (PID) of the SCTE-35 stream in the transport stream.

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

**nielsenId3**
If `INSERT`, Nielsen inaudible tones for media tracking will be detected in the input audio and an equivalent ID3 tag will be inserted in the output.

- **Type**: `M3u8NielsenId3 (p. 592)`  
- **Required**: False

**timedMetadata**
Applies only to HLS outputs. Use this setting to specify whether the service inserts the ID3 timed metadata from the input in this output.

- **Type**: `TimedMetadata (p. 628)`  
- **Required**: False

**timedMetadataPid**
Packet Identifier (PID) of the timed metadata stream in the transport stream.

- **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.

- **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.

- **Type**: Array of type integer
- **Required**: False
- **Minimum**: 32
- **Maximum**: 8182

MotionImageInserter
Overlay motion graphics on top of your video at the time that you specify.

insertionMode
Choose the type of motion graphic asset that you are providing for your overlay. You can choose either a .mov file or a series of .png files.

- **Type**: MotionImageInsertionMode (p. 597)
- **Required**: False

input
Specify the .mov file or series of .png files that you want to overlay on your video. For .png files, provide the file name of the first file in the series. Make sure that the names of the .png files end with sequential numbers that specify the order that they are played in. For example, overlay_000.png, overlay_001.png, overlay_002.png, and so on. The sequence must start at zero, and each image file name must have the same number of digits. Pad your initial file names with enough zeros to complete the sequence. For example, if the first image is overlay_0.png, there can be only 10 images in the sequence, with the last image being overlay_9.png. But if the first image is overlay_00.png, there can be 100 images in the sequence.

- **Type**: string
- **Required**: False
- **Pattern**: `^(s3://)(.*)(.mov|\.[0-9]+\.png)$`
- **MinLength**: 14
- **MaxLength**: 1285

offset
Use Offset to specify the placement of your motion graphic overlay on the video frame. Specify in pixels, from the upper-left corner of the frame. If you don’t specify an offset, the service scales your overlay to the full size of the frame. Otherwise, the service inserts the overlay at its native resolution and scales the size up or down with any video scaling.

- **Type**: MotionImageInsertionOffset (p. 597)
- **Required**: False

startTime
Specify when the motion overlay begins. Use timecode format (HH:MM:SS:FF or HH:MM:SS;FF). Make sure that the timecode you provide here takes into account how you have set up your timecode configuration under both job settings and input settings. The simplest way to do that is to set both to start at 0. If you need to set up your job to follow timecodes embedded in your source that don’t start
at zero, make sure that you specify a start time that is after the first embedded timecode. For more information, see https://docs.aws.amazon.com/mediaconvert/latest/ug/setting-up-timecode.html

Find job-wide and input timecode configuration settings in your JSON job settings specification at settings>timecodeConfig>source and settings>inputs>timecodeSource.

Type: string
Required: False
Pattern: ^((((0-1)\d)|(2[0-3]))(:[0-5]\d){2}(;\d)?([0-5]\d))$
MinLength: 11
MaxLength: 11

playback

Specify whether your motion graphic overlay repeats on a loop or plays only once.

Type: MotionImagePlayback (p. 597)
Required: False

framerate

If your motion graphic asset is a .mov file, keep this setting unspecified. If your motion graphic asset is a series of .png files, specify the frame rate of the overlay in frames per second, as a fraction. For example, specify 24 fps as 24/1. Make sure that the number of images in your series matches the frame rate and your intended overlay duration. For example, if you want a 30-second overlay at 30 fps, you should have 900 .png images. This overlay frame rate doesn't need to match the frame rate of the underlying video.

Type: MotionImageInsertionFramerate (p. 596)
Required: False

MotionImageInsertionFramerate

For motion overlays that don't have a built-in frame rate, specify the frame rate of the overlay in frames per second, as a fraction. For example, specify 24 fps as 24/1. The overlay frame rate doesn't need to match the frame rate of the underlying video.

framerateNumerator

The top of the fraction that expresses your overlay frame rate. For example, if your frame rate is 24 fps, set this value to 24.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483640

framerateDenominator

The bottom of the fraction that expresses your overlay frame rate. For example, if your frame rate is 24 fps, set this value to 1.

Type: integer
Required: False
Minimum: 1
Maximum: 17895697
MotionImageInsertionMode

Choose the type of motion graphic asset that you are providing for your overlay. You can choose either a .mov file or a series of .png files.

- MOV
- PNG

MotionImageInsertionOffset

Specify the offset between the upper-left corner of the video frame and the top left corner of the overlay.

imageX

Set the distance, in pixels, between the overlay and the left edge of the video frame.

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

imageY

Set the distance, in pixels, between the overlay and the top edge of the video frame.

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

MotionImagePlayback

Specify whether your motion graphic overlay repeats on a loop or plays only once.

- ONCE
- REPEAT

MovClapAtom

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

- INCLUDE
- EXCLUDE

MovCslgAtom

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

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

If set to OMNEON, inserts Omneon-compatible padding

- OMNEON
- NONE

### MovReference

Always keep the default value (SELF_CONTAINED) for this setting.

- SELF_CONTAINED
- EXTERNAL

### MovSettings

Settings for MOV Container.

#### clapAtom

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

- **Type:** MovClapAtom (p. 597)
- **Required:** False

#### cslgAtom

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.

- **Type:** MovCslgAtom (p. 597)
- **Required:** False

#### paddingControl

If set to OMNEON, inserts Omneon-compatible padding

- **Type:** MovPaddingControl (p. 598)
- **Required:** False

#### reference

Always keep the default value (SELF_CONTAINED) for this setting.
Type: MovReference (p. 598)
Required: False

mpeg2FourCCControl

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.

Type: MovMpeg2FourCCControl (p. 598)
Required: False

Mp2Settings

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value MP2.

bitrate

Average bitrate in bits/second.

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

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

sampleRate

Sample rate in hz.

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

Mp4CslgAtom

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**

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

- INCLUDE
- EXCLUDE

**Mp4MoovPlacement**

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**

Settings for MP4 Container

**cslgAtom**

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.

- **Type:** Mp4CslgAtom (p. 599)
- **Required:** False

**freeSpaceBox**

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

- **Type:** Mp4FreeSpaceBox (p. 600)
- **Required:** False

**mp4MajorBrand**

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

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

**moovPlacement**

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.

- **Type:** Mp4MoovPlacement (p. 600)
- **Required:** False

**Mpeg2AdaptiveQuantization**

Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.
Mpeg2CodecLevel

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

- AUTO
- LOW
- MAIN
- HIGH
- MAIN1440
- HIGH

Mpeg2CodecProfile

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

- MAIN
- PROFILE_422

Mpeg2DynamicSubGop

Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting B frames between reference frames (numberBFramesBetweenReferenceFrames).

- ADAPTIVE
- STATIC

Mpeg2FramerateControl

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

- INITIALIZE_FROM_SOURCE
- SPECIFIED

Mpeg2FramerateConversionAlgorithm

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

- DUPLICATE_DROP
**Mpeg2GopSizeUnits**

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**

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**

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**

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**

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

- SINGLE_PASS
MULTI_PASS

**Mpeg2RateControlMode**

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

- VBR
- CBR

**Mpeg2SceneChangeDetect**

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

- DISABLED
- ENABLED

**Mpeg2Settings**

Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value MPEG2.

**interlaceMode**

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.

- **Type:** Mpeg2InterlaceMode (p. 602)
- **Required:** False

**parNumerator**

Pixel Aspect Ratio numerator.

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

**syntax**

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

- **Type:** Mpeg2Syntax (p. 608)
- **Required:** False

**softness**

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

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

framerateDenominator
Frame rate denominator.
Type: integer
Required: False
Minimum: 1
Maximum: 1001

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

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

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

hrdBufferSize
Size of buffer (HRD buffer model) in bits. For example, enter five megabits as 5000000.
Type: integer
Required: False
Minimum: 0
Maximum: 47185920

maxBitrate
Maximum bitrate in bits/second. For example, enter five megabits per second as 5000000.
Properties

**Type**: integer
**Required**: False
**Minimum**: 1000
**Maximum**: 300000000

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

**Type**: Mpeg2SlowPal (p. 608)
**Required**: False

**parDenominator**
Pixel Aspect Ratio denominator.

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

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

**Type**: Mpeg2SpatialAdaptiveQuantization (p. 608)
**Required**: False

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

**Type**: Mpeg2TemporalAdaptiveQuantization (p. 609)
**Required**: False

**bitrate**
Average bitrate in bits/second. Required for VBR and CBR. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.

**Type**: integer
**Required**: False
**Minimum**: 1000
**Maximum**: 288000000

**intraDcPrecision**
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.

**Type**: Mpeg2IntraDcPrecision (p. 602)
**Required**: False
framerateControl

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

  Type: Mpeg2FramerateControl (p. 601)
  Required: False

rateControlMode

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

  Type: Mpeg2RateControlMode (p. 603)
  Required: False

codecProfile

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

  Type: Mpeg2CodecProfile (p. 601)
  Required: False

telecine

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.

  Type: Mpeg2Telecine (p. 608)
  Required: False

framerateNumerator

Frame rate numerator - frame rate is a fraction, e.g. 24000 / 1001 = 23.976 fps.

  Type: integer
  Required: False
  Minimum: 24
  Maximum: 60000

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
**Properties**

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

**adaptiveQuantization**

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

**Type**: Mpeg2AdaptiveQuantization (p. 600)  
**Required**: False

**codecLevel**

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

**Type**: Mpeg2CodecLevel (p. 601)  
**Required**: False

**sceneChangeDetect**

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

**Type**: Mpeg2SceneChangeDetect (p. 603)  
**Required**: False

**qualityTuningLevel**

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

**Type**: Mpeg2QualityTuningLevel (p. 602)  
**Required**: False

**framerateConversionAlgorithm**

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

**Type**: Mpeg2FramerateConversionAlgorithm (p. 601)  
**Required**: False

**gopSizeUnits**

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.

**Type**: Mpeg2GopSizeUnits (p. 602)  
**Required**: False

**parControl**

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.
**Properties**

**Type:** Mpeg2ParControl (p. 602)
**Required:** False

**numberBFramesBetweenReferenceFrames**

Number of B-frames between reference frames.

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

**dynamicSubGop**

Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting B frames between reference frames (numberBFramesBetweenReferenceFrames).

**Type:** Mpeg2DynamicSubGop (p. 601)
**Required:** False

**Mpeg2SlowPal**

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

DISABLED
ENABLED

**Mpeg2SpatialAdaptiveQuantization**

Adjust quantization within each frame based on spatial variation of content complexity.

DISABLED
ENABLED

**Mpeg2Syntax**

Produces a Type D-10 compatible bitstream (SMPTE 356M-2001).

DEFAULT
D_10

**Mpeg2Telecine**

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**

Adjust quantization within each frame based on temporal variation of content complexity.

- DISABLED
- ENABLED

**MsSmoothAudioDeduplication**

COMBINE_DUPLICATE_STREAMS combines identical audio encoding settings across a Microsoft Smooth output group into a single audio stream.

- COMBINE_DUPLICATE_STREAMS
- NONE

**MsSmoothEncryptionSettings**

If you are using DRM, set DRM System (MsSmoothEncryptionSettings) to specify the value SpekeKeyProvider.

**spekeKeyProvider**

Settings for use with a SPEKE key provider

- **Type:** SpekeKeyProvider (p. 624)
- **Required:** False

**MsSmoothGroupSettings**

Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to MS_SMOOTH_GROUP_SETTINGS.

**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:** False
- **Pattern:** ^s3:\\/

**destinationSettings**

Settings associated with the destination. Will vary based on the type of destination

- **Type:** DestinationSettings (p. 513)
- **Required:** False
**fragmentLength**

Use Fragment length (FragmentLength) to specify the mp4 fragment sizes in seconds. Fragment length must be compatible with GOP size and frame rate.

- **Type:** integer
- **Required:** False
- **Minimum:** 1
- **Maximum:** 2147483647

**encryption**

If you are using DRM, set DRM System (MsSmoothEncryptionSettings) to specify the value SpekeKeyProvider.

- **Type:** MsSmoothEncryptionSettings (p. 609)
- **Required:** False

**manifestEncoding**

Use Manifest encoding (MsSmoothManifestEncoding) to specify the encoding format for the server and client manifest. Valid options are utf8 and utf16.

- **Type:** MsSmoothManifestEncoding (p. 610)
- **Required:** False

**audioDeduplication**

COMBINE_DUPLICATE_STREAMS combines identical audio encoding settings across a Microsoft Smooth output group into a single audio stream.

- **Type:** MsSmoothAudioDeduplication (p. 609)
- **Required:** False

**MsSmoothManifestEncoding**

Use Manifest encoding (MsSmoothManifestEncoding) to specify the encoding format for the server and client manifest. Valid options are utf8 and utf16.

- UTF8
- UTF16

**NielsenConfiguration**

Settings for Nielsen Configuration

**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
**distributorId**

Use Distributor ID (DistributorID) to specify the distributor ID that is assigned to your organization by Neilsen.

*Type: string*
*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. When you enable Noise reducer (NoiseReducer), you must also select a value for Noise reducer filter (NoiseReducerFilter).

**filter**

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 a frequency-domain filter based on JND principles.

*Type: NoiseReducerFilter* (p. 611)
*Required: False*

**filterSettings**

Settings for a noise reducer filter

*Type: NoiseReducerFilterSettings* (p. 612)
*Required: False*

**spatialFilterSettings**

Noise reducer filter settings for spatial filter.

*Type: NoiseReducerSpatialFilterSettings* (p. 612)
*Required: False*

**NoiseReducerFilter**

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 a frequency-domain filter based on JND principles.

BILATERAL
MEAN
GAUSSIAN
LANCZOS
SHARPEN
CONSERVE
SPATIAL

**NoiseReducerFilterSettings**

Settings for a noise reducer filter

**strength**

Relative strength of noise reducing filter. Higher values produce stronger filtering.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 3

**NoiseReducerSpatialFilterSettings**

Noise reducer filter settings for spatial filter.

**strength**

Relative strength of noise reducing filter. Higher values produce stronger filtering.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 16

**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

**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

**Order**

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**

An output object describes the settings for a single output file or stream in an output group.

**containerSettings**

Container specific settings.

- **Type**: ContainerSettings (p. 506)
- **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
- **MinLength**: 0

**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. 631)
- **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. 483)
- **Required**: False

**outputSettings**

Specific settings for this type of output.

- **Type**: OutputSettings (p. 617)
- **Required**: False

**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
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 (NameModifier) 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
MinLength: 1

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. 494)
Required: False

OutputChannelMapping

OutputChannel mapping settings.

inputChannels

List of input channels

Type: Array of type integer
Required: False
Minimum: -60
Maximum: 6

OutputDetail

Details regarding output

durationInMs

Duration in milliseconds

Type: integer
Required: False

videoDetails

Contains details about the output's video stream

Type: VideoDetail (p. 634)
Required: False

OutputGroup

Group of outputs
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

name

Name of the output group

  Type: string  
  Required: False

outputs

This object holds groups of encoding settings, one group of settings per output.

  Type: Array of type Output (p. 613)  
  Required: False

outputGroupSettings

Output Group settings, including type

  Type: OutputGroupSettings (p. 615)  
  Required: False

OutputGroupDetail

Contains details about the output groups specified in the job settings.

outputDetails

Details about the output

  Type: Array of type OutputDetail (p. 614)  
  Required: False

OutputGroupSettings

Output Group settings, including type

  Type: OutputGroupType (p. 616)  
  Required: False

hlsGroupSettings

Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to HLS_GROUP_SETTINGS.
**Properties**

**dashIsoGroupSettings**

Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to DASH_ISO_GROUP_SETTINGS.

*Type:* DashIsoGroupSettings (p. 509)  
*Required:* False

**fileGroupSettings**

Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to FILE_GROUP_SETTINGS.

*Type:* FileGroupSettings (p. 528)  
*Required:* False

**msSmoothGroupSettings**

Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to MS_SMOOTH_GROUP_SETTINGS.

*Type:* MsSmoothGroupSettings (p. 609)  
*Required:* False

**cmafGroupSettings**

Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to CMAF_GROUP_SETTINGS. Each output in a CMAF Output Group may only contain a single video, audio, or caption output.

*Type:* CmafGroupSettings (p. 500)  
*Required:* False

**OutputGroupType**

Type of output group (File group, Apple HLS, DASH ISO, Microsoft Smooth Streaming, CMAF)

- HLS_GROUP_SETTINGS
- DASH_ISO_GROUP_SETTINGS
- FILE_GROUP_SETTINGS
- MS_SMOOTH_GROUP_SETTINGS
- CMAF_GROUP_SETTINGS

**OutputSdt**

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*
OutputSettings

Specific settings for this type of output.

**hlsSettings**

Settings for HLS output groups

- **Type**: HlsSettings (p. 564)
- **Required**: False

**ProresCodecProfile**

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**

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

- INITIALIZE_FROM_SOURCE
- SPECIFIED

**ProresFramerateConversionAlgorithm**

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

- DUPLICATE_DROP
- INTERPOLATE

**ProresInterlaceMode**

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**

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**

Required when you set (Codec) under (VideoDescription)> (CodecSettings) to the value PRORES.

**interlaceMode**

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.

- **Type**: ProresInterlaceMode (p. 617)
- **Required**: False

**parNumerator**

Pixel Aspect Ratio numerator.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 2147483647

**framerateDenominator**

Frame rate denominator.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 2147483647
**codecProfile**

Use Profile (ProResCodecProfile) to specify the type of Apple ProRes codec to use for this output.

- **Type:** ProresCodecProfile (p. 617)
- **Required:** False

**slowPal**

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

- **Type:** ProresSlowPal (p. 620)
- **Required:** False

**parDenominator**

Pixel Aspect Ratio denominator.

- **Type:** integer
- **Required:** False
- **Minimum:** 1
- **Maximum:** 2147483647

**framerateControl**

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

- **Type:** ProresFramerateControl (p. 617)
- **Required:** False

**telecine**

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.

- **Type:** ProresTelecine (p. 620)
- **Required:** False

**framerateNumerator**

When you use the API for transcode jobs that use frame rate conversion, specify the frame rate 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

**framerateConversionAlgorithm**

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

- **Type**: ProresFramerateConversionAlgorithm  (p. 617)  
- **Required**: False

**parControl**

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).

- **Type**: ProresParControl  (p. 618)  
- **Required**: False

**ProresSlowPal**

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

- DISABLED
- ENABLED

**ProresTelecine**

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**

Use Rectangle to identify a specific area of the video frame.

**height**

Height of rectangle in pixels. Specify only even numbers.

- **Type**: integer  
- **Required**: False  
- **Minimum**: 2  
- **Maximum**: 2147483647

**width**

Width of rectangle in pixels. Specify only even numbers.
**Type:** integer  
**Required:** False  
**Minimum:** 2  
**Maximum:** 2147483647

**x**

The distance, in pixels, between the rectangle and the left edge of the video frame. Specify only even numbers.

**Type:** integer  
**Required:** False  
**Minimum:** 0  
**Maximum:** 2147483647

**y**

The distance, in pixels, between the rectangle and the top edge of the video frame. Specify only even numbers.

**Type:** integer  
**Required:** False  
**Minimum:** 0  
**Maximum:** 2147483647

**RemixSettings**

Use Manual audio remixing (RemixSettings) to adjust audio levels for each audio channel in each output of your job. With audio remixing, you can output more or fewer audio channels than your input audio source provides.

**channelMapping**

Channel mapping (ChannelMapping) contains the group of fields that hold the remixing value for each channel. Units are in dB. Acceptable values are within the range from -60 (mute) through 6. A setting of 0 passes the input channel unchanged to the output channel (no attenuation or amplification).

**Type:** ChannelMapping (p. 498)  
**Required:** False

**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:** False  
**Minimum:** 1  
**Maximum:** 16

**channelsOut**

Specify the number of channels in this output after remixing. Valid values: 1, 2, 4, 6, 8
**Properties**

**Type**
- integer

**Required**: False

**Minimum**: 1

**Maximum**: 8

**RespondToAfd**

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

**S3DestinationSettings**

Settings associated with S3 destination

**encryption**

Settings for how your job outputs are encrypted as they are uploaded to Amazon S3.

- **Type**: S3EncryptionSettings (p. 622)
- **Required**: False

**S3EncryptionSettings**

Settings for how your job outputs are encrypted as they are uploaded to Amazon S3.

**encryptionType**

Specify how you want your data keys managed. AWS uses data keys to encrypt your content. AWS also encrypts the data keys themselves, using a customer master key (CMK), and then stores the encrypted data keys alongside your encrypted content. Use this setting to specify which AWS service manages the CMK. For simplest set up, choose Amazon S3 (SERVER_SIDE_ENCRYPTION_S3). If you want your master key to be managed by AWS Key Management Service (KMS), choose AWS KMS (SERVER_SIDE_ENCRYPTION_KMS). By default, when you choose AWS KMS, KMS uses the AWS managed customer master key (CMK) associated with Amazon S3 to encrypt your data keys. You can optionally choose to specify a different, customer managed CMK. Do so by specifying the Amazon Resource Name (ARN) of the key for the setting KMS ARN (kmsKeyArn).

- **Type**: S3ServerSideEncryptionType (p. 623)
- **Required**: False

**kmsKeyArn**

Optional, specify the customer master key (CMK) that you want to use to encrypt the data key that AWS uses to encrypt your output content. Enter the Amazon Resource Name (ARN) of the CMK. To use this setting, you must also set Server-side encryption (S3ServerSideEncryptionType) to AWS KMS (SERVER_SIDE_ENCRYPTION_KMS). If you set Server-side encryption to AWS KMS but don’t specify a CMK here, AWS uses the AWS managed CMK associated with Amazon S3.
Type: string
Required: False
Pattern: ^arn:aws(-us-gov)?:kms:[a-z-]{2,6}-(east|west|central|((north|south)(east|west)?)\-)[1-9]{1,2}:\d{12}:key/[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{12}$

S3ServerSideEncryptionType

Specify how you want your data keys managed. AWS uses data keys to encrypt your content. AWS also encrypts the data keys themselves, using a customer master key (CMK), and then stores the encrypted data keys alongside your encrypted content. Use this setting to specify which AWS service manages the CMK. For simplest set up, choose Amazon S3 (SERVER_SIDE_ENCRYPTION_S3). If you want your master key to be managed by AWS Key Management Service (KMS), choose AWS KMS (SERVER_SIDE_ENCRYPTION_KMS). By default, when you choose AWS KMS, KMS uses the AWS managed customer master key (CMK) associated with Amazon S3 to encrypt your data keys. You can optionally choose to specify a different, customer managed CMK. Do so by specifying the Amazon Resource Name (ARN) of the key for the setting KMS ARN (kmsKeyArn).

SERVER_SIDE_ENCRYPTION_S3
SERVER_SIDE_ENCRYPTION_KMS

ScalingBehavior

Applies only if your input aspect ratio is different from your output aspect ratio. Choose "Stretch to output" to have the service stretch your video image to fit. Keep the setting "Default" 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

Set Framerate (SccDestinationFramerate) to make sure that the captions and the video are synchronized in the output. Specify a frame rate that matches the frame rate of the associated video. If the video frame rate 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

Settings for SCC caption output.

framerate

Set Framerate (SccDestinationFramerate) to make sure that the captions and the video are synchronized in the output. Specify a frame rate that matches the frame rate of the associated video. If the video frame rate 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).

**Type:** SccDestinationFramerate (p. 623)  
**Required:** False

### SpekeKeyProvider

Settings for use with a SPEKE key provider

**resourceld**

The SPEKE-compliant server uses Resource ID (ResourceId) to identify content.

**Type:** string  
**Required:** False

**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:** False  
**Pattern:** ^[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}$

**url**

Use URL (Url) to specify the SPEKE-compliant server that will provide keys for content.

**Type:** string  
**Required:** False  
**Format:** uri  
**Pattern:** ^https:/\/

**certificateArn**

Optional AWS Certificate Manager ARN for a certificate to send to the keyprovider. The certificate holds a key used by the keyprovider to encrypt the keys in its response.

**Type:** string  
**Required:** False  
**Pattern:** ^arn:aws(-us-gov)?:acm:

### StaticKeyProvider

Use these settings to set up encryption with a static key provider.

**staticKeyValue**

Relates to DRM implementation. Use a 32-character hexadecimal string to specify Key Value (StaticKeyValue).
**Properties**

**Type**: string  
**Required**: False  
**Pattern**: `^[A-Za-z0-9]{32}$`

**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**: `^([^A-Za-z]{2,6}([A-Za-z0-9-]{1,63})+)$`

**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+)*$`

**url**

Relates to DRM implementation. The location of the license server used for protecting content.

**Type**: string  
**Required**: False  
**Format**: uri

**StatusUpdateInterval**

Specify how often MediaConvert sends STATUS_UPDATE events to Amazon CloudWatch Events. Set the interval, in seconds, between status updates. MediaConvert sends an update at this interval from the time the service begins processing your job to the time it completes the transcode or encounters an error.

- SECONDS_10
- SECONDS_12
- SECONDS_15
- SECONDS_20
- SECONDS_30
- SECONDS_60
- SECONDS_120
- SECONDS_180
- SECONDS_240
- SECONDS_300
- SECONDS_360
- SECONDS_420
- SECONDS_480
- SECONDS_540
- SECONDS_600
TeletextDestinationSettings

Settings for Teletext caption output

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]$
MinLength: 3
MaxLength: 3

TeletextSourceSettings

Settings specific to Teletext caption sources, including Page number.

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]$
MinLength: 3
MaxLength: 3

TimecodeBurnin

Timecode burn-in (TimecodeBurnIn)--Burns the output timecode and specified prefix into the output.

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

Use Position (Position) under under Timecode burn-in (TimecodeBurnIn) to specify the location the burned-in timecode on output video.

Type: TimecodeBurninPosition (p. 627)
Required: False
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: ^[\ -~]+$

TimecodeBurninPosition

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

These settings control how the service handles timecodes throughout the job. These settings don't affect input clipping.

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 frame rate conversion. System behavior for Anchor Timecode varies depending on your setting for Source (TimecodeSource).

* If 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 Source (TimecodeSource) is set to Start at 0 (ZEROBASED) the first frame is 00:00:00:00. * If 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][0-9]:[0-9]{2}$

source

Use Source (TimecodeSource) to set how timecodes are handled within this job. 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.

**Type:** TimecodeSource (p. 628)
**Required:** False

### start

Only use when you set 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}$

### timestampOffset

Only applies to outputs that support program-date-time stamp. Use Timestamp offset (TimestampOffset) to overwrite the timecode date without affecting the time and frame number. Provide the new date as a string in the format "yyyy-mm-dd". To use Time stamp offset, you must also enable Insert program-date-time (InsertProgramDateTime) in the output settings. For example, if the date part of your timecodes is 2002-1-25 and you want to change it to one year later, set Timestamp offset (TimestampOffset) to 2003-1-25.

**Type:** string
**Required:** False
**Pattern:** ^([0-9]{4})-(0[1-9]|1[0-2])-(0[1-9]|[12][0-9]|3[01])$

**TimecodeSource**

Use Source (TimecodeSource) to set how timecodes are handled within this job. 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**

Applies only to HLS outputs. Use this setting to specify whether the service inserts the ID3 timed metadata from the input in this output.

PASSTHROUGH
NONE
**TimedMetadataInsertion**

Enable Timed metadata insertion (TimedMetadataInsertion) to include ID3 tags in your job. To include timed metadata, you must enable it here, enable it in each output container, and specify tags and timecodes in ID3 insertion (Id3Insertion) objects.

**id3Insertions**

Id3Insertions contains the array of Id3Insertion instances.

- **Type:** Array of type Id3Insertion (p. 565)
- **Required:** False

**Timing**

Information about when jobs are submitted, started, and finished is specified in Unix epoch format in seconds.

**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

**finishTime**

The time, in Unix epoch format, that the transcoding job finished.

- **Type:** string
- **Required:** False
- **Format:** date-time

**TrackSourceSettings**

Settings specific to caption sources that are specified by track number. Sources include IMSC in IMF.

**trackNumber**

Use this setting to select a single captions track from a source. Track numbers correspond to the order in the captions source file. For IMF sources, track numbering is based on the order that the captions appear in the CPL. For example, use 1 to select the captions asset that is listed first in the CPL. To include more than one captions track in your job outputs, create multiple input captions selectors. Specify one track per selector.

- **Type:** integer
**Required**: False  
**Minimum**: 1  
**Maximum**: 2147483647  

**TtmlDestinationSettings**  
Settings specific to TTML caption outputs, including Pass style information (TtmlStylePassthrough).

**stylePassthrough**  
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.

- **Type**: TtmlStylePassthrough (p. 630)  
- **Required**: False  

**TtmlStylePassthrough**  
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**  
Type of video codec

- `FRAME_CAPTURE`  
- `H_264`  
- `H_265`  
- `MPEG2`  
- `PRORES`  

**VideoCodecSettings**  
Video codec settings, (CodecSettings) under (VideoDescription), contains the group of settings related to video encoding. The settings in this group vary depending on the value you choose for Video codec (Codec). For each codec enum you choose, define the corresponding settings object. The following lists the codec enum, settings object pairs. * `H_264`, H264Settings * `H_265`, H265Settings * `MPEG2`, Mpeg2Settings * `PRORES`, ProresSettings * `FRAME_CAPTURE`, FrameCaptureSettings  

**codec**  
Specifies the video codec. This must be equal to one of the enum values defined by the object VideoCodec.

- **Type**: VideoCodec (p. 630)  
- **Required**: False  

**frameCaptureSettings**  
Required when you set (Codec) under (VideoDescription)> (CodecSettings) to the value FRAME_CAPTURE.
Type: FrameCaptureSettings (p. 529)  
Required: False

h264Settings
Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value H_264.

Type: H264Settings (p. 534)  
Required: False

h265Settings
Settings for H265 codec

Type: H265Settings (p. 545)  
Required: False

mpeg2Settings
Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value MPEG2.

Type: Mpeg2Settings (p. 603)  
Required: False

proresSettings
Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value PRORES.

Type: ProresSettings (p. 618)  
Required: False

VideoDescription
Settings for video outputs

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

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
**Properties**

**Maximum**: 4096

**scalingBehavior**

Applies only if your input aspect ratio is different from your output aspect ratio. Choose "Stretch to output" to have the service stretch your video image to fit. Keep the setting "Default" to allow the service to letterbox your video instead. This setting overrides any positioning value you specify elsewhere in the job.

Type: ScalingBehavior (p. 623)
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. If you crop your input frame size to smaller than your output frame size, make sure to specify the behavior you want in your output setting "Scaling behavior".

Type: Rectangle (p. 620)
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

**videoPreprocessors**

Find additional transcoding features under Preprocessors (VideoPreprocessors). Enable the features at each output individually. These features are disabled by default.

Type: VideoPreprocessor (p. 634)
Required: False

**timecodeInsertion**

Applies only to H.264, H.265, MPEG2, and ProRes outputs. Only enable Timecode insertion when the input frame rate is identical to the output frame rate. To include timecodes in this output, set Timecode insertion (VideoTimecodeInsertion) to PIC_TIMING_SEI. To leave them out, set it to DISABLED. Default is DISABLED. When the service inserts timecodes in an output, by default, it uses any embedded timecodes from the input. If none are present, the service will set the timecode for the first output frame to zero. To change this default behavior, adjust the settings under Timecode configuration (TimecodeConfig). In the console, these settings are located under Job > Job settings > Timecode configuration. Note - Timecode source under input settings (InputTimecodeSource) does not affect the timecodes that are inserted in the output. Source under Job settings > Timecode configuration (TimecodeSource) does.

Type: VideoTimecodeInsertion (p. 636)
Required: False
antiAlias

The anti-alias filter is automatically applied to all outputs. The service no longer accepts the value DISABLED for AntiAlias. If you specify that in your job, the service will ignore the setting.

Type: AntiAlias (p. 481)
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. 620)
Required: False

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. 0 is the softest setting, 100 the sharpest, and 50 recommended for most content.

Type: integer
Required: False
Minimum: 0
Maximum: 100

codecSettings

Video codec settings, (CodecSettings) under (VideoDescription), contains the group of settings related to video encoding. The settings in this group vary depending on the value you choose for Video codec (Codec). For each codec enum you choose, define the corresponding settings object. The following lists the codec enum, settings object pairs. * H_264, H264Settings * H_265, H265Settings * MPEG2, Mpeg2Settings * PRORES, ProresSettings * FRAME_CAPTURE, FrameCaptureSettings

Type: VideoCodecSettings (p. 630)
Required: False

afdSignaling

This setting only applies to H.264, H.265, and MPEG2 outputs. Use Insert AFD signaling (AfdSignaling) to specify whether the service includes 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.

Type: AfdSignaling (p. 480)
Required: False

dropFrameTimecode

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.
Properties

**Type**: DropFrameTimecode (p. 513)
**Required**: False

**respondToAfd**
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.

**Type**: RespondToAfd (p. 622)
**Required**: False

**colorMetadata**
Enable Insert color metadata (ColorMetadata) to include color metadata in this output. This setting is enabled by default.

**Type**: ColorMetadata (p. 505)
**Required**: False

**VideoDetail**
Contains details about the output's video stream

**widthInPx**
Width in pixels for the output

**Type**: integer
**Required**: False

**heightInPx**
Height in pixels for the output

**Type**: integer
**Required**: False

**VideoPreprocessor**
Find additional transcoding features under Preprocessors (VideoPreprocessors). Enable the features at each output individually. These features are disabled by default.

**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. 504)
**Required**: False
deinterlacer

Use Deinterlacer (Deinterlacer) to produce smoother motion and a clearer picture.

Type: Deinterlacer (p. 512)
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. 565)
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. 611)
Required: False

timecodeBurnin

Timecode burn-in (TimecodeBurnIn)–Burns the output timecode and specified prefix into the output.

Type: TimecodeBurnin (p. 626)
Required: False

VideoSelector

Selector for video.

colorSpace

If your input video has accurate color space metadata, or if you don't know about color space, leave this set to the default value FOLLOW. The service will automatically detect your input color space. If your input video has metadata indicating the wrong color space, or if your input video is missing color space metadata that should be there, specify the accurate color space here. If you choose HDR10, you can also correct inaccurate color space coefficients, using the HDR master display information controls. You must also set Color space usage (ColorSpaceUsage) to FORCE for the service to use these values.

Type: ColorSpace (p. 505)
Required: False

rotate

Use Rotate (InputRotate) to specify how the service rotates your video. You can choose automatic rotation or specify a rotation. You can specify a clockwise rotation of 0, 90, 180, or 270 degrees. If your input video container is .mov or .mp4 and your input has rotation metadata, you can choose Automatic to have the service rotate your video according to the rotation specified in the metadata. The rotation must be within one degree of 90, 180, or 270 degrees. If the rotation metadata specifies any other rotation, the service will default to no rotation. By default, the service does no rotation, even if your input video has rotation metadata. The service doesn't pass through rotation metadata.
**Properties**

**Type**: InputRotate (p. 571)
**Required**: False

**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

**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

**colorSpaceUsage**

There are two sources for color metadata, the input file and the job configuration (in the Color space and HDR master display information settings). The Color space usage setting controls which takes precedence. FORCE: The 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: The 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.

**Type**: ColorSpaceUsage (p. 506)
**Required**: False

**hdr10Metadata**

Use the "HDR master display information" (Hdr10Metadata) settings to correct HDR metadata or to provide missing metadata. These values vary depending on the input video and must be provided by a color grader. Range is 0 to 50,000; each increment represents 0.00002 in CIE1931 color coordinate. Note that these settings are not color correction. Note that if you are creating HDR outputs inside of an HLS CMAF package, to comply with the Apple specification, you must use the following settings. Set "MP4 packaging type" (writeMp4PackagingType) to HVC1 (HVC1). Set "Profile" (H265Settings > codecProfile) to Main10/High (MAIN10_HIGH). Set "Level" (H265Settings > codecLevel) to 5 (LEVEL_5).

**Type**: Hdr10Metadata (p. 553)
**Required**: False

**VideoTimecodeInsertion**

Applies only to H.264, H.265, MPEG2, and ProRes outputs. Only enable Timecode insertion when the input frame rate is identical to the output frame rate. To include timecodes in this output, set Timecode insertion (VideoTimecodeInsertion) to PIC_TIMING_SEI. To leave them out, set it to DISABLED. Default is DISABLED. When the service inserts timecodes in an output, by default, it uses any embedded timecodes.
from the input. If none are present, the service will set the timecode for the first output frame to zero. To change this default behavior, adjust the settings under Timecode configuration (TimecodeConfig). In the console, these settings are located under Job > Job settings > Timecode configuration. Note - Timecode source under input settings (InputTimecodeSource) does not affect the timecodes that are inserted in the output. Source under Job settings > Timecode configuration (TimecodeSource) does.

DISABLED  
PIC_TIMING_SEI

**WavFormat**

The service defaults to using RIFF for WAV outputs. If your output audio is likely to exceed 4 GB in file size, or if you otherwise need the extended support of the RF64 format, set your output WAV file format to RF64.

- RIFF  
- RF64

**WavSettings**

Required when you set (Codec) under (AudioDescriptions)->(CodecSettings) to the value WAV.

**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

**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

**sampleRate**

Sample rate in Hz.

- **Type**: integer  
- **Required**: False  
- **Minimum**: 8000  
- **Maximum**: 192000

**format**

The service defaults to using RIFF for WAV outputs. If your output audio is likely to exceed 4 GB in file size, or if you otherwise need the extended support of the RF64 format, set your output WAV file format to RF64.
Type: WavFormat (p. 637)
Required: False

See Also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

ListJobs

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Go - Pilot
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

CreateJob

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Go - Pilot
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

Jobs id

URI

/2017-08-29/jobs/ld

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. 640)</td>
<td>200 response</td>
</tr>
<tr>
<td>400</td>
<td>ExceptionBody (p. 653)</td>
<td>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>403</td>
<td>ExceptionBody (p. 653)</td>
<td>You don't have permissions for this action with the credentials you sent.</td>
</tr>
<tr>
<td>404</td>
<td>ExceptionBody (p. 653)</td>
<td>The resource you requested does not exist.</td>
</tr>
<tr>
<td>409</td>
<td>ExceptionBody (p. 653)</td>
<td>The service could not complete your request because there is a conflict with the current state of the resource.</td>
</tr>
<tr>
<td>429</td>
<td>ExceptionBody (p. 653)</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>
<tr>
<td>500</td>
<td>ExceptionBody (p. 653)</td>
<td>The service encountered an unexpected condition and cannot fulfill your request.</td>
</tr>
</tbody>
</table>

**DELETE**

Operation ID: CancelJob

Permanently cancel a job. Once you have canceled a job, you can't start it again.

**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>202</td>
<td>CancelJobResponse (p. 653)</td>
<td>202 response</td>
</tr>
</tbody>
</table>
### Status Code

<table>
<thead>
<tr>
<th>Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>400</td>
<td>ExceptionBody (p. 653)</td>
<td>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>403</td>
<td>ExceptionBody (p. 653)</td>
<td>You don't have permissions for this action with the credentials you sent.</td>
</tr>
<tr>
<td>404</td>
<td>ExceptionBody (p. 653)</td>
<td>The resource you requested does not exist.</td>
</tr>
<tr>
<td>409</td>
<td>ExceptionBody (p. 653)</td>
<td>The service could not complete your request because there is a conflict with the current state of the resource.</td>
</tr>
<tr>
<td>429</td>
<td>ExceptionBody (p. 653)</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>
<tr>
<td>500</td>
<td>ExceptionBody (p. 653)</td>
<td>The service encountered an unexpected condition and cannot fulfill your request.</td>
</tr>
</tbody>
</table>

### Schemas

#### Request Bodies

**Example GET**

```json
{
  "id": "string"
}
```

**Example DELETE**

```json
{
  "id": "string"
}
```

#### Response Bodies

**Example GetJobResponse**

```json
{
  "job": {
    "arn": "string",
    "id": "string",
    "createdAt": "string",
    "jobTemplate": "string",
    "queue": "string",
```
"userMetadata": {
},
"role": "string",
"settings": {
"timecodeConfig": {
"anchor": "string",
"source": enum,
"start": "string",
"timestampOffset": "string"
},
"outputGroups": [
{
"customName": "string",
"name": "string",
"outputs": [
{
"containerSettings": {
"container": enum,
"m3u8Settings": {
"audioFramesPerPes": integer,
"pcrControl": enum,
"pcrPid": integer,
"pmtPid": integer,
"privateMetadataPid": integer,
"programNumber": integer,
"patInterval": integer,
"pmtInterval": integer,
"scte35Source": enum,
"scte35Pid": integer,
"nielsenId3": enum,
"timedMetadata": enum,
"timedMetadataPid": integer,
"transportStreamId": integer,
"videoPid": integer,
"audioPids": [integer]
},
"f4vSettings": {
"moovPlacement": enum
},
"m2tsSettings": {
"audioBufferModel": enum,
"minEbPiInterval": integer,
"esRateInPes": enum,
"patInterval": integer,
"dvbNitSettings": {
"nitInterval": integer,
"networkId": integer,
"networkName": "string"
},
"dvbSdtSettings": {
"outputSdt": enum,
"sdtInterval": integer,
"serviceName": "string",
"serviceProviderName": "string"
},
"scte35Source": enum,
"scte35Pid": integer,
"scte35Esam": {
"scte35EsamPid": integer
},
"videoPid": integer,
"dvbTdtSettings": {
"tdtInterval": integer
}]}]}
"pmtInterval": integer,
"segmentationStyle": enum,
"segmentationTime": number,
"pmtPid": integer,
"bitrate": integer,
"audioPids": [
  integer
],
"privateMetadataPid": integer,
"nielsenId3": enum,
"timedMetadataPid": integer,
"maxPcrInterval": integer,
"transportStreamId": integer,
"dvbSubPids": [
  integer
],
"rateMode": enum,
"audioFramesPerPes": integer,
"pcrControl": enum,
"segmentationMarkers": enum,
"ebpAudioInterval": enum,
"forceToVideoEbpOrder": enum,
"programNumber": integer,
"pcrPid": integer,
"bufferModel": enum,
"dvbTeletextPid": integer,
"fragmentTime": number,
"ebpPlacement": enum,
"nullPacketBitrate": number
},
"movSettings": {
  "clapAtom": enum,
  "cslgAtom": enum,
  "paddingControl": enum,
  "reference": enum,
  "mpeg2FourCCControl": enum
},
"mp4Settings": {
  "cslgAtom": enum,
  "freeSpaceBox": enum,
  "mp4MajorBrand": "string",
  "moovPlacement": enum
}
},
"preset": "string",
"videoDescription": {
  "fixedAfd": integer,
  "width": integer,
  "scalingBehavior": enum,
  "crop": {
    "height": integer,
    "width": integer,
    "x": integer,
    "y": integer
  },
  "height": integer,
  "videoPreprocessors": {
    "colorCorrector": {
      "brightness": integer,
      "colorSpaceConversion": enum,
      "contrast": integer,
      "hue": integer,
      "saturation": integer,
      "hdr10Metadata": {
        "redPrimaryX": integer,
        "redPrimaryY": integer,
"greenPrimaryX": integer,
"greenPrimaryY": integer,
"bluePrimaryX": integer,
"bluePrimaryY": integer,
"whitePointX": integer,
"whitePointY": integer,
"maxFrameAverageLightLevel": integer,
"maxContentLightLevel": integer,
"maxLuminance": integer,
"minLuminance": integer
}
},
"deinterlacer": {
  "algorithm": enum,
  "mode": enum,
  "control": enum
},
"imageInserter": {
  "insertableImages": [
    {
      "width": integer,
      "height": integer,
      "imageX": integer,
      "imageY": integer,
      "duration": integer,
      "fadeIn": integer,
      "layer": integer,
      "imageInserterInput": "string",
      "startTime": "string",
      "fadeOut": integer,
      "opacity": integer
    }
  ]
},
"noiseReducer": {
  "filter": enum,
  "filterSettings": {
    "strength": integer
  },
  "spatialFilterSettings": {
    "strength": integer,
    "speed": integer,
    "postFilterSharpenStrength": integer
  }
},
"timecodeBurnin": {
  "fontSize": integer,
  "position": enum,
  "prefix": "string"
},
"timecodeInsertion": enum,
"antiAlias": enum,
"position": {
  "height": integer,
  "width": integer,
  "x": integer,
  "y": integer
},
"sharpness": integer,
"codecSettings": {
  "codec": enum,
  "frameCaptureSettings": {
    "framerateNumerator": integer,
    "framerateDenominator": integer,
    "maxCaptures": integer
  }
}
"quality": integer
},
"h264Settings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "numberReferenceFrames": integer,
  "syntax": enum,
  "softness": integer,
  "framerateDenominator": integer,
  "gopClosedCadence": integer,
  "hrdBufferInitialFillPercentage": integer,
  "gopSize": number,
  "slices": integer,
  "gopBReference": enum,
  "hrdBufferSize": integer,
  "maxBitrate": integer,
  "slowPal": enum,
  "parDenominator": integer,
  "spatialAdaptiveQuantization": enum,
  "temporalAdaptiveQuantization": enum,
  "flickerAdaptiveQuantization": enum,
  "entropyEncoding": enum,
  "bitrate": integer,
  "framerateControl": enum,
  "rateControlMode": enum,
  "qvbrSettings": {
    "qvbrQualityLevel": integer,
    "maxAverageBitrate": integer
  },
  "codecProfile": enum,
  "telecine": enum,
  "framerateNumerator": integer,
  "minIInterval": integer,
  "adaptiveQuantization": enum,
  "codecLevel": enum,
  "fieldEncoding": enum,
  "sceneChangeDetect": enum,
  "qualityTuningLevel": enum,
  "framerateConversionAlgorithm": enum,
  "unregisteredSeiTimecode": enum,
  "gopSizeUnits": enum,
  "parControl": enum,
  "numberBFramesBetweenReferenceFrames": integer,
  "repeatPps": enum,
  "dynamicSubGop": enum
},
"h265Settings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "numberReferenceFrames": integer,
  "framerateDenominator": integer,
  "gopClosedCadence": integer,
  "alternateTransferFunctionSei": enum,
  "hrdBufferInitialFillPercentage": integer,
  "gopSize": number,
  "slices": integer,
  "gopBReference": enum,
  "hrdBufferSize": integer,
  "maxBitrate": integer,
  "slowPal": enum,
  "parDenominator": integer,
  "spatialAdaptiveQuantization": enum,
  "temporalAdaptiveQuantization": enum,
  "flickerAdaptiveQuantization": enum,
  "bitrate": integer,
  "framerateControl": enum,
"rateControlMode": enum,
"qvbrSettings": {
  "qvbrQualityLevel": integer,
  "maxAverageBitrate": integer
},
"codecProfile": enum,
"tiles": enum,
"telecine": enum,
"framerateNumerator": integer,
"minIInterval": integer,
"adaptiveQuantization": enum,
"codecLevel": enum,
"sceneChangeDetect": enum,
"qualityTuningLevel": enum,
"framerateConversionAlgorithm": enum,
"unregisteredSeiTimecode": enum,
"gopSizeUnits": enum,
"parControl": enum,
"numberBFramesBetweenReferenceFrames": integer,
"temporalIds": enum,
"sampleAdaptiveOffsetFilterMode": enum,
"writeMp4PackagingType": enum,
"dynamicSubGop": enum
},
"mpeg2Settings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "syntax": enum,
  "softness": integer,
  "framerateDenominator": integer,
  "gopClosedCadence": integer,
  "hrdBufferInitialFillPercentage": integer,
  "gopSize": number,
  "hrdBufferSize": integer,
  "maxBitrate": integer,
  "intraDcPrecision": enum,
  "framerateControl": enum,
  "telecine": enum,
  "framerateNumerator": integer,
  "minIInterval": integer,
  "adaptiveQuantization": enum,
  "codecLevel": enum,
  "sceneChangeDetect": enum,
  "qualityTuningLevel": enum,
  "framerateConversionAlgorithm": enum,
  "gopSizeUnits": enum,
  "parControl": enum,
  "numberBFramesBetweenReferenceFrames": integer,
  "dynamicSubGop": enum
},
"proresSettings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "framerateDenominator": integer,
  "codecProfile": enum,
  "slowPal": enum,
  "parDenominator": integer,
  "spatialAdaptiveQuantization": enum,
  "temporalAdaptiveQuantization": enum,
  "bitrate": integer,
  "intraDcPrecision": enum,
  "framerateControl": enum,
  "telecine": enum,
  "framerateNumerator": integer,
  "minIInterval": integer,
  "adaptiveQuantization": enum,
  "codecLevel": enum,
  "sceneChangeDetect": enum,
  "qualityTuningLevel": enum,
  "framerateConversionAlgorithm": enum,
  "gopSizeUnits": enum,
  "parControl": enum,
  "numberBFramesBetweenReferenceFrames": integer,
  "dynamicSubGop": enum
}
"framerateNumerator": integer,
"framerateConversionAlgorithm": enum,
"parControl": enum
}
],
"afdsignaling": enum,
"dropFrameTimecode": enum,
"respondToAf": enum,
"colorMetadata": enum
],
"audioDescriptions": [
{
 "audioTypeControl": enum,
 "audioSourceName": "string",
 "audioNormalizationSettings": {
  "algorithm": enum,
  "algorithmControl": enum,
  "correctionGateLevel": integer,
  "loudnessLogging": enum,
  "targetLkfs": number,
  "peakCalculation": enum
  },
  "codecSettings": {
  "codec": enum,
  "aacSettings": {
  "audioDescriptionBroadcasterMix": enum,
  "vbrrQuality": enum,
  "bitrate": integer,
  "rateControlMode": enum,
  "codecProfile": enum,
  "codingMode": enum,
  "rawFormat": enum,
  "sampleRate": integer,
  "specification": enum
  },
  "ac3Settings": {
  "bitrate": integer,
  "bitstreamMode": enum,
  "codingMode": enum,
  "dialnorm": integer,
  "dynamicRangeCompressionProfile": enum,
  "metadataControl": enum,
  "lfeFilter": enum,
  "sampleRate": integer
  },
  "aiffSettings": {
  "bitDepth": integer,
  "channels": integer,
  "sampleRate": integer
  },
  "eac3Settings": {
  "metadataControl": enum,
  "surroundexMode": enum,
  "loRoSurroundMixLevel": number,
  "phaseControl": enum,
  "dialnorm": integer,
  "lrltSurroundMixLevel": number,
  "bitrate": integer,
  "lrltCenterMixLevel": number,
  "passthroughControl": enum,
  "lfeControl": enum,
  "loRoCenterMixLevel": number,
  "attenuationControl": enum,
  "codingMode": enum,
  "surroundMode": enum,
  "bitstreamMode": enum,
"lfeFilter": enum,
"stereoDownmix": enum,
"dynamicRangeCompressionRf": enum,
"sampleRate": integer,
"dynamicRangeCompressionLine": enum,
"dcFilter": enum
},
"mp2Settings": {
"bitrate": integer,
"channels": integer,
"sampleRate": integer
},
"wavSettings": {
"bitDepth": integer,
"channels": integer,
"sampleRate": integer,
"format": enum
},
"remixSettings": {
"channelMapping": {
"outputChannels": [
{"inputChannels": [integer
]
}
],
"channelsIn": integer,
"channelsOut": integer
},
"streamName": "string",
"languageCodeControl": enum,
"audioType": integer,
"customLanguageCode": "string",
"languageCode": enum
}
],
"outputSettings": {
"hlsSettings": {
"audioGroupId": "string",
"audioRenditionSets": "string",
"audioTrackType": enum,
"iFrameOnlyManifest": enum,
"segmentModifier": "string"
}
},
"extension": "string",
"nameModifier": "string",
"captionDescriptions": [
{
"captionSelectorName": "string",
"destinationSettings": {
"destinationType": enum,
"burninDestinationSettings": {
"backgroundOpacity": integer,
"shadowXOffset": integer,
"teletextSpacing": enum,
"alignment": enum,
"outlineSize": integer,
"yPosition": integer,
"shadowColor": enum,
"fontOpacity": integer,
"fontSize": integer,
"fontScript": enum,
"fontColor": enum,
"backgroundColor": enum,
"fontResolution": integer,
"outlineColor": enum,
"shadowYOffset": integer,
"xPosition": integer,
"shadowOpacity": integer
},
"dvbSubDestinationSettings": {
"backgroundOpacity": integer,
"shadowXOffset": integer,
"teletextSpacing": enum,
"alignment": enum,
"outlineSize": integer,
"yPosition": integer,
"shadowColor": enum,
"fontOpacity": integer,
"fontSize": integer,
"fontScript": enum,
"fontColor": enum,
"backgroundColor": enum,
"fontResolution": integer,
"outlineColor": enum,
"shadowYOffset": integer,
"xPosition": integer,
"shadowOpacity": integer
},
"sccDestinationSettings": {
"framerate": enum
},
"teletextDestinationSettings": {
"pageNumber": "string"
},
"ttmlDestinationSettings": {
"stylePassthrough": enum
},
"embeddedDestinationSettings": {
"destination608ChannelNumber": integer
},
"customLanguageCode": "string",
"languageCode": enum,
"languageDescription": "string"

],
"outputGroupSettings": {
"type": enum,
"hlsGroupSettings": {
"manifestDurationFormat": enum,
"segmentLength": integer,
"timedMetadataId3Period": integer,
"captionLanguageSetting": enum,
"captionLanguageMappings": [
{
"captionChannel": integer,
"customLanguageCode": "string",
"languageCode": enum,
"languageDescription": "string"
}
],
"destination": "string",
"destinationSettings": {
"s3Settings": {
"encryption": {
""fontColor": enum,
"backgroundColor": enum,
"fontResolution": integer,
"outlineColor": enum,
"shadowYOffset": integer,
"xPosition": integer,
"shadowOpacity": integer
},
"dvbSubDestinationSettings": {
"backgroundOpacity": integer,
"shadowXOffset": integer,
"teletextSpacing": enum,
"alignment": enum,
"outlineSize": integer,
"yPosition": integer,
"shadowColor": enum,
"fontOpacity": integer,
"fontSize": integer,
"fontScript": enum,
"fontColor": enum,
"backgroundColor": enum,
"fontResolution": integer,
"outlineColor": enum,
"shadowYOffset": integer,
"xPosition": integer,
"shadowOpacity": integer
},
"sccDestinationSettings": {
"framerate": enum
},
"teletextDestinationSettings": {
"pageNumber": "string"
},
"ttmlDestinationSettings": {
"stylePassthrough": enum
},
"embeddedDestinationSettings": {
"destination608ChannelNumber": integer
},
"customLanguageCode": "string",
"languageCode": enum,
"languageDescription": "string"}
"encryptionType": enum,
  "kmsKeyArn": "string"
}
}

"encryption": {
  "encryptionMethod": enum,
  "constantInitializationVector": "string",
  "initializationVectorInManifest": enum,
  "offlineEncrypted": enum,
  "spekeKeyProvider": {
    "resourceId": "string",
    "systemIds": [
      "string"
    ],
    "url": "string",
    "certificateArn": "string"
  },
  "staticKeyProvider": {
    "staticKeyValue": "string",
    "keyFormat": "string",
    "keyFormatVersions": "string",
    "url": "string"
  },
  "type": enum
},
"timedMetadataId3Frame": enum,
"baseUrl": "string",
"codecSpecification": enum,
"outputSelection": enum,
"programDateTimePeriod": integer,
"segmentsPerSubdirectory": integer,
"minSegmentLength": integer,
"minFinalSegmentLength": number,
"directoryStructure": enum,
"programDateTime": enum,
"adMarkers": [
  enum
],
"segmentControl": enum,
"timestampDeltaMilliseconds": integer,
"manifestCompression": enum,
"clientCache": enum,
"streamInfResolution": enum
},
"dashIsoGroupSettings": {
  "segmentLength": integer,
  "destination": "string",
  "destinationSettings": {
    "s3Settings": {
      "encryption": {
        "encryptionType": enum,
        "kmsKeyArn": "string"
      }
    },
  },
  "encryption": {
    "spekeKeyProvider": {
      "resourceId": "string",
      "systemIds": [
        "string"
      ],
      "url": "string",
      "certificateArn": "string"
    }
  }
},
"minBufferTime": integer,
"fragmentLength": integer,
"baseUrl": "string",
"segmentControl": enum,
"hbbtvCompliance": enum,
"writeSegmentTimelineInRepresentation": enum
},
"fileGroupSettings": {
  "destination": "string",
  "destinationSettings": {
    "s3Settings": {
      "encryption": {
        "encryptionType": enum,
        "kmsKeyArn": "string"
      }
    }
  }
},
"msSmoothGroupSettings": {
  "destination": "string",
  "destinationSettings": {
    "s3Settings": {
      "encryption": {
        "encryptionType": enum,
        "kmsKeyArn": "string"
      }
    }
  },
  "fragmentLength": integer,
  "encryption": {
    "spekeKeyProvider": {
      "resourceId": "string",
      "systemIds": [
        "string"
      ],
      "url": "string",
      "certificateArn": "string"
    }
  },
  "manifestEncoding": enum,
  "audioDeduplication": enum
},
"cmafGroupSettings": {
  "writeHlsManifest": enum,
  "writeDashManifest": enum,
  "segmentLength": integer,
  "minFinalSegmentLength": number,
  "destination": "string",
  "destinationSettings": {
    "s3Settings": {
      "encryption": {
        "encryptionType": enum,
        "kmsKeyArn": "string"
      }
    }
  },
  "encryption": {
    "encryptionMethod": enum,
    "constantInitializationVector": "string",
    "initializationVectorInManifest": enum,
    "staticKeyProvider": {
      "staticKeyValue": "string",
      "keyFormat": "string",
      "keyFormatVersions": "string",
      "url": "string"
    }
  }
}
"type": enum,
"minBufferTime": integer,
"fragmentLength": integer,
"baseUrl": "string",
"segmentControl": enum,
"manifestDurationFormat": enum,
"streamInfResolution": enum,
"clientCache": enum,
"manifestCompression": enum,
"codecSpecification": enum
}
]
",
"adAvailOffset": integer,
"availBlanking": {
    "availBlankingImage": "string"
},
"timedMetadataInsertion": {
    "id3Insertions": [
        {
            "timecode": "string",
            "id3": "string"
        }
    ]
},
"nielsenConfiguration": {
    "breakoutCode": integer,
    "distributorId": "string"
},
"motionImageInserter": {
    "insertionMode": enum,
    "input": "string",
    "offset": {
        "imageX": integer,
        "imageY": integer
    },
    "startTime": "string",
    "playback": enum,
    "framerate": {
        "framerateNumerator": integer,
        "framerateDenominator": integer
    }
},
"esam": {
    "signalProcessingNotification": {
        "sccXml": "string"
    },
    "manifestConfirmConditionNotification": {
        "mccXml": "string"
    },
    "responseSignalPreroll": integer
},
"inputs": [
    {
        "inputClippings": [
            {
                "endTimecode": "string",
                "startTimecode": "string"
            }
        ],
        "audioSelectors": {
        },
        "audioSelectorGroups": {
        }
    }
]
"programNumber": integer,
"videoSelector": {  
  "colorSpace": enum,
  "rotate": enum,
  "pid": integer,
  "programNumber": integer,
  "colorSpaceUsage": enum,
  "hdr10Metadata": { 
    "redPrimaryX": integer,
    "redPrimaryY": integer,
    "greenPrimaryX": integer,
    "greenPrimaryY": integer,
    "bluePrimaryX": integer,
    "bluePrimaryY": integer,
    "whitePointX": integer,
    "whitePointY": integer,
    "maxFrameAverageLightLevel": integer,
    "maxContentLightLevel": integer,
    "maxLuminance": integer,
    "minLuminance": integer
  },
  "filterEnable": enum,
  "psiControl": enum,
  "filterStrength": integer,
  "deblockFilter": enum,
  "denoiseFilter": enum,
  "timecodeSource": enum,
  "captionSelectors": { 
  },
  "imageInserter": { 
    "insertableImages": [ 
      { 
        "width": integer,
        "height": integer,
        "imageX": integer,
        "imageY": integer,
        "duration": integer,
        "fafeed": integer,
        "layer": integer,
        "imageInserterInput": "string",
        "startTime": "string",
        "fadeOut": integer,
        "opacity": integer
      } 
    ],
    "fileInput": "string",
    "decryptionSettings": { 
      "decryptionMode": enum,
      "encryptedDecryptionKey": "string",
      "initializationVector": "string",
      "kmsKeyRegion": "string"
    },
    "supplementalImps": [ 
      "string"
    ]
  },
  "status": enum,
  "errorCode": integer,
  "errorMessage": "string",
  "timing": { 
    "submitTime": "string",
    "startTime": "string",
    "endTime": "string",
    "duration": integer,
    "pendingDuration": integer
  }
}


"finishTime": "string",
"outputGroupDetails": [
  {
    "outputDetails": [
      {
        "durationInMs": integer,
        "videoDetails": {
          "widthInPx": integer,
          "heightInPx": integer
        }
      }
    ]
  }
],
"billingTagsSource": enum,
"accelerationSettings": {
  "mode": enum
},
"statusUpdateInterval": enum,
"jobPercentComplete": integer,
"currentPhase": enum,
"retryCount": integer
}
}

Example CancelJobResponse

{
  
}

Example ExceptionBody

{
  "message": "string"
}

Properties

AacAudioDescriptionBroadcasterMix

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

AAC Profile.
LC
HEV1
HEV2

**AacCodingMode**

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**

Rate Control Mode.

- **CBR**
- **VBR**

**AacRawFormat**

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**

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value AAC. The service accepts one of two mutually exclusive groups of AAC settings--VBR and CBR. To select one of these modes, set the value of Bitrate control mode (rateControlMode) to "VBR" or "CBR". In VBR mode, you control the audio quality with the setting VBR quality (vbrQuality). In CBR mode, you use the setting Bitrate (bitrate). Defaults and valid values depend on the rate control mode.

**audioDescriptionBroadcasterMix**

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.

*Type: AacAudioDescriptionBroadcasterMix (p. 653)*
**Required**: False

**vbrQuality**

VBR Quality Level - Only used if rate_control_mode is VBR.

- **Type**: AacVbrQuality (p. 656)
- **Required**: False

**bitrate**

Average bitrate in bits/second. The set of valid values for this setting is: 6000, 8000, 10000, 12000, 14000, 16000, 20000, 24000, 28000, 32000, 40000, 48000, 56000, 64000, 80000, 96000, 112000, 128000, 160000, 192000, 224000, 256000, 288000, 320000, 384000, 448000, 512000, 576000, 640000, 768000, 896000, 1024000. The value you set is also constrained by the values you choose for Profile (codecProfile), Bitrate control mode (codingMode), and Sample rate (sampleRate). Default values depend on Bitrate control mode and Profile.

- **Type**: integer
  - **Required**: False
  - **Minimum**: 6000
  - **Maximum**: 1024000

**rateControlMode**

Rate Control Mode.

- **Type**: AacRateControlMode (p. 654)
- **Required**: False

**codecProfile**

AAC Profile.

- **Type**: AacCodecProfile (p. 653)
- **Required**: False

**codingMode**

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.

- **Type**: AacCodingMode (p. 654)
- **Required**: False

**rawFormat**

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.

- **Type**: AacRawFormat (p. 654)
- **Required**: False
sampleRate

Sample rate in Hz. Valid values depend on rate control mode and profile.

Type: integer
Required: False
Minimum: 8000
Maximum: 96000

specification

Use MPEG-2 AAC instead of MPEG-4 AAC audio for raw or MPEG-2 Transport Stream containers.

Type: AacSpecification (p. 656)
Required: False

AacSpecification

Use MPEG-2 AAC instead of MPEG-4 AAC audio for raw or MPEG-2 Transport Stream containers.

MPEG2
MPEG4

AacVbrQuality

VBR Quality Level - Only used if rate_control_mode is VBR.

LOW
MEDIUM_LOW
MEDIUM_HIGH
HIGH

Ac3BitstreamMode

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

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**

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**

Applies a 120Hz lowpass filter to the LFE channel prior to encoding. Only valid with 3_2_LFE coding mode.

ENABLED
DISABLED

**Ac3MetadataControl**

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**

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value AC3.

**bitrate**

Average bitrate in bits/second. Valid bitrates depend on the coding mode.

Type: integer
Required: False
Minimum: 64000
Maximum: 640000

**bitstreamMode**

Specifies the "Bitstream Mode" (bsmod) for the emitted AC-3 stream. See ATSC A/52-2012 for background on these values.

Type: Ac3BitstreamMode (p. 656)
Required: False

**codingMode**

Dolby Digital coding mode. Determines number of channels.
**Type**: Ac3CodingMode (p. 656)  
**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

**dynamicRangeCompressionProfile**

If set to FILM_STANDARD, adds dynamic range compression signaling to the output bitstream as defined in the Dolby Digital specification.

**Type**: Ac3DynamicRangeCompressionProfile (p. 657)  
**Required**: False

**metadataControl**

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.

**Type**: Ac3MetadataControl (p. 657)  
**Required**: False

**lfeFilter**

Applies a 120Hz lowpass filter to the LFE channel prior to encoding. Only valid with 3_2_LFE coding mode.

**Type**: Ac3LfeFilter (p. 657)  
**Required**: False

**sampleRate**

Sample rate in hz. Sample rate is always 48000.

**Type**: integer  
**Required**: False  
**Minimum**: 48000  
**Maximum**: 48000

**AccelerationMode**

Enable Acceleration (AccelerationMode) on any job that you want processed with accelerated transcoding.

**DISABLED**
Properties

**ENABLED**

**AccelerationSettings**

Accelerated transcoding can significantly speed up jobs with long, visually complex content. Outputs that use this feature incur pro-tier pricing. For information about feature limitations, see the AWS Elemental MediaConvert User Guide.

**mode**

Acceleration configuration for the job.

- **Type**: AccelerationMode (p. 658)
- **Required**: True

**AfdSignaling**

This setting only applies to H.264, H.265, and MPEG2 outputs. Use Insert AFD signaling (AfdSignaling) to specify whether the service includes 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**

Required when you set (Codec) under (AudioDescriptions)>({CodecSettings}) to the value AIFF.

**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

**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

**sampleRate**

Sample rate in hz.
Properties

**Type**: integer
**Required**: False
**Minimum**: 8000
**Maximum**: 192000

**AncillarySourceSettings**

Settings for ancillary captions source.

**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**

The anti-alias filter is automatically applied to all outputs. The service no longer accepts the value DISABLED for AntiAlias. If you specify that in your job, the service will ignore the setting.

- DISABLED
- ENABLED

**AudioCodec**

Type of Audio codec.

- AAC
- MP2
- WAV
- AIFF
- AC3
- EAC3
- PASSTHROUGH

**AudioCodecSettings**

Audio codec settings (CodecSettings) under (AudioDescriptions) contains the group of settings related to audio encoding. The settings in this group vary depending on the value you choose for Audio codec (Codec). For each codec enum you choose, define the corresponding settings object. The following lists the codec enum, settings object pairs. * AAC, AacSettings * MP2, Mp2Settings * WAV, WavSettings * AIFF, AiffSettings * AC3, Ac3Settings * EAC3, Eac3Settings

**codec**

Type of Audio codec.

**Type**: AudioCodec (p. 660)
Required: False

aacSettings

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value AAC. The service accepts one of two mutually exclusive groups of AAC settings--VBR and CBR. To select one of these modes, set the value of Bitrate control mode (rateControlMode) to "VBR" or "CBR". In VBR mode, you control the audio quality with the setting VBR quality (vbrQuality). In CBR mode, you use the setting Bitrate (bitrate). Defaults and valid values depend on the rate control mode.

Type: AacSettings (p. 654)
Required: False

ac3Settings

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value AC3.

Type: Ac3Settings (p. 657)
Required: False

aiffSettings

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value AIFF.

Type: AiffSettings (p. 659)
Required: False

eac3Settings

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value EAC3.

Type: Eac3Settings (p. 699)
Required: False

mp2Settings

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value MP2.

Type: Mp2Settings (p. 775)
Required: False

wavSettings

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value WAV.

Type: WavSettings (p. 814)
Required: False

AudioDefaultSelection

Enable this setting on one audio selector to set it as the default for the job. The service uses this default for outputs where it can't find the specified input audio. If you don't set a default, those outputs have no audio.
AudioDescription

Description of audio output

audioTypeControl

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.

Type: AudioTypeControl (p. 668)
Required: False

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

audioNormalizationSettings

Advanced audio normalization settings.

Type: AudioNormalizationSettings (p. 664)
Required: False

codecSettings

Audio codec settings (CodecSettings) under (AudioDescriptions) contains the group of settings related to audio encoding. The settings in this group vary depending on the value you choose for Audio codec (Codec). For each codec enum you choose, define the corresponding settings object. The following lists the codec enum, settings object pairs. * AAC, AacSettings * MP2, Mp2Settings * WAV, WavSettings * AIFF, AiffSettings * AC3, Ac3Settings * EAC3, Eac3Settings

Type: AudioCodecSettings (p. 660)
Required: False

remixSettings

Advanced audio remixing settings.

Type: RemixSettings (p. 798)
Required: False
streamName

Used for MS Smooth 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: ^[^\s\w]*$

languageCodeControl

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.

Type: AudioLanguageCodeControl (p. 663)
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

customLanguageCode

Specify the language for this audio output track, using the ISO 639-2 or ISO 639-3 three-letter language code. The language specified 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: string
Required: False
Pattern: ^[A-Za-z]{3}$
MinLength: 3
MaxLength: 3

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. 756)
Required: False

AudioLanguageCodeControl

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**

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**

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**

If set to LOG, log each output's audio track loudness to a CSV file.

- LOG
- DONT_LOG

**AudioNormalizationPeakCalculation**

If set to TRUE_PEAK, calculate and log the TruePeak for each output's audio track loudness.

- TRUE_PEAK
- NONE

**AudioNormalizationSettings**

Advanced audio normalization settings.

**algorithm**

Audio normalization algorithm to use. 1770-1 conforms to the CALM Act specification, 1770-2 conforms to the EBU R-128 specification.

- **Type**: AudioNormalizationAlgorithm (p. 664)
- **Required**: False

**algorithmControl**

When enabled the output audio is corrected using the chosen algorithm. If disabled, the audio will be measured but not adjusted.
Properties

**Type**: AudioNormalizationAlgorithmControl (p. 664)

**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

### loudnessLogging

If set to LOG, log each output's audio track loudness to a CSV file.

**Type**: AudioNormalizationLoudnessLogging (p. 664)

**Required**: False

### 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

### peakCalculation

If set to TRUE_PEAK, calculate and log the TruePeak for each output's audio track loudness.

**Type**: AudioNormalizationPeakCalculation (p. 664)

**Required**: False

---

**AudioSelector**

Selector for Audio

### tracks

Identify a track from the input audio to include in this selector by entering the track index number. To include several tracks in a single audio selector, specify multiple tracks as follows. Using the console, enter a comma-separated list. For example, type "1,2,3" to include tracks 1 through 3. Specifying directly in your JSON job file, provide the track numbers in an array. For example, "tracks": [1,2,3].

**Type**: Array of type integer

**Required**: False

**Minimum**: 1

**Maximum**: 2147483647
offset

Specifies a time delta in milliseconds to offset the audio from the input video.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

defaultSelection

Enable this setting on one audio selector to set it as the default for the job. The service uses this default for outputs where it can’t find the specified input audio. If you don’t set a default, those outputs have no audio.

Type: AudioDefaultSelection (p. 661)
Required: False

selectorType

Specifies the type of the audio selector.

Type: AudioSelectorType (p. 667)
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
Minimum: 1
Maximum: 2147483647

externalAudioFileInput

Specifies audio data from an external file source.

Type: string
Required: False
Pattern:
^s3:\/\/[a-zA-Z0-9]+(\/\/[a-zA-Z0-9]+)*\/.\/.\/.*\([mM][2-4]([vV]|pP)|[mM][0-9]+([vV]|pP)|[mM][kK][vV]|[mM][oO][vV]|[mM][tT][sS]|[mM][2-4][tT]|wW|mM[vV]|aA][sS][fF]|VV|oO][bB]|3[gG][pP]|3[mM][xX][fF]|dD][iI][vV]|xX][xX][vV]|iI][dD]|rR][aA][wW]|dD][vV]|gG][xX][fF]|mM][1][vV]|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]|aA][aA][cC]|aA][iI][fF]|fF|mM][pP][2]|aA][cC]|eE][cC]|3][dD][tT][sS][eE]])$

programSelection

Use this setting for input streams that contain Dolby E, to have the service extract specific program data from the track. To select multiple programs, create multiple selectors with the same Track and different Program numbers. In the console, this setting is visible when you set Selector type to Track. Choose the
program number from the dropdown list. If you are sending a JSON file, provide the program ID, which is part of the audio metadata. If your input file has incorrect metadata, you can choose All channels instead of a program number to have the service ignore the program IDs and include all the programs in the track.

**Type:** integer  
**Required:** False  
**Minimum:** 0  
**Maximum:** 8

### customLanguageCode

Selects a specific language code from within an audio source, using the ISO 639-2 or ISO 639-3 three-letter language code

**Type:** string  
**Required:** False  
**Pattern:** ^[A-Za-z]{3}$  
**MinLength:** 3  
**MaxLength:** 3

### languageCode

Selects a specific language code from within an audio source.

**Type:** LanguageCode (p. 756)  
**Required:** False

### remixSettings

Use these settings to reorder the audio channels of one input to match those of another input. This allows you to combine the two files into a single output, one after the other.

**Type:** RemixSettings (p. 798)  
**Required:** False

### AudioSelectorGroup

Group of Audio Selectors

### audioSelectorNames

Name of an 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:** False  
**MinLength:** 1

### AudioSelectorType

Specifies the type of the audio selector.
Properties

PID
TRACK
LANGUAGE_CODE

AudioTypeControl

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

Settings for Avail Blanking

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)$
MinLength: 14

BillingTagsSource

Optional. Choose a tag type that AWS Billing and Cost Management will use to sort your AWS Elemental MediaConvert costs on any billing report that you set up. Any transcoding outputs that don't have an associated tag will appear in your billing report unsorted. If you don't choose a valid value for this field, your job outputs will appear on the billing report unsorted.

QUEUE
PRESET
JOB_TEMPLATE

BurninDestinationSettings

Burn-In Destination Settings.

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
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

**teletextSpacing**

Only applies to jobs with input captions in Teletext or STL formats. Specify whether the spacing between letters in your captions is set by the captions grid or varies depending on letter width. Choose fixed grid to conform to the spacing specified in the captions file more accurately. Choose proportional to make the text easier to read if the captions are closed caption.

- **Type**: BurninSubtitleTeletextSpacing (p. 673)
- **Required**: False

**alignment**

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.

- **Type**: BurninSubtitleAlignment (p. 672)
- **Required**: False

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**: False
- **Minimum**: 0
- **Maximum**: 10

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

**shadowColor**

Specifies the color of the shadow cast by the captions. All burn-in and DVB-Sub font settings must match.

- **Type**: BurninSubtitleShadowColor (p. 672)
- **Required**: False

**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**: False
- **Minimum**: 0
- **Maximum**: 255

**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

**fontScript**

Provide the font script, using an ISO 15924 script code, if the LanguageCode is not sufficient for determining the script type. Where LanguageCode or CustomLanguageCode is sufficient, use “AUTOMATIC” or leave unset. This is used to help determine the appropriate font for rendering burn-in captions.

- **Type**: FontScript (p. 706)
- **Required**: False

**fontColor**

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.

- **Type**: BurninSubtitleFontColor (p. 672)
- **Required**: False

**backgroundColor**

Specifies the color of the rectangle behind the captions. All burn-in and DVB-Sub font settings must match.
**Properties**

**Type**

**BurninSubtitleBackgroundColor** (p. 672)

**Required**: False

**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

**outlineColor**

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.

**Type**: **BurninSubtitleOutlineColor** (p. 672)

**Required**: False

**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

**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

**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
BurninSubtitleAlignment

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

Specifies the color of the rectangle behind the captions. All burn-in and DVB-Sub font settings must match.

- NONE
- BLACK
- WHITE

BurninSubtitleFontColor

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

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

Specifies the color of the shadow cast by the captions. All burn-in and DVB-Sub font settings must match.

- NONE
BLACK
WHITE

**BurninSubtitleTeletextSpacing**

Only applies to jobs with input captions in Teletext or STL formats. Specify whether the spacing between letters in your captions is set by the captions grid or varies depending on letter width. Choose fixed grid to conform to the spacing specified in the captions file more accurately. Choose proportional to make the text easier to read if the captions are closed caption.

FIXED_GRID
PROPORTIONAL

**CancelJobRequest**

Cancel a job by sending a request with the job ID

'id'

The Job ID of the job to be cancelled.

*Type:* string
*Required:* False

**CancelJobResponse**

A cancel job request will receive a response with an empty body.

**CaptionDescription**

Description of Caption output

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:* False
*MinLength:* 1

**destinationSettings**

Specific settings required by destination type. Note that burnin_destination_settings are not available if the source of the caption data is Embedded or Teletext.

*Type:* CaptionDestinationSettings (p. 674)
*Required:* False

**customLanguageCode**

Indicates the language of the caption output track, using the ISO 639-2 or ISO 639-3 three-letter language code. For most captions output formats, the encoder puts this language information in the
output captions metadata. If your output captions format is DVB-Sub or Burn in, the encoder uses this language information to choose the font language for rendering the captions text.

- **Type**: string
  - **Required**: False
  - **Pattern**: ^[A-Za-z]{3}$
  - **MinLength**: 3
  - **MaxLength**: 3

**languageCode**

Specify the language of this captions output track. For most captions output formats, the encoder puts this language information in the output captions metadata. If your output captions format is DVB-Sub or Burn in, the encoder uses this language information to choose the font language for rendering the captions text.

- **Type**: LanguageCode (p. 756)
  - **Required**: False

**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

**CaptionDestinationSettings**

Specific settings required by destination type. Note that burnin_destination_settings are not available if the source of the caption data is Embedded or Teletext.

**destinationType**

Specify the format for this set of captions on this output. The default format is embedded without SCTE-20. Other options are embedded with SCTE-20, burn-in, DVB-sub, SCC, SRT, teletext, TTML, and web-VTT. If you are using SCTE-20, choose SCTE-20 plus embedded (SCTE20_PLUS_EMBEDDED) to create an output that complies with the SCTE-43 spec. To create a non-compliant output where the embedded captions come first, choose Embedded plus SCTE-20 (EMBEDDED_PLUS_SCTE20).

- **Type**: CaptionDestinationType (p. 675)
  - **Required**: False

**burninDestinationSettings**

Burn-In Destination Settings.

- **Type**: BurninDestinationSettings (p. 668)
  - **Required**: False

**dvbSubDestinationSettings**

DVB-Sub Destination Settings

- **Type**: DvbSubDestinationSettings (p. 692)
Properties

**Required**: False

**sccDestinationSettings**

Settings for SCC caption output.

*Type*: SccDestinationSettings (p. 800)
*Required*: False

**teletextDestinationSettings**

Settings for Teletext caption output

*Type*: TeletextDestinationSettings (p. 802)
*Required*: False

**ttmlDestinationSettings**

Settings specific to TTML caption outputs, including Pass style information (TtmlStylePassthrough).

*Type*: TtmlDestinationSettings (p. 806)
*Required*: False

**embeddedDestinationSettings**

Settings specific to embedded/ancillary caption outputs, including 608/708 Channel destination number.

*Type*: EmbeddedDestinationSettings (p. 703)
*Required*: False

**CaptionDestinationType**

Specify the format for this set of captions on this output. The default format is embedded without SCTE-20. Other options are embedded with SCTE-20, burn-in, DVB-sub, SCC, SRT, teletext, TTML, and web-VTT. If you are using SCTE-20, choose SCTE-20 plus embedded (SCTE20_PLUS_EMBEDDED) to create an output that complies with the SCTE-43 spec. To create a non-compliant output where the embedded captions come first, choose Embedded plus SCTE-20 (EMBEDDED_PLUS_SCTE20).

- BURN_IN
- DVB_SUB
- EMBEDDED
- EMBEDDED_PLUS_SCTE20
- SCTE20_PLUS_EMBEDDED
- SCC
- SRT
- SMI
- TELETEXT
- TTML
- WEBVTT

**CaptionSelector**

Set up captions in your outputs by first selecting them from your input here.
customLanguageCode

The specific language to extract from source, using the ISO 639-2 or ISO 639-3 three-letter language code. 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: string
Required: False
Pattern: ^[A-Za-z]{3}$
MinLength: 3
MaxLength: 3

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. 756)
Required: False

sourceSettings

Source settings (SourceSettings) contains the group of settings for captions in the input.

Type: CaptionSourceSettings (p. 676)
Required: False

CaptionSourceSettings

Source settings (SourceSettings) contains the group of settings for captions in the input.

sourceType

Use Source (SourceType) to identify the format of your input captions. The service cannot auto-detect caption format.

Type: CaptionSourceType (p. 677)
Required: False

ancillarySourceSettings

Settings for ancillary captions source.

Type: AncillarySourceSettings (p. 660)
Required: False

dvbSubSourceSettings

DVB Sub Source Settings
**Properties**

**Type**: DvbSubSourceSettings (p. 695)
**Required**: False

**embeddedSourceSettings**

Settings for embedded captions Source

**Type**: EmbeddedSourceSettings (p. 703)
**Required**: False

**fileSourceSettings**

Settings for File-based Captions in Source

**Type**: FileSourceSettings (p. 706)
**Required**: False

**teletextSourceSettings**

Settings specific to Teletext caption sources, including Page number.

**Type**: TeletextSourceSettings (p. 803)
**Required**: False

**trackSourceSettings**

Settings specific to caption sources that are specified by track number. Sources include IMSC in IMF.

**Type**: TrackSourceSettings (p. 806)
**Required**: False

**CaptionSourceType**

Use Source (SourceType) to identify the format of your input captions. The service cannot auto-detect caption format.

ANCILLARY
DVB_SUB
EMBEDDED
SCTE20
SCC
TTML
STL
SRT
SMI
TELETEXT
NULL_SOURCE
IMSC

**ChannelMapping**

Channel mapping (ChannelMapping) contains the group of fields that hold the remixing value for each channel. Units are in dB. Acceptable values are within the range from -60 (mute) through 6. A setting of 0 passes the input channel unchanged to the output channel (no attenuation or amplification).
outputChannels

List of output channels

  Type: Array of type OutputChannelMapping (p. 791)
  Required: False

CmafClientCache

When set to ENABLED, sets #EXT-X-ALLOW-CACHE:no tag, which prevents client from saving media segments for later replay.

  DISABLED
  ENABLED

CmafCodecSpecification

Specification to use (RFC-6381 or the default RFC-4281) during m3u8 playlist generation.

  RFC_6381
  RFC_4281

CmafEncryptionSettings

Settings for CMAF encryption

encryptionMethod

Encrypts the segments with the given encryption scheme. Leave blank to disable. Selecting 'Disabled' in the web interface also disables encryption.

  Type: CmafEncryptionType (p. 679)
  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}$
  MinLength: 32
  MaxLength: 32

initializationVectorInManifest

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.

  Type: CmafInitializationVectorInManifest (p. 682)
**staticKeyProvider**

Use these settings to set up encryption with a static key provider.

*Type:* StaticKeyProvider (p. 801)
*Required:* False

**type**

Indicates which type of key provider is used for encryption.

*Type:* CmafKeyProviderType (p. 682)
*Required:* False

**CmafEncryptionType**

Encrypts the segments with the given encryption scheme. Leave blank to disable. Selecting 'Disabled' in the web interface also disables encryption.

SAMPLE_AES

**CmafGroupSettings**

Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to CMAF_GROUP_SETTINGS. Each output in a CMAF Output Group may only contain a single video, audio, or caption output.

**writeHlsManifest**

When set to ENABLED, an Apple HLS manifest will be generated for this output.

*Type:* CmafWriteHLSManifest (p. 683)
*Required:* False

**writeDashManifest**

When set to ENABLED, a DASH MPD manifest will be generated for this output.

*Type:* CmafWriteDASHManifest (p. 683)
*Required:* False

**segmentLength**

Use this setting to specify the length, in seconds, of each individual CMAF segment. This value applies to the whole package; that is, to every output in the output group. Note that segments end on the first keyframe after this number of seconds, so the actual segment length might be slightly longer. If you set Segment control (CmafSegmentControl) to single file, the service puts the content of each output in a single file that has metadata that marks these segments. If you set it to segmented files, the service creates multiple files for each output, each with the content of one segment.

*Type:* integer
**Properties**

**minFinalSegmentLength**

Keep this setting at the default value of 0, unless you are troubleshooting a problem with how devices play back the end of your video asset. If you know that player devices are hanging on the final segment of your video because the length of your final segment is too short, use this setting to specify a minimum final segment length, in seconds. Choose a value that is greater than or equal to 1 and less than your segment length. When you specify a value for this setting, the encoder will combine any final segment that is shorter than the length that you specify with the previous segment. For example, your segment length is 3 seconds and your final segment is .5 seconds without a minimum final segment length; when you set the minimum final segment length to 1, your final segment is 3.5 seconds.

- **Type**: number
- **Required**: False
- **Format**: float
- **Minimum**: 0.0
- **Maximum**: 2147483647

**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**: False
- **Pattern**: ^s3://

**destinationSettings**

Settings associated with the destination. Will vary based on the type of destination

- **Type**: DestinationSettings (p. 690)
- **Required**: False

**encryption**

DRM settings.

- **Type**: CmafEncryptionSettings (p. 678)
- **Required**: False

**minBufferTime**

Minimum time of initially buffered media that is needed to ensure smooth playout.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 2147483647
**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:** False
- **Minimum:** 1
- **Maximum:** 2147483647

**baseUrl**

A partial URI prefix that will be put in the manifest 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

**segmentControl**

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.

- **Type:** CmafSegmentControl (p. 682)
- **Required:** False

**manifestDurationFormat**

Indicates whether the output manifest should use floating point values for segment duration.

- **Type:** CmafManifestDurationFormat (p. 682)
- **Required:** False

**streamInfResolution**

Include or exclude RESOLUTION attribute for video in EXT-X-STREAM-INF tag of variant manifest.

- **Type:** CmafStreamInfResolution (p. 682)
- **Required:** False

**clientCache**

When set to ENABLED, sets #EXT-X-ALLOW-CACHE:no tag, which prevents client from saving media segments for later replay.

- **Type:** CmafClientCache (p. 678)
- **Required:** False

**manifestCompression**

When set to GZIP, compresses HLS playlist.
Type: CmafManifestCompression (p. 682)
Required: False

codecSpecification
Specification to use (RFC-6381 or the default RFC-4281) during m3u8 playlist generation.
Type: CmafCodecSpecification (p. 678)
Required: False

CmafInitializationVectorInManifest
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

CmafKeyProviderType
Indicates which type of key provider is used for encryption.

STATIC_KEY

CmafManifestCompression
When set to GZIP, compresses HLS playlist.

GZIP
NONE

CmafManifestDurationFormat
Indicates whether the output manifest should use floating point values for segment duration.

FLOATING_POINT
INTEGER

CmafSegmentControl
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

CmafStreamInfResolution
Include or exclude RESOLUTION attribute for video in EXT-X-STREAM-INF tag of variant manifest.
CmafWriteDASHManifest

When set to ENABLED, a DASH MPD manifest will be generated for this output.

DISABLED
ENABLED

CmafWriteHLSManifest

When set to ENABLED, an Apple HLS manifest will be generated for this output.

DISABLED
ENABLED

ColorCorrector

Settings for color correction.

brightness

Brightness level.

Type: integer
Required: False
Minimum: 1
Maximum: 100

colorSpaceConversion

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.

Type: ColorSpaceConversion (p. 684)
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

**saturation**

Saturation level.

Type: integer
Required: False
Minimum: 1
Maximum: 100

**hdr10Metadata**

Use the HDR master display (Hdr10Metadata) settings to correct HDR metadata or to provide missing metadata. Note that these settings are not color correction.

Type: Hdr10Metadata (p. 731)
Required: False

**ColorMetadata**

Enable insert color metadata (ColorMetadata) to include color metadata in this output. This setting is enabled by default.

IGNORE
INSERT

**ColorSpace**

If your input video has accurate color space metadata, or if you don't know about color space, leave this set to the default value FOLLOW. The service will automatically detect your input color space. If your input video has metadata indicating the wrong color space, or if your input video is missing color space metadata that should be there, specify the accurate color space here. If you choose HDR10, you can also correct inaccurate color space coefficients, using the HDR master display information controls. You must also set Color space usage (ColorSpaceUsage) to FORCE for the service to use these values.

FOLLOW
REC_601
REC_709
HDR10
HLG_2020

**ColorSpaceConversion**

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

There are two sources for color metadata, the input file and the job configuration (in the Color space and HDR master display information settings). The Color space usage setting controls which takes precedence. FORCE: The 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: The 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 specific settings.

container

Container for this output. Some containers require a container settings object. If not specified, the default object will be created.

Type: ContainerType (p. 686)
Required: False

m3u8Settings

Settings for TS segments in HLS

Type: M3u8Settings (p. 769)
Required: False

f4vSettings

Settings for F4v container

Type: F4vSettings (p. 705)
Required: False

m2tsSettings

MPEG-2 TS container settings. These apply to outputs in a File output group when the output's container (ContainerType) is MPEG-2 Transport Stream (M2TS). In these assets, data is organized by the program map table (PMT). Each transport stream program contains subsets of data, including audio, video, and metadata. Each of these subsets of data has a numerical label called a packet identifier (PID). Each transport stream program corresponds to one MediaConvert output. The PMT lists the types of data in a program along with their PID. Downstream systems and players use the program map table to look up the PID for each type of data it accesses and then uses the PIDs to locate specific data within the asset.

Type: M2tsSettings (p. 762)
**Properties**

**Required:** False

**movSettings**
Settings for MOV Container.

*Type:* MovSettings (p. 775)
*Required:* False

**mp4Settings**
Settings for MP4 Container

*Type:* Mp4Settings (p. 776)
*Required:* False

**ContainerType**
Container for this output. Some containers require a container settings object. If not specified, the default object will be created.

F4V
ISMV
M2TS
M3U8
CMFC
MOV
MP4
MPD
MXF
RAW

**DashIsoEncryptionSettings**
Specifies DRM settings for DASH outputs.

**spekeKeyProvider**
Settings for use with a SPEKE key provider

*Type:* SpekeKeyProvider (p. 800)
*Required:* False

**DashIsoGroupSettings**
Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to DASH_ISO_GROUP_SETTINGS.

**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.
**Properties**

**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:** False
- **Pattern:** `s3://`

**destinationSettings**

Settings associated with the destination. Will vary based on the type of destination.

- **Type:** DestinationSettings (p. 690)
- **Required:** False

**encryption**

DRM settings.

- **Type:** DashIsoEncryptionSettings (p. 686)
- **Required:** False

**minBufferTime**

Minimum time of initially buffered media that is needed to ensure smooth playout.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 2147483647

**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:** False
- **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

**segmentControl**

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.

Type: DashIsoSegmentControl  
Required: False

**hbbtvCompliance**

Supports HbbTV specification as indicated

Type: DashIsoHbbtvCompliance  
Required: False

**writeSegmentTimelineInRepresentation**

When you enable Precise segment duration in manifests (writeSegmentTimelineInRepresentation), your DASH manifest shows precise segment durations. The segment duration information appears inside the SegmentTimeline element, inside SegmentTemplate at the Representation level. When this feature isn't enabled, the segment durations in your DASH manifest are approximate. The segment duration information appears in the duration attribute of the SegmentTemplate element.

Type: DashIsoWriteSegmentTimelineInRepresentation  
Required: False

**DashIsoHbbtvCompliance**

Supports HbbTV specification as indicated

HBBTV_1_5
NONE

**DashIsoSegmentControl**

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

**DashIsoWriteSegmentTimelineInRepresentation**

When you enable Precise segment duration in manifests (writeSegmentTimelineInRepresentation), your DASH manifest shows precise segment durations. The segment duration information appears inside the SegmentTimeline element, inside SegmentTemplate at the Representation level. When this feature isn't enabled, the segment durations in your DASH manifest are approximate. The segment duration information appears in the duration attribute of the SegmentTemplate element.
ENABLED
DISABLED

**DecryptionMode**

Specify the encryption mode that you used to encrypt your input files.

- AES_CTR
- AES_CBC
- AES_GCM

**DeinterlaceAlgorithm**

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**

Settings for deinterlacer algorithm

- **Mode**
  
  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.

  - **Type**: DeinterlaceMode (p. 689)
  - **Required**: False

- **Control**

  - 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.
**Type**: DeinterlacerControl (p. 690)  
**Required**: False

**DeinterlacerControl**

- 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**

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

**DestinationSettings**

Settings associated with the destination. Will vary based on the type of destination

**s3Settings**

Settings associated with S3 destination

  **Type**: S3DestinationSettings (p. 799)  
  **Required**: False

**DropFrameTimecode**

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**

Inserts DVB Network Information Table (NIT) at the specified table repetition interval.

  **nitInterval**  
  The number of milliseconds between instances of this table in the output transport stream.
**Properties**

**Type**: integer  
**Required**: False  
**Minimum**: 25  
**Maximum**: 10000

**networkId**

The numeric value placed in the Network Information Table (NIT).

**Type**: integer  
**Required**: False  
**Minimum**: 0  
**Maximum**: 65535

**networkName**

The network name text placed in the network_name_descriptor inside the Network Information Table. Maximum length is 256 characters.

**Type**: string  
**Required**: False  
**MinLength**: 1  
**MaxLength**: 256

**DvbSdtSettings**

Inserts DVB Service Description Table (NIT) at the specified table repetition interval.

**outputSdt**

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.

**Type**: OutputSdt (p. 793)  
**Required**: False

**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
**serviceProviderName**

The service provider name placed in the service_descriptor in the Service Description Table. Maximum length is 256 characters.

- **Type:** string
- **Required:** False
- **MinLength:** 1
- **MaxLength:** 256

**DvbSubDestinationSettings**

DVB-Sub Destination Settings

**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

**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

**teletextSpacing**

Only applies to jobs with input captions in Teletext or STL formats. Specify whether the spacing between letters in your captions is set by the captions grid or varies depending on letter width. Choose fixed grid to conform to the spacing specified in the captions file more accurately. Choose proportional to make the text easier to read if the captions are closed caption.

- **Type:** DvbSubtitleTeletextSpacing (p. 696)
- **Required:** False

**alignment**

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.

Type: DvbSubtitleAlignment (p. 695)
Required: False

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: False
Minimum: 0
Maximum: 10

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

shadowColor

Specifies the color of the shadow cast by the captions. All burn-in and DVB-Sub font settings must match.

Type: DvbSubtitleShadowColor (p. 696)
Required: False

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: False
Minimum: 0
Maximum: 255

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
Properties

**fontScript**

Provide the font script, using an ISO 15924 script code, if the LanguageCode is not sufficient for determining the script type. Where LanguageCode or CustomLanguageCode is sufficient, use "AUTOMATIC" or leave unset. This is used to help determine the appropriate font for rendering DVB-Sub captions.

*Required: False
Type: FontScript (p. 706)

**fontColor**

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.

*Required: False
Type: DvbSubtitleFontColor (p. 696)

**backgroundColor**

Specifies the color of the rectangle behind the captions. All burn-in and DVB-Sub font settings must match.

*Required: False
Type: DvbSubtitleBackgroundColor (p. 696)

**fontResolution**

Font resolution in DPI (dots per inch); default is 96 dpi. All burn-in and DVB-Sub font settings must match.

*Required: False
Type: integer
Minimum: 96
Maximum: 600

**outlineColor**

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.

*Required: False
Type: DvbSubtitleOutlineColor (p. 696)

**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.

*Required: False
**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**: -2147483648
- **Maximum**: 2147483647

**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

**DvbSubSourceSettings**

DVB Sub Source Settings

**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**

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**

Specifies the color of the rectangle behind the captions. All burn-in and DVB-Sub font settings must match.

- NONE
- BLACK
- WHITE

**DvbSubtitleFontColor**

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**

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**

Specifies the color of the shadow cast by the captions. All burn-in and DVB-Sub font settings must match.

- NONE
- BLACK
- WHITE

**DvbSubtitleTeletextSpacing**

Only applies to jobs with input captions in Teletext or STL formats. Specify whether the spacing between letters in your captions is set by the captions grid or varies depending on letter width. Choose fixed grid to conform to the spacing specified in the captions file more accurately. Choose proportional to make the text easier to read if the captions are closed caption.

- FIXED_GRID
**DvbTdtSettings**

Inserts DVB Time and Date Table (TDT) at the specified table repetition interval.

**tdtInterval**

The number of milliseconds between instances of this table in the output transport stream.

- **Type**: integer
- **Required**: False
- **Minimum**: 1000
- **Maximum**: 30000

**Eac3AttenuationControl**

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**

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**

Dolby Digital Plus coding mode. Determines number of channels.

- CODING_MODE_1_0
- CODING_MODE_2_0
- CODING_MODE_3_2

**Eac3DcFilter**

Activates a DC highpass filter for all input channels.

- ENABLED
- DISABLED

**Eac3DynamicRangeCompressionLine**

Enables Dynamic Range Compression that restricts the absolute peak level for a signal.
NONE
FILM_STANDARD
FILM_LIGHT
MUSIC_STANDARD
MUSIC_LIGHT
SPEECH

**Eac3DynamicRangeCompressionRf**

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**

When encoding 3/2 audio, controls whether the LFE channel is enabled

LFE
NO_LFE

**Eac3LfeFilter**

Applies a 120Hz lowpass filter to the LFE channel prior to encoding. Only valid with 3_2_LFE coding mode.

ENABLED
DISABLED

**Eac3MetadataControl**

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**

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**

Controls the amount of phase-shift applied to the surround channels. Only used for 3/2 coding mode.

- SHIFT_90_DEGREES
- NO_SHIFT

**Eac3Settings**

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value EAC3.

**metadataControl**

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.

- **Type**: Eac3MetadataControl (p. 698)
- **Required**: False

**surroundExMode**

When encoding 3/2 audio, sets whether an extra center back surround channel is matrix encoded into the left and right surround channels.

- **Type**: Eac3SurroundExMode (p. 702)
- **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

**phaseControl**

Controls the amount of phase-shift applied to the surround channels. Only used for 3/2 coding mode.

- **Type**: Eac3PhaseControl (p. 699)
- **Required**: False

**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
**Properties**

**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

**bitrate**
Average bitrate in bits/second. Valid bitrates depend on the coding mode.

- **Type**: integer
- **Required**: False
- **Minimum**: 64000
- **Maximum**: 640000

**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

**passthroughControl**
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.

- **Type**: Eac3PassthroughControl (p. 698)
- **Required**: False

**lfeControl**
When encoding 3/2 audio, controls whether the LFE channel is enabled

- **Type**: Eac3LfeControl (p. 698)
- **Required**: False

**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
Properties

Maximum: 3.0

**attenuationControl**

If set to ATTENUATE_3_DB, applies a 3 dB attenuation to the surround channels. Only used for 3/2 coding mode.

*Type: Eac3AttenuationControl (p. 697)*

*Required: False*

**codingMode**

Dolby Digital Plus coding mode. Determines number of channels.

*Type: Eac3CodingMode (p. 697)*

*Required: False*

**surroundMode**

When encoding 2/0 audio, sets whether Dolby Surround is matrix encoded into the two channels.

*Type: Eac3SurroundMode (p. 702)*

*Required: False*

**bitstreamMode**

Specifies the "Bitstream Mode" (bsmod) for the emitted E-AC-3 stream. See ATSC A/52-2012 (Annex E) for background on these values.

*Type: Eac3BitstreamMode (p. 697)*

*Required: False*

**lfeFilter**

Applies a 120Hz lowpass filter to the LFE channel prior to encoding. Only valid with 3_2_LFE coding mode.

*Type: Eac3LfeFilter (p. 698)*

*Required: False*

**stereoDownmix**

Stereo downmix preference. Only used for 3/2 coding mode.

*Type: Eac3StereoDownmix (p. 702)*

*Required: False*

**dynamicRangeCompressionRf**

Enables Heavy Dynamic Range Compression, ensures that the instantaneous signal peaks do not exceed specified levels.

*Type: Eac3DynamicRangeCompressionRf (p. 698)*

*Required: False*
sampleRate

Sample rate in hz. Sample rate is always 48000.

- **Type**: integer
- **Required**: False
- **Minimum**: 48000
- **Maximum**: 48000

**dynamicRangeCompressionLine**

Enables Dynamic Range Compression that restricts the absolute peak level for a signal.

- **Type**: Eac3DynamicRangeCompressionLine (p. 697)
- **Required**: False

**dcFilter**

Activates a DC highpass filter for all input channels.

- **Type**: Eac3DcFilter (p. 697)
- **Required**: False

**Eac3StereoDownmix**

Stereo downmix preference. Only used for 3/2 coding mode.

- NOT_INDICATED
- LO_RO
- LT_RT
- DPL2

**Eac3SurroundExMode**

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**

When encoding 2/0 audio, sets whether Dolby Surround is matrix encoded into the two channels.

- NOT_INDICATED
- ENABLED
- DISABLED

**EmbeddedConvert608To708**

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.
EmbeddedDestinationSettings

Settings specific to embedded/ancillary caption outputs, including 608/708 Channel destination number.

destination608ChannelNumber

Ignore this setting unless your input captions are SCC format and your output container is MXF. With this combination of input captions format and output container, you can optionally use this setting to replace the input channel number with the track number that you specify. Specify a different number for each output captions track. If you don't specify an output track number, the system uses the input channel number for the output channel number. This setting applies to each output individually. You can optionally combine two captions channels in your output. The two output channel numbers can be one of the following pairs: 1,3; 2,4; 1,4; or 2,3.

Type: integer
Required: False
Minimum: 1
Maximum: 4

EmbeddedSourceSettings

Settings for embedded captions Source

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

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

convert608To708

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.

Type: EmbeddedConvert608To708 (p. 702)
**Properties**

**Required**: False

**EsamManifestConfirmConditionNotification**

ESAM ManifestConfirmConditionNotification defined by OC-SP-ESAM-API-I03-131025.

**mccXml**

Provide your ESAM ManifestConfirmConditionNotification XML document inside your JSON job settings. Form the XML document as per OC-SP-ESAM-API-I03-131025. The transcoder will use the Manifest Conditioning instructions in the message that you supply.

**Type**: string  
**Required**: False  
**Pattern**: ^\s*<(.|\n)*ManifestConfirmConditionNotification(.|\n)*>\s*$

**EsamSettings**

Settings for Event Signaling And Messaging (ESAM). If you don't do ad insertion, you can ignore these settings.

**signalProcessingNotification**

Specifies an ESAM SignalProcessingNotification XML as per OC-SP-ESAM-API-I03-131025. The transcoder uses the signal processing instructions that you provide in the setting SCC XML (sccXml).

**Type**: EsamSignalProcessingNotification (p. 704)  
**Required**: False

**manifestConfirmConditionNotification**

Specifies an ESAM ManifestConfirmConditionNotification XML as per OC-SP-ESAM-API-I03-131025. The transcoder uses the manifest conditioning instructions that you provide in the setting MCC XML (mccXml).

**Type**: EsamManifestConfirmConditionNotification (p. 704)  
**Required**: False

**responseSignalPreroll**

Specifies the stream distance, in milliseconds, between the SCTE 35 messages that the transcoder places and the splice points that they refer to. If the time between the start of the asset and the SCTE-35 message is less than this value, then the transcoder places the SCTE-35 marker at the beginning of the stream.

**Type**: integer  
**Required**: False  
**Minimum**: 0  
**Maximum**: 30000

**EsamSignalProcessingNotification**

ESAM SignalProcessingNotification data defined by OC-SP-ESAM-API-I03-131025.
**sccXml**

Provide your ESAM SignalProcessingNotification XML document inside your JSON job settings. Form the XML document as per OC-SP-ESAM-API-I03-131025. The transcoder will use the signal processing instructions in the message that you supply. Provide your ESAM SignalProcessingNotification XML document inside your JSON job settings. If you want the service to place SCTE-35 markers at the insertion points you specify in the XML document, you must also enable SCTE-35 ESAM (scte35Esam). Note that you can either specify an ESAM XML document or enable SCTE-35 passthrough. You can't do both.

- **Type:** string
- **Required:** False
- **Pattern:** `^\s*<(.|\n)*SignalProcessingNotification(.|\n)*>\s*$`

**ExceptionBody**

**message**

- **Type:** string
- **Required:** False

**F4vMoovPlacement**

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**

Settings for F4v container

**moovPlacement**

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.

- **Type:** `F4vMoovPlacement (p. 705)`
- **Required:** False

**FileGroupSettings**

Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to FILE_GROUP_SETTINGS.

**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:** False
- **Pattern:** `^s3:\/\/$`
destinationSettings
Settings associated with the destination. Will vary based on the type of destination

Type: DestinationSettings (p. 690)
Required: False

FileNotFoundException
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
Settings for File-based Captions in Source

sourceFile
External caption file used for loading captions. Accepted file extensions are 'scc', 'ttml', 'dfxp', 'stl', 'srt', and 'smi'.

Type: string
Required: False
Pattern: ^(s3:\/\/)(.*?)\.\(scc|SCC|ttml|TTML|dfxp|DFXP|stl|STL|srt|SRT|smi|SMI)$
MinLength: 14

timeDelta
Specifies a time delta in seconds to offset the captions from the source file.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

convert608To708
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.

Type: FileSourceConvert608To708 (p. 706)
Required: False

FontScript
Provide the font script, using an ISO 15924 script code, if the LanguageCode is not sufficient for determining the script type. Where LanguageCode or CustomLanguageCode is sufficient, use "AUTOMATIC" or leave unset.

AUTOMATIC
HANS
HANT
**FrameCaptureSettings**

Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value FRAME_CAPTURE.

**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

**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

**quality**

JPEG Quality - a higher value equals higher quality.

- **Type:** integer
- **Required:** False
- **Minimum:** 1
- **Maximum:** 100

**GetJobRequest**

Query a job by sending a request with the job ID.

**id**

the job ID of the job.
GetJobResponse

Successful get job requests will return an OK message and the job JSON.

job

Each job converts an input file into an output file or files. For more information, see the User Guide at http://docs.aws.amazon.com/mediaconvert/latest/ug/what-is.html

Type: Job (p. 751)
Required: False

H264AdaptiveQuantization

Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

OFF
LOW
MEDIUM
HIGH
HIGHER
MAX

H264CodecLevel

Specify an H.264 level that is consistent with your output video settings. If you aren't sure what level to specify, choose Auto (AUTO).

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

H.264 Profile. High 4:2:2 and 10-bit profiles are only available with the AVC-I License.

BASELINE
**H264DynamicSubGop**

Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting B frames between reference frames (numberBFramesBetweenReferenceFrames).

- ADAPTIVE
- STATIC

**H264EntropyEncoding**

Entropy encoding mode. Use CABAC (must be in Main or High profile) or CAVLC.

- CABAC
- CAVLC

**H264FieldEncoding**

Choosing FORCE_FIELD disables PAFF encoding for interlaced outputs.

- PAFF
- FORCE_FIELD

**H264FlickerAdaptiveQuantization**

Adjust quantization within each frame to reduce flicker or 'pop' on I-frames.

- DISABLED
- ENABLED

**H264FramerateControl**

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

- INITIALIZE_FROM_SOURCE
- SPECIFIED
**H264FramerateConversionAlgorithm**

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

- DUPPLICATE_DROP
- INTERPOLATE

**H264GopBReference**

If enable, use reference B frames for GOP structures that have B frames > 1.

- DISABLED
- ENABLED

**H264GopSizeUnits**

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**

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, as follows. - 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**

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**

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

**H264QvbrSettings**

Settings for quality-defined variable bitrate encoding with the H.264 codec. Required when you set Rate control mode to QVBR. Not valid when you set Rate control mode to a value other than QVBR, or when you don’t define Rate control mode.

**qvbrQualityLevel**

Required when you use QVBR rate control mode. That is, when you specify qvbrSettings within h264Settings. Specify the target quality level for this output, from 1 to 10. Use higher numbers for greater quality. Level 10 results in nearly lossless compression. The quality level for most broadcast-quality transcodes is between 6 and 9.

*Type: integer*
*Required: False*
*Minimum: 1*
*Maximum: 10*

**maxAverageBitrate**

Use this setting only when Rate control mode is QVBR and Quality tuning level is Multi-pass HQ. For Max average bitrate values suited to the complexity of your input video, the service limits the average bitrate of the video part of this output to the value you choose. That is, the total size of the video element is less than or equal to the value you set multiplied by the number of seconds of encoded output.

*Type: integer*
*Required: False*
*Minimum: 1000*
*Maximum: 1152000000*

**H264RateControlMode**

Use this setting to specify whether this output has a variable bitrate (VBR), constant bitrate (CBR) or quality-defined variable bitrate (QVBR).

VBR
CBR
QVBR

**H264RepeatPps**

Places a PPS header on each encoded picture, even if repeated.

DISABLED
ENABLED

**H264SceneChangeDetect**

Scene change detection (inserts I-frames on scene changes).
DISABLED
ENABLED

**H264Settings**

Required when you set (Codec) under (VideoDescription)->(CodecSettings) to the value H_264.

**interlaceMode**

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, as follows. - 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.

- **Type:** H264InterlaceMode (p. 710)
- **Required:** False

**parNumerator**

Pixel Aspect Ratio numerator.

- **Type:** integer
- **Required:** False
- **Minimum:** 1
- **Maximum:** 2147483647

**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

**syntax**

Produces a bitstream compliant with SMPTE RP-2027.

- **Type:** H264Syntax (p. 718)
- **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

framerateDenominator

When you use the API for transcode jobs that use frame rate conversion, specify the frame rate 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 frame rate conversion, provide the value as a decimal number for Framerate. In this example, specify 23.976.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

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

hrdBufferInitialFillPercentage

Percentage of the buffer that should initially be filled (HRD buffer model).

Type: integer
Required: False
Minimum: 0
Maximum: 100

gopSize

GOP Length (keyframe interval) in frames or seconds. Must be greater than zero.

Type: number
Required: False
Format: float
Minimum: 0.0

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
gopBReference
If enable, use reference B frames for GOP structures that have B frames > 1.

Type: H264GopBReference (p. 710)
Required: False

hrdBufferSize
Size of buffer (HRD buffer model) in bits. For example, enter five megabits as 5000000.

Type: integer
Required: False
Minimum: 0
Maximum: 1152000000

maxBitrate
Maximum bitrate in bits/second. For example, enter five megabits per second as 5000000. Required when Rate control mode is QVBR.

Type: integer
Required: False
Minimum: 1000
Maximum: 1152000000

slowPal
Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

Type: H264SlowPal (p. 718)
Required: False

parDenominator
Pixel Aspect Ratio denominator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

spatialAdaptiveQuantization
Adjust quantization within each frame based on spatial variation of content complexity.

Type: H264SpatialAdaptiveQuantization (p. 718)
Required: False

temporalAdaptiveQuantization
Adjust quantization within each frame based on temporal variation of content complexity.

Type: H264TemporalAdaptiveQuantization (p. 719)
**Properties**

**flickerAdaptiveQuantization**

Adjust quantization within each frame to reduce flicker or 'pop' on I-frames.

- **Type:** H264FlickerAdaptiveQuantization (p. 709)
- **Required:** False

**entropyEncoding**

Entropy encoding mode. Use CABAC (must be in Main or High profile) or CAVLC.

- **Type:** H264EntropyEncoding (p. 709)
- **Required:** False

**bitrate**

Average bitrate in bits/second. Required for VBR and CBR. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.

- **Type:** integer
- **Required:** False
- **Minimum:** 1000
- **Maximum:** 115200000

**framerateControl**

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

- **Type:** H264FramerateControl (p. 709)
- **Required:** False

**rateControlMode**

Use this setting to specify whether this output has a variable bitrate (VBR), constant bitrate (CBR) or quality-defined variable bitrate (QVBR).

- **Type:** H264RateControlMode (p. 711)
- **Required:** False

**qvbrSettings**

Settings for quality-defined variable bitrate encoding with the H.264 codec. Required when you set Rate control mode to QVBR. Not valid when you set Rate control mode to a value other than QVBR, or when you don't define Rate control mode.
Properties

Type: H264QvbrSettings (p. 711)
Required: False

codecProfile

H.264 Profile. High 4:2:2 and 10-bit profiles are only available with the AVC-I License.

Type: H264CodecProfile (p. 708)
Required: False

telecine

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.

Type: H264Telecine (p. 718)
Required: False

framerateNumerator

Frame rate numerator - frame rate is a fraction, e.g. 24000 / 1001 = 23.976 fps.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

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

adaptiveQuantization

Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

Type: H264AdaptiveQuantization (p. 708)
Required: False

codecLevel

Specify an H.264 level that is consistent with your output video settings. If you aren't sure what level to specify, choose Auto (AUTO).
Properties

Type: H264CodecLevel (p. 708)
Required: False

fieldEncoding
Choosing FORCE_FIELD disables PAFF encoding for interlaced outputs.

Type: H264FieldEncoding (p. 709)
Required: False

sceneChangeDetect
Scene change detection (inserts I-frames on scene changes).

Type: H264SceneChangeDetect (p. 711)
Required: False

qualityTuningLevel
Use Quality tuning level (H264QualityTuningLevel) to specify whether to use fast single-pass, high-quality singlepass, or high-quality multipass video encoding.

Type: H264QualityTuningLevel (p. 710)
Required: False

framerateConversionAlgorithm
When set to INTERPOLATE, produces smoother motion during frame rate conversion.

Type: H264FramerateConversionAlgorithm (p. 710)
Required: False

unregisteredSeiTimecode
Inserts timecode for each frame as 4 bytes of an unregistered SEI message.

Type: H264UnregisteredSeiTimecode (p. 719)
Required: False

gopSizeUnits
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.

Type: H264GopSizeUnits (p. 710)
Required: False

parControl
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.

Type: H264ParControl (p. 710)
Required: False
numberBFramesBetweenReferenceFrames

Number of B-frames between reference frames.

Type: integer
Required: False
Minimum: 0
Maximum: 7

repeatPps

Places a PPS header on each encoded picture, even if repeated.

Type: H264RepeatPps (p. 711)
Required: False

dynamicSubGop

Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting B frames between reference frames (numberBFramesBetweenReferenceFrames).

Type: H264DynamicSubGop (p. 709)
Required: False

H264SlowPal

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

DISABLED
ENABLED

H264SpatialAdaptiveQuantization

Adjust quantization within each frame based on spatial variation of content complexity.

DISABLED
ENABLED

H264Syntax

Produces a bitstream compliant with SMPTE RP-2027.

DEFAULT
RP2027

H264Telecine

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**

Adjust quantization within each frame based on temporal variation of content complexity.

- DISABLED
- ENABLED

**H264UnregisteredSeiTimecode**

Inserts timecode for each frame as 4 bytes of an unregistered SEI message.

- DISABLED
- ENABLED

**H265AdaptiveQuantization**

Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

- OFF
- LOW
- MEDIUM
- HIGH
- HIGHER
- MAX

**H265AlternateTransferFunctionSei**

Enables Alternate Transfer Function SEI message for outputs using Hybrid Log Gamma (HLG) Electro-Optical Transfer Function (EOTF).

- DISABLED
- ENABLED

**H265CodecLevel**

H.265 Level.

- AUTO
- LEVEL_1
- LEVEL_2
- LEVEL_2_1
- LEVEL_3
- LEVEL_3_1
H265CodecProfile

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

H265DynamicSubGop

Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting B frames between reference frames (numberBFramesBetweenReferenceFrames).

- ADAPTIVE
- STATIC

H265FlickerAdaptiveQuantization

Adjust quantization within each frame to reduce flicker or 'pop' on I-frames.

- DISABLED
- ENABLED

H265FramerateControl

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose
SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

INITIALIZE_FROM_SOURCE
SPECIFIED

H265FramerateConversionAlgorithm

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

DUPLICATE_DROP
INTERPOLATE

H265GopBReference

If enable, use reference B frames for GOP structures that have B frames > 1.

DISABLED
ENABLED

H265GopSizeUnits

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

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

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**

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

**H265QvbrSettings**

Settings for quality-defined variable bitrate encoding with the H.265 codec. Required when you set Rate control mode to QVBR. Not valid when you set Rate control mode to a value other than QVBR, or when you don’t define Rate control mode.

**qvbrQualityLevel**

Required when you use QVBR rate control mode. That is, when you specify qvbrSettings within h265Settings. Specify the target quality level for this output, from 1 to 10. Use higher numbers for greater quality. Level 10 results in nearly lossless compression. The quality level for most broadcast-quality transcodes is between 6 and 9.

- **Type:** integer
- **Required:** False
- **Minimum:** 1
- **Maximum:** 10

**maxAverageBitrate**

Use this setting only when Rate control mode is QVBR and Quality tuning level is Multi-pass HQ. For Max average bitrate values suited to the complexity of your input video, the service limits the average bitrate of the video part of this output to the value you choose. That is, the total size of the video element is less than or equal to the value you set multiplied by the number of seconds of encoded output.

- **Type:** integer
- **Required:** False
- **Minimum:** 1000
- **Maximum:** 1466400000

**H265RateControlMode**

Use this setting to specify whether this output has a variable bitrate (VBR), constant bitrate (CBR) or quality-defined variable bitrate (QVBR).

- **VBR**
- **CBR**
- **QVBR**

**H265SampleAdaptiveOffsetFilterMode**

Specify Sample Adaptive Offset (SAO) filter strength. Adaptive mode dynamically selects best strength based on content

- **DEFAULT**
ADAPTIVE
OFF

H265SceneChangeDetect

Scene change detection (inserts I-frames on scene changes).

DISABLED
ENABLED

H265Settings

Settings for H265 codec

interlaceMode

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.

Type: H265InterlaceMode (p. 721)
Required: False

parNumerator

Pixel Aspect Ratio numerator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

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

framerateDenominator

Frame rate denominator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

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

alternateTransferFunctionSei

Enables Alternate Transfer Function SEI message for outputs using Hybrid Log Gamma (HLG) Electro-Optical Transfer Function (EOTF).

Type: H265AlternateTransferFunctionSei (p. 719)
Required: False

hrdBufferInitialFillPercentage

Percentage of the buffer that should initially be filled (HRD buffer model).

Type: integer
Required: False
Minimum: 0
Maximum: 100

gopSize

GOP Length (keyframe interval) in frames or seconds. Must be greater than zero.

Type: number
Required: False
Format: float
Minimum: 0.0

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

gopBReference

If enable, use reference B frames for GOP structures that have B frames > 1.
**Type:** H265GopBReference (p. 721)  
**Required:** False

**hrdBufferSize**
Size of buffer (HRD buffer model) in bits. For example, enter five megabits as 5000000.

**Type:** integer  
**Required:** False  
**Minimum:** 0  
**Maximum:** 1466400000

**maxBitrate**
Maximum bitrate in bits/second. For example, enter five megabits per second as 5000000. Required when Rate control mode is QVBR.

**Type:** integer  
**Required:** False  
**Minimum:** 1000  
**Maximum:** 1466400000

**slowPal**
Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

**Type:** H265SlowPal (p. 729)  
**Required:** False

**parDenominator**
Pixel Aspect Ratio denominator.

**Type:** integer  
**Required:** False  
**Minimum:** 1  
**Maximum:** 2147483647

**spatialAdaptiveQuantization**
Adjust quantization within each frame based on spatial variation of content complexity.

**Type:** H265SpatialAdaptiveQuantization (p. 730)  
**Required:** False

**temporalAdaptiveQuantization**
Adjust quantization within each frame based on temporal variation of content complexity.

**Type:** H265TemporalAdaptiveQuantization (p. 730)  
**Required:** False
flickerAdaptiveQuantization

Adjust quantization within each frame to reduce flicker or 'pop' on I-frames.

Type: H265FlickerAdaptiveQuantization (p. 720)
Required: False

bitrate

Average bitrate in bits/second. Required for VBR and CBR. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.

Type: integer
Required: False
Minimum: 1000
Maximum: 1466400000

framerateControl

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

Type: H265FramerateControl (p. 720)
Required: False

rateControlMode

Use this setting to specify whether this output has a variable bitrate (VBR), constant bitrate (CBR) or quality-defined variable bitrate (QVBR).

Type: H265RateControlMode (p. 722)
Required: False

qvbrSettings

Settings for quality-defined variable bitrate encoding with the H.265 codec. Required when you set Rate control mode to QVBR. Not valid when you set Rate control mode to a value other than QVBR, or when you don't define Rate control mode.

Type: H265QvbrSettings (p. 722)
Required: False

codecProfile

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.

Type: H265CodecProfile (p. 720)
Required: False

tiles
Enable use of tiles, allowing horizontal as well as vertical subdivision of the encoded pictures.

Type: H265Tiles (p. 730)
Required: False

telecine
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.

Type: H265Telecine (p. 730)
Required: False

framerateNumerator
Frame rate numerator - frame rate is a fraction, e.g. 24000 / 1001 = 23.976 fps.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

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

adaptiveQuantization
Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

Type: H265AdaptiveQuantization (p. 719)
Required: False

codecLevel
H.265 Level.

Type: H265CodecLevel (p. 719)
Required: False

**sceneChangeDetect**

Scene change detection (inserts I-frames on scene changes).

Type: H265SceneChangeDetect (p. 723)
Required: False

**qualityTuningLevel**

Use Quality tuning level (H265QualityTuningLevel) to specify whether to use fast single-pass, high-quality singlepass, or high-quality multipass video encoding.

Type: H265QualityTuningLevel (p. 722)
Required: False

**framerateConversionAlgorithm**

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

Type: H265FramerateConversionAlgorithm (p. 721)
Required: False

**unregisteredSeiTimecode**

Inserts timecode for each frame as 4 bytes of an unregistered SEI message.

Type: H265UnregisteredSeiTimecode (p. 730)
Required: False

**gopSizeUnits**

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.

Type: H265GopSizeUnits (p. 721)
Required: False

**parControl**

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.

Type: H265ParControl (p. 721)
Required: False

**numberBFramesBetweenReferenceFrames**

Number of B-frames between reference frames.

Type: integer
Required: False
Minimum: 0
Maximum: 7

temporalIds
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.

Type: H265TemporalIds (p. 730)
Required: False

sampleAdaptiveOffsetFilterMode
Specify Sample Adaptive Offset (SAO) filter strength. Adaptive mode dynamically selects best strength based on content

Type: H265SampleAdaptiveOffsetFilterMode (p. 722)
Required: False

writeMp4PackagingType
Use this setting only for outputs encoded with H.265 that are in CMAF or DASH output groups. If you include writeMp4PackagingType in your JSON job specification for other outputs, your video might not work properly with downstream systems and video players. If the location of parameter set NAL units don't matter in your workflow, ignore this setting. The service defaults to marking your output as HEV1. Choose HVC1 to mark your output as HVC1. This makes your output compliant with this specification: ISO IEC JTC1 SC29 N13798 Text ISO/IEC FDIS 14496-15 3rd Edition. For these outputs, the service stores parameter set NAL units in the sample headers but not in the samples directly. Keep the default HEV1 to mark your output as HEV1. For these outputs, the service writes parameter set NAL units directly into the samples.

Type: H265WriteMp4PackagingType (p. 731)
Required: False

dynamicSubGop
Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting B frames between reference frames (numberBFramesBetweenReferenceFrames).

Type: H265DynamicSubGop (p. 720)
Required: False

H265SlowPal
Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

DISABLED
ENABLED
H265SpatialAdaptiveQuantization

Adjust quantization within each frame based on spatial variation of content complexity.

- DISABLED
- ENABLED

H265Telecine

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

Adjust quantization within each frame based on temporal variation of content complexity.

- DISABLED
- ENABLED

H265TemporalIds

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

Enable use of tiles, allowing horizontal as well as vertical subdivision of the encoded pictures.

- DISABLED
- ENABLED

H265UnregisteredSeiTimecode

Inserts timecode for each frame as 4 bytes of an unregistered SEI message.

- DISABLED
- ENABLED
H265WriteMp4PackagingType

Use this setting only for outputs encoded with H.265 that are in CMAF or DASH output groups. If you include writeMp4PackagingType in your JSON job specification for other outputs, your video might not work properly with downstream systems and video players. If the location of parameter set NAL units don’t matter in your workflow, ignore this setting. The service defaults to marking your output as HEV1. Choose HVC1 to mark your output as HVC1. This makes your output compliant with this specification: ISO IEC/JTC1 SC29 N13798 Text ISO/IEC FDIS 14496-15 3rd Edition. For these outputs, the service stores parameter set NAL units in the sample headers but not in the samples directly. Keep the default HEV1 to mark your output as HEV1. For these outputs, the service writes parameter set NAL units directly into the samples.

    HVC1
    HEV1

Hdr10Metadata

Use the "HDR master display information" (Hdr10Metadata) settings to correct HDR metadata or to provide missing metadata. These values vary depending on the input video and must be provided by a color grader. Range is 0 to 50,000; each increment represents 0.00002 in CIE1931 color coordinate. Note that these settings are not color correction. Note that if you are creating HDR outputs inside of an HLS CMAF package, to comply with the Apple specification, you must use the following settings. Set "MP4 packaging type" (writeMp4PackagingType) to HVC1 (HVC1). Set "Profile" (H265Settings > codecProfile) to Main10/High (MAIN10_HIGH). Set "Level" (H265Settings > codecLevel) to 5 (LEVEL_5).

    redPrimaryX

    HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

    Type: integer
    Required: False
    Minimum: 0
    Maximum: 50000

    redPrimaryY

    HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

    Type: integer
    Required: False
    Minimum: 0
    Maximum: 50000

    greenPrimaryX

    HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

    Type: integer
    Required: False
greenPrimaryY

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

Type: integer
Required: False
Minimum: 0
Maximum: 50000

bluePrimaryX

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

Type: integer
Required: False
Minimum: 0
Maximum: 50000

bluePrimaryY

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

Type: integer
Required: False
Minimum: 0
Maximum: 50000

whitePointX

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

Type: integer
Required: False
Minimum: 0
Maximum: 50000

whitePointY

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

Type: integer
Required: False
**Properties**

- **maxFrameAverageLightLevel**
  - Maximum average light level of any frame in the coded video sequence, in units of candelas per square meter.
  - **Type:** integer
  - **Required:** False
  - **Minimum:** 0
  - **Maximum:** 65535

- **maxContentLightLevel**
  - Maximum light level among all samples in the coded video sequence, in units of candelas per square meter.
  - **Type:** integer
  - **Required:** False
  - **Minimum:** 0
  - **Maximum:** 65535

- **maxLuminance**
  - Nominal maximum mastering display luminance in units of 0.0001 candelas per square meter.
  - **Type:** integer
  - **Required:** False
  - **Minimum:** 0
  - **Maximum:** 2147483647

- **minLuminance**
  - Nominal minimum mastering display luminance in units of 0.0001 candelas per square meter.
  - **Type:** integer
  - **Required:** False
  - **Minimum:** 0
  - **Maximum:** 2147483647

- **HlsAdMarkers**
  - ELEMENTAL
  - ELEMENTAL_SCTE35

- **HlsAudioTrackType**
  - 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

Caption Language Mapping

captionChannel

Caption channel.

Type: integer
Required: False
Minimum: -2147483648
Maximum: 2147483647

customLanguageCode

Specify the language for this caption channel, using the ISO 639-2 or ISO 639-3 three-letter language code

Type: string
Required: False
Pattern: ^[A-Za-z]{3}$
MinLength: 3
MaxLength: 3

languageCode


Type: LanguageCode (p. 756)
Required: False

languageDescription

Caption language description.

Type: string
Required: False

HlsCaptionLanguageSetting

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-CAPITIONS=None line in the manifest. Omit: Omit any CLOSED-CAPTIONS line from the manifest.

INSERT
OMIT
NONE

HlsClientCache

When set to ENABLED, sets #EXT-X-ALLOW-CACHE:no tag, which prevents client from saving media segments for later replay.

DISABLED
ENABLED

HlsCodecSpecification

Specification to use (RFC-6381 or the default RFC-4281) during m3u8 playlist generation.

RFC_6381
RFC_4281

HlsDirectoryStructure

Indicates whether segments should be placed in subdirectories.

SINGLE_DIRECTORY
SUBDIRECTORY_PER_STREAM

HlsEncryptionSettings

Settings for HLS encryption

encryptionMethod

Encrypts the segments with the given encryption scheme. Leave blank to disable. Selecting 'Disabled' in the web interface also disables encryption.

Type: HlsEncryptionType (p. 736)
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}$
MinLength: 32
MaxLength: 32
**initializationVectorInManifest**

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.

*Type:* HlsInitializationVectorInManifest (p. 740)
*Required:* False

**offlineEncrypted**

Enable this setting to insert the EXT-X-SESSION-KEY element into the master playlist. This allows for offline Apple HLS FairPlay content protection.

*Type:* HlsOfflineEncrypted (p. 741)
*Required:* False

**spekeKeyProvider**

Settings for use with a SPEKE key provider

*Type:* SpekeKeyProvider (p. 800)
*Required:* False

**staticKeyProvider**

Use these settings to set up encryption with a static key provider.

*Type:* StaticKeyProvider (p. 801)
*Required:* False

**type**

Indicates which type of key provider is used for encryption.

*Type:* HlsKeyProviderType (p. 741)
*Required:* False

**HlsEncryptionType**

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**

Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to HLS_GROUP_SETTINGS.

**manifestDurationFormat**

Indicates whether the output manifest should use floating point values for segment duration.

*Type:* HlsManifestDurationFormat (p. 741)
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:** False
- **Minimum:** 1
- **Maximum:** 2147483647

**timedMetadataId3Period**

Timed Metadata interval in seconds.

- **Type:** integer
- **Required:** False
- **Minimum:** -2147483648
- **Maximum:** 2147483647

**captionLanguageSetting**

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.

- **Type:** HlsCaptionLanguageSetting (p. 734)
- **Required:** False

**captionLanguageMappings**

Language to be used on Caption outputs

- **Type:** Array of type HlsCaptionLanguageMapping (p. 734)
- **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:** False
- **Pattern:** ^s3:\/\/

**destinationSettings**

Settings associated with the destination. Will vary based on the type of destination
### Properties

- **Type**: DestinationSettings (p. 690)
  - **Required**: False

- **encryption**
  DRM settings.
  - **Type**: HlsEncryptionSettings (p. 735)
  - **Required**: False

- **timedMetadataId3Frame**
  Indicates ID3 frame that has the timecode.
  - **Type**: HlsTimedMetadataId3Frame (p. 743)
  - **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

- **codecSpecification**
  Specification to use (RFC-6381 or the default RFC-4281) during m3u8 playlist generation.
  - **Type**: HlsCodecSpecification (p. 735)
  - **Required**: False

- **outputSelection**
  Indicates whether the .m3u8 manifest file should be generated for this HLS output group.
  - **Type**: HlsOutputSelection (p. 741)
  - **Required**: False

- **programDateTimePeriod**
  Period of insertion of EXT-X-PROGRAM-DATE-TIME entry, in seconds.
  - **Type**: integer
  - **Required**: False
  - **Minimum**: 0
  - **Maximum**: 3600

- **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
**Properties**

**Required**: False
**Minimum**: 1
**Maximum**: 2147483647

**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**: False
**Minimum**: 0
**Maximum**: 2147483647

**minFinalSegmentLength**

Keep this setting at the default value of 0, unless you are troubleshooting a problem with how devices play back the end of your video asset. If you know that player devices are hanging on the final segment of your video because the length of your final segment is too short, use this setting to specify a minimum final segment length, in seconds. Choose a value that is greater than or equal to 1 and less than your segment length. When you specify a value for this setting, the encoder will combine any final segment that is shorter than the length that you specify with the previous segment. For example, your segment length is 3 seconds and your final segment is .5 seconds without a minimum final segment length; when you set the minimum final segment length to 1, your final segment is 3.5 seconds.

**Type**: number
**Required**: False
**Format**: float
**Minimum**: 0.0
**Maximum**: 2147483647

**directoryStructure**

Indicates whether segments should be placed in subdirectories.

**Type**: HlsDirectoryStructure (p. 735)
**Required**: False

**programDateTime**

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.

**Type**: HlsProgramDateTime (p. 741)
**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. 733)
**Required**: False
**segmentControl**

When set to SINGLE_FILE, emits program as a single media resource (.ts) file, uses #EXT-X-BYTERANGE tags to index segment for playback.

- **Type**: HlsSegmentControl (p. 741)
- **Required**: False

**timestampDeltaMilliseconds**

Provides an extra millisecond delta offset to fine tune the timestamps.

- **Type**: integer
- **Required**: False
- **Minimum**: -2147483648
- **Maximum**: 2147483647

**manifestCompression**

When set to GZIP, compresses HLS playlist.

- **Type**: HlsManifestCompression (p. 741)
- **Required**: False

**clientCache**

When set to ENABLED, sets #EXT-X-ALLOW-CACHE:no tag, which prevents client from saving media segments for later replay.

- **Type**: HlsClientCache (p. 735)
- **Required**: False

**streamInfResolution**

Include or exclude RESOLUTION attribute for video in EXT-X-STREAM-INF tag of variant manifest.

- **Type**: HlsStreamInfResolution (p. 742)
- **Required**: False

**HlsIFrameOnlyManifest**

When set to INCLUDE, writes I-Frame Only Manifest in addition to the HLS manifest

- INCLUDE
- EXCLUDE

**HlsInitializationVectorInManifest**

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**

Indicates which type of key provider is used for encryption.

- SPEKE
- STATIC_KEY

**HlsManifestCompression**

When set to GZIP, compresses HLS playlist.

- GZIP
- NONE

**HlsManifestDurationFormat**

Indicates whether the output manifest should use floating point values for segment duration.

- FLOATING_POINT
- INTEGER

**HlsOfflineEncrypted**

Enable this setting to insert the EXT-X-SESSION-KEY element into the master playlist. This allows for offline Apple HLS FairPlay content protection.

- ENABLED
- DISABLED

**HlsOutputSelection**

Indicates whether the .m3u8 manifest file should be generated for this HLS output group.

- MANIFESTS_AND_SEGMENTS
- SEGMENTS_ONLY

**HlsProgramDateTime**

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**

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**

Settings for HLS output groups

**audioGroupId**

Specifies the group to which the audio Rendition belongs.

*Type:* string  
*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

**audioTrackType**

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=NO 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

*Type:* HlsAudioTrackType (p. 733)  
*Required:* False

**iFrameOnlyManifest**

When set to INCLUDE, writes I-Frame Only Manifest in addition to the HLS manifest

*Type:* HlsIFrameOnlyManifest (p. 740)  
*Required:* False

**segmentModifier**

String concatenated to end of segment filenames. Accepts "Format Identifiers":#format_identifier_parameters.

*Type:* string  
*Required:* False

**HlsStreamInfResolution**

Include or exclude RESOLUTION attribute for video in EXT-X-STREAM-INF tag of variant manifest.

*INCLUDE*  
*EXCLUDE*
HlsTimedMetadataId3Frame

Indicates ID3 frame that has the timecode.

- NONE
- PRIV
- TDRL

Id3Insertion

To insert ID3 tags in your output, specify two values. Use ID3 tag (Id3) to specify the base 64 encoded string and use Timecode (TimeCode) to specify the time when the tag should be inserted. To insert multiple ID3 tags in your output, create multiple instances of ID3 insertion (Id3Insertion).

**timecode**

Provide a Timecode (TimeCode) in HH:MM:SS:FF or HH:MM:SS;FF format.

- **Type**: string
- **Required**: False
- **Format**: timecode
- **Pattern**: ^([01][0-9]|2[0-4]):[0-5][0-9]:[0-5][0-9]:[;][0-9]{2}$

**id3**

Use ID3 tag (Id3) to provide a tag value in base64-encode format.

- **Type**: string
- **Required**: False
- **Pattern**: ^[A-Za-z0-9+\-/]+={0,2}$

ImageInserter

Enable the image inserter feature to include a graphic overlay on your video. Enable or disable this feature for each input or output individually. This setting is disabled by default.

**insertableImages**

Specify the images that you want to overlay on your video. The images must be PNG or TGA files.

- **Type**: Array of type InsertableImage (p. 749)
- **Required**: False

Input

Specifies media input

**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.
Properties

Type: Array of type InputClipping (p. 746)
Required: False

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

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: 1
Maximum: 2147483647

videoSelector

Selector for video.

Type: VideoSelector (p. 812)
Required: False

filterEnable

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.

Type: InputFilterEnable (p. 748)
Required: False

psiControl

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.

Type: InputPsiControl (p. 748)
**Properties**

**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

**deblockFilter**

Enable Deblock (InputDeblockFilter) to produce smoother motion in the output. Default is disabled. Only manually controllable for MPEG2 and uncompressed video inputs.

- **Type**: InputDeblockFilter (p. 747)
- **Required**: False

**denoiseFilter**

Enable Denoise (InputDenoiseFilter) to filter noise from the input. Default is disabled. Only applicable to MPEG2, H.264, H.265, and uncompressed video inputs.

- **Type**: InputDenoiseFilter (p. 748)
- **Required**: False

**timecodeSource**

Timecode source under input settings (InputTimecodeSource) only affects the behavior of features that apply to a single input at a time, such as input clipping and synchronizing some captions formats. Use this setting to specify whether the service counts frames by timecodes embedded in the video (EMBEDDED) or by starting the first frame at zero (ZEROBASED). In both cases, the timecode format is HH:MM:SS:FF or HH:MM:SS;FF, where FF is the frame number. Only set this to EMBEDDED if your source video has embedded timecodes.

- **Type**: InputTimecodeSource (p. 749)
- **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

**imageInserter**

Enable the image inserter feature to include a graphic overlay on your video. Enable or disable this feature for each input individually. This setting is disabled by default.

- **Type**: ImageInserter (p. 743)
fileInput

Specify the source file for your transcoding job. You can use multiple inputs in a single job. The service concatenates these inputs, in the order that you specify them in the job, to create the outputs. If your input format is IMF, specify your input by providing the path to your CPL. For example, "s3://bucket/vf/cpl.xml". If the CPL is in an incomplete IMP, make sure to use "Supplemental IMPs" (SupplementalImps) to specify any supplemental IMPs that contain assets referenced by the CPL.

**Required:** False

**Type:** string

**Pattern:** ^s3://[^/]*(ASSETMAP.xml)?$
embedded timecodes that start at 01:00:00:00 and you want your clip to end six minutes into the video, use 01:06:00:00.

**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. Use the format HH:MM:SS:FF or HH:MM:SS;FF, where HH is the hour, MM is the minute, SS is the second, and FF is the frame number. 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 01:05:00:00.

**InputDeblockFilter**

Enable Deblock (InputDeblockFilter) to produce smoother motion in the output. Default is disabled. Only manually controllable for MPEG2 and uncompressed video inputs.

**InputDecryptionSettings**

Settings for decrypting any input files that you encrypt before you upload them to Amazon S3. MediaConvert can decrypt files only when you use AWS Key Management Service (KMS) to encrypt the data key that you use to encrypt your content.

**decryptionMode**

Specify the encryption mode that you used to encrypt your input files.

**encryptedDecryptionKey**

Warning! Don't provide your encryption key in plaintext. Your job settings could be intercepted, making your encrypted content vulnerable. Specify the encrypted version of the data key that you used to encrypt your content. The data key must be encrypted by AWS Key Management Service (KMS). The key can be 128, 192, or 256 bits.
**MaxLength**: 512

**initializationVector**

Specify the initialization vector that you used when you encrypted your content before uploading it to Amazon S3. You can use a 16-byte initialization vector with any encryption mode. Or, you can use a 12-byte initialization vector with GCM or CTR. MediaConvert accepts only initialization vectors that are base64-encoded.

*Type*: string  
*Required*: False  
*Pattern*: `^[A-Za-z0-9+/]{22}==$|^[A-Za-z0-9+/]{16}$`  
*MinLength*: 16  
*MaxLength*: 24

**kmsKeyRegion**

Specify the AWS Region for AWS Key Management Service (KMS) that you used to encrypt your data key, if that Region is different from the one you are using for AWS Elemental MediaConvert.

*Type*: string  
*Required*: False  
*Pattern*: `^[a-z-]{2,6}-(east|west|central|((north|south)(east|west)?)-[1-9]{1,2}$`  
*MinLength*: 9  
*MaxLength*: 19

**InputDenoiseFilter**

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**

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**

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

**InputRotate**

Use Rotate (InputRotate) to specify how the service rotates your video. You can choose automatic rotation or specify a rotation. You can specify a clockwise rotation of 0, 90, 180, or 270 degrees. If your input video container is .mov or .mp4 and your input has rotation metadata, you can choose Automatic to have the service rotate your video according to the rotation specified in the metadata. The rotation must be within one degree of 90, 180, or 270 degrees. If the rotation metadata specifies any other rotation, the service will default to no rotation. By default, the service does no rotation, even if your input video has rotation metadata. The service doesn't pass through rotation metadata.

- DEGREE_0
- DEGREES_90
- DEGREES_180
- DEGREES_270
- AUTO

**InputTimecodeSource**

Timecode source under input settings (InputTimecodeSource) only affects the behavior of features that apply to a single input at a time, such as input clipping and synchronizing some captions formats. Use this setting to specify whether the service counts frames by timecodes embedded in the video (EMBEDDED) or by starting the first frame at zero (ZEROBASED). In both cases, the timecode format is HH:MM:SS:FF or HH:MM:SS;FF, where FF is the frame number. Only set this to EMBEDDED if your source video has embedded timecodes.

- EMBEDDED
- ZEROBASED
- SPECIFIEDSTART

**InsertableImage**

Settings that specify how your still graphic overlay appears.

**width**

Specify the width of the inserted image in pixels. If you specify a value that's larger than the video resolution width, the service will crop your overlaid image to fit. To use the native width of the image, keep this setting blank.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 2147483647

**height**

Specify the height of the inserted image in pixels. If you specify a value that's larger than the video resolution height, the service will crop your overlaid image to fit. To use the native height of the image, keep this setting blank.

- **Type**: integer
- **Required**: False
Properties

Minimum: 0
Maximum: 2147483647

**imageX**

Specify the distance, in pixels, between the inserted image and the left edge of the video frame. Required for any image overlay that you specify.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 2147483647

**imageY**

Specify the distance, in pixels, between the overlaid image and the top edge of the video frame. Required for any image overlay that you specify.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 2147483647

**duration**

Specify the time, in milliseconds, for the image to remain on the output video. This duration includes fade-in time but not fade-out time.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 2147483647

**fadeIn**

Specify the length of time, in milliseconds, between the Start time that you specify for the image insertion and the time that the image appears at full opacity. Full opacity is the level that you specify for the opacity setting. If you don't specify a value for Fade-in, the image will appear abruptly at the overlay start time.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 2147483647

**layer**

Specify how overlapping inserted images appear. Images with higher values for Layer appear on top of images with lower values for Layer.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 99
imageInserterInput

Specify the Amazon S3 location of the image that you want to overlay on the video. Use a PNG or TGA file.

Type: string
Required: False
Pattern: ^(s3:\/\//)(.*?\.(bmp|BMP|png|PNG|tga|TGA))$
MinLength: 14

startTime

Specify the timecode of the frame that you want the overlay to first appear on. This must be in timecode (HH:MM:SS:FF or HH:MM:SS;FF) format. Remember to take into account your timecode source settings.

Type: string
Required: False
Pattern: ^(((\[0-1\d]|2\d)(\[0-5\d]{2}(\[;\d\[0-5\d\d]))$^

fadeOut

Specify the length of time, in milliseconds, between the end of the time that you have specified for the image overlay Duration and when the overlaid image has faded to total transparency. 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: 0
Maximum: 2147483647

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: False
Minimum: 0
Maximum: 100

Job

Each job converts an input file into an output file or files. For more information, see the User Guide at http://docs.aws.amazon.com/mediaconvert/latest/ug/what-is.html

arn

An identifier for this resource that is unique within all of AWS.

Type: string
Required: False

id

A portion of the job's ARN, unique within your AWS Elemental MediaConvert resources
**Properties**

**createdAt**

The time, in Unix epoch format in seconds, when the job got created.

- **Type**: string
- **Required**: False
- **Format**: date-time

**jobTemplate**

The job template that the job is created from, if it is created from a job template.

- **Type**: string
- **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

**userMetadata**

User-defined metadata that you want to associate with an MediaConvert job. You specify metadata in key/value pairs.

- **Type**: object
- **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

**settings**

JobSettings contains all the transcode settings for a job.

- **Type**: JobSettings (p. 754)
- **Required**: True

**status**

A job's status can be SUBMITTED, PROGRESSING, COMPLETE, CANCELED, or ERROR.

- **Type**: JobStatus (p. 756)
Properties

Required: False

code

Error code for the job

Type: integer
  Required: False
  Format: int32

errorMessage

Error message of Job

Type: string
  Required: False

timing

Information about when jobs are submitted, started, and finished is specified in Unix epoch format in seconds.

Type: Timing (p. 806)
  Required: False

outputGroupDetails

List of output group details

Type: Array of type OutputGroupDetail (p. 792)
  Required: False

billingTagsSource

Optional. Choose a tag type that AWS Billing and Cost Management will use to sort your AWS Elemental MediaConvert costs on any billing report that you set up. Any transcoding outputs that don't have an associated tag will appear in your billing report unsorted. If you don't choose a valid value for this field, your job outputs will appear on the billing report unsorted.

Type: BillingTagsSource (p. 668)
  Required: False

accelerationSettings

Accelerated transcoding can significantly speed up jobs with long, visually complex content.

Type: AccelerationSettings (p. 659)
  Required: False

statusUpdateInterval

Specify how often MediaConvert sends STATUS_UPDATE events to Amazon CloudWatch Events. Set the interval, in seconds, between status updates. MediaConvert sends an update at this interval from the time the service begins processing your job to the time it completes the transcode or encounters an error.
**jobPercentComplete**

An estimate of how far your job has progressed. This estimate is shown as a percentage of the total time from when your job leaves its queue to when your output files appear in your output Amazon S3 bucket. AWS Elemental MediaConvert provides jobPercentComplete in CloudWatch STATUS_UPDATE events and in the response to GetJob and ListJobs requests. The jobPercentComplete estimate is reliable for the following input containers: Quicktime, Transport Stream, MP4, and MXF. For some jobs, including audio-only jobs and jobs that use input clipping, the service can't provide information about job progress. In those cases, jobPercentComplete returns a null value.

**Type**: integer  
**Required**: False

**currentPhase**

A job's phase can be PROBING, TRANSCODING OR UPLOADING

**Type**: JobPhase (p. 754)  
**Required**: False

**retryCount**

The number of times that the service automatically attempted to process your job after encountering an error.

**Type**: integer  
**Required**: False

**JobPhase**

A job's phase can be PROBING, TRANSCODING OR UPLOADING

- PROBING  
- TRANSCODING  
- UPLOADING

**JobSettings**

JobSettings contains all the transcode settings for a job.

**timecodeConfig**

Contains settings used to acquire and adjust timecode information from inputs.

**Type**: TimecodeConfig (p. 804)  
**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)
Properties

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 * CMAF_GROUP_SETTINGS, CmAfGroupSettings

  Type: Array of type OutputGroup (p. 791)
  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

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. 668)
  Required: False

timedMetadataInsertion

Enable Timed metadata insertion (TimedMetadataInsertion) to include ID3 tags in your job. To include timed metadata, you must enable it here, enable it in each output container, and specify tags and timecodes in ID3 insertion (Id3Insertion) objects.

  Type: TimedMetadataInsertion (p. 805)
  Required: False

nielsenConfiguration

Settings for Nielsen Configuration

  Type: NielsenConfiguration (p. 787)
  Required: False

motionImageInserter

Overlay motion graphics on top of your video. The motion graphics that you specify here appear on all outputs in all output groups.

  Type: MotionImageInserter (p. 771)
  Required: False

esam

Settings for Event Signaling And Messaging (ESAM).
Type: EsamSettings (p. 704)
Required: False

inputs

Use Inputs (inputs) to define source file used in the transcode job. There can be multiple inputs add in a job. These inputs will be concatenated together to create the output.

Type: Array of type Input (p. 743)
Required: False

JobStatus

A job's status can be SUBMITTED, PROGRESSING, COMPLETE, CANCELED, or ERROR.

SUBMITTED
PROGRESSING
COMPLETE
CANCELED
ERROR

LanguageCode


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
<table>
<thead>
<tr>
<th>Language Code</th>
<th>Language Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>AYM</td>
<td>Aym</td>
</tr>
<tr>
<td>AZE</td>
<td>Aze</td>
</tr>
<tr>
<td>BAM</td>
<td>Bam</td>
</tr>
<tr>
<td>BAK</td>
<td>Bak</td>
</tr>
<tr>
<td>EUS</td>
<td>Eus</td>
</tr>
<tr>
<td>BEL</td>
<td>Bel</td>
</tr>
<tr>
<td>BEN</td>
<td>Ben</td>
</tr>
<tr>
<td>BIH</td>
<td>Bih</td>
</tr>
<tr>
<td>BIS</td>
<td>Bis</td>
</tr>
<tr>
<td>BOS</td>
<td>Bos</td>
</tr>
<tr>
<td>BRE</td>
<td>Bre</td>
</tr>
<tr>
<td>BUL</td>
<td>Bul</td>
</tr>
<tr>
<td>MYA</td>
<td>Mya</td>
</tr>
<tr>
<td>CAT</td>
<td>Cat</td>
</tr>
<tr>
<td>KHM</td>
<td>Khm</td>
</tr>
<tr>
<td>CHA</td>
<td>Cha</td>
</tr>
<tr>
<td>CHE</td>
<td>Che</td>
</tr>
<tr>
<td>NYA</td>
<td>nya</td>
</tr>
<tr>
<td>CHU</td>
<td>chu</td>
</tr>
<tr>
<td>CHV</td>
<td>chv</td>
</tr>
<tr>
<td>COR</td>
<td>cor</td>
</tr>
<tr>
<td>COS</td>
<td>cos</td>
</tr>
<tr>
<td>CRE</td>
<td>cre</td>
</tr>
<tr>
<td>HRV</td>
<td>hrv</td>
</tr>
<tr>
<td>CES</td>
<td>ces</td>
</tr>
<tr>
<td>DAN</td>
<td>dan</td>
</tr>
<tr>
<td>DIV</td>
<td>div</td>
</tr>
<tr>
<td>NLD</td>
<td>nld</td>
</tr>
<tr>
<td>DZ0</td>
<td>dz0</td>
</tr>
<tr>
<td>ENM</td>
<td>enm</td>
</tr>
<tr>
<td>EPO</td>
<td>epo</td>
</tr>
<tr>
<td>EST</td>
<td>est</td>
</tr>
<tr>
<td>EWE</td>
<td>ewe</td>
</tr>
<tr>
<td>FAO</td>
<td>fao</td>
</tr>
<tr>
<td>FIJ</td>
<td>fij</td>
</tr>
<tr>
<td>FIN</td>
<td>fin</td>
</tr>
<tr>
<td>FRM</td>
<td>frm</td>
</tr>
<tr>
<td>FUL</td>
<td>ful</td>
</tr>
<tr>
<td>GLA</td>
<td>gla</td>
</tr>
<tr>
<td>GLG</td>
<td>glg</td>
</tr>
<tr>
<td>LUG</td>
<td>lug</td>
</tr>
<tr>
<td>KAT</td>
<td>kat</td>
</tr>
<tr>
<td>ELL</td>
<td>ell</td>
</tr>
<tr>
<td>GRN</td>
<td>grn</td>
</tr>
<tr>
<td>GUJ</td>
<td>guj</td>
</tr>
<tr>
<td>HAT</td>
<td>hat</td>
</tr>
<tr>
<td>HAU</td>
<td>hau</td>
</tr>
<tr>
<td>HER</td>
<td>her</td>
</tr>
<tr>
<td>HMO</td>
<td>hmo</td>
</tr>
<tr>
<td>HUN</td>
<td>hun</td>
</tr>
<tr>
<td>ISL</td>
<td>isl</td>
</tr>
<tr>
<td>IDO</td>
<td>ido</td>
</tr>
<tr>
<td>IBO</td>
<td>ibo</td>
</tr>
<tr>
<td>Property</td>
<td></td>
</tr>
<tr>
<td>----------</td>
<td></td>
</tr>
<tr>
<td>IND</td>
<td></td>
</tr>
<tr>
<td>INA</td>
<td></td>
</tr>
<tr>
<td>ILE</td>
<td></td>
</tr>
<tr>
<td>IKU</td>
<td></td>
</tr>
<tr>
<td>IPK</td>
<td></td>
</tr>
<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>
</tbody>
</table>
TNG

**M2tsAudioBufferModel**

Selects between the DVB and ATSC buffer models for Dolby Digital audio.

- DVB
- ATSC

**M2tsBufferModel**

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 playback the stream without interruptions.

- MULTIPLEX
- NONE

**M2tsEbpAudioInterval**

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**

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**

Controls whether to include the ES Rate field in the PES header.

- INCLUDE
- EXCLUDE

**M2tsForceTsVideoEbpOrder**

Keep the default value (DEFAULT) unless you know that your audio EBP markers are incorrectly appearing before your video EBP markers. To correct this problem, set this value to Force (FORCE).

- FORCE
- DEFAULT
**M2tsNielsenId3**

If INSERT, Nielsen inaudible tones for media tracking will be detected in the input audio and an equivalent ID3 tag will be inserted in the output.

- INSERT
- NONE

**M2tsPcrControl**

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**

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

**M2tsScte35Esam**

Settings for SCTE-35 signals from ESAM. Include this in your job settings to put SCTE-35 markers in your HLS and transport stream outputs at the insertion points that you specify in an ESAM XML document. Provide the document in the setting SCC XML (sccXml).

- **scte35EsamPid**
  - Packet Identifier (PID) of the SCTE-35 stream in the transport stream generated by ESAM.
  
  - *Type:* integer
  - *Required:* False
  - *Minimum:* 32
  - *Maximum:* 8182

**M2tsScte35Source**

Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output.

- PASSTHROUGH
- NONE

**M2tsSegmentationMarkers**

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**

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**

MPEG-2 TS container settings. These apply to outputs in a File output group when the output's container (ContainerType) is MPEG-2 Transport Stream (M2TS). In these assets, data is organized by the program map table (PMT). Each transport stream program contains subsets of data, including audio, video, and metadata. Each of these subsets of data has a numerical label called a packet identifier (PID). Each transport stream program corresponds to one MediaConvert output. The PMT lists the types of data in a program along with their PID. Downstream systems and players use the program map table to look up the PID for each type of data it accesses and then uses the PIDs to locate specific data within the asset.

**audioBufferModel**

Selects between the DVB and ATSC buffer models for Dolby Digital audio.

*Type: M2tsAudioBufferModel (p. 760)*

*Required: False*

**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*
**esRateInPes**
Controls whether to include the ES Rate field in the PES header.

- **Type**: M2tsEsRateInPes (p. 760)
- **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

**dvbNitSettings**
Inserts DVB Network Information Table (NIT) at the specified table repetition interval.

- **Type**: DvbNitSettings (p. 690)
- **Required**: False

**dvbSdtSettings**
Inserts DVB Service Description Table (NIT) at the specified table repetition interval.

- **Type**: DvbSdtSettings (p. 691)
- **Required**: False

**scte35Source**
Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output.

- **Type**: M2tsScte35Source (p. 761)
- **Required**: False

**scte35Pid**
Specify the packet identifier (PID) of the SCTE-35 stream in the transport stream.

- **Type**: integer
  - **Required**: False
  - **Minimum**: 32
  - **Maximum**: 8182

**scte35Esam**
Include this in your job settings to put SCTE-35 markers in your HLS and transport stream outputs at the insertion points that you specify in an ESAM XML document. Provide the document in the setting SCC XML (sccXml).

- **Type**: M2tsScte35Esam (p. 761)
- **Required**: False
videoPid

Specify the packet identifier (PID) of the elementary video stream in the transport stream.

- **Type:** integer
- **Required:** False
- **Minimum:** 32
- **Maximum:** 8182

**dvbTdtSettings**

Inserts DVB Time and Date Table (TDT) at the specified table repetition interval.

- **Type:** DvbTdtSettings (p. 697)
- **Required:** False

**pmtInterval**

Specify the number of milliseconds between instances of the program map table (PMT) in the output transport stream.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 1000

**segmentationStyle**

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.

- **Type:** M2tsSegmentationStyle (p. 762)
- **Required:** False

**segmentationTime**

Specify the length, in seconds, of each segment. Required unless markers is set to _none_.

- **Type:** number
- **Required:** False
- **Format:** float
- **Minimum:** 0.0

**pmtPid**

Specify the packet identifier (PID) for the program map table (PMT) itself. Default is 480.

- **Type:** integer
Required: False
Minimum: 32
Maximum: 8182

**bitrate**
Specify 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

**audioPids**
Specify the packet identifiers (PIDs) for any elementary audio streams you include in this output. Specify multiple PIDs as a JSON array. Default is the range 482-492.

Type: Array of type integer
Required: False
Minimum: 32
Maximum: 8182

**privateMetadataPid**
Specify the packet identifier (PID) of the private metadata stream. Default is 503.

Type: integer
Required: False
Minimum: 32
Maximum: 8182

**nielsenId3**
If INSERT, Nielsen inaudible tones for media tracking will be detected in the input audio and an equivalent ID3 tag will be inserted in the output.

Type: M2tsNielsenId3 (p. 761)
Required: False

**timedMetadataPid**
Specify the packet identifier (PID) for timed metadata in this output. Default is 502.

Type: integer
Required: False
Minimum: 32
Maximum: 8182

**maxPcrInterval**
Specify the maximum time, in milliseconds, between Program Clock References (PCRs) inserted into the transport stream.
**Transport Stream Id**

Specify the ID for the transport stream itself in the program map table for this output. Transport stream IDs and program map tables are parts of MPEG-2 transport stream containers, used for organizing data.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 500

**DVB Sub PIDs**

Specify the packet identifiers (PIDs) for DVB subtitle data included in this output. Specify multiple PIDs as a JSON array. Default is the range 460-479.

- **Type:** Array of type integer
- **Required:** False
- **Minimum:** 32
- **Maximum:** 8182

**Rate Mode**

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.

- **Type:** M2tsRateMode (p. 761)
- **Required:** False

**Audio Frames per PES**

The number of audio frames to insert for each PES packet.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 2147483647

**PCR Control**

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.

- **Type:** M2tsPcrControl (p. 761)
- **Required:** False

**Segmentation Markers**

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.

**Type**: M2tsSegmentationMarkers (p. 761)
**Required**: False

**ebpAudioInterval**

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).

**Type**: M2tsEbpAudioInterval (p. 760)
**Required**: False

**forceTsVideoEbpOrder**

Keep the default value (DEFAULT) unless you know that your audio EBP markers are incorrectly appearing before your video EBP markers. To correct this problem, set this value to Force (FORCE).

**Type**: M2tsForceTsVideoEbpOrder (p. 760)
**Required**: False

**programNumber**

Use Program number (programNumber) to specify the program number used in the program map table (PMT) for this output. Default is 1. Program numbers and program map tables are parts of MPEG-2 transport stream containers, used for organizing data.

**Type**: integer
**Required**: False
**Minimum**: 0
**Maximum**: 65535

**pcrPid**

Specify the packet identifier (PID) for the program clock reference (PCR) in this output. If you do not specify a value, the service will use the value for Video PID (VideoPid).

**Type**: integer
**Required**: False
**Minimum**: 32
**Maximum**: 8182

**bufferModel**

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.

**Type**: M2tsBufferModel (p. 760)
**Required:** False

**dvbTeletextPid**

Specify the packet identifier (PID) for DVB teletext data you include in this output. Default is 499.

**Type:** integer
**Required:** False
**Minimum:** 32
**Maximum:** 8182

**fragmentTime**

The length, in seconds, of each fragment. Only used with EBP markers.

**Type:** number
**Required:** False
**Format:** float
**Minimum:** 0.0

**ebpPlacement**

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 selected (segmentationMarkers is EBP or EBP_LEGACY).

**Type:** M2tsEbpPlacement (p. 760)
**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

**M3u8NielsenId3**

If INSERT, Nielsen inaudible tones for media tracking will be detected in the input audio and an equivalent ID3 tag will be inserted in the output.

INSERT
NONE

**M3u8PcrControl**

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**

Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output.

- PASSTHROUGH
- NONE

**M3u8Settings**

Settings for TS segments in HLS

**audioFramesPerPes**

The number of audio frames to insert for each PES packet.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 2147483647

**pcrControl**

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.

- **Type**: M3u8PcrControl (p. 768)
- **Required**: False

**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.

- **Type**: integer
- **Required**: False
- **Minimum**: 32
- **Maximum**: 8182

**pmtPid**

Packet Identifier (PID) for the Program Map Table (PMT) in the transport stream.

- **Type**: integer
- **Required**: False
- **Minimum**: 32
- **Maximum**: 8182

**privateMetadataPid**

Packet Identifier (PID) of the private metadata stream in the transport stream.
**Type**: integer  
**Required**: False  
**Minimum**: 32  
**Maximum**: 8182

### programNumber

The value of the program number field in the Program Map Table.

**Type**: integer  
**Required**: False  
**Minimum**: 0  
**Maximum**: 65535

### patInterval

The number of milliseconds between instances of this table in the output transport stream.

**Type**: integer  
**Required**: False  
**Minimum**: 0  
**Maximum**: 1000

### pmtInterval

The number of milliseconds between instances of this table in the output transport stream.

**Type**: integer  
**Required**: False  
**Minimum**: 0  
**Maximum**: 1000

### scte35Source

Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output.

**Type**: M3u8Scte35Source (p. 769)  
**Required**: False

### scte35Pid

Packet Identifier (PID) of the SCTE-35 stream in the transport stream.

**Type**: integer  
**Required**: False  
**Minimum**: 32  
**Maximum**: 8182

### nielsenId3

If INSERT, Nielsen inaudible tones for media tracking will be detected in the input audio and an equivalent ID3 tag will be inserted in the output.

**Type**: M3u8NielsenId3 (p. 768)  
**Required**: False
timedMetadata
Applies only to HLS outputs. Use this setting to specify whether the service inserts the ID3 timed metadata from the input in this output.

Type: TimedMetadata (p. 805)
Required: False

timedMetadataPid
Packet Identifier (PID) of the timed metadata stream in the transport stream.

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.

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.

Type: Array of type integer
Required: False
Minimum: 32
Maximum: 8182

MotionImageInserter
Overlay motion graphics on top of your video at the time that you specify.

insertionMode
Choose the type of motion graphic asset that you are providing for your overlay. You can choose either a .mov file or a series of .png files.

Type: MotionImageInsertionMode (p. 773)
Required: False
Properties

input

Specify the .mov file or series of .png files that you want to overlay on your video. For .png files, provide the file name of the first file in the series. Make sure that the names of the .png files end with sequential numbers that specify the order that they are played in. For example, overlay_000.png, overlay_001.png, overlay_002.png, and so on. The sequence must start at zero, and each image file name must have the same number of digits. Pad your initial file names with enough zeros to complete the sequence. For example, if the first image is overlay_0.png, there can be only 10 images in the sequence, with the last image being overlay_9.png. But if the first image is overlay_00.png, there can be 100 images in the sequence.

Type: string
Required: False
Pattern: ^s3:\/[\s.]*\.(mov|\[0-9]+\.)\.
MinLength: 14
MaxLength: 1285

offset

Use Offset to specify the placement of your motion graphic overlay on the video frame. Specify in pixels, from the upper-left corner of the frame. If you don't specify an offset, the service scales your overlay to the full size of the frame. Otherwise, the service inserts the overlay at its native resolution and scales the size up or down with any video scaling.

Type: MotionImageInsertionOffset (p. 773)
Required: False

startTime

Specify when the motion overlay begins. Use timecode format (HH:MM:SS:FF or HH:MM:SS;FF). Make sure that the timecode you provide here takes into account how you have set up your timecode configuration under both job settings and input settings. The simplest way to do that is to set both to start at 0. If you need to set up your job to follow timecodes embedded in your source that don't start at zero, make sure that you specify a start time that is after the first embedded timecode. For more information, see https://docs.aws.amazon.com/mediaconvert/latest/ug/setting-up-timecode.html Find job-wide and input timecode configuration settings in your JSON job settings specification at settings>timecodeConfig>source and settings>inputs>timecodeSource.

Type: string
Required: False
Pattern: ^(((\[0-1\d]|2\[0-3]))(:[0-5]\d){2}(:;)[0-5]\d)$
MinLength: 11
MaxLength: 11

playback

Specify whether your motion graphic overlay repeats on a loop or plays only once.

Type: MotionImagePlayback (p. 774)
Required: False

framerate

If your motion graphic asset is a .mov file, keep this setting unspecified. If your motion graphic asset is a series of .png files, specify the frame rate of the overlay in frames per second, as a fraction. For example,
specify 24 fps as 24/1. Make sure that the number of images in your series matches the frame rate and your intended overlay duration. For example, if you want a 30-second overlay at 30 fps, you should have 900 .png images. This overlay frame rate doesn't need to match the frame rate of the underlying video.

**Type:** MotionImageInsertionFramerate (p. 773)  
**Required:** False

### MotionImageInsertionFramerate

For motion overlays that don't have a built-in frame rate, specify the frame rate of the overlay in frames per second, as a fraction. For example, specify 24 fps as 24/1. The overlay frame rate doesn't need to match the frame rate of the underlying video.

#### framerateNumerator

The top of the fraction that expresses your overlay frame rate. For example, if your frame rate is 24 fps, set this value to 24.

**Type:** integer  
**Required:** False  
**Minimum:** 1  
**Maximum:** 2147483640

#### framerateDenominator

The bottom of the fraction that expresses your overlay frame rate. For example, if your frame rate is 24 fps, set this value to 1.

**Type:** integer  
**Required:** False  
**Minimum:** 1  
**Maximum:** 17895697

### MotionImageInsertionMode

Choose the type of motion graphic asset that you are providing for your overlay. You can choose either a .mov file or a series of .png files.

- MOV
- PNG

### MotionImageInsertionOffset

Specify the offset between the upper-left corner of the video frame and the top left corner of the overlay.

#### imageX

Set the distance, in pixels, between the overlay and the left edge of the video frame.

**Type:** integer  
**Required:** False  
**Minimum:** 0  
**Maximum:** 2147483647
imageY
Set the distance, in pixels, between the overlay and the top edge of the video frame.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 2147483647

MotionImagePlayback
Specify whether your motion graphic overlay repeats on a loop or plays only once.

- ONCE
- REPEAT

MovClapAtom
When enabled, include 'clap' atom if appropriate for the video output settings.

- INCLUDE
- EXCLUDE

MovCslgAtom
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
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
If set to OMNEON, inserts Omneon-compatible padding

- OMNEON
- NONE

MovReference
Always keep the default value (SELF_CONTAINED) for this setting.
**MovSettings**

Settings for MOV Container.

**clapAtom**

When enabled, include 'clap' atom if appropriate for the video output settings.

Type: MovClapAtom (p. 774)

Required: False

**cslgAtom**

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.

Type: MovCslgAtom (p. 774)

Required: False

**paddingControl**

If set to OMNEON, inserts Omneon-compatible padding.

Type: MovPaddingControl (p. 774)

Required: False

**reference**

Always keep the default value (SELF_CONTAINED) for this setting.

Type: MovReference (p. 774)

Required: False

**mpeg2FourCCControl**

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.

Type: MovMpeg2FourCCControl (p. 774)

Required: False

**Mp2Settings**

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value MP2.

**bitrate**

Average bitrate in bits/second.
**Type**: integer  
**Required**: False  
**Minimum**: 32000  
**Maximum**: 384000

### 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

### sampleRate

Sample rate in hz.

**Type**: integer  
**Required**: False  
**Minimum**: 32000  
**Maximum**: 48000

### Mp4CslgAtom

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

Inserts a free-space box immediately after the moov box.

INCLUDE
EXCLUDE

### Mp4MoovPlacement

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

Settings for MP4 Container
**cslgAtom**

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.

- **Type**: Mp4CslgAtom (p. 776)
- **Required**: False

**freeSpaceBox**

Inserts a free-space box immediately after the moov box.

- **Type**: Mp4FreeSpaceBox (p. 776)
- **Required**: False

**mp4MajorBrand**

Overrides the "Major Brand" field in the output file. Usually not necessary to specify.

- **Type**: string
- **Required**: False

**moovPlacement**

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.

- **Type**: Mp4MoovPlacement (p. 776)
- **Required**: False

**Mpeg2AdaptiveQuantization**

Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

- OFF
- LOW
- MEDIUM
- HIGH

**Mpeg2CodecLevel**

Use Level (Mpeg2CodecLevel) to set the MPEG-2 level for the video output.

- AUTO
- LOW
- MAIN
- HIGH1440
- HIGH

**Mpeg2CodecProfile**

Use Profile (Mpeg2CodecProfile) to set the MPEG-2 profile for the video output.
**Mpeg2DynamicSubGop**

Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting B frames between reference frames (numberBFramesBetweenReferenceFrames).

- ADAPTIVE
- STATIC

**Mpeg2FramerateControl**

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

- INITIALIZE_FROM_SOURCE
- SPECIFIED

**Mpeg2FramerateConversionAlgorithm**

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

- DUPLICATE_DROP
- INTERPOLATE

**Mpeg2GopSizeUnits**

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**

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**

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**

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**

Use Quality tuning level (Mpeg2QualityTuningLevel) to specify whether to use single-pass or multipass video encoding.

- SINGLE_PASS
- MULTI_PASS

**Mpeg2RateControlMode**

Use Rate control mode (Mpeg2RateControlMode) to specify whether the bitrate is variable (vbr) or constant (cbr).

- VBR
- CBR

**Mpeg2SceneChangeDetect**

Scene change detection (inserts I-frames on scene changes).

- DISABLED
**ENABLED**

**Mpeg2Settings**

Required when you set (Codec) under (VideoDescription)> (CodecSettings) to the value MPEG2.

**interlaceMode**

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.

_Type:_ Mpeg2InterlaceMode (p. 778)
_Required:_ False

**parNumerator**

Pixel Aspect Ratio numerator.

_Type:_ integer
_Required:_ False
_Minimum:_ 1
_Maximum:_ 2147483647

**syntax**

Produces a Type D-10 compatible bitstream (SMPTE 356M-2001).

_Type:_ Mpeg2Syntax (p. 785)
_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

**framerateDenominator**

Frame rate denominator.

_Type:_ integer
_Required:_ False
_Minimum:_ 1
_Maximum:_ 1001
**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

**hrdBufferInitialFillPercentage**

Percentage of the buffer that should initially be filled (HRD buffer model).

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 100

**gopSize**

GOP Length (keyframe interval) in frames or seconds. Must be greater than zero.

- **Type**: number
- **Required**: False
- **Format**: float
- **Minimum**: 0.0

**hrdBufferSize**

Size of buffer (HRD buffer model) in bits. For example, enter five megabits as 5000000.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 47185920

**maxBitrate**

Maximum bitrate in bits/second. For example, enter five megabits per second as 5000000.

- **Type**: integer
- **Required**: False
- **Minimum**: 1000
- **Maximum**: 300000000

**slowPal**

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

- **Type**: Mpeg2SlowPal (p. 785)
Properties

Required: False

parDenominator

Pixel Aspect Ratio denominator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

spatialAdaptiveQuantization

Adjust quantization within each frame based on spatial variation of content complexity.

Type: Mpeg2SpatialAdaptiveQuantization (p. 785)
Required: False

temporalAdaptiveQuantization

Adjust quantization within each frame based on temporal variation of content complexity.

Type: Mpeg2TemporalAdaptiveQuantization (p. 785)
Required: False

bitrate

Average bitrate in bits/second. Required for VBR and CBR. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.

Type: integer
Required: False
Minimum: 1000
Maximum: 288000000

intraDcPrecision

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.

Type: Mpeg2IntraDcPrecision (p. 779)
Required: False

framerateControl

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job sepecification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.
rateControlMode

Use Rate control mode (Mpeg2RateControlMode) to specify whether the bitrate is variable (vbr) or constant (cbr).

Type: Mpeg2RateControlMode (p. 779)  
Required: False

codecProfile

Use Profile (Mpeg2CodecProfile) to set the MPEG-2 profile for the video output.

Type: Mpeg2CodecProfile (p. 777)  
Required: False

telecine

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.

Type: Mpeg2Telecine (p. 785)  
Required: False

framerateNumerator

Frame rate numerator - frame rate is a fraction, e.g. 24000 / 1001 = 23.976 fps.

Type: integer  
Required: False  
Minimum: 24  
Maximum: 60000

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

adaptiveQuantization

Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

Type: Mpeg2AdaptiveQuantization (p. 777)
Properties

**Required**: False

**codecLevel**

Use Level (Mpeg2CodecLevel) to set the MPEG-2 level for the video output.

- **Type**: Mpeg2CodecLevel (p. 777)
- **Required**: False

**sceneChangeDetect**

Scene change detection (inserts I-frames on scene changes).

- **Type**: Mpeg2SceneChangeDetect (p. 779)
- **Required**: False

**qualityTuningLevel**

Use Quality tuning level (Mpeg2QualityTuningLevel) to specify whether to use single-pass or multipass video encoding.

- **Type**: Mpeg2QualityTuningLevel (p. 779)
- **Required**: False

**framerateConversionAlgorithm**

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

- **Type**: Mpeg2FramerateConversionAlgorithm (p. 778)
- **Required**: False

**gopSizeUnits**

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.

- **Type**: Mpeg2GopSizeUnits (p. 778)
- **Required**: False

**parControl**

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.

- **Type**: Mpeg2ParControl (p. 779)
- **Required**: False

**numberBFramesBetweenReferenceFrames**

Number of B-frames between reference frames.

- **Type**: integer
- **Required**: False
**Minimum**: 0  
**Maximum**: 7

**dynamicSubGop**

Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting B frames between reference frames (numberOfFramesBetweenReferenceFrames).

**Type**: Mpeg2DynamicSubGop (p. 778)  
**Required**: False

**Mpeg2SlowPal**

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

- DISABLED
- ENABLED

**Mpeg2SpatialAdaptiveQuantization**

Adjust quantization within each frame based on spatial variation of content complexity.

- DISABLED
- ENABLED

**Mpeg2Syntax**

Produces a Type D-10 compatible bitstream (SMPTE 356M-2001).

- DEFAULT
- D_10

**Mpeg2Telecine**

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**

Adjust quantization within each frame based on temporal variation of content complexity.

- DISABLED
ENABLED

**MsSmoothAudioDeduplication**

COMBINE_DUPLICATE_STREAMS combines identical audio encoding settings across a Microsoft Smooth output group into a single audio stream.

- COMBINE_DUPLICATE_STREAMS
- NONE

**MsSmoothEncryptionSettings**

If you are using DRM, set DRM System (MsSmoothEncryptionSettings) to specify the value SpekeKeyProvider.

**spekeKeyProvider**

Settings for use with a SPEKE key provider

- **Type**: SpekeKeyProvider (p. 800)
- **Required**: False

**MsSmoothGroupSettings**

Required when you set (Type) under (OutputGroups)(OutputGroupSettings) to MS_SMOOTH_GROUP_SETTINGS.

**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**: False
- **Pattern**: ^s3: / / / 

**destinationSettings**

Settings associated with the destination. Will vary based on the type of destination

- **Type**: DestinationSettings (p. 690)
- **Required**: False

**fragmentLength**

Use Fragment length (FragmentLength) to specify the mp4 fragment sizes in seconds. Fragment length must be compatible with GOP size and frame rate.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 2147483647
encryption

If you are using DRM, set DRM System (MsSmoothEncryptionSettings) to specify the value SpekeKeyProvider.

    Type: MsSmoothEncryptionSettings (p. 786)
    Required: False

manifestEncoding

Use Manifest encoding (MsSmoothManifestEncoding) to specify the encoding format for the server and client manifest. Valid options are utf8 and utf16.

    Type: MsSmoothManifestEncoding (p. 787)
    Required: False

audioDeduplication

COMBINE_DUPLICATE_STREAMS combines identical audio encoding settings across a Microsoft Smooth output group into a single audio stream.

    Type: MsSmoothAudioDeduplication (p. 786)
    Required: False

MsSmoothManifestEncoding

Use Manifest encoding (MsSmoothManifestEncoding) to specify the encoding format for the server and client manifest. Valid options are utf8 and utf16.

    UTF8
    UTF16

NielsenConfiguration

Settings for Nielsen Configuration

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

distributorId

Use Distributor ID (DistributorID) to specify the distributor ID that is assigned to your organization by Neilsen.

    Type: string
    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. When you enable Noise reducer (NoiseReducer), you must also select a value for Noise reducer filter (NoiseReducerFilter).

filter

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 a frequency-domain filter based on JND principles.

- **BILATERAL**
- **MEAN**
- **GAUSSIAN**
- **LANCZOS**
- **SHARPEN**
- **CONSERVE**
- **SPATIAL**

NoiseReducerFilterSettings

Settings for a noise reducer filter

**strength**

Relative strength of noise reducing filter. Higher values produce stronger filtering.
Properties

Type: integer  
Required: False  
Minimum: 0  
Maximum: 3

**NoiseReducerSpatialFilterSettings**

Noise reducer filter settings for spatial filter.

**strength**

Relative strength of noise reducing filter. Higher values produce stronger filtering.

Type: integer  
Required: False  
Minimum: 0  
Maximum: 16

**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

**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

Output

An output object describes the settings for a single output file or stream in an output group.

**containerSettings**

Container specific settings.

Type: ContainerSettings (p. 685)  
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
Properties

Required: False
MinLength: 0

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. 808)
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. 662)
Required: False

outputSettings

Specific settings for this type of output.

Type: OutputSettings (p. 793)
Required: False

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

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 (NameModifier) 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
MinLength: 1

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**

- Type: Array of type `CaptionDescription` (p. 673)
- Required: False

**OutputChannelMapping**

OutputChannel mapping settings.

**inputChannels**

List of input channels

- Type: Array of type integer
- Required: False
- Minimum: -60
- Maximum: 6

**OutputDetail**

Details regarding output

**durationInMs**

Duration in milliseconds

- Type: integer
- Required: False

**videoDetails**

Contains details about the output's video stream

- Type: `VideoDetail` (p. 811)
- Required: False

**OutputGroup**

Group of outputs

**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

**name**

Name of the output group

- Type: string
- Required: False
outputs
This object holds groups of encoding settings, one group of settings per output.

  Type: Array of type Output (p. 789)
  Required: False

outputGroupSettings
Output Group settings, including type

  Type: OutputGroupSettings (p. 792)
  Required: False

OutputGroupDetail
Contains details about the output groups specified in the job settings.

outputDetails
Details about the output

  Type: Array of type OutputDetail (p. 791)
  Required: False

OutputGroupSettings
Output Group settings, including type

type
Type of output group (File group, Apple HLS, DASH ISO, Microsoft Smooth Streaming, CMAF)

  Type: OutputGroupType (p. 793)
  Required: False

hlsGroupSettings
Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to HLS_GROUP_SETTINGS.

  Type: HlsGroupSettings (p. 736)
  Required: False

dashIsoGroupSettings
Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to DASH_ISO_GROUP_SETTINGS.

  Type: DashIsoGroupSettings (p. 686)
  Required: False

fileGroupSettings
Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to FILE_GROUP_SETTINGS.
**Properties**

**Type**: FileGroupSettings (p. 705)  
**Required**: False

**msSmoothGroupSettings**

Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to MS_SMOOTH_GROUP_SETTINGS.

**Type**: MsSmoothGroupSettings (p. 786)  
**Required**: False

**cmafGroupSettings**

Required when you set (Type) under (OutputGroups)>(OutputGroupSettings) to CMAF_GROUP_SETTINGS. Each output in a CMAF Output Group may only contain a single video, audio, or caption output.

**Type**: CmafGroupSettings (p. 679)  
**Required**: False

**OutputGroupType**

Type of output group (File group, Apple HLS, DASH ISO, Microsoft Smooth Streaming, CMAF)

- HLS_GROUP_SETTINGS
- DASH_ISO_GROUP_SETTINGS
- FILE_GROUP_SETTINGS
- MS_SMOOTH_GROUP_SETTINGS
- CMAF_GROUP_SETTINGS

**OutputSdt**

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**

Specific settings for this type of output.

**hlsSettings**

Settings for HLS output groups

**Type**: HlsSettings (p. 742)  
**Required**: False
**ProresCodecProfile**

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**

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

INITIALIZE_FROM_SOURCE
SPECIFIED

**ProresFramerateConversionAlgorithm**

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

DUPLICATE_DROP
INTERPOLATE

**ProresInterlaceMode**

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**

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

Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value PRORES.

interlaceMode

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.

Type: ProresInterlaceMode (p. 794)
Required: False

parNumerator

Pixel Aspect Ratio numerator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

framerateDenominator

Frame rate denominator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

codecProfile

Use Profile (ProResCodecProfile) to specify the type of Apple ProRes codec to use for this output.

Type: ProresCodecProfile (p. 794)
Required: False

slowPal

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

Type: ProresSlowPal (p. 797)
Required: False
parDenominator

Pixel Aspect Ratio denominator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

framerateControl

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

Type: ProresFramerateControl (p. 794)
Required: False

telecine

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.

Type: ProresTelecine (p. 797)
Required: False

framerateNumerator

When you use the API for transcode jobs that use frame rate conversion, specify the frame rate 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

framerateConversionAlgorithm

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

Type: ProresFramerateConversionAlgorithm (p. 794)
Required: False

parControl

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).

   Type: ProresParControl (p. 794)  
   Required: False

**ProresSlowPal**

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

   DISABLED  
   ENABLED

**ProresTelecine**

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**

Use Rectangle to identify a specific area of the video frame.

**height**

Height of rectangle in pixels. Specify only even numbers.

   Type: integer  
   Required: False  
   Minimum: 2  
   Maximum: 2147483647

**width**

Width of rectangle in pixels. Specify only even numbers.

   Type: integer  
   Required: False  
   Minimum: 2  
   Maximum: 2147483647

**x**

The distance, in pixels, between the rectangle and the left edge of the video frame. Specify only even numbers.

   Type: integer  
   Required: False  
   Minimum: 0
**Maximum:** 2147483647

y

The distance, in pixels, between the rectangle and the top edge of the video frame. Specify only even numbers.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 2147483647

**RemixSettings**

Use Manual audio remixing (RemixSettings) to adjust audio levels for each audio channel in each output of your job. With audio remixing, you can output more or fewer audio channels than your input audio source provides.

**channelMapping**

Channel mapping (ChannelMapping) contains the group of fields that hold the remixing value for each channel. Units are in dB. Acceptable values are within the range from -60 (mute) through 6. A setting of 0 passes the input channel unchanged to the output channel (no attenuation or amplification).

- **Type:** ChannelMapping (p. 677)
- **Required:** False

**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:** False
- **Minimum:** 1
- **Maximum:** 16

**channelsOut**

Specify the number of channels in this output after remixing. Valid values: 1, 2, 4, 6, 8

- **Type:** integer
- **Required:** False
- **Minimum:** 1
- **Maximum:** 8

**RespondToAfd**

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
**Properties**

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

**S3DestinationSettings**

Settings associated with S3 destination

encryption

Settings for how your job outputs are encrypted as they are uploaded to Amazon S3.

* Type: S3EncryptionSettings (p. 799)
* Required: False

**S3EncryptionSettings**

Settings for how your job outputs are encrypted as they are uploaded to Amazon S3.

encryptionType

Specify how you want your data keys managed. AWS uses data keys to encrypt your content. AWS also encrypts the data keys themselves, using a customer master key (CMK), and then stores the encrypted data keys alongside your encrypted content. Use this setting to specify which AWS service manages the CMK. For simplest set up, choose Amazon S3 (SERVER_SIDE_ENCRYPTION_S3). If you want your master key to be managed by AWS Key Management Service (KMS), choose AWS KMS (SERVER_SIDE_ENCRYPTION_KMS). By default, when you choose AWS KMS, KMS uses the AWS managed customer master key (CMK) associated with Amazon S3 to encrypt your data keys. You can optionally choose to specify a different, customer managed CMK. Do so by specifying the Amazon Resource Name (ARN) of the key for the setting KMS ARN (kmsKeyArn).

* Type: S3ServerSideEncryptionType (p. 799)
* Required: False

kmsKeyArn

Optionally, specify the customer master key (CMK) that you want to use to encrypt the data key that AWS uses to encrypt your output content. Enter the Amazon Resource Name (ARN) of the CMK. To use this setting, you must also set Server-side encryption (S3ServerSideEncryptionType) to AWS KMS (SERVER_SIDE_ENCRYPTION_KMS). If you set Server-side encryption to AWS KMS but don't specify a CMK here, AWS uses the AWS managed CMK associated with Amazon S3.

* Type: string
* Required: False
* Pattern: ^arn:aws(-us-gov)?:kms:[a-z-]{2,6}-(east|west|central|((north|south) (east|west)?))[1-9][1,2]:\d{12}:key/[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{12}$

**S3ServerSideEncryptionType**

Specify how you want your data keys managed. AWS uses data keys to encrypt your content. AWS also encrypts the data keys themselves, using a customer master key (CMK), and then stores the
encrypted data keys alongside your encrypted content. Use this setting to specify which AWS service manages the CMK. For simplest set up, choose Amazon S3 (SERVER_SIDE_ENCRYPTION_S3). If you want your master key to be managed by AWS Key Management Service (KMS), choose AWS KMS (SERVER_SIDE_ENCRYPTION_KMS). By default, when you choose AWS KMS, KMS uses the AWS managed customer master key (CMK) associated with Amazon S3 to encrypt your data keys. You can optionally choose to specify a different, customer managed CMK. Do so by specifying the Amazon Resource Name (ARN) of the key for the setting KMS ARN (kmsKeyArn).

- SERVER_SIDE_ENCRYPTION_S3
- SERVER_SIDE_ENCRYPTION_KMS

**ScalingBehavior**

Applies only if your input aspect ratio is different from your output aspect ratio. Choose "Stretch to output" to have the service stretch your video image to fit. Keep the setting "Default" 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**

Set Framerate (SccDestinationFramerate) to make sure that the captions and the video are synchronized in the output. Specify a frame rate that matches the frame rate of the associated video. If the video frame rate 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**

Settings for SCC caption output.

**framerate**

Set Framerate (SccDestinationFramerate) to make sure that the captions and the video are synchronized in the output. Specify a frame rate that matches the frame rate of the associated video. If the video frame rate 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).

- **Type:** SccDestinationFramerate (p. 800)
- **Required:** False

**SpekeKeyProvider**

Settings for use with a SPEKE key provider
resourceld

The SPEKE-compliant server uses Resource ID (ResourceId) to identify content.

  Type: string
  Required: False

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: False
  Pattern: ^[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}$

url

Use URL (Url) to specify the SPEKE-compliant server that will provide keys for content.

  Type: string
  Required: False
  Format: uri
  Pattern: ^https:/$

certificateArn

Optional AWS Certificate Manager ARN for a certificate to send to the keyprovider. The certificate holds a key used by the keyprovider to encrypt the keys in its response.

  Type: string
  Required: False
  Pattern: ^arn:aws(-us-gov)?:acm:

**StaticKeyProvider**

Use these settings to set up encryption with a static key provider.

**staticKeyValue**

Relates to DRM implementation. Use a 32-character hexadecimal string to specify Key Value (StaticKeyValue).

  Type: string
  Required: False
  Pattern: ^[A-Za-z0-9]{32}$

**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
Properties

**Required**: False  
**Pattern**: `^identity|[A-Za-z]{2,6}(\.[A-Za-z0-9-]{1,63})+$`

**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+)*$`

**url**

Relates to DRM implementation. The location of the license server used for protecting content.

- **Type**: string  
- **Required**: False  
- **Format**: uri

**StatusUpdateInterval**

Specify how often MediaConvert sends STATUS_UPDATE events to Amazon CloudWatch Events. Set the interval, in seconds, between status updates. MediaConvert sends an update at this interval from the time the service begins processing your job to the time it completes the transcode or encounters an error.

- SECONDS_10  
- SECONDS_12  
- SECONDS_15  
- SECONDS_20  
- SECONDS_30  
- SECONDS_60  
- SECONDS_120  
- SECONDS_180  
- SECONDS_240  
- SECONDS_300  
- SECONDS_360  
- SECONDS_420  
- SECONDS_480  
- SECONDS_540  
- SECONDS_600

**TeletextDestinationSettings**

Settings for Teletext caption output

**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]$`  
**MinLength**: 3  
**MaxLength**: 3

**TeletextSourceSettings**

Settings specific to Teletext caption sources, including Page number.

**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]$`  
**MinLength**: 3  
**MaxLength**: 3

**TimecodeBurnin**

Timecode burn-in (TimecodeBurnIn)--Burns the output timecode and specified prefix into the output.

**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**

Use Position (Position) under Timecode burn-in (TimecodeBurnIn) to specify the location the burned-in timecode on output video.

**Type**: `TimecodeBurninPosition (p. 804)`  
**Required**: False

**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**: `^[\ -~]+$`
**TimecodeBurninPosition**

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**

These settings control how the service handles timecodes throughout the job. These settings don't affect input clipping.

**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 frame rate conversion.

System behavior for Anchor Timecode varies depending on your setting for `Source (TimecodeSource)`.

* If `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 `Source (TimecodeSource)` is set to `Start at 0 (ZEROBASED)` the first frame is `00:00:00:00`. * If `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}$`

**source**

Use `Source (TimecodeSource)` to set how timecodes are handled within this job. 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.

**Type:** TimecodeSource (p. 805)

**Required:** False

**start**

Only use when you set `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)`. 
**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}$`

**timestampOffset**

Only applies to outputs that support program-date-time stamp. Use Timestamp offset (TimestampOffset) to overwrite the timecode date without affecting the time and frame number. Provide the new date as a string in the format "yyyy-mm-dd". To use Timestamp offset, you must also enable Insert program-date-time (InsertProgramDateTime) in the output settings. For example, if the date part of your timecodes is 2002-1-25 and you want to change it to one year later, set Timestamp offset (TimestampOffset) to 2003-1-25.

**Type**: string  
**Required**: False  
**Pattern**: `^([0-9]{4})-(0[1-9]|1[0-2])-([0-9][1][0-9]|[2][0-3][0-9]|3[01][0-9])$`

**TimecodeSource**

Use Source (TimecodeSource) to set how timecodes are handled within this job. 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**

Applies only to HLS outputs. Use this setting to specify whether the service inserts the ID3 timed metadata from the input in this output.

PASSTHROUGH  
NONE

**TimedMetadataInsertion**

Enable Timed metadata insertion (TimedMetadataInsertion) to include ID3 tags in your job. To include timed metadata, you must enable it here, enable it in each output container, and specify tags and timecodes in ID3 insertion (Id3Insertion) objects.

**id3Insertions**

Id3Insertions contains the array of Id3Insertion instances.

**Type**: Array of type Id3Insertion (p. 743)  
**Required**: False
## Timing

Information about when jobs are submitted, started, and finished is specified in Unix epoch format in seconds.

**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

**finishTime**

The time, in Unix epoch format, that the transcoding job finished.

- **Type:** string
- **Required:** False
- **Format:** date-time

## TrackSourceSettings

Settings specific to caption sources that are specified by track number. Sources include IMSC in IMF.

**trackNumber**

Use this setting to select a single captions track from a source. Track numbers correspond to the order in the captions source file. For IMF sources, track numbering is based on the order that the captions appear in the CPL. For example, use 1 to select the captions asset that is listed first in the CPL. To include more than one captions track in your job outputs, create multiple input captions selectors. Specify one track per selector.

- **Type:** integer
- **Required:** False
- **Minimum:** 1
- **Maximum:** 2147483647

## TtmlDestinationSettings

Settings specific to TTML caption outputs, including Pass style information (TtmlStylePassthrough).

**stylePassthrough**

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.
**Properties**

**Type:** TtmlStylePassthrough (p. 807)

**Required:** False

**TtmlStylePassthrough**

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**

Type of video codec

FRAME_CAPTURE
H_264
H_265
MPEG2
PRORES

**VideoCodecSettings**

Video codec settings, (CodecSettings) under (VideoDescription), contains the group of settings related to video encoding. The settings in this group vary depending on the value you choose for Video codec (Codec). For each codec enum you choose, define the corresponding settings object. The following lists the codec enum, settings object pairs. * H_264, H264Settings * H_265, H265Settings * MPEG2, Mpeg2Settings * PRORES, ProresSettings * FRAME_CAPTURE, FrameCaptureSettings

**codec**

Specifies the video codec. This must be equal to one of the enum values defined by the object VideoCodec.

**Type:** VideoCodec (p. 807)

**Required:** False

**frameCaptureSettings**

Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value FRAME_CAPTURE.

**Type:** FrameCaptureSettings (p. 707)

**Required:** False

**h264Settings**

Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value H_264.

**Type:** H264Settings (p. 712)

**Required:** False

**h265Settings**

Settings for H265 codec
Properties

Type: **H265Settings** *(p. 723)*  
**Required:** False

**mpeg2Settings**

Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value MPEG2.

Type: **Mpeg2Settings** *(p. 780)*  
**Required:** False

**proresSettings**

Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value PRORES.

Type: **ProresSettings** *(p. 795)*  
**Required:** False

**VideoDescription**

Settings for video outputs

**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

**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

**scalingBehavior**

Applies only if your input aspect ratio is different from your output aspect ratio. Choose "Stretch to output" to have the service stretch your video image to fit. Keep the setting "Default" to allow the service to letterbox your video instead. This setting overrides any positioning value you specify elsewhere in the job.

Type: **ScalingBehavior** *(p. 800)*  
**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. If you crop your input frame size to smaller than
your output frame size, make sure to specify the behavior you want in your output setting "Scaling
behavior".

Type: Rectangle (p. 797)
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

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

timecodeInsertion

Applies only to H.264, H.265, MPEG2, and ProRes outputs. Only enable Timecode insertion when the
input frame rate is identical to the output frame rate. To include timecodes in this output, set Timecode
insertion (VideoTimecodeInsertion) to PIC_TIMING_SEI. To leave them out, set it to DISABLED. Default is
DISABLED. When the service inserts timecodes in an output, by default, it uses any embedded timecodes
from the input. If none are present, the service will set the timecode for the first output frame to zero. To
change this default behavior, adjust the settings under Timecode configuration (TimecodeConfig). In the
console, these settings are located under Job > Job settings > Timecode configuration. Note - Timecode
source under input settings (InputTimecodeSource) does not affect the timecodes that are inserted in the
output. Source under Job settings > Timecode configuration (TimecodeSource) does.

Type: VideoTimecodeInsertion (p. 813)
Required: False

antiAlias

The anti-alias filter is automatically applied to all outputs. The service no longer accepts the value
DISABLED for AntiAlias. If you specify that in your job, the service will ignore the setting.

Type: AntiAlias (p. 660)
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. 797)
Properties

**Required:** False

### 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. 0 is the softest setting, 100 the sharpest, and 50 recommended for most content.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 100

### codecSettings

Video codec settings, (CodecSettings) under (VideoDescription), contains the group of settings related to video encoding. The settings in this group vary depending on the value you choose for Video codec (Codec). For each codec enum you choose, define the corresponding settings object. The following lists the codec enum, settings object pairs. * H_264, H264Settings * H_265, H265Settings * MPEG2, Mpeg2Settings * PRORES, ProresSettings * FRAME_CAPTURE, FrameCaptureSettings

- **Type:** VideoCodecSettings (p. 807)
- **Required:** False

### afdSignaling

This setting only applies to H.264, H.265, and MPEG2 outputs. Use Insert AFD signaling (AfdSignaling) to specify whether the service includes 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.

- **Type:** AfdSignaling (p. 659)
- **Required:** False

### dropFrameTimecode

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.

- **Type:** DropFrameTimecode (p. 690)
- **Required:** False

### respondToAfd

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.

- **Type:** RespondToAfd (p. 798)
Required: False

colorMetadata
Enable Insert color metadata (ColorMetadata) to include color metadata in this output. This setting is enabled by default.

Type: ColorMetadata (p. 684)
Required: False

VideoDetail
Contains details about the output's video stream

widthInPx
Width in pixels for the output

Type: integer
Required: False

heightInPx
Height in pixels for the output

Type: integer
Required: False

VideoPreprocessor
Find additional transcoding features under Preprocessors (VideoPreprocessors). Enable the features at each output individually. These features are disabled by default.

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. 683)
Required: False

deinterlacer
Use Deinterlacer (Deinterlacer) to produce smoother motion and a clearer picture.

Type: Deinterlacer (p. 689)
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. 743)
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. 788)
Required: False

timecodeBurnin

Timecode burn-in (TimecodeBurnIn)--Burns the output timecode and specified prefix into the output.

Type: TimecodeBurnin (p. 803)
Required: False

VideoSelector

Selector for video.

colorSpace

If your input video has accurate color space metadata, or if you don’t know about color space, leave this set to the default value FOLLOW. The service will automatically detect your input color space. If your input video has metadata indicating the wrong color space, or if your input video is missing color space metadata that should be there, specify the accurate color space here. If you choose HDR10, you can also correct inaccurate color space coefficients, using the HDR master display information controls. You must also set Color space usage (ColorSpaceUsage) to FORCE for the service to use these values.

Type: ColorSpace (p. 684)
Required: False

rotate

Use Rotate (InputRotate) to specify how the service rotates your video. You can choose automatic rotation or specify a rotation. You can specify a clockwise rotation of 0, 90, 180, or 270 degrees. If your input video container is .mov or .mp4 and your input has rotation metadata, you can choose Automatic to have the service rotate your video according to the rotation specified in the metadata. The rotation must be within one degree of 90, 180, or 270 degrees. If the rotation metadata specifies any other rotation, the service will default to no rotation. By default, the service does no rotation, even if your input video has rotation metadata. The service doesn't pass through rotation metadata.

Type: InputRotate (p. 749)
Required: False

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
**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

**colorSpaceUsage**

There are two sources for color metadata, the input file and the job configuration (in the Color space and HDR master display information settings). The Color space usage setting controls which takes precedence. **FORCE:** The 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:** The 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.

- **Type:** ColorSpaceUsage (p. 685)
- **Required:** False

**hdr10Metadata**

Use the "HDR master display information" (Hdr10Metadata) settings to correct HDR metadata or to provide missing metadata. These values vary depending on the input video and must be provided by a color grader. Range is 0 to 50,000; each increment represents 0.00002 in CIE1931 color coordinate. Note that these settings are not color correction. Note that if you are creating HDR outputs inside of an HLS CMAF package, to comply with the Apple specification, you must use the following settings. Set "MP4 packaging type" (writeMp4PackagingType) to HVC1 (HVC1). Set "Profile" (H265Settings > codecProfile) to Main10/High (MAIN10_HIGH). Set "Level" (H265Settings > codecLevel) to 5 (LEVEL_5).

- **Type:** Hdr10Metadata (p. 731)
- **Required:** False

**VideoTimecodeInsertion**

Applies only to H.264, H.265, MPEG2, and ProRes outputs. Only enable Timecode insertion when the input frame rate is identical to the output frame rate. To include timecodes in this output, set Timecode insertion (VideoTimecodeInsertion) to PIC_TIMING_SEI. To leave them out, set it to DISABLED. Default is DISABLED. When the service inserts timecodes in an output, by default, it uses any embedded timecodes from the input. If none are present, the service will set the timecode for the first output frame to zero. To change this default behavior, adjust the settings under Timecode configuration (TimecodeConfig). In the console, these settings are located under Job > Job settings > Timecode configuration. Note - Timecode source under input settings (InputTimecodeSource) does not affect the timecodes that are inserted in the output. Source under Job settings > Timecode configuration (TimecodeSource) does.

- **DISABLED**
- **PIC_TIMING_SEI**

**WavFormat**

The service defaults to using RIFF for WAV outputs. If your output audio is likely to exceed 4 GB in file size, or if you otherwise need the extended support of the RF64 format, set your output WAV file format to RF64.
RIFF
RF64

**WavSettings**

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value WAV.

**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

**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

**sampleRate**

Sample rate in Hz.

- **Type:** integer
- **Required:** False
- **Minimum:** 8000
- **Maximum:** 192000

**format**

The service defaults to using RIFF for WAV outputs. If your output audio is likely to exceed 4 GB in file size, or if you otherwise need the extended support of the RF64 format, set your output WAV file format to RF64.

- **Type:** WavFormat (p. 813)
- **Required:** False

**See Also**

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

**GetJob**

- AWS Command Line Interface
Presets

URI

/2017-08-29/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>
</tbody>
</table>
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>ListPresetsResponse</td>
<td>200 response</td>
</tr>
<tr>
<td>400</td>
<td>ExceptionBody</td>
<td>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>403</td>
<td>ExceptionBody</td>
<td>You don't have permissions for this action with the credentials you sent.</td>
</tr>
<tr>
<td>404</td>
<td>ExceptionBody</td>
<td>The resource you requested does not exist.</td>
</tr>
<tr>
<td>409</td>
<td>ExceptionBody</td>
<td>The service could not complete your request because there is a conflict with the current state of the resource.</td>
</tr>
<tr>
<td>429</td>
<td>ExceptionBody</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>
<tr>
<td>500</td>
<td>ExceptionBody</td>
<td>The service encountered an unexpected condition and cannot fulfill your request.</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</td>
<td>201 response</td>
</tr>
<tr>
<td>400</td>
<td>ExceptionBody</td>
<td>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>403</td>
<td>ExceptionBody</td>
<td>You don't have permissions for this action with the credentials you sent.</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>404</td>
<td>ExceptionBody (p. 840)</td>
<td>The resource you requested does not exist.</td>
</tr>
<tr>
<td>409</td>
<td>ExceptionBody (p. 840)</td>
<td>The service could not complete your request because there is a conflict with the current state of the resource.</td>
</tr>
<tr>
<td>429</td>
<td>ExceptionBody (p. 840)</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>
<tr>
<td>500</td>
<td>ExceptionBody (p. 840)</td>
<td>The service encountered an unexpected condition and cannot fulfill your request.</td>
</tr>
</tbody>
</table>

### Schemas

#### Request Bodies

**Example GET**

```json
{
    "listBy": enum,
    "category": "string",
    "order": enum,
    "nextToken": "string",
    "maxResults": integer
}
```

**Example POST**

```json
{
    "description": "string",
    "category": "string",
    "name": "string",
    "settings": {
        "videoDescription": {
            "fixedAfd": integer,
            "width": integer,
            "scalingBehavior": enum,
            "crop": {
                "height": integer,
                "width": integer,
                "x": integer,
                "y": integer
            },
            "height": integer,
            "videoPreprocessors": {
                "colorCorrector": {
                    "brightness": integer,
                    "colorSpaceConversion": enum,
                    "contrast": integer,
                    "hue": integer,
                    "saturation": integer,
```
"hdr10Metadata": {
  "redPrimaryX": integer,
  "redPrimaryY": integer,
  "greenPrimaryX": integer,
  "greenPrimaryY": integer,
  "bluePrimaryX": integer,
  "bluePrimaryY": integer,
  "whitePointX": integer,
  "whitePointY": integer,
  "maxFrameAverageLightLevel": integer,
  "maxContentLightLevel": integer,
  "maxLuminance": integer,
  "minLuminance": integer
},
"deinterlacer": {
  "algorithm": enum,
  "mode": enum,
  "control": enum
},
"imageInserter": {
  "insertableImages": [
    {
      "width": integer,
      "height": integer,
      "imageX": integer,
      "imageY": integer,
      "duration": integer,
      "fadeIn": integer,
      "layer": integer,
      "imageInserterInput": "string",
      "startTime": "string",
      "fadeOut": integer,
      "opacity": integer
    }
  ]
},
"noiseReducer": {
  "filter": enum,
  "filterSettings": {
    "strength": integer
  },
  "spatialFilterSettings": {
    "strength": integer,
    "speed": integer,
    "postFilterSharpenStrength": integer
  }
},
"timecodeBurnin": {
  "fontSize": integer,
  "position": enum,
  "prefix": "string"
},
"timecodeInsertion": enum,
"antiAlias": enum,
"position": {
  "height": integer,
  "width": integer,
  "x": integer,
  "y": integer
},
"sharpness": integer,
"codecSettings": {
  "codec": enum,
  "frameCaptureSettings": {

"framerateNumerator": integer,
"framerateDenominator": integer,
"maxCaptures": integer,
"quality": integer
},
"h264Settings": {
"interlaceMode": enum,
"parNumerator": integer,
"numberReferenceFrames": integer,
"syntax": enum,
"softness": integer,
"framerateDenominator": integer,
"gopClosedCadence": integer,
"hrdBufferInitialFillPercentage": integer,
"gopSize": number,
"slices": integer,
"gopBReference": enum,
"hrdBufferSize": integer,
"maxBitrate": integer,
"slowPal": enum,
"parDenominator": integer,
"spatialAdaptiveQuantization": enum,
"temporalAdaptiveQuantization": enum,
"entropyEncoding": enum,
"framerateControl": enum,
"rateControlMode": enum,
"qvbrSettings": {
  "qvbrQualityLevel": integer,
  "maxAverageBitrate": integer
},
"codecProfile": enum,
"telecine": enum,
"framerateNumerator": integer,
"minIInterval": integer,
"adaptiveQuantization": enum,
"codecLevel": enum,
"fieldEncoding": enum,
"sceneChangeDetect": enum,
"qualityTuningLevel": enum,
"framerateConversionAlgorithm": enum,
"unregisteredSeiTimecode": enum,
"gopSizeUnits": enum,
"parControl": enum,
"numberBFramesBetweenReferenceFrames": integer,
"repeatPps": enum,
"dynamicSubGop": enum
},
"h265Settings": {
"interlaceMode": enum,
"parNumerator": integer,
"numberReferenceFrames": integer,
"framerateDenominator": integer,
"gopClosedCadence": integer,
"alternateTransferFunctionSei": enum,
"hrdBufferInitialFillPercentage": integer,
"gopSize": number,
"slices": integer,
"gopBReference": enum,
"hrdBufferSize": integer,
"maxBitrate": integer,
"slowPal": enum,
"parDenominator": integer,
"spatialAdaptiveQuantization": enum,
"temporalAdaptiveQuantization": enum,
"flickerAdaptiveQuantization": enum,
"bitrate": integer,
"framerateControl": enum,
"rateControlMode": enum,
"qvbrSettings": {
  "qvbrQualityLevel": integer,
  "maxAverageBitrate": integer
},
"codecProfile": enum,
"tiles": enum,
"telecine": enum,
"framerateNumerator": integer,
"minIInterval": integer,
"adaptiveQuantization": enum,
"codecLevel": enum,
"sceneChangeDetect": enum,
"qualityTuningLevel": enum,
"framerateConversionAlgorithm": enum,
"unregisteredSeiTimecode": enum,
"gopSizeUnits": enum,
"parControl": enum,
"numberBFramesBetweenReferenceFrames": integer,
"temporalIds": enum,
"sampleAdaptiveOffsetFilterMode": enum,
"writeMp4PackagingType": enum,
"dynamicSubGop": enum
},
"mpeg2Settings": {
"interlaceMode": enum,
"parNumerator": integer,
"syntax": enum,
"softness": integer,
"framerateDenominator": integer,
"gopClosedCadence": integer,
"hrdBufferInitialFillPercentage": integer,
"gopSize": number,
"hrdBufferSize": integer,
"maxBitrate": integer,
"slowPal": enum,
"parDenominator": integer,
"spatialAdaptiveQuantization": enum,
"temporalAdaptiveQuantization": enum,
"bitrate": integer,
"intraDcPrecision": enum,
"framerateControl": enum,
"rateControlMode": enum,
"codecProfile": enum,
"telecine": enum,
"framerateNumerator": integer,
"minIInterval": integer,
"adaptiveQuantization": enum,
"codecLevel": enum,
"sceneChangeDetect": enum,
"qualityTuningLevel": enum,
"framerateConversionAlgorithm": enum,
"gopSizeUnits": enum,
"parControl": enum,
"numberBFramesBetweenReferenceFrames": integer,
"dynamicSubGop": enum
},
"proresSettings": {
"interlaceMode": enum,
"parNumerator": integer,
"framerateDenominator": integer,
"codecProfile": enum,
"slowPal": enum,
"parDenominator": integer,
"framerateControl": enum,
"telecine": enum,
"framerateNumerator": integer,
"framerateConversionAlgorithm": enum,
"parControl": enum
},
"afdSignaling": enum,
"dropFrameTimecode": enum,
"respondToAfd": enum,
"colorMetadata": enum
},
"audioDescriptions": [
{
"audioTypeControl": enum,
"audioSourceName": "string",
"audioNormalizationSettings": {
"algorithm": enum,
"algorithmControl": enum,
"correctionGateLevel": integer,
"loudnessLogging": enum,
"targetLkfs": number,
"peakCalculation": enum
},
"codecSettings": {
"codec": enum,
"aacSettings": {
"audioDescriptionBroadcasterMix": enum,
"vbrQuality": enum,
"bitrate": integer,
"rateControlMode": enum,
"codecProfile": enum,
"codingMode": enum,
"rawFormat": enum,
"sampleRate": integer,
"specification": enum
},
"ac3Settings": {
"bitrate": integer,
"bitstreamMode": enum,
"codingMode": enum,
"dialnorm": integer,
"dynamicRangeCompressionProfile": enum,
"metadataControl": enum,
"lfeFilter": enum,
"sampleRate": integer
},
"siffSettings": {
"bitDepth": integer,
"channels": integer,
"sampleRate": integer
},
"eac3Settings": {
"metadataControl": enum,
"surroundEXMode": enum,
"loRoSurroundMixLevel": number,
"phaseControl": enum,
"dialnorm": integer,
"ltRtSurroundMixLevel": number,
"bitrate": integer,
"ltRtCenterMixLevel": number,
"passthroughControl": enum,
"lfeControl": enum,
"loRoCenterMixLevel": number,
"attenuationControl": enum,
"codingMode": enum,
"surroundMode": enum,
"bitstreamMode": enum,
"lfeFilter": enum,
"stereoDownmix": enum,
"dynamicRangeCompressionRf": enum,
"sampleRate": integer,
"dynamicRangeCompressionLine": enum,
"dcFilter": enum
},
"mp2Settings": {
  "bitrate": integer,
  "channels": integer,
  "sampleRate": integer
},
"wavSettings": {
  "bitDepth": integer,
  "channels": integer,
  "sampleRate": integer,
  "format": enum
},
"remixSettings": {
  "channelMapping": {
    "outputChannels": [
      "inputChannels": [
        integer
      ]
    ]
  },
  "channelsIn": integer,
  "channelsOut": integer
},
"streamName": "string",
"languageCodeControl": enum,
"audioType": integer,
"customLanguageCode": "string",
"languageCode": enum
},
"containerSettings": {
  "container": enum,
  "m3u8Settings": {
    "audioFramesPerPes": integer,
    "pcrControl": enum,
    "pcrPid": integer,
    "pmtPid": integer,
    "privateMetadataPid": integer,
    "programNumber": integer,
    "patInterval": integer,
    "pmtInterval": integer,
    "scte35Source": enum,
    "scte35Pid": integer,
    "nielsenId3": enum,
    "timedMetadata": enum,
    "timedMetadataPid": integer,
    "transportStreamId": integer,
    "videoPid": integer,
    "audioPids": [
      integer
    ]
  },
  "f4vSettings": {
    "moovPlacement": enum
"m2tsSettings": {
  "audioBufferModel": enum,
  "minEbpInterval": integer,
  "esRateInPes": enum,
  "patInterval": integer,
  "dvbNltSettings": {
    "nitInterval": integer,
    "networkId": integer,
    "networkName": "string"
  },
  "dvbSdtSettings": {
    "outputSdt": enum,
    "sdtInterval": integer,
    "serviceName": "string",
    "serviceProviderName": "string"
  },
  "scte35Source": enum,
  "scte35Pid": integer,
  "scte35Esam": {
    "scte35EsamPid": integer
  },
  "videoPid": integer,
  "dvbTdtSettings": {
    "tdtInterval": integer
  },
  "pmtInterval": integer,
  "segmentationStyle": enum,
  "segmentationTime": number,
  "pmtPid": integer,
  "bitrate": integer,
  "audioPids": [ integer
  ],
  "privateMetadataPid": integer,
  "nsielsenid3": enum,
  "timedMetadataPid": integer,
  "maxPcrInterval": integer,
  "transportStreamId": integer,
  "dvbSubPids": [ integer
  ],
  "rateMode": enum,
  "audioFramesPerPes": integer,
  "pcrControl": enum,
  "segmentationMarkers": enum,
  "ebpAudioInterval": enum,
  "forceTsVideoEbpOrder": enum,
  "programNumber": integer,
  "pcrPid": integer,
  "bufferModel": enum,
  "dvbTeletextPid": integer,
  "fragmentTime": number,
  "ebpPlacement": enum,
  "nullPacketBitrate": number
},
"movSettings": {
  "clapAtom": enum,
  "cslgAtom": enum,
  "paddingControl": enum,
  "reference": enum,
  "mpeg2FourCCControl": enum
},
"mp4Settings": {
  "cslgAtom": enum,
  "freeSpaceBox": enum,

823
"mp4MajorBrand": "string",
"moovPlacement": enum
},
"captionDescriptions": [
{
"destinationSettings": {
"destinationType": enum,
"burninDestinationSettings": {
"backgroundOpacity": integer,
"shadowXOffset": integer,
"teletextSpacing": enum,
"alignment": enum,
"outlineSize": integer,
"yPosition": integer,
"shadowColor": enum,
"fontOpacity": integer,
"fontSize": integer,
"fontScript": enum,
"fontColor": enum,
"backgroundColor": enum,
"fontResolution": integer,
"outlineColor": enum,
"shadowYOffset": integer,
"xPosition": integer,
"shadowOpacity": integer
},
"dvbSubDestinationSettings": {
"backgroundOpacity": integer,
"shadowXOffset": integer,
"teletextSpacing": enum,
"alignment": enum,
"outlineSize": integer,
"yPosition": integer,
"shadowColor": enum,
"fontOpacity": integer,
"fontSize": integer,
"fontScript": enum,
"fontColor": enum,
"backgroundColor": enum,
"fontResolution": integer,
"outlineColor": enum,
"shadowYOffset": integer,
"xPosition": integer,
"shadowOpacity": integer
},
"sccDestinationSettings": {
"framerate": enum
},
"teletextDestinationSettings": {
"pageNumber": "string"
},
"ttmlDestinationSettings": {
"stylePassthrough": enum
},
"embeddedDestinationSettings": {
"destination608ChannelNumber": integer
},
"customLanguageCode": "string",
"languageCode": enum,
"languageDescription": "string"
}
],
"tags": {
Response Bodies

Example ListPresetsResponse

```json
{
  "presets": [
    {
      "arn": "string",
      "createdAt": "string",
      "lastUpdated": "string",
      "description": "string",
      "category": "string",
      "name": "string",
      "type": enum,
      "settings": {
        "videoDescription": {
          "fixedAfd": integer,
          "width": integer,
          "scalingBehavior": enum,
          "crop": {
            "height": integer,
            "width": integer,
            "x": integer,
            "y": integer
          },
          "height": integer,
          "videoPreprocessors": {
            "colorCorrector": {
              "brightness": integer,
              "colorSpaceConversion": enum,
              "contrast": integer,
              "hue": integer,
              "saturation": integer,
              "hdr10Metadata": {
                "redPrimaryX": integer,
                "redPrimaryY": integer,
                "greenPrimaryX": integer,
                "greenPrimaryY": integer,
                "bluePrimaryX": integer,
                "bluePrimaryY": integer,
                "whitePointX": integer,
                "whitePointY": integer,
                "maxFrameAverageLightLevel": integer,
                "maxContentLightLevel": integer,
                "maxLuminance": integer,
                "minLuminance": integer
              }
            },
            "deinterlacer": {
              "algorithm": enum,
              "mode": enum,
              "control": enum
            },
            "imageInserter": {
              "insertableImages": [
                {
                  "width": integer,
                  "height": integer,
                  "imageX": integer,
                  "imageY": integer
                }
              ]
            }
          }
        }
      }
    }
  ]
}```
"duration": integer,
"fadeIn": integer,
"layer": integer,
"imageInserterInput": "string",
"startTime": "string",
"fadeOut": integer,
"opacity": integer
}

"noiseReducer": {
  "filter": enum,
  "filterSettings": {
    "strength": integer
  },
  "spatialFilterSettings": {
    "strength": integer,
    "speed": integer,
    "postFilterSharpenStrength": integer
  }
},
"timecodeBurnin": {
  "fontSize": integer,
  "position": enum,
  "prefix": "string"
},
"timecodeInsertion": enum,
"antiAlias": enum,
"position": {
  "height": integer,
  "width": integer,
  "x": integer,
  "y": integer
},
"sharpness": integer,
"codecSettings": {
  "codec": enum,
  "frameCaptureSettings": {
    "framerateNumerator": integer,
    "framerateDenominator": integer,
    "maxCaptures": integer,
    "quality": integer
  },
  "h264Settings": {
    "interlaceMode": enum,
    "parNumerator": integer,
    "numberReferenceFrames": integer,
    "syntax": enum,
    "softness": integer,
    "framerateDenominator": integer,
    "gopClosedCadence": integer,
    "hrdBufferInitialFillPercentage": integer,
    "gopSize": number,
    "slices": integer,
    "gopBReference": enum,
    "hrdBufferSize": integer,
    "maxBitrate": integer,
    "slowPal": enum,
    "parDenominator": integer,
    "spatialAdaptiveQuantization": enum,
    "temporalAdaptiveQuantization": enum,
    "flickerAdaptiveQuantization": enum,
    "entropyEncoding": enum,
    "bitrate": integer,
    "framerateControl": enum,
"rateControlMode": enum,
"qvbrSettings": {
  "qvbrQualityLevel": integer,
  "maxAverageBitrate": integer
},
"codecProfile": enum,
"telecine": enum,
"framerateNumerator": integer,
"minIInterval": integer,
"adaptiveQuantization": enum,
"codecLevel": enum,
"fieldEncoding": enum,
"sceneChangeDetect": enum,
"qualityTuningLevel": enum,
"framerateConversionAlgorithm": enum,
"unregisteredSeiTimecode": enum,
"gopSizeUnits": enum,
"parControl": enum,
"numberBFramesBetweenReferenceFrames": integer,
"repeatPps": enum,
"dynamicSubGop": enum
},
"h265Settings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "numberReferenceFrames": integer,
  "framerateDenominator": integer,
  "gopClosedCadence": integer,
  "alternateTransferFunctionSei": enum,
  "hrdBufferInitialFillPercentage": integer,
  "gopSize": number,
  "slices": integer,
  "gopBReference": enum,
  "hrdBufferSize": integer,
  "maxBitrate": integer,
  "slowPal": enum,
  "parDenominator": integer,
  "spatialAdaptiveQuantization": enum,
  "temporalAdaptiveQuantization": enum,
  "flickerAdaptiveQuantization": enum,
  "bitrate": integer,
  "framerateControl": enum,
  "rateControlMode": enum,
  "qvbrSettings": {
    "qvbrQualityLevel": integer,
    "maxAverageBitrate": integer
  },
  "codecProfile": enum,
  "tiles": enum,
  "telecine": enum,
  "framerateNumerator": integer,
  "minIInterval": integer,
  "adaptiveQuantization": enum,
  "codecLevel": enum,
  "sceneChangeDetect": enum,
  "qualityTuningLevel": enum,
  "framerateConversionAlgorithm": enum,
  "unregisteredSeiTimecode": enum,
  "gopSizeUnits": enum,
  "parControl": enum,
  "numberBFramesBetweenReferenceFrames": integer,
  "temporalIds": enum,
  "sampleAdaptiveOffsetFilterMode": enum,
  "writeMp4PackagingType": enum,
  "dynamicSubGop": enum
},
"mpeg2Settings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "syntax": enum,
  "softness": integer,
  "framerateDenominator": integer,
  "gopClosedCadence": integer,
  "hrdBufferInitialFillPercentage": integer,
  "gopSize": number,
  "hrdBufferSize": integer,
  "maxBitrate": integer,
  "slowPal": enum,
  "parDenominator": integer,
  "spatialAdaptiveQuantization": enum,
  "temporalAdaptiveQuantization": enum,
  "bitrate": integer,
  "intraDcPrecision": enum,
  "framerateControl": enum,
  "rateControlMode": enum,
  "codecProfile": enum,
  "telecine": enum,
  "framerateNumerator": integer,
  "minIInterval": integer,
  "adaptiveQuantization": enum,
  "codecLevel": enum,
  "sceneChangeDetect": enum,
  "qualityTuningLevel": enum,
  "framerateConversionAlgorithm": enum,
  "gopSizeUnits": enum,
  "parControl": enum,
  "numberBFramesBetweenReferenceFrames": integer,
  "dynamicSubGop": enum
},

"proresSettings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "framerateDenominator": integer,
  "codecProfile": enum,
  "slowPal": enum,
  "parDenominator": integer,
  "framerateControl": enum,
  "telecine": enum,
  "framerateNumerator": integer,
  "framerateConversionAlgorithm": enum,
  "parControl": enum
}
},

"afdSignaling": enum,
"dropFrameTimecode": enum,
"respondToAfd": enum,
"colorMetadata": enum
},

"audioDescriptions": [
{
  "audioTypeControl": enum,
  "audioSourceName": "string",
  "audioNormalizationSettings": {
    "algorithm": enum,
    "algorithmControl": enum,
    "correctionGateLevel": integer,
    "loudnessLogging": enum,
    "targetLkfs": number,
    "peakCalculation": enum
  },
  "codecSettings": {
    "codec": enum,
  }
}
"aacSettings": {
  "audioDescriptionBroadcasterMix": enum,
  "vbrQuality": enum,
  "bitrate": integer,
  "rateControlMode": enum,
  "codecProfile": enum,
  "codingMode": enum,
  "rawFormat": enum,
  "sampleRate": integer,
  "specification": enum
},
"ac3Settings": {
  "bitrate": integer,
  "bitstreamMode": enum,
  "codingMode": enum,
  "dialnorm": integer,
  "dynamicRangeCompressionProfile": enum,
  "metadataControl": enum,
  "lfeFilter": enum,
  "sampleRate": integer
},
"aiffSettings": {
  "bitDepth": integer,
  "channels": integer,
  "sampleRate": integer
},
"eac3Settings": {
  "metadataControl": enum,
  "surroundExMode": enum,
  "loRoSurroundMixLevel": number,
  "phaseControl": enum,
  "dialnorm": integer,
  "ltRtSurroundMixLevel": number,
  "bitrate": integer,
  "ltRtCenterMixLevel": number,
  "passthroughControl": enum,
  "lfeControl": enum,
  "loRoCenterMixLevel": number,
  "attenuationControl": enum,
  "codingMode": enum,
  "surroundMode": enum,
  "bitstreamMode": enum,
  "lfeFilter": enum,
  "stereoDownmix": enum,
  "dynamicRangeCompressionRef": enum,
  "sampleRate": integer,
  "dynamicRangeCompressionLine": enum,
  "dcFilter": enum
},
"mp2Settings": {
  "bitrate": integer,
  "channels": integer,
  "sampleRate": integer
},
"wavSettings": {
  "bitDepth": integer,
  "channels": integer,
  "sampleRate": integer,
  "format": enum
},
"remixSettings": {
  "channelMapping": {
    "outputChannels": [ 
      { 
        "inputChannels": [ 
        ]
      }
    ]
  }
}
integer
]
}
]
"channelsIn": integer,
"channelsOut": integer,
"streamName": "string",
"languageCodeControl": enum,
"audioType": integer,
"customLanguageCode": "string",
"languageCode": enum
],
"containerSettings": {
"container": enum,
"m3u8Settings": {
"audioFramesPerPes": integer,
"pcrControl": enum,
"pcrPid": integer,
"pmtPid": integer,
"privateMetadataPid": integer,
"programNumber": integer,
"patInterval": integer,
"pmtInterval": integer,
"scte35Source": enum,
"scte35Pid": integer,
"nielsenId3": enum,
"timedMetadata": enum,
"timedMetadataPid": integer,
"transportStreamId": integer,
"videoPid": integer,
"audioPids": [
integers
]
},
"f4vSettings": {
"moovPlacement": enum
},
"m2tsSettings": {
"audioBufferModel": enum,
"minEbpInterval": integer,
"esRateInPes": enum,
"patInterval": integer,
"dvbNitSettings": {
"nitInterval": integer,
"networkId": integer,
"networkName": "string"
},
"dvbSdtSettings": {
"outputSdt": enum,
"sdtInterval": integer,
"serviceName": "string",
"serviceProviderName": "string"
},
"scte35Source": enum,
"scte35Pid": integer,
"scte35Esam": {
"scte35EsampId": integer
},
"videoPid": integer,
"dvbTdtSettings": {
"tdtInterval": integer
},
"pmtInterval": integer,
"segmentationStyle": enum,
"segmentationTime": number,
"pmtPid": integer,
"bitrate": integer,
"audioPids": [ integer
 ],
"privateMetadataPid": integer,
"nielsen2Id": enum,
"timeMetadataPid": integer,
"maxPcrInterval": integer,
"transportStreamId": integer,
"dvbSubPids": [ integer
 ],
"rateMode": enum,
"audioFramesPerPes": integer,
"segmentationMarkers": enum,
"ebpAudioInterval": enum,
"forceTsVideoEbpOrder": enum,
"programNumber": integer,
"pcrPid": integer,
"bufferModel": enum,
"dvbTeletextPid": integer,
"fragmentTime": number,
"ebpPlacement": enum,
"nullPacketBitrate": number
},
"movSettings": {
  "clapAtom": enum,
  "cslgAtom": enum,
  "paddingControl": enum,
  "reference": enum,
  "mpeg2FourCCControl": enum
},
"mp4Settings": {
  "cslgAtom": enum,
  "freeSpaceBox": enum,
  "mp4MajorBrand": "string",
  "moovPlacement": enum
},
"captionDescriptions": [
  {
    "destinationSettings": {
      "destinationType": enum,
      "burninDestinationSettings": {
        "backgroundColor": integer,
        "shadowXOffset": integer,
        "teletextSpacing": enum,
        "alignment": enum,
        "outlineSize": integer,
        "yPosition": integer,
        "shadowColor": enum,
        "fontOpacity": integer,
        "fontSize": integer,
        "fontScript": enum,
        "fontColor": enum,
        "backgroundColor": enum,
        "fontResolution": integer,
        "outlineColor": enum,
        "shadowYOffset": integer,
        "xPosition": integer,
        "shadowOpacity": integer
      }
    }
  }
]
"dvbSubDestinationSettings": {  "backgroundOpacity": integer,  "shadowXOffset": integer,  "teletextSpacing": enum,  "alignment": enum,  "outlineSize": integer,  "yPosition": integer,  "teletextSpacing": integer,  "shadowColor": enum,  "fontOpacity": integer,  "fontSize": integer,  "fontScript": enum,  "fontColor": enum,  "fontResolution": integer,  "backgroundColor": enum,  "outlineColor": enum,  "shadowYOffset": integer,  "xPosition": integer,  "shadowOpacity": integer},  
"sccDestinationSettings": {  "framerate": enum},  
"teletextDestinationSettings": {  "pageNumber": "string"},  
"ttmlDestinationSettings": {  "stylePassthrough": enum},  
"embeddedDestinationSettings": {  "destination608ChannelNumber": integer}
"customLanguageCode": "string",  "languageCode": enum,  "languageDescription": "string"
}

Example CreatePresetResponse

{
  "preset": {
    "arn": "string",
    "createdAt": "string",
    "lastUpdated": "string",
    "description": "string",
    "category": "string",
    "name": "string",
    "type": enum,
    "settings": {  "videoDescription": {
      "fixedAfd": integer,
      "width": integer,
      "scalingBehavior": enum,
      "crop": {
        "height": integer,
        "width": integer,
        "x": integer,
        "y": integer
      }
    }  
  }  
}
"videoPreprocessors": {
  "colorCorrector": {
    "brightness": integer,
    "colorSpaceConversion": enum,
    "contrast": integer,
    "hue": integer,
    "saturation": integer,
    "hdr10Metadata": {
      "redPrimaryX": integer,
      "redPrimaryY": integer,
      "greenPrimaryX": integer,
      "greenPrimaryY": integer,
      "bluePrimaryX": integer,
      "bluePrimaryY": integer,
      "whitePointX": integer,
      "whitePointY": integer,
      "maxFrameAverageLightLevel": integer,
      "maxContentLightLevel": integer,
      "maxLuminance": integer,
      "minLuminance": integer
    }
  },
  "deinterlacer": {
    "algorithm": enum,
    "mode": enum,
    "control": enum
  },
  "imageInserter": {
    "insertableImages": [
      {
        "width": integer,
        "height": integer,
        "imageX": integer,
        "imageY": integer,
        "duration": integer,
        "fadeIn": integer,
        "layer": integer,
        "imageInserterInput": "string",
        "startTime": "string",
        "fadeOut": integer,
        "opacity": integer
      }
    ]
  },
  "noiseReducer": {
    "filter": enum,
    "filterSettings": {
      "strength": integer
    },
    "spatialFilterSettings": {
      "strength": integer,
      "speed": integer,
      "postFilterSharpenStrength": integer
    }
  },
  "timecodeBurnin": {
    "fontSize": integer,
    "position": enum,
    "prefix": "string"
  }
},
"timecodeInsertion": enum,
"antiAlias": enum,
"position": {
"height": integer,
"width": integer,
"x": integer,
"y": integer
},
"sharpness": integer,
"codecSettings": {
  "codec": enum,
  "frameCaptureSettings": {
    "framerateNumerator": integer,
    "framerateDenominator": integer,
    "maxCaptures": integer,
    "quality": integer
  },
  "h264Settings": {
    "interlaceMode": enum,
    "parNumerator": integer,
    "numberReferenceFrames": integer,
    "syntax": enum,
    "softness": integer,
    "framerateNumerator": integer,
    "gopClosedCadence": integer,
    "hrdBufferInitialFillPercentage": integer,
    "gopSize": number,
    "slices": integer,
    "gopSReference": enum,
    "hrdBufferSize": integer,
    "maxBitrate": integer,
    "slowPal": enum,
    "parDenominator": integer,
    "spatialAdaptiveQuantization": enum,
    "temporalAdaptiveQuantization": enum,
    "flickerAdaptiveQuantization": enum,
    "entropyEncoding": enum,
    "bitrate": integer,
    "framerateControl": enum,
    "rateControlMode": enum,
    "qvbrSettings": {
      "qvbrQualityLevel": integer,
      "maxAverageBitrate": integer
    },
    "codecProfile": enum,
    "telecine": enum,
    "framerateNumerator": integer,
    "minInterval": integer,
    "adaptiveQuantization": enum,
    "codecLevel": enum,
    "fieldEncoding": enum,
    "sceneChangeDetect": enum,
    "qualityTuningLevel": enum,
    "framerateConversionAlgorithm": enum,
    "unregisteredSeiTimecode": enum,
    "gopSizeUnits": enum,
    "parControl": enum,
    "numberBFramesBetweenReferenceFrames": integer,
    "repeatPps": enum,
    "dynamicSubGop": enum
  },
  "h265Settings": {
    "interlaceMode": enum,
    "parNumerator": integer,
    "numberReferenceFrames": integer,
    "framerateNumerator": integer,
    "framerateDenominator": integer,
    "gopClosedCadence": integer,
    "alternateTransferFunctionSei": enum,
    "hrdBufferInitialFillPercentage": integer,
"gopSize": number,
"slices": integer,
"gopReference": enum,
"hrdBufferSize": number,
"maxBitrate": integer,
"slowPal": enum,
"parDenominator": integer,
"spatialAdaptiveQuantization": enum,
"temporalAdaptiveQuantization": enum,
"flickerAdaptiveQuantization": enum,
"bitrate": integer,
"framerateControl": enum,
"rateControlMode": enum,
"qvbrSettings": {
  "qvbrQualityLevel": integer,
  "maxAverageBitrate": integer
},
"codecProfile": enum,
"tiles": enum,
"telecine": enum,
"framerateNumerator": integer,
"minIInterval": integer,
"adaptiveQuantization": enum,
"codecLevel": enum,
"sceneChangeDetect": enum,
"qualityTuningLevel": enum,
"framerateConversionAlgorithm": enum,
"unregisteredSeiTimecode": enum,
"gopsSizeUnits": enum,
"parControl": enum,
"numberBFramesBetweenReferenceFrames": integer,
"temporalIds": enum,
"sampleAdaptiveOffsetFilterMode": enum,
"writeMp4PackagingType": enum,
"dynamicSubGop": enum
},
"mpeg2Settings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "syntax": enum,
  "softness": integer,
  "framerateDenominator": integer,
  "gopClosedCadence": integer,
  "hrdBufferInitialFillPercentage": integer,
  "gopSize": number,
  "hrdBufferSize": number,
  "maxBitrate": integer,
  "slowPal": enum,
  "parDenominator": integer,
  "spatialAdaptiveQuantization": enum,
  "temporalAdaptiveQuantization": enum,
  "intraDcPrecision": enum,
  "framerateControl": enum,
  "rateControlMode": enum,
  "codecProfile": enum,
  "telecine": enum,
  "framerateNumerator": integer,
  "minIInterval": integer,
  "adaptiveQuantization": enum,
  "codecLevel": enum,
  "sceneChangeDetect": enum,
  "qualityTuningLevel": enum,
  "framerateConversionAlgorithm": enum,
  "gopSizeUnits": enum,
  "parControl": enum
}
"numberOfBFramesBetweenReferenceFrames": integer,
"dynamicSubGop": enum
),
"proresSettings": {
"interlaceMode": enum,
"parNumerator": integer,
"framerateDenominator": integer,
"codecProfile": enum,
"slowPal": enum,
"parDenominator": integer,
"framerateControl": enum,
"telecine": enum,
"framerateNumerator": integer,
"framerateConversionAlgorithm": enum,
"parControl": enum
}
},
"afdSignaling": enum,
"dropFrameTimecode": enum,
"respondToAfd": enum,
"colorMetadata": enum
},
"audioDescriptions": [
{
"audioTypeControl": enum,
"audioSourceName": "string",
"audioNormalizationSettings": {
"algorithm": enum,
"algorithmControl": enum,
"correctionGateLevel": integer,
"loudnessLogging": enum,
"targetLkfs": number,
"peakCalculation": enum
}
},
"codecsSettings": {
"code": enum,
"aacSettings": {
"audioDescriptionBroadcasterMix": enum,
"vbrQuality": enum,
"bitrate": integer,
"rateControlMode": enum,
"codecProfile": enum,
"codingMode": enum,
"rawFormat": enum,
"sampleRate": integer,
"specification": enum
}
},
"ac3Settings": {
"bitrate": integer,
"bitstreamMode": enum,
"codingMode": enum,
"dialnorm": integer,
"dynamicRangeCompressionProfile": enum,
"metadataControl": enum,
"lfeFilter": enum,
"sampleRate": integer
}
},
"siffSettings": {
"bitDepth": integer,
"channels": integer,
"sampleRate": integer
}
},
"eac3Settings": {
"metadataControl": enum,
"surroundExMode": enum,
"loRoSurroundMixLevel": number,
"numberBFramesBetweenReferenceFrames": integer,
"dynamicSubGop": enum
),
"phaseControl": enum,
"dialnorm": integer,
"ltRtSurroundMixLevel": number,
"bitrate": integer,
"ltRtCenterMixLevel": number,
"passthroughControl": enum,
"lfeControl": enum,
"loRoCenterMixLevel": number,
"attenuationControl": enum,
"codingMode": enum,
"surroundMode": enum,
"bitstreamMode": enum,
"lfeFilter": enum,
"stereoDownmix": enum,
"dynamicRangeCompressionRf": enum,
"sampleRate": integer,
"dynamicRangeCompressionLine": enum,
"dcFilter": enum
},
"mp2Settings": {
"bitrate": integer,
"channels": integer,
"sampleRate": integer
},
"wavSettings": {
"bitDepth": integer,
"channels": integer,
"sampleRate": integer,
"format": enum
},
"remixSettings": {
"channelMapping": {
"outputChannels": [
{
"inputChannels": [integer
]
}
]
},
"channelsIn": integer,
"channelsOut": integer
},
"streamName": "string",
"languageCodeControl": enum,
"audioType": integer,
"customLanguageCode": "string",
"languageCode": enum
},
"containerSettings": {
"container": enum,
"m3u8Settings": {
"audioFramesPerPes": integer,
"pcrControl": enum,
"pcrPid": integer,
"pmtPid": integer,
"privateMetadataPid": integer,
"programNumber": integer,
"patInterval": integer,
"pmtInterval": integer,
"scte35Source": enum,
"scte35Pid": integer,
"nielsenId3": enum,
"timedMetadata": enum,
"timedMetadataPid": integer,
"transportStreamId": integer,
"videoPid": integer,
"audioPids": [
    integer
]
},
"f4vSettings": {
    "moovPlacement": enum
},
"m2tsSettings": {
    "audioBufferModel": enum,
    "minEbpInterval": integer,
    "esRateInPes": enum,
    "patInterval": integer,
    "dvbNitSettings": {
        "nitInterval": integer,
        "networkId": integer,
        "networkName": "string"
    },
    "dvbSdtSettings": {
        "outputSdt": enum,
        "sdtInterval": integer,
        "serviceName": "string",
        "serviceProviderName": "string"
    },
    "scte35Source": enum,
    "scte35Pid": integer,
    "scte35Esam": {
        "scte35EsamPid": integer
    },
    "videoPid": integer,
    "dvbTdtSettings": {
        "tdtInterval": integer
    },
    "pmtInterval": integer,
    "segmentationStyle": enum,
    "segmentationTime": number,
    "pmtPid": integer,
    "bitrate": integer,
    "audioPids": [
        integer
    ],
    "privateMetadataPid": integer,
    "nielsenId3": enum,
    "timedMetadataPid": integer,
    "maxPcrInterval": integer,
    "transportStreamId": integer,
    "dvbSubPids": [
        integer
    ],
    "rateMode": enum,
    "audioFramesPerPes": integer,
    "pcrControl": enum,
    "segmentationMarkers": enum,
    "ebpAudioInterval": enum,
    "forceTsVideoEbpOrder": enum,
    "programNumber": integer,
    "pcrPid": integer,
    "bufferModel": enum,
    "dvbTeletextPid": integer,
    "fragmentTime": number,
    "ebpPlacement": enum,
    "nullPacketBitrate": number
},
"movSettings": {

"clapAtom": enum,
"cslgAtom": enum,
"paddingControl": enum,
"reference": enum,
"mpeg2FourCCControl": enum
},
"mp4Settings": {
"cslgAtom": enum,
"freeSpaceBox": enum,
"mp4MajorBrand": "string",
"moovPlacement": enum
},
"captionDescriptions": [
{
"destinationSettings": {
"destinationType": enum,
"burninDestinationSettings": {
"backgroundColor": enum,
"shadowXOffset": integer,
"teletextSpacing": enum,
"alignment": enum,
"outlineSize": integer,
"yPosition": integer,
"shadowColor": enum,
"fontOpacity": integer,
"fontSize": integer,
"fontScript": enum,
"fontColor": enum,
"backgroundColor": enum,
"fontResolution": integer,
"outlineColor": enum,
"shadowYOffset": integer,
"xPosition": integer,
"shadowOpacity": integer
},
"dvbSubDestinationSettings": {
"backgroundColor": enum,
"shadowXOffset": integer,
"teletextSpacing": enum,
"alignment": enum,
"outlineSize": integer,
"yPosition": integer,
"shadowColor": enum,
"fontOpacity": integer,
"fontSize": integer,
"fontScript": enum,
"fontColor": enum,
"backgroundColor": enum,
"fontResolution": integer,
"outlineColor": enum,
"shadowYOffset": integer,
"xPosition": integer,
"shadowOpacity": integer
},
"sccDestinationSettings": {
"framerate": enum
},
"teletextDestinationSettings": {
"pageNumber": "string"
},
"ttmlDestinationSettings": {
"stylePassthrough": enum
},
"embeddedDestinationSettings": {
"destination608ChannelNumber": integer
}
Example ExceptionBody

```json
{
  "message": "string"
}
```

## Properties

### AacAudioDescriptionBroadcasterMix

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

AAC Profile.

- LC
- HEV1
- HEV2

### AacCodingMode

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**

Rate Control Mode.
- CBR
- VBR

**AacRawFormat**

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**

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value AAC. The service accepts one of two mutually exclusive groups of AAC settings--VBR and CBR. To select one of these modes, set the value of Bitrate control mode (rateControlMode) to "VBR" or "CBR". In VBR mode, you control the audio quality with the setting VBR quality (vbrQuality). In CBR mode, you use the setting Bitrate (bitrate). Defaults and valid values depend on the rate control mode.

**audioDescriptionBroadcasterMix**

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.

- **Type**: AacAudioDescriptionBroadcasterMix (p. 840)
- **Required**: False

**vbrQuality**

VBR Quality Level - Only used if rate_control_mode is VBR.

- **Type**: AacVbrQuality (p. 843)
- **Required**: False

**bitrate**

Average bitrate in bits/second. The set of valid values for this setting is: 6000, 8000, 10000, 12000, 14000, 16000, 20000, 24000, 28000, 32000, 40000, 48000, 56000, 64000, 80000, 96000, 112000, 128000, 160000, 192000, 224000, 256000, 288000, 320000, 384000, 448000, 512000, 576000, 640000, 768000, 896000, 1024000. The value you set is also constrained by the values you choose for Profile (codecProfile), Bitrate control mode (codingMode), and Sample rate (sampleRate). Default values depend on Bitrate control mode and Profile.

- **Type**: integer
- **Required**: False
Minimum: 6000
Maximum: 1024000

rateControlMode
Rate Control Mode.

Type: AacRateControlMode (p. 841)
Required: False

codecProfile
AAC Profile.

Type: AacCodecProfile (p. 840)
Required: False

codingMode
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.

Type: AacCodingMode (p. 840)
Required: False

rawFormat
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.

Type: AacRawFormat (p. 841)
Required: False

sampleRate
Sample rate in Hz. Valid values depend on rate control mode and profile.

Type: integer
Required: False
Minimum: 8000
Maximum: 96000

specification
Use MPEG-2 AAC instead of MPEG-4 AAC audio for raw or MPEG-2 Transport Stream containers.

Type: AacSpecification (p. 842)
Required: False

AacSpecification
Use MPEG-2 AAC instead of MPEG-4 AAC audio for raw or MPEG-2 Transport Stream containers.
**MPEG2**

**MPEG4**

**AacVbrQuality**

VBR Quality Level - Only used if rate_control_mode is VBR.

- LOW
- MEDIUM_LOW
- MEDIUM_HIGH
- HIGH

**Ac3BitstreamMode**

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**

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**

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**

Applies a 120Hz lowpass filter to the LFE channel prior to encoding. Only valid with 3_2_LFE coding mode.

- ENABLED
- DISABLED
Ac3MetadataControl

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

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value AC3.

bitrate

Average bitrate in bits/second. Valid bitrates depend on the coding mode.

   Type: integer
   Required: False
   Minimum: 64000
   Maximum: 640000

bitstreamMode

Specifies the "Bitstream Mode" (bsmod) for the emitted AC-3 stream. See ATSC A/52-2012 for background on these values.

   Type: Ac3BitstreamMode (p. 843)
   Required: False

codingMode

Dolby Digital coding mode. Determines number of channels.

   Type: Ac3CodingMode (p. 843)
   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

dynamicRangeCompressionProfile

If set to FILM_STANDARD, adds dynamic range compression signaling to the output bitstream as defined in the Dolby Digital specification.

   Type: Ac3DynamicRangeCompressionProfile (p. 843)
   Required: False
metadataControl

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.

  Type: Ac3MetadataControl (p. 844)
  Required: False

lfeFilter

Applies a 120Hz lowpass filter to the LFE channel prior to encoding. Only valid with 3_2_LFE coding mode.

  Type: Ac3LfeFilter (p. 843)
  Required: False

tSampleRate

Sample rate in hz. Sample rate is always 48000.

  Type: integer
  Required: False
  Minimum: 48000
  Maximum: 48000

AfdSignaling

This setting only applies to H.264, H.265, and MPEG2 outputs. Use Insert AFD signaling (AfdSignaling) to specify whether the service includes 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

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value AIFF.

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

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

sampleRate
Sample rate in hz.

- Type: integer
- Required: False
- Minimum: 8000
- Maximum: 192000

AntiAlias
The anti-alias filter is automatically applied to all outputs. The service no longer accepts the value DISABLED for AntiAlias. If you specify that in your job, the service will ignore the setting.

- DISABLED
- ENABLED

AudioCodec
Type of Audio codec.

- AAC
- MP2
- WAV
- AIFF
- AC3
- EAC3
- PASSTHROUGH

AudioCodecSettings
Audio codec settings (CodecSettings) under (AudioDescriptions) contains the group of settings related to audio encoding. The settings in this group vary depending on the value you choose for Audio codec (Codec). For each codec enum you choose, define the corresponding settings object. The following lists the codec enum, settings object pairs. * AAC, AacSettings * MP2, Mp2Settings * WAV, WavSettings * AIFF, AiffSettings * AC3, Ac3Settings * EAC3, Eac3Settings

- codec
  - Type: AudioCodec (p. 846)
  - Required: False

- aacSettings
  - Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value AAC. The service accepts one of two mutually exclusive groups of AAC settings--VBR and CBR. To select one of these
modes, set the value of Bitrate control mode (rateControlMode) to "VBR" or "CBR". In VBR mode, you control the audio quality with the setting VBR quality (vbrQuality). In CBR mode, you use the setting Bitrate (bitrate). Defaults and valid values depend on the rate control mode.

**Type:** AacSettings (p. 841)  
**Required:** False

### ac3Settings
Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value AC3.

**Type:** Ac3Settings (p. 844)  
**Required:** False

### aiffSettings
Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value AIFF.

**Type:** AiffSettings (p. 845)  
**Required:** False

### eac3Settings
Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value EAC3.

**Type:** Eac3Settings (p. 871)  
**Required:** False

### mp2Settings
Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value MP2.

**Type:** Mp2Settings (p. 924)  
**Required:** False

### wavSettings
Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value WAV.

**Type:** WavSettings (p. 951)  
**Required:** False

### AudioDescription
Description of audio output

### audioTypeControl
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.

**Type:** AudioTypeControl (p. 851)
Properties

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

audioNormalizationSettings

Advanced audio normalization settings.

Type: AudioNormalizationSettings (p. 850)  
Required: False

codecSettings

Audio codec settings (CodecSettings) under (AudioDescriptions) contains the group of settings related to audio encoding. The settings in this group vary depending on the value you choose for Audio codec (Codec). For each codec enum you choose, define the corresponding settings object. The following lists the codec enum, settings object pairs. * AAC, AacSettings * MP2, Mp2Settings * WAV, WavSettings * AIFF, AiffSettings * AC3, Ac3Settings * EAC3, Eac3Settings

Type: AudioCodecSettings (p. 846)  
Required: False

remixSettings

Advanced audio remixing settings.

Type: RemixSettings (p. 942)  
Required: False

streamName

Used for MS Smooth 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]*$

languageCodeControl

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.

**Type:** AudioLanguageCodeControl (p. 849)  
**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

### customLanguageCode

Specify the language for this audio output track, using the ISO 639-2 or ISO 639-3 three-letter language code. The language specified 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:** string  
**Required:** False  
**Pattern:** ^[A-Za-z]{3}$  
**MinLength:** 3  
**MaxLength:** 3

### 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. 905)  
**Required:** False

### AudioLanguageCodeControl

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

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**

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**

If set to LOG, log each output's audio track loudness to a CSV file.

- LOG
- DONT_LOG

**AudioNormalizationPeakCalculation**

If set to TRUE_PEAK, calculate and log the TruePeak for each output's audio track loudness.

- TRUE_PEAK
- NONE

**AudioNormalizationSettings**

Advanced audio normalization settings.

**algorithm**

Audio normalization algorithm to use. 1770-1 conforms to the CALM Act specification, 1770-2 conforms to the EBU R-128 specification.

- **Type:** AudioNormalizationAlgorithm (p. 849)
- **Required:** False

**algorithmControl**

When enabled the output audio is corrected using the chosen algorithm. If disabled, the audio will be measured but not adjusted.

- **Type:** AudioNormalizationAlgorithmControl (p. 850)
- **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
**Properties**

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Minimum</strong></td>
<td></td>
<td></td>
<td>-70                                                                aversal. 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.</td>
</tr>
<tr>
<td><strong>Maximum</strong></td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td><strong>loudnessLogging</strong></td>
<td>AudioNormalizationLoudnessLogging (p. 850)</td>
<td>False</td>
<td>If set to LOG, log each output's audio track loudness to a CSV file.</td>
</tr>
<tr>
<td><strong>targetLkfs</strong></td>
<td>number</td>
<td>False</td>
<td>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.</td>
</tr>
<tr>
<td><strong>peakCalculation</strong></td>
<td>AudioNormalizationPeakCalculation (p. 850)</td>
<td>False</td>
<td>If set to TRUEPEATK, calculate and log the TruePeak for each output's audio track loudness.</td>
</tr>
<tr>
<td><strong>AudioTypeControl</strong></td>
<td>FOLLOW_INPUT USE_CONFIGURED</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>BurninDestinationSettings</strong></td>
<td></td>
<td></td>
<td>Burn-In Destination Settings.</td>
</tr>
<tr>
<td><strong>backgroundOpacity</strong></td>
<td></td>
<td>False</td>
<td>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.</td>
</tr>
</tbody>
</table>
Maximum: 255

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

teletextSpacing

Only applies to jobs with input captions in Teletext or STL formats. Specify whether the spacing between letters in your captions is set by the captions grid or varies depending on letter width. Choose fixed grid to conform to the spacing specified in the captions file more accurately. Choose proportional to make the text easier to read if the captions are closed caption.

Type: BurninSubtitleTeletextSpacing (p. 856)
Required: False

alignment

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.

Type: BurninSubtitleAlignment (p. 855)
Required: False

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: False
Minimum: 0
Maximum: 10

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
shadowColor

Specifies the color of the shadow cast by the captions. All burn-in and DVB-Sub font settings must match.

Type: BurninSubtitleShadowColor (p. 855)
Required: False

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: False
Minimum: 0
Maximum: 255

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

fontScript

Provide the font script, using an ISO 15924 script code, if the LanguageCode is not sufficient for determining the script type. Where LanguageCode or CustomLanguageCode is sufficient, use "AUTOMATIC" or leave unset. This is used to help determine the appropriate font for rendering burn-in captions.

Type: FontScript (p. 876)
Required: False

fontColor

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.

Type: BurninSubtitleFontColor (p. 855)
Required: False

backgroundColor

Specifies the color of the rectangle behind the captions. All burn-in and DVB-Sub font settings must match.
**Properties**

**Type**: BurninSubtitleBackgroundColor (p. 855)  
**Required**: False

**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

**outlineColor**

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.

- **Type**: BurninSubtitleOutlineColor (p. 855)  
- **Required**: False

**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

**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

**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
**BurninSubtitleAlignment**

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**

Specifies the color of the rectangle behind the captions. All burn-in and DVB-Sub font settings must match.

- NONE
- BLACK
- WHITE

**BurninSubtitleFontColor**

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**

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**

Specifies the color of the shadow cast by the captions. All burn-in and DVB-Sub font settings must match.

- NONE
BurninSubtitleTeletextSpacing

Only applies to jobs with input captions in Teletext or STL formats. Specify whether the spacing between letters in your captions is set by the captions grid or varies depending on letter width. Choose fixed grid to conform to the spacing specified in the captions file more accurately. Choose proportional to make the text easier to read if the captions are closed caption.

- FIXED_GRID
- PROPORTIONAL

CaptionDescriptionPreset

Caption Description for preset

destinationSettings

Specific settings required by destination type. Note that burnin_destination_settings are not available if the source of the caption data is Embedded or Teletext.

- Type: CaptionDestinationSettings (p. 857)
- Required: False

customLanguageCode

Indicates the language of the caption output track, using the ISO 639-2 or ISO 639-3 three-letter language code. For most captions output formats, the encoder puts this language information in the output captions metadata. If your output captions format is DVB-Sub or Burn in, the encoder uses this language information to choose the font language for rendering the captions text.

- Type: string
- Required: False
- Pattern: ^[A-Za-z]{3}$
- MinLength: 3
- MaxLength: 3

languageCode

Specify the language of this captions output track. For most captions output formats, the encoder puts this language information in the output captions metadata. If your output captions format is DVB-Sub or Burn in, the encoder uses this language information to choose the font language for rendering the captions text.

- Type: LanguageCode (p. 905)
- Required: False

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
CaptionDestinationSettings

Specific settings required by destination type. Note that burnin_destination_settings are not available if the source of the caption data is Embedded or Teletext.

destinationType

Specify the format for this set of captions on this output. The default format is embedded without SCTE-20. Other options are embedded with SCTE-20, burn-in, DVB-sub, SCC, SRT, teletext, TTML, and web-VTT. If you are using SCTE-20, choose SCTE-20 plus embedded (SCTE20_PLUS_EMBEDDED) to create an output that complies with the SCTE-43 spec. To create a non-compliant output where the embedded captions come first, choose Embedded plus SCTE-20 (EMBEDDED_PLUS_SCTE20).

Type: CaptionDestinationType (p. 858)
Required: False

burninDestinationSettings

Burn-In Destination Settings.

Type: BurninDestinationSettings (p. 851)
Required: False

dvbSubDestinationSettings

DVB-Sub Destination Settings

Type: DvbSubDestinationSettings (p. 865)
Required: False

sccDestinationSettings

Settings for SCC caption output.

Type: SccDestinationSettings (p. 943)
Required: False

teletextDestinationSettings

Settings for Teletext caption output

Type: TeletextDestinationSettings (p. 944)
Required: False

ttmlDestinationSettings

Settings specific to TTML caption outputs, including Pass style information (TtmlStylePassthrough).

Type: TtmlDestinationSettings (p. 945)
Required: False

embeddedDestinationSettings

Settings specific to embedded/ancillary caption outputs, including 608/708 Channel destination number.
**Type:** EmbeddedDestinationSettings (p. 875)  
**Required:** False

### CaptionDestinationType

Specify the format for this set of captions on this output. The default format is embedded without SCTE-20. Other options are embedded with SCTE-20, burn-in, DVB-sub, SCC, SRT, teletext, TTML, and web-VTT. If you are using SCTE-20, choose SCTE-20 plus embedded (SCTE20_PLUS_EMBEDDED) to create an output that complies with the SCTE-43 spec. To create a non-compliant output where the embedded captions come first, choose Embedded plus SCTE-20 (EMBEDDED_PLUS_SCTE20).

- BURN_IN
- DVB_SUB
- EMBEDDED
- EMBEDDED_PLUS_SCTE20
- SCTE20_PLUS_EMBEDDED
- SCC
- SRT
- SMI
- TELETEXT
- TTML
- WEBVTT

### ChannelMapping

Channel mapping (ChannelMapping) contains the group of fields that hold the remixing value for each channel. Units are in dB. Acceptable values are within the range from -60 (mute) through 6. A setting of 0 passes the input channel unchanged to the output channel (no attenuation or amplification).

**outputChannels**

List of output channels

**Type:** Array of type OutputChannelMapping (p. 935)  
**Required:** False

### ColorCorrector

Settings for color correction.

**brightness**

Brightness level.

**Type:** integer  
**Required:** False  
**Minimum:** 1  
**Maximum:** 100

**colorSpaceConversion**

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 colorspace. An input's colorspace can be specified explicitly in the "Video Selector":#inputs-video_selector if necessary.

- **Type**: ColorSpaceConversion (p. 859)
- **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

**saturation**

Saturation level.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 100

**hdr10Metadata**

Use the HDR master display (Hdr10Metadata) settings to correct HDR metadata or to provide missing metadata. Note that these settings are not color correction.

- **Type**: Hdr10Metadata (p. 900)
- **Required**: False

**ColorMetadata**

Enable Insert color metadata (ColorMetadata) to include color metadata in this output. This setting is enabled by default.

- **IGNORE**
- **INSERT**

**ColorSpaceConversion**

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 specific settings.

**container**

Container for this output. Some containers require a container settings object. If not specified, the default object will be created.

*Type: ContainerType (p. 861)*
*Required: False*

**m3u8Settings**

Settings for TS segments in HLS

*Type: M3u8Settings (p. 919)*
*Required: False*

**f4vSettings**

Settings for F4v container

*Type: F4vSettings (p. 876)*
*Required: False*

**m2tsSettings**

MPEG-2 TS container settings. These apply to outputs in a File output group when the output's container (ContainerType) is MPEG-2 Transport Stream (M2TS). In these assets, data is organized by the program map table (PMT). Each transport stream program contains subsets of data, including audio, video, and metadata. Each of these subsets of data has a numerical label called a packet identifier (PID). Each transport stream program corresponds to one MediaConvert output. The PMT lists the types of data in a program along with their PID. Downstream systems and players use the program map table to look up the PID for each type of data it accesses and then uses the PIDs to locate specific data within the asset.

*Type: M2tsSettings (p. 913)*
*Required: False*

**movSettings**

Settings for MOV Container.

*Type: MovSettings (p. 923)*
*Required: False*
mp4Settings
Settings for MP4 Container

Type: Mp4Settings (p. 925)
Required: False

ContainerType
Container for this output. Some containers require a container settings object. If not specified, the
default object will be created.

F4V
ISMV
M2TS
M3U8
CMFC
MOV
MP4
MPD
MXF
RAW

CreatePresetRequest
Send your create preset request with the name of the preset and the JSON for the output settings
specified by the preset.

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

name
The name of the preset you are creating.

Type: string
Required: True

settings
Settings for preset

Type: PresetSettings (p. 937)
Required: True
tags

The tags that you want to add to the resource. You can tag resources with a key-value pair or with only a key.

Type: object
Required: False

CreatePresetResponse

Successful create preset requests will return the preset JSON.

preset

A preset is a collection of preconfigured media conversion settings that you want MediaConvert to apply to the output during the conversion process.

Type: Preset (p. 936)
Required: False

DeinterlaceAlgorithm

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

Settings for deinterlacer

algorithm

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.

Type: DeinterlaceAlgorithm (p. 862)
Required: False

mode

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.

Type: DeinterlacerMode (p. 863)
Required: False
control

- 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.

  **Type**: DeinterlacerControl (p. 863)
  **Required**: False

DeinterlacerControl

- 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

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

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

Inserts DVB Network Information Table (NIT) at the specified table repetition interval.

  **nitInterval**

The number of milliseconds between instances of this table in the output transport stream.

  **Type**: integer
  **Required**: False
Properties

**networkId**

The numeric value placed in the Network Information Table (NIT).

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 65535

**networkName**

The network name text placed in the network_name_descriptor inside the Network Information Table. Maximum length is 256 characters.

- **Type**: string
- **Required**: False
- **MinLength**: 1
- **MaxLength**: 256

**DvbSdtSettings**

Inserts DVB Service Description Table (NIT) at the specified table repetition interval.

**outputSdt**

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.

- **Type**: OutputSdt (p. 936)
- **Required**: False

**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
- **MinLength**: 1
MaxLength: 256

**serviceProviderName**

The service provider name placed in the service_descriptor in the Service Description Table. Maximum length is 256 characters.

- **Type**: string
- **Required**: False
- **MinLength**: 1
- **MaxLength**: 256

### DvbSubDestinationSettings

DVB-Sub Destination Settings

**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

**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

**teletextSpacing**

Only applies to jobs with input captions in Teletext or STL formats. Specify whether the spacing between letters in your captions is set by the captions grid or varies depending on letter width. Choose fixed grid to conform to the spacing specified in the captions file more accurately. Choose proportional to make the text easier to read if the captions are closed caption.

- **Type**: DvbSubtitleTeletextSpacing (p. 869)
- **Required**: False

**alignment**

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.
**Properties**

**Type**: DvbSubtitleAlignment (p. 868)
**Required**: False

**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**: False
**Minimum**: 0
**Maximum**: 10

**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

**shadowColor**

Specifies the color of the shadow cast by the captions. All burn-in and DVB-Sub font settings must match.

**Type**: DvbSubtitleShadowColor (p. 869)
**Required**: False

**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**: False
**Minimum**: 0
**Maximum**: 255

**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
**fontScript**

Provide the font script, using an ISO 15924 script code, if the LanguageCode is not sufficient for determining the script type. Where LanguageCode or CustomLanguageCode is sufficient, use "AUTOMATIC" or leave unset. This is used to help determine the appropriate font for rendering DVB-Sub captions.

*Type: FontScript (p. 876)*  
*Required: False*

**fontColor**

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.

*Type: DvbSubtitleFontColor (p. 868)*  
*Required: False*

**backgroundColor**

Specifies the color of the rectangle behind the captions. All burn-in and DVB-Sub font settings must match.

*Type: DvbSubtitleBackgroundColor (p. 868)*  
*Required: False*

**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*

**outlineColor**

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.

*Type: DvbSubtitleOutlineColor (p. 869)*  
*Required: False*

**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*
**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

**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

**DvbSubtitleAlignment**

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**

Specifies the color of the rectangle behind the captions. All burn-in and DVB-Sub font settings must match.

- NONE
- BLACK
- WHITE

**DvbSubtitleFontColor**

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
**DvbSubtitleOutlineColor**

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**

Specifies the color of the shadow cast by the captions. All burn-in and DVB-Sub font settings must match.

- NONE
- BLACK
- WHITE

**DvbSubtitleTeletextSpacing**

Only applies to jobs with input captions in Teletext or STL formats. Specify whether the spacing between letters in your captions is set by the captions grid or varies depending on letter width. Choose fixed grid to conform to the spacing specified in the captions file more accurately. Choose proportional to make the text easier to read if the captions are closed caption.

- FIXED_GRID
- PROPORTIONAL

**DvbTdtSettings**

Inserts DVB Time and Date Table (TDT) at the specified table repetition interval.

**tdtInterval**

The number of milliseconds between instances of this table in the output transport stream.

- **Type**: integer
- **Required**: False
- **Minimum**: 1000
- **Maximum**: 30000

**Eac3AttenuationControl**

If set to ATTENUATE_3_DB, applies a 3 dB attenuation to the surround channels. Only used for 3/2 coding mode.
### Properties

#### Eac3BitstreamMode

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

Dolby Digital Plus coding mode. Determines number of channels.

- CODING_MODE_1_0
- CODING_MODE_2_0
- CODING_MODE_3_2

#### Eac3DcFilter

Activates a DC highpass filter for all input channels.

- ENABLED
- DISABLED

#### Eac3DynamicRangeCompressionLine

Enables Dynamic Range Compression that restricts the absolute peak level for a signal.

- NONE
- FILM_STANDARD
- FILM_LIGHT
- MUSIC_STANDARD
- MUSIC_LIGHT
- SPEECH

#### Eac3DynamicRangeCompressionRf

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

When encoding 3/2 audio, controls whether the LFE channel is enabled

- LFE
- NO_LFE

Eac3LfeFilter

Applies a 120Hz lowpass filter to the LFE channel prior to encoding. Only valid with 3_2_LFE coding mode.

- ENABLED
- DISABLED

Eac3MetadataControl

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

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

Controls the amount of phase-shift applied to the surround channels. Only used for 3/2 coding mode.

- SHIFT_90_DEGREES
- NO_SHIFT

Eac3Settings

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value EAC3.

metadataControl

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.

Type: Eac3MetadataControl (p. 871)
**Properties**

**Required:** False

**surroundExMode**

When encoding 3/2 audio, sets whether an extra center back surround channel is matrix encoded into the left and right surround channels.

**Type:** Eac3SurroundExMode (p. 875)

**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

**phaseControl**

Controls the amount of phase-shift applied to the surround channels. Only used for 3/2 coding mode.

**Type:** Eac3PhaseControl (p. 871)

**Required:** False

**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

**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

**bitrate**

Average bitrate in bits/second. Valid bitrates depend on the coding mode.

**Type:** integer
**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

**passthroughControl**

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.

- **Type**: Eac3PassthroughControl (p. 871)
- **Required**: False

**lfeControl**

When encoding 3/2 audio, controls whether the LFE channel is enabled

- **Type**: Eac3LfeControl (p. 871)
- **Required**: False

**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

**attenuationControl**

If set to ATTENUATE_3_DB, applies a 3 dB attenuation to the surround channels. Only used for 3/2 coding mode.

- **Type**: Eac3AttenuationControl (p. 869)
- **Required**: False

**codingMode**

Dolby Digital Plus coding mode. Determines number of channels.

- **Type**: Eac3CodingMode (p. 870)
**Properties**

**surroundMode**

When encoding 2/0 audio, sets whether Dolby Surround is matrix encoded into the two channels.

- **Type**: Eac3SurroundMode (p. 875)
- **Required**: False

**bitstreamMode**

Specifies the "Bitstream Mode" (bsmod) for the emitted E-AC-3 stream. See ATSC A/52-2012 (Annex E) for background on these values.

- **Type**: Eac3BitstreamMode (p. 870)
- **Required**: False

**lfeFilter**

Applies a 120Hz lowpass filter to the LFE channel prior to encoding. Only valid with 3_2_LFE coding mode.

- **Type**: Eac3LfeFilter (p. 871)
- **Required**: False

**stereoDownmix**

Stereo downmix preference. Only used for 3/2 coding mode.

- **Type**: Eac3StereoDownmix (p. 875)
- **Required**: False

**dynamicRangeCompressionRf**

Enables Heavy Dynamic Range Compression, ensures that the instantaneous signal peaks do not exceed specified levels.

- **Type**: Eac3DynamicRangeCompressionRf (p. 870)
- **Required**: False

**sampleRate**

Sample rate in Hz. Sample rate is always 48000.

- **Type**: integer
- **Required**: False
- **Minimum**: 48000
- **Maximum**: 48000

**dynamicRangeCompressionLine**

Enables Dynamic Range Compression that restricts the absolute peak level for a signal.

- **Type**: Eac3DynamicRangeCompressionLine (p. 870)
**Required:** False

**dcFilter**
Activates a DC highpass filter for all input channels.

**Type:** `Eac3DcFilter` (p. 870)
**Required:** False

**Eac3StereoDownmix**
Stereo downmix preference. Only used for 3/2 coding mode.

- NOT_INDICATED
- LO_RO
- LT_RT
- DPL2

**Eac3SurroundExMode**
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**
When encoding 2/0 audio, sets whether Dolby Surround is matrix encoded into the two channels.

- NOT_INDICATED
- ENABLED
- DISABLED

**EmbeddedDestinationSettings**
Settings specific to embedded/ancillary caption outputs, including 608/708 Channel destination number.

**destination608ChannelNumber**
Ignore this setting unless your input captions are SCC format and your output container is MXF. With this combination of input captions format and output container, you can optionally use this setting to replace the input channel number with the track number that you specify. Specify a different number for each output captions track. If you don't specify an output track number, the system uses the input channel number for the output channel number. This setting applies to each output individually. You can optionally combine two captions channels in your output. The two output channel numbers can be one of the following pairs: 1,3; 2,4; 1,4; or 2,3.

**Type:** integer
**Required:** False
**Minimum:** 1

875
Maximum: 4

**ExceptionBody**

*message*

- **Type**: string
- **Required**: False

**F4vMoovPlacement**

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**

Settings for F4v container

*moovPlacement*

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.

- **Type**: `F4vMoovPlacement` (p. 876)
- **Required**: False

**FontScript**

Provide the font script, using an ISO 15924 script code, if the LanguageCode is not sufficient for determining the script type. Where LanguageCode or CustomLanguageCode is sufficient, use "AUTOMATIC" or leave unset.

- `AUTOMATIC`
- `HANS`
- `HANT`

**FrameCaptureSettings**

Required when you set `(Codec)` under `(VideoDescription)>(CodecSettings)` to the value FRAME_CAPTURE.

*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
Properties

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

quality

JPEG Quality - a higher value equals higher quality.

Type: integer
Required: False
Minimum: 1
Maximum: 100

H264AdaptiveQuantization

Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

OFF
LOW
MEDIUM
HIGH
HIGHER
MAX

H264CodecLevel

Specify an H.264 level that is consistent with your output video settings. If you aren't sure what level to
specify, choose Auto (AUTO).

AUTO
LEVEL_1
H264CodecProfile

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

H264DynamicSubGop

Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting B frames between reference frames (numberBFramesBetweenReferenceFrames).

- ADAPTIVE
- STATIC

H264EntropyEncoding

Entropy encoding mode. Use CABAC (must be in Main or High profile) or CAVLC.

- CABAC
- CAVLC

H264FieldEncoding

Choosing FORCE_FIELD disables PAFF encoding for interlaced outputs.

- PAFF
- FORCE_FIELD
H264FlickerAdaptiveQuantization

Adjust quantization within each frame to reduce flicker or 'pop' on I-frames.

DISABLED
ENABLED

H264FramerateControl

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

INITIALIZE_FROM_SOURCE
SPECIFIED

H264FramerateConversionAlgorithm

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

DUPLICATE_DROP
INTERPOLATE

H264GopBReference

If enable, use reference B frames for GOP structures that have B frames > 1.

DISABLED
ENABLED

H264GopSizeUnits

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

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, as follows. - 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_FIELDS
  BOTTOM_FIELD
  FOLLOW_TOP_FIELD
  FOLLOW_BOTTOM_FIELD

H264ParControl

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

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

H264QvbrSettings

Settings for quality-defined variable bitrate encoding with the H.264 codec. Required when you set Rate control mode to QVBR. Not valid when you set Rate control mode to a value other than QVBR, or when you don't define Rate control mode.

qvbrQualityLevel

Required when you use QVBR rate control mode. That is, when you specify qvbrSettings within h264Settings. Specify the target quality level for this output, from 1 to 10. Use higher numbers for greater quality. Level 10 results in nearly lossless compression. The quality level for most broadcast-quality transcodes is between 6 and 9.

  Type: integer
  Required: False
  Minimum: 1
  Maximum: 10

maxAverageBitrate

Use this setting only when Rate control mode is QVBR and Quality tuning level is Multi-pass HQ. For Max average bitrate values suited to the complexity of your input video, the service limits the average bitrate of the video part of this output to the value you choose. That is, the total size of the video element is less than or equal to the value you set multiplied by the number of seconds of encoded output.

  Type: integer
  Required: False
Minimum: 1000
Maximum: 1152000000

**H264RateControlMode**

Use this setting to specify whether this output has a variable bitrate (VBR), constant bitrate (CBR) or quality-defined variable bitrate (QVBR).

- VBR
- CBR
- QVBR

**H264RepeatPps**

Places a PPS header on each encoded picture, even if repeated.

- DISABLED
- ENABLED

**H264SceneChangeDetect**

Scene change detection (inserts I-frames on scene changes).

- DISABLED
- ENABLED

**H264Settings**

Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value H_264.

**interlaceMode**

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, as follows. - 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.

- **Type:** H264InterlaceMode (p. 879)
- **Required:** False

**parNumerator**

Pixel Aspect Ratio numerator.

- **Type:** integer
- **Required:** False
- **Minimum:** 1
- **Maximum:** 2147483647
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

syntax

Produces a bitstream compliant with SMPTE RP-2027.

- **Type**: H264Syntax (p. 888)
- **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

framerateDenominator

When you use the API for transcode jobs that use frame rate conversion, specify the frame rate 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 frame rate conversion, provide the value as a decimal number for Framerate. In this example, specify 23.976.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 2147483647

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

hrdBufferInitialFillPercentage

Percentage of the buffer that should initially be filled (HRD buffer model).

- **Type**: integer
gopSize

GOP Length (keyframe interval) in frames or seconds. Must be greater than zero.

  Type: number
  Required: False
  Format: float
  Minimum: 0.0

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

gopBReference

If enable, use reference B frames for GOP structures that have B frames > 1.

  Type: H264GopBReference (p. 879)
  Required: False

hrdBufferSize

Size of buffer (HRD buffer model) in bits. For example, enter five megabits as 5000000.

  Type: integer
  Required: False
  Minimum: 0
  Maximum: 1152000000

maxBitrate

Maximum bitrate in bits/second. For example, enter five megabits per second as 5000000. Required when Rate control mode is QVBR.

  Type: integer
  Required: False
  Minimum: 1000
  Maximum: 1152000000

slowPal

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.
Type: H264SlowPal (p. 887)
Required: False

parDenominator
Pixel Aspect Ratio denominator.

  Type: integer
  Required: False
  Minimum: 1
  Maximum: 2147483647

spatialAdaptiveQuantization
Adjust quantization within each frame based on spatial variation of content complexity.

  Type: H264SpatialAdaptiveQuantization (p. 888)
  Required: False

temporalAdaptiveQuantization
Adjust quantization within each frame based on temporal variation of content complexity.

  Type: H264TemporalAdaptiveQuantization (p. 888)
  Required: False

flickerAdaptiveQuantization
Adjust quantization within each frame to reduce flicker or 'pop' on I-frames.

  Type: H264FlickerAdaptiveQuantization (p. 879)
  Required: False

entropyEncoding
Entropy encoding mode. Use CABAC (must be in Main or High profile) or CAVLC.

  Type: H264EntropyEncoding (p. 878)
  Required: False

bitrate
Average bitrate in bits/second. Required for VBR and CBR. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.

  Type: integer
  Required: False
  Minimum: 1000
  Maximum: 1152000000

framerateControl
If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame...
rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

**Type:** H264FramerateControl (p. 879)
**Required:** False

**rateControlMode**

Use this setting to specify whether this output has a variable bitrate (VBR), constant bitrate (CBR) or quality-defined variable bitrate (QVBR).

**Type:** H264RateControlMode (p. 881)
**Required:** False

**qvbrSettings**

Settings for quality-defined variable bitrate encoding with the H.264 codec. Required when you set Rate control mode to QVBR. Not valid when you set Rate control mode to a value other than QVBR, or when you don't define Rate control mode.

**Type:** H264QvbrSettings (p. 880)
**Required:** False

**codecProfile**

H.264 Profile. High 4:2:2 and 10-bit profiles are only available with the AVC-I License.

**Type:** H264CodecProfile (p. 878)
**Required:** False

**telecine**

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.

**Type:** H264Telecine (p. 888)
**Required:** False

**framerateNumerator**

Frame rate numerator - frame rate is a fraction, e.g. 24000 / 1001 = 23.976 fps.

**Type:** integer
**Required:** False
**Minimum:** 1
**Maximum:** 2147483647
**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

**adaptiveQuantization**

Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

- **Type**: H264AdaptiveQuantization (p. 877)
- **Required**: False

**codecLevel**

Specify an H.264 level that is consistent with your output video settings. If you aren't sure what level to specify, choose Auto (AUTO).

- **Type**: H264CodecLevel (p. 877)
- **Required**: False

**fieldEncoding**

Choosing FORCE_FIELD disables PAFF encoding for interlaced outputs.

- **Type**: H264FieldEncoding (p. 878)
- **Required**: False

**sceneChangeDetect**

Scene change detection (inserts I-frames on scene changes).

- **Type**: H264SceneChangeDetect (p. 881)
- **Required**: False

**qualityTuningLevel**

Use Quality tuning level (H264QualityTuningLevel) to specify whether to use fast single-pass, high-quality singlepass, or high-quality multipass video encoding.

- **Type**: H264QualityTuningLevel (p. 880)
- **Required**: False

**framerateConversionAlgorithm**

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

- **Type**: H264FramerateConversionAlgorithm (p. 879)
- **Required**: False
unregisteredSeiTimecode

Inserts timecode for each frame as 4 bytes of an unregistered SEI message.

- **Type:** H264UnregisteredSeiTimecode (p. 888)
- **Required:** False

**gopSizeUnits**

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.

- **Type:** H264GopSizeUnits (p. 879)
- **Required:** False

**parControl**

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.

- **Type:** H264ParControl (p. 880)
- **Required:** False

**numberBFramesBetweenReferenceFrames**

Number of B-frames between reference frames.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 7

**repeatPps**

Places a PPS header on each encoded picture, even if repeated.

- **Type:** H264RepeatPps (p. 881)
- **Required:** False

**dynamicSubGop**

Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting B frames between reference frames (numberBFramesBetweenReferenceFrames).

- **Type:** H264DynamicSubGop (p. 878)
- **Required:** False

**H264SlowPal**

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.
**H264SpatialAdaptiveQuantization**
Adjust quantization within each frame based on spatial variation of content complexity.

- DISABLED
- ENABLED

**H264Syntax**
Produces a bitstream compliant with SMPTE RP-2027.

- DEFAULT
- RP2027

**H264Telecine**
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**
Adjust quantization within each frame based on temporal variation of content complexity.

- DISABLED
- ENABLED

**H264UnregisteredSeiTimecode**
Inserts timecode for each frame as 4 bytes of an unregistered SEI message.

- DISABLED
- ENABLED

**H265AdaptiveQuantization**
Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

- OFF
- LOW
- MEDIUM
- HIGH
**H265AlternateTransferFunctionSei**

Enables Alternate Transfer Function SEI message for outputs using Hybrid Log Gamma (HLG) Electro-Optical Transfer Function (EOTF).

- DISABLED
- ENABLED

**H265CodecLevel**

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**

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

**H265DynamicSubGop**

Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting B frames between reference frames (numberBFramesBetweenReferenceFrames).
**ADAPTIVE**
**STATIC**

**H265FlickerAdaptiveQuantization**
Adjust quantization within each frame to reduce flicker or 'pop' on I-frames.

- **DISABLED**
- **ENABLED**

**H265FramerateControl**
If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

- **INITIALIZE_FROM_SOURCE**
- **SPECIFIED**

**H265FramerateConversionAlgorithm**
When set to INTERPOLATE, produces smoother motion during frame rate conversion.

- **DUPLICATE_DROP**
- **INTERPOLATE**

**H265GopBReference**
If enable, use reference B frames for GOP structures that have B frames > 1.

- **DISABLED**
- **ENABLED**

**H265GopSizeUnits**
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**
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

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

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

H265QvbrSettings

Settings for quality-defined variable bitrate encoding with the H.265 codec. Required when you set Rate control mode to QVBR. Not valid when you set Rate control mode to a value other than QVBR, or when you don’t define Rate control mode.

qvbrQualityLevel

Required when you use QVBR rate control mode. That is, when you specify qvbrSettings within h265Settings. Specify the target quality level for this output, from 1 to 10. Use higher numbers for greater quality. Level 10 results in nearly lossless compression. The quality level for most broadcast-quality transcodes is between 6 and 9.

Type: integer
Required: False
Minimum: 1
Maximum: 10

maxAverageBitrate

Use this setting only when Rate control mode is QVBR and Quality tuning level is Multi-pass HQ. For Max average bitrate values suited to the complexity of your input video, the service limits the average bitrate of the video part of this output to the value you choose. That is, the total size of the video element is less than or equal to the value you set multiplied by the number of seconds of encoded output.
**Type**: integer  
**Required**: False  
**Minimum**: 1000  
**Maximum**: 1466400000

**H265RateControlMode**

Use this setting to specify whether this output has a variable bitrate (VBR), constant bitrate (CBR) or quality-defined variable bitrate (QVBR).

- VBR
- CBR
- QVBR

**H265SampleAdaptiveOffsetFilterMode**

Specify Sample Adaptive Offset (SAO) filter strength. Adaptive mode dynamically selects best strength based on content.

- DEFAULT
- ADAPTIVE
- OFF

**H265SceneChangeDetect**

Scene change detection (inserts I-frames on scene changes).

- DISABLED
- ENABLED

**H265Settings**

Settings for H265 codec

**interlaceMode**

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.

- **Type**: H265InterlaceMode (p. 890)  
- **Required**: False

**parNumerator**

Pixel Aspect Ratio numerator.

- **Type**: integer
Properties

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**: 2147483647

framerateDenominator
Frame rate denominator.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 2147483647

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

alternateTransferFunctionSei
Enables Alternate Transfer Function SEI message for outputs using Hybrid Log Gamma (HLG) Electro-Optical Transfer Function (EOTF).

- **Type**: H265AlternateTransferFunctionSei (p. 889)
- **Required**: False

hrdBufferInitialFillPercentage
Percentage of the buffer that should initially be filled (HRD buffer model).

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 100

gopSize
GOP Length (keyframe interval) in frames or seconds. Must be greater than zero.
Type: number
Required: False
Format: float
Minimum: 0.0

**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

**gopBReference**

If enable, use reference B frames for GOP structures that have B frames > 1.

Type: H265GopBReference (p. 890)
Required: False

**hrdBufferSize**

Size of buffer (HRD buffer model) in bits. For example, enter five megabits as 5000000.

Type: integer
Required: False
Minimum: 0
Maximum: 1466400000

**maxBitrate**

Maximum bitrate in bits/second. For example, enter five megabits per second as 5000000. Required when Rate control mode is QVBR.

Type: integer
Required: False
Minimum: 1000
Maximum: 1466400000

**slowPal**

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

Type: H265SlowPal (p. 899)
Required: False

**parDenominator**

Pixel Aspect Ratio denominator.

Type: integer
**spatialAdaptiveQuantization**

Adjust quantization within each frame based on spatial variation of content complexity.

- **Type**: H265SpatialAdaptiveQuantization (p. 899)
- **Required**: False

**temporalAdaptiveQuantization**

Adjust quantization within each frame based on temporal variation of content complexity.

- **Type**: H265TemporalAdaptiveQuantization (p. 899)
- **Required**: False

**flickerAdaptiveQuantization**

Adjust quantization within each frame to reduce flicker or 'pop' on I-frames.

- **Type**: H265FlickerAdaptiveQuantization (p. 890)
- **Required**: False

**bitrate**

Average bitrate in bits/second. Required for VBR and CBR. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.

- **Type**: integer
- **Required**: False
- **Minimum**: 1000
- **Maximum**: 1466400000

**framerateControl**

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job sepcification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

- **Type**: H265FramerateControl (p. 890)
- **Required**: False

**rateControlMode**

Use this setting to specify whether this output has a variable bitrate (VBR), constant bitrate (CBR) or quality-defined variable bitrate (QVBR).
Properties

Type: H265RateControlMode (p. 892)
Required: False

qvbrSettings

Settings for quality-defined variable bitrate encoding with the H.265 codec. Required when you set Rate control mode to QVBR. Not valid when you set Rate control mode to a value other than QVBR, or when you don’t define Rate control mode.

Type: H265QvbrSettings (p. 891)
Required: False

codecProfile

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.

Type: H265CodecProfile (p. 889)
Required: False

tiles

Enable use of tiles, allowing horizontal as well as vertical subdivision of the encoded pictures.

Type: H265Tiles (p. 900)
Required: False

telecine

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.

Type: H265Telecine (p. 899)
Required: False

framerateNumerator

Frame rate numerator - frame rate is a fraction, e.g. 24000 / 1001 = 23.976 fps.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

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

### adaptiveQuantization
Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

- **Type**: H265AdaptiveQuantization (p. 888)
- **Required**: False

### codecLevel
H.265 Level.

- **Type**: H265CodecLevel (p. 889)
- **Required**: False

### sceneChangeDetect
Scene change detection (inserts I-frames on scene changes).

- **Type**: H265SceneChangeDetect (p. 892)
- **Required**: False

### qualityTuningLevel
Use Quality tuning level (H265QualityTuningLevel) to specify whether to use fast single-pass, high-quality singlepass, or high-quality multipass video encoding.

- **Type**: H265QualityTuningLevel (p. 891)
- **Required**: False

### framerateConversionAlgorithm
When set to INTERPOLATE, produces smoother motion during frame rate conversion.

- **Type**: H265FramerateConversionAlgorithm (p. 890)
- **Required**: False

### unregisteredSeiTimecode
Inserts timecode for each frame as 4 bytes of an unregistered SEI message.

- **Type**: H265UnregisteredSeiTimecode (p. 900)
- **Required**: False

### gopSizeUnits
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.
Properties

**Type**: H265GopSizeUnits (p. 890)
**Required**: False

**parControl**

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.

**Type**: H265ParControl (p. 891)
**Required**: False

**numberBFramesBetweenReferenceFrames**

Number of B-frames between reference frames.

**Type**: integer
**Required**: False
**Minimum**: 0
**Maximum**: 7

**temporalIds**

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.

**Type**: H265TemporalIds (p. 899)
**Required**: False

**sampleAdaptiveOffsetFilterMode**

Specify Sample Adaptive Offset (SAO) filter strength. Adaptive mode dynamically selects best strength based on content.

**Type**: H265SampleAdaptiveOffsetFilterMode (p. 892)
**Required**: False

**writeMp4PackagingType**

Use this setting only for outputs encoded with H.265 that are in CMAF or DASH output groups. If you include writeMp4PackagingType in your JSON job specification for other outputs, your video might not work properly with downstream systems and video players. If the location of parameter set NAL units don't matter in your workflow, ignore this setting. The service defaults to marking your output as HEV1. Choose HVC1 to mark your output as HVC1. This makes your output compliant with this specification: ISO IEC/JTC1 SC29 N13798 Text ISO/IEC FDIS 14496-15 3rd Edition. For these outputs, the service stores parameter set NAL units in the sample headers but not in the samples directly. Keep the default HEV1 to mark your output as HEV1. For these outputs, the service writes parameter set NAL units directly into the samples.

**Type**: H265WriteMp4PackagingType (p. 900)
**Required**: False
dynamicSubGop

Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting B frames between reference frames (numberBFramesBetweenReferenceFrames).

Type: H265DynamicSubGop (p. 889)
Required: False

H265SlowPal

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

DISABLED
ENABLED

H265SpatialAdaptiveQuantization

Adjust quantization within each frame based on spatial variation of content complexity.

DISABLED
ENABLED

H265Telecine

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

Adjust quantization within each frame based on temporal variation of content complexity.

DISABLED
ENABLED

H265TemporalIds

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

Enable use of tiles, allowing horizontal as well as vertical subdivision of the encoded pictures.

   DISABLED
   ENABLED

H265UnregisteredSeiTimecode

Inserts timecode for each frame as 4 bytes of an unregistered SEI message.

   DISABLED
   ENABLED

H265WriteMp4PackagingType

Use this setting only for outputs encoded with H.265 that are in CMAF or DASH output groups. If you include writeMp4PackagingType in your JSON job specification for other outputs, your video might not work properly with downstream systems and video players. If the location of parameter set NAL units don't matter in your workflow, ignore this setting. The service defaults to marking your output as HEV1. Choose HVC1 to mark your output as HVC1. This makes your output compliant with this specification: ISO IECJTC1 SC29 N13798 Text ISO/IEC FDIS 14496-15 3rd Edition. For these outputs, the service stores parameter set NAL units in the sample headers but not in the samples directly. Keep the default HEV1 to mark your output as HEV1. For these outputs, the service writes parameter set NAL units directly into the samples.

   HVC1
   HEV1

Hdr10Metadata

Use the "HDR master display information" (Hdr10Metadata) settings to correct HDR metadata or to provide missing metadata. These values vary depending on the input video and must be provided by a color grader. Range is 0 to 50,000; each increment represents 0.00002 in CIE1931 color coordinate. Note that these settings are not color correction. Note that if you are creating HDR outputs inside of an HLS CMAF package, to comply with the Apple specification, you must use the following settings. Set "MP4 packaging type" (writeMp4PackagingType) to HVC1 (HVC1). Set "Profile" (H265Settings > codecProfile) to Main10/High (MAIN10_HIGH). Set "Level" (H265Settings > codecLevel) to 5 (LEVEL_5).

   redPrimaryX

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

   Type: integer
   Required: False
Minimum: 0
Maximum: 50000

**redPrimaryY**

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

  Type: integer
  Required: False
  Minimum: 0
  Maximum: 50000

**greenPrimaryX**

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

  Type: integer
  Required: False
  Minimum: 0
  Maximum: 50000

**greenPrimaryY**

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

  Type: integer
  Required: False
  Minimum: 0
  Maximum: 50000

**bluePrimaryX**

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

  Type: integer
  Required: False
  Minimum: 0
  Maximum: 50000

**bluePrimaryY**

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

  Type: integer
Properties

**Required**: False
**Minimum**: 0
**Maximum**: 50000

**whitePointX**

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 50000

**whitePointY**

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

- **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**: False
- **Minimum**: 0
- **Maximum**: 65535

**maxContentLightLevel**

Maximum light level among all samples in the coded video sequence, in units of candelas per square meter.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 65535

**maxLuminance**

Nominal maximum mastering display luminance in units of 0.0001 candelas per square meter.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
**Maximum**: 2147483647

**minLuminance**
Nominal minimum mastering display luminance in units of 0.0001 candelas per square meter

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 2147483647

**ImageInserter**
Enable the image inserter feature to include a graphic overlay on your video. Enable or disable this feature for each input or output individually. This setting is disabled by default.

**insertableImages**
Specify the images that you want to overlay on your video. The images must be PNG or TGA files.

- **Type**: Array of type InsertableImage (p. 903)
- **Required**: False

**InsertableImage**
Settings that specify how your still graphic overlay appears.

**width**
Specify the width of the inserted image in pixels. If you specify a value that's larger than the video resolution width, the service will crop your overlaid image to fit. To use the native width of the image, keep this setting blank.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 2147483647

**height**
Specify the height of the inserted image in pixels. If you specify a value that's larger than the video resolution height, the service will crop your overlaid image to fit. To use the native height of the image, keep this setting blank.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 2147483647

**imageX**
Specify the distance, in pixels, between the inserted image and the left edge of the video frame. Required for any image overlay that you specify.
Properties

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

imageY
Specify the distance, in pixels, between the overlaid image and the top edge of the video frame. Required for any image overlay that you specify.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

duration
Specify the time, in milliseconds, for the image to remain on the output video. This duration includes fade-in time but not fade-out time.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

fadeIn
Specify the length of time, in milliseconds, between the Start time that you specify for the image insertion and the time that the image appears at full opacity. Full opacity is the level that you specify for the opacity setting. If you don't specify a value for Fade-in, the image will appear abruptly at the overlay start time.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

layer
Specify how overlapping inserted images appear. Images with higher values for Layer appear on top of images with lower values for Layer.

Type: integer
Required: False
Minimum: 0
Maximum: 99

imageInserterInput
Specify the Amazon S3 location of the image that you want to overlay on the video. Use a PNG or TGA file.

Type: string
Required: False
Properties

startTime

Specify the timecode of the frame that you want the overlay to first appear on. This must be in timecode (HH:MM:SS:FF or HH:MM:SS;FF) format. Remember to take into account your timecode source settings.

Type: string
Required: False
Pattern: ^((((0-1)\d)|(2[0-3]))(0-5)\d):(0-5)\d)$

fadeOut

Specify the length of time, in milliseconds, between the end of the time that you have specified for the image overlay Duration and when the overlaid image has faded to total transparency. 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: 0
Maximum: 2147483647

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: False
Minimum: 0
Maximum: 100

LanguageCode


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
<table>
<thead>
<tr>
<th>Code</th>
<th>Language</th>
</tr>
</thead>
<tbody>
<tr>
<td>LUG</td>
<td>LUG</td>
</tr>
<tr>
<td>KAT</td>
<td>KAT</td>
</tr>
<tr>
<td>ELL</td>
<td>ELL</td>
</tr>
<tr>
<td>GRN</td>
<td>GRN</td>
</tr>
<tr>
<td>GUJ</td>
<td>GUJ</td>
</tr>
<tr>
<td>HAT</td>
<td>HAT</td>
</tr>
<tr>
<td>HAU</td>
<td>HAU</td>
</tr>
<tr>
<td>HEB</td>
<td>HEB</td>
</tr>
<tr>
<td>HER</td>
<td>HER</td>
</tr>
<tr>
<td>HMO</td>
<td>HMO</td>
</tr>
<tr>
<td>HUN</td>
<td>HUN</td>
</tr>
<tr>
<td>ISL</td>
<td>ISL</td>
</tr>
<tr>
<td>IDO</td>
<td>IDO</td>
</tr>
<tr>
<td>IBO</td>
<td>IBO</td>
</tr>
<tr>
<td>IND</td>
<td>IND</td>
</tr>
<tr>
<td>INA</td>
<td>INA</td>
</tr>
<tr>
<td>ILE</td>
<td>ILE</td>
</tr>
<tr>
<td>IKU</td>
<td>IKU</td>
</tr>
<tr>
<td>IPK</td>
<td>IPK</td>
</tr>
<tr>
<td>GLE</td>
<td>GLE</td>
</tr>
<tr>
<td>JAV</td>
<td>JAV</td>
</tr>
<tr>
<td>KAL</td>
<td>KAL</td>
</tr>
<tr>
<td>KAN</td>
<td>KAN</td>
</tr>
<tr>
<td>KAU</td>
<td>KAU</td>
</tr>
<tr>
<td>KAS</td>
<td>KAS</td>
</tr>
<tr>
<td>KAZ</td>
<td>KAZ</td>
</tr>
<tr>
<td>KIK</td>
<td>KIK</td>
</tr>
<tr>
<td>KIN</td>
<td>KIN</td>
</tr>
<tr>
<td>KIR</td>
<td>KIR</td>
</tr>
<tr>
<td>KOM</td>
<td>KOM</td>
</tr>
<tr>
<td>KON</td>
<td>KON</td>
</tr>
<tr>
<td>KUA</td>
<td>KUA</td>
</tr>
<tr>
<td>KUR</td>
<td>KUR</td>
</tr>
<tr>
<td>LAO</td>
<td>LAO</td>
</tr>
<tr>
<td>LAT</td>
<td>LAT</td>
</tr>
<tr>
<td>LAV</td>
<td>LAV</td>
</tr>
<tr>
<td>LIM</td>
<td>LIM</td>
</tr>
<tr>
<td>LIN</td>
<td>LIN</td>
</tr>
<tr>
<td>LIT</td>
<td>LIT</td>
</tr>
<tr>
<td>LUB</td>
<td>LUB</td>
</tr>
<tr>
<td>LTZ</td>
<td>LTZ</td>
</tr>
<tr>
<td>MKD</td>
<td>MKD</td>
</tr>
<tr>
<td>MLG</td>
<td>MLG</td>
</tr>
<tr>
<td>MSA</td>
<td>MSA</td>
</tr>
<tr>
<td>MAL</td>
<td>MAL</td>
</tr>
<tr>
<td>MLT</td>
<td>MLT</td>
</tr>
<tr>
<td>GLV</td>
<td>GLV</td>
</tr>
<tr>
<td>MRI</td>
<td>MRI</td>
</tr>
<tr>
<td>MAR</td>
<td>MAR</td>
</tr>
<tr>
<td>MAH</td>
<td>MAH</td>
</tr>
<tr>
<td>MON</td>
<td>MON</td>
</tr>
<tr>
<td>NAU</td>
<td>NAU</td>
</tr>
<tr>
<td>NAV</td>
<td>NAV</td>
</tr>
</tbody>
</table>
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
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
ListPresetsRequest

You can send list presets requests with an empty body. Optionally, you can filter the response by category by specifying it in your request body. You can also optionally specify the maximum number, up to twenty, of queues to be returned.

listBy

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.

Type: PresetListBy (p. 937)
Required: False

category

Optionally, specify a preset category to limit responses to only presets from that category.

Type: string
Required: False

order

When you request lists of resources, you can optionally specify whether they are sorted in ASCENDING or DESCENDING order. Default varies by resource.

Type: Order (p. 935)
Required: False

nextToken

Use this string, provided with the response to a previous request, to request the next batch of presets.

Type: string
**Properties**

**maxResults**
Optional. Number of presets, up to twenty, that will be returned at one time

- **Type:** integer
- **Required:** False
- **Format:** int32
- **Minimum:** 1
- **Maximum:** 20

**ListPresetsResponse**
Successful list presets requests return a JSON array of presets. If you don't specify how they are ordered, you will receive them alphabetically by name.

**presets**
List of presets

- **Type:** Array of type Preset (p. 936)
- **Required:** False

**nextToken**
Use this string to request the next batch of presets.

- **Type:** string
- **Required:** False

**M2tsAudioBufferModel**
Selects between the DVB and ATSC buffer models for Dolby Digital audio.

- **DVB**
- **ATSC**

**M2tsBufferModel**
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**
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**

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**

Controls whether to include the ES Rate field in the PES header.

INCLUDE
EXCLUDE

**M2tsForceTsVideoEbpOrder**

Keep the default value (DEFAULT) unless you know that your audio EBP markers are incorrectly appearing before your video EBP markers. To correct this problem, set this value to Force (FORCE).

FORCE
DEFAULT

**M2tsNielsenId3**

If INSERT, Nielsen inaudible tones for media tracking will be detected in the input audio and an equivalent ID3 tag will be inserted in the output.

INSERT
NONE

**M2tsPcrControl**

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**

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

**M2tsScte35Esam**

Settings for SCTE-35 signals from ESAM. Include this in your job settings to put SCTE-35 markers in your HLS and transport stream outputs at the insertion points that you specify in an ESAM XML document. Provide the document in the setting SCC XML (sccXml).

**scte35EsamPid**

Packet Identifier (PID) of the SCTE-35 stream in the transport stream generated by ESAM.

- **Type**: integer
- **Required**: False
- **Minimum**: 32
- **Maximum**: 8182

**M2tsScte35Source**

Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output.

- PASSTHROUGH
- NONE

**M2tsSegmentationMarkers**

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-101-130118. 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**

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

MPEG-2 TS container settings. These apply to outputs in a File output group when the output's container (ContainerType) is MPEG-2 Transport Stream (M2TS). In these assets, data is organized by the program map table (PMT). Each transport stream program contains subsets of data, including audio, video, and metadata. Each of these subsets of data has a numerical label called a packet identifier (PID). Each transport stream program corresponds to one MediaConvert output. The PMT lists the types of data in a program along with their PID. Downstream systems and players use the program map table to look up the PID for each type of data it accesses and then uses the PIDs to locate specific data within the asset.

audioBufferModel

Selects between the DVB and ATSC buffer models for Dolby Digital audio.

Type: M2tsAudioBufferModel (p. 910)
Required: False

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

esRateInPes

Controls whether to include the ES Rate field in the PES header.

Type: M2tsEsRateInPes (p. 911)
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

dvbNitSettings

Inserts DVB Network Information Table (NIT) at the specified table repetition interval.

Type: DvbNitSettings (p. 863)
Required: False
**dvbSdtSettings**

Inserts DVB Service Description Table (NIT) at the specified table repetition interval.

- **Type:** DvbSdtSettings (p. 864)
- **Required:** False

**scte35Source**

Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output.

- **Type:** M2tsScte35Source (p. 912)
- **Required:** False

**scte35Pid**

Specify the packet identifier (PID) of the SCTE-35 stream in the transport stream.

- **Type:** integer
- **Required:** False
- **Minimum:** 32
- **Maximum:** 8182

**scte35Esam**

Include this in your job settings to put SCTE-35 markers in your HLS and transport stream outputs at the insertion points that you specify in an ESAM XML document. Provide the document in the setting SCC XML (sccXml).

- **Type:** M2tsScte35Esam (p. 912)
- **Required:** False

**videoPid**

Specify the packet identifier (PID) of the elementary video stream in the transport stream.

- **Type:** integer
- **Required:** False
- **Minimum:** 32
- **Maximum:** 8182

**dvbTdtSettings**

Inserts DVB Time and Date Table (TDT) at the specified table repetition interval.

- **Type:** DvbTdtSettings (p. 869)
- **Required:** False

**pmtInterval**

Specify the number of milliseconds between instances of the program map table (PMT) in the output transport stream.

- **Type:** integer
Properties

**Required**: False
**Minimum**: 0
**Maximum**: 1000

**segmentationStyle**

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.

**Type**: M2tsSegmentationStyle (p. 912)
**Required**: False

**segmentationTime**

Specify the length, in seconds, of each segment. Required unless markers is set to _none_.

**Type**: number
**Required**: False
**Format**: float
**Minimum**: 0.0

**pmtPid**

Specify the packet identifier (PID) for the program map table (PMT) itself. Default is 480.

**Type**: integer
**Required**: False
**Minimum**: 32
**Maximum**: 8182

**bitrate**

Specify 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

**audioPids**

Specify the packet identifiers (PIDs) for any elementary audio streams you include in this output. Specify multiple PIDs as a JSON array. Default is the range 482-492.

**Type**: Array of type integer
Required: False
Minimum: 32
Maximum: 8182

privateMetadataPid

Specify the packet identifier (PID) of the private metadata stream. Default is 503.

Type: integer
Required: False
Minimum: 32
Maximum: 8182

nielsenId3

If INSERT, Nielsen inaudible tones for media tracking will be detected in the input audio and an equivalent ID3 tag will be inserted in the output.

Type: M2tsNielsenId3 (p. 911)
Required: False

timedMetadataPid

Specify the packet identifier (PID) for timed metadata in this output. Default is 502.

Type: integer
Required: False
Minimum: 32
Maximum: 8182

maxPcrInterval

Specify the maximum time, in milliseconds, between Program Clock References (PCRs) inserted into the transport stream.

Type: integer
Required: False
Minimum: 0
Maximum: 500

transportStreamId

Specify the ID for the transport stream itself in the program map table for this output. Transport stream IDs and program map tables are parts of MPEG-2 transport stream containers, used for organizing data.

Type: integer
Required: False
Minimum: 0
Maximum: 65535

dvbSubPids

Specify the packet identifiers (PIDs) for DVB subtitle data included in this output. Specify multiple PIDs as a JSON array. Default is the range 460-479.
Type: Array of type integer
Required: False
Minimum: 32
Maximum: 8182

rateMode

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.

Type: M2tsRateMode (p. 911)
Required: False

audioFramesPerPes

The number of audio frames to insert for each PES packet.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

pcrControl

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.

Type: M2tsPcrControl (p. 911)
Required: False

segmentationMarkers

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.

Type: M2tsSegmentationMarkers (p. 912)
Required: False

ebpAudioInterval

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).

Type: M2tsEbpAudioInterval (p. 910)
Required: False
forceTsVideoEbpOrder

Keep the default value (DEFAULT) unless you know that your audio EBP markers are incorrectly appearing before your video EBP markers. To correct this problem, set this value to Force (FORCE).

**Type:** M2tsForceTsVideoEbpOrder (p. 911)
**Required:** False

programNumber

Use Program number (programNumber) to specify the program number used in the program map table (PMT) for this output. Default is 1. Program numbers and program map tables are parts of MPEG-2 transport stream containers, used for organizing data.

**Type:** integer
**Required:** False
**Minimum:** 0
**Maximum:** 65535

pcrPid

Specify the packet identifier (PID) for the program clock reference (PCR) in this output. If you do not specify a value, the service will use the value for Video PID (VideoPid).

**Type:** integer
**Required:** False
**Minimum:** 32
**Maximum:** 8182

bufferModel

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.

**Type:** M2tsBufferModel (p. 910)
**Required:** False

dvbTeletextPid

Specify the packet identifier (PID) for DVB teletext data you include in this output. Default is 499.

**Type:** integer
**Required:** False
**Minimum:** 32
**Maximum:** 8182

fragmentTime

The length, in seconds, of each fragment. Only used with EBP markers.

**Type:** number
**Required:** False
**Format:** float
**Minimum:** 0.0
Properties

**ebpPlacement**
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).

*Type:* M2tsEbpPlacement (p. 911)
*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

**M3u8NielsenId3**
If INSERT, Nielsen inaudible tones for media tracking will be detected in the input audio and an equivalent ID3 tag will be inserted in the output.

INSERT
NONE

**M3u8PcrControl**
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**
Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output.

PASSTHROUGH
NONE

**M3u8Settings**
Settings for TS segments in HLS

**audioFramesPerPes**
The number of audio frames to insert for each PES packet.

*Type:* integer
*Required:* False
*Minimum:* 0
Maximum: 2147483647

pcrControl

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.

Type: M3u8PcrControl (p. 919)
Required: False

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.

Type: integer
Required: False
Minimum: 32
Maximum: 8182

pmtPid

Packet Identifier (PID) for the Program Map Table (PMT) in the transport stream.

Type: integer
Required: False
Minimum: 32
Maximum: 8182

privateMetadataPid

Packet Identifier (PID) of the private metadata stream in the transport stream.

Type: integer
Required: False
Minimum: 32
Maximum: 8182

programNumber

The value of the program number field in the Program Map Table.

Type: integer
Required: False
Minimum: 0
Maximum: 65535

patInterval

The number of milliseconds between instances of this table in the output transport stream.

Type: integer
Required: False
Minimum: 0
Properties

**Maximum**

1000

**pmtInterval**

The number of milliseconds between instances of this table in the output transport stream.

- Type: integer
- Required: False
- Minimum: 0
- Maximum: 1000

**scte35Source**

Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output.

- Type: M3u8Scte35Source (p. 919)
- Required: False

**scte35Pid**

Packet Identifier (PID) of the SCTE-35 stream in the transport stream.

- Type: integer
- Required: False
- Minimum: 32
- Maximum: 8182

**nielsenId3**

If INSERT, Nielsen inaudible tones for media tracking will be detected in the input audio and an equivalent ID3 tag will be inserted in the output.

- Type: M3u8NielsenId3 (p. 919)
- Required: False

**timedMetadata**

Applies only to HLS outputs. Use this setting to specify whether the service inserts the ID3 timed metadata from the input in this output.

- Type: TimedMetadata (p. 945)
- Required: False

**timedMetadataPid**

Packet Identifier (PID) of the timed metadata stream in the transport stream.

- Type: integer
- Required: False
- Minimum: 32
- Maximum: 8182

**transportStreamId**

The value of the transport stream ID field in the Program Map Table.
**Properties**

**videoPid**

Packet Identifier (PID) of the elementary video stream in the transport stream.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 65535

**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.

- **Type**: Array of type integer
- **Required**: False
- **Minimum**: 32
- **Maximum**: 8182

**MovClapAtom**

When enabled, include 'clap' atom if appropriate for the video output settings.

- **include**
- **exclude**

**MovCslgAtom**

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**

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**

If set to OMNEON, inserts Omneon-compatible padding
OMNEON
NONE

MovReference
Always keep the default value (SELF_CONTAINED) for this setting.

  SELF_CONTAINED
  EXTERNAL

MovSettings
Settings for MOV Container.

clapAtom
When enabled, include 'clap' atom if appropriate for the video output settings.

  Type: MovClapAtom (p. 922)
  Required: False

cslgAtom
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.

  Type: MovCslgAtom (p. 922)
  Required: False

paddingControl
If set to OMNEON, inserts Omneon-compatible padding

  Type: MovPaddingControl (p. 922)
  Required: False

reference
Always keep the default value (SELF_CONTAINED) for this setting.

  Type: MovReference (p. 923)
  Required: False

mpeg2FourCCControl
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.

  Type: MovMpeg2FourCCControl (p. 922)
  Required: False
**Mp2Settings**

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value MP2.

**bitrate**

Average bitrate in bits/second.

- **Type:** integer
- **Required:** False
- **Minimum:** 32000
- **Maximum:** 384000

**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

**sampleRate**

Sample rate in hz.

- **Type:** integer
- **Required:** False
- **Minimum:** 32000
- **Maximum:** 48000

**Mp4CslgAtom**

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**

Inserts a free-space box immediately after the moov box.

- INCLUDE
- EXCLUDE

**Mp4MoovPlacement**

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

Settings for MP4 Container

cslgAtom

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.

  Type: Mp4CslgAtom (p. 924)
  Required: False

freeSpaceBox

Inserts a free-space box immediately after the moov box.

  Type: Mp4FreeSpaceBox (p. 924)
  Required: False

mp4MajorBrand

Overrides the "Major Brand" field in the output file. Usually not necessary to specify.

  Type: string
  Required: False

moovPlacement

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.

  Type: Mp4MoovPlacement (p. 924)
  Required: False

Mpeg2AdaptiveQuantization

Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

  OFF
  LOW
  MEDIUM
  HIGH

Mpeg2CodecLevel

Use Level (Mpeg2CodecLevel) to set the MPEG-2 level for the video output.

  AUTO
**Mpeg2CodecProfile**

Use Profile (Mpeg2CodecProfile) to set the MPEG-2 profile for the video output.

- MAIN
- PROFILE_422

**Mpeg2DynamicSubGop**

Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting B frames between reference frames (numberBFramesBetweenReferenceFrames).

- ADAPTIVE
- STATIC

**Mpeg2FramerateControl**

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

- INITIALIZE_FROM_SOURCE
- SPECIFIED

**Mpeg2FramerateConversionAlgorithm**

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

- DUPLICATE_DROP
- INTERPOLATE

**Mpeg2GopSizeUnits**

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**

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**

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**

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**

Use Quality tuning level (Mpeg2QualityTuningLevel) to specify whether to use single-pass or multipass video encoding.

- SINGLE_PASS
- MULTI_PASS

**Mpeg2RateControlMode**

Use Rate control mode (Mpeg2RateControlMode) to specify whether the bitrate is variable (vbr) or constant (cbr).

- VBR
- CBR
Mpeg2SceneChangeDetect

Scene change detection (inserts I-frames on scene changes).

- DISABLED
- ENABLED

Mpeg2Settings

Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value MPEG2.

interlaceMode

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.

- **Type**: Mpeg2InterlaceMode (p. 927)
- **Required**: False

parNumerator

Pixel Aspect Ratio numerator.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 2147483647

syntax

Produces a Type D-10 compatible bitstream (SMPTE 356M-2001).

- **Type**: Mpeg2Syntax (p. 933)
- **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

framerateDenominator

Frame rate denominator.

- **Type**: integer
Required: False
Minimum: 1
Maximum: 1001

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

hrdBufferInitialFillPercentage
Percentage of the buffer that should initially be filled (HRD buffer model).

Type: integer
Required: False
Minimum: 0
Maximum: 100

gopSize
GOP Length (keyframe interval) in frames or seconds. Must be greater than zero.

Type: number
Required: False
Format: float
Minimum: 0.0

hrdBufferSize
Size of buffer (HRD buffer model) in bits. For example, enter five megabits as 5000000.

Type: integer
Required: False
Minimum: 0
Maximum: 47185920

maxBitrate
Maximum bitrate in bits/second. For example, enter five megabits per second as 5000000.

Type: integer
Required: False
Minimum: 1000
Maximum: 300000000

slowPal
Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.
Type: Mpeg2SlowPal (p. 933)
Required: False

parDenominator

Pixel Aspect Ratio denominator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

spatialAdaptiveQuantization

Adjust quantization within each frame based on spatial variation of content complexity.

Type: Mpeg2SpatialAdaptiveQuantization (p. 933)
Required: False

temporalAdaptiveQuantization

Adjust quantization within each frame based on temporal variation of content complexity.

Type: Mpeg2TemporalAdaptiveQuantization (p. 933)
Required: False

bitrate

Average bitrate in bits/second. Required for VBR and CBR. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.

Type: integer
Required: False
Minimum: 1000
Maximum: 288000000

intraDcPrecision

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.

Type: Mpeg2IntraDcPrecision (p. 927)
Required: False

framerateControl

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose
SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

**Type:** Mpeg2FramerateControl (p. 926)  
**Required:** False

### rateControlMode

Use Rate control mode (Mpeg2RateControlMode) to specify whether the bitrate is variable (vbr) or constant (cbr).

**Type:** Mpeg2RateControlMode (p. 927)  
**Required:** False

### codecProfile

Use Profile (Mpeg2CodecProfile) to set the MPEG-2 profile for the video output.

**Type:** Mpeg2CodecProfile (p. 926)  
**Required:** False

### telecine

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.

**Type:** Mpeg2Telecine (p. 933)  
**Required:** False

### framerateNumerator

Frame rate numerator - frame rate is a fraction, e.g. 24000 / 1001 = 23.976 fps.

**Type:** integer  
**Required:** False  
**Minimum:** 24  
**Maximum:** 60000

### 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

### adaptiveQuantization

Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.
**Properties**

**Type**: Mpeg2AdaptiveQuantization (p. 925)
**Required**: False

**codecLevel**
Use Level (Mpeg2CodecLevel) to set the MPEG-2 level for the video output.
**Type**: Mpeg2CodecLevel (p. 925)
**Required**: False

**sceneChangeDetect**
Scene change detection (inserts I-frames on scene changes).
**Type**: Mpeg2SceneChangeDetect (p. 928)
**Required**: False

**qualityTuningLevel**
Use Quality tuning level (Mpeg2QualityTuningLevel) to specify whether to use single-pass or multipass video encoding.
**Type**: Mpeg2QualityTuningLevel (p. 927)
**Required**: False

**framerateConversionAlgorithm**
When set to INTERPOLATE, produces smoother motion during frame rate conversion.
**Type**: Mpeg2FramerateConversionAlgorithm (p. 926)
**Required**: False

**gopSizeUnits**
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.
**Type**: Mpeg2GopSizeUnits (p. 926)
**Required**: False

**parControl**
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.
**Type**: Mpeg2ParControl (p. 927)
**Required**: False

**numberBFramesBetweenReferenceFrames**
Number of B-frames between reference frames.
**Type**: integer
**Required**: False
Minimum: 0  
Maximum: 7

dynamicSubGop

Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting B frames between reference frames (numberBFramesBetweenReferenceFrames).

Type: Mpeg2DynamicSubGop (p. 926)  
Required: False

Mpeg2SlowPal

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

DISABLED  
ENABLED

Mpeg2SpatialAdaptiveQuantization

Adjust quantization within each frame based on spatial variation of content complexity.

DISABLED  
ENABLED

Mpeg2Syntax

Produces a Type D-10 compatible bitstream (SMPTE 356M-2001).

DEFAULT  
D_10

Mpeg2Telecine

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

Adjust quantization within each frame based on temporal variation of content complexity.

DISABLED
ENABLED

**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. When you enable Noise reducer (NoiseReducer), you must also select a value for Noise reducer filter (NoiseReducerFilter).

**filter**

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 a frequency-domain filter based on JND principles.

**Type:** NoiseReducerFilter (p. 934)

**Required:** False

**filterSettings**

Settings for a noise reducer filter

**Type:** NoiseReducerFilterSettings (p. 934)

**Required:** False

**.spatialFilterSettings**

Noise reducer filter settings for spatial filter.

**Type:** NoiseReducerSpatialFilterSettings (p. 935)

**Required:** False

**NoiseReducerFilter**

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 a frequency-domain filter based on JND principles.

**BILATERAL**

**MEAN**

**GAUSSIAN**

**LANCZOS**

**SHARPEN**

**CONSERVE**

**SPATIAL**

**NoiseReducerFilterSettings**

Settings for a noise reducer filter
strength

Relative strength of noise reducing filter. Higher values produce stronger filtering.

  Type: integer
  Required: False
  Minimum: 0
  Maximum: 3

NoiseReducerSpatialFilterSettings

Noise reducer filter settings for spatial filter.

strength

Relative strength of noise reducing filter. Higher values produce stronger filtering.

  Type: integer
  Required: False
  Minimum: 0
  Maximum: 16

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

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

Order

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

OutputChannel mapping settings.
inputChannels

List of input channels

Type: Array of type integer
Required: False
Minimum: -60
Maximum: 6

OutputSdt

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

A preset is a collection of preconfigured media conversion settings that you want MediaConvert to apply to the output during the conversion process.

arn

An identifier for this resource that is unique within all of AWS.

Type: string
Required: False

createdAt

The timestamp in epoch seconds for preset creation.

Type: string
Required: False
Format: date-time

lastUpdated

The timestamp in epoch seconds when the preset was last updated.

Type: string
Required: False
Format: date-time

description

An optional description you create for each preset.
Properties

Type: string  
Required: False

category

An optional category you create to organize your presets.

Type: string  
Required: False

name

A name you create for each preset. Each name must be unique within your account.

Type: string  
Required: True

type

A preset can be of two types: system or custom. System or built-in preset can't be modified or deleted by the user.

Type: Type (p. 945)  
Required: False

settings

Settings for preset

Type: PresetSettings (p. 937)  
Required: True

PresetListBy

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

Settings for preset

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. 947)  
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. 847)
  Required: False

containerSettings

Container specific settings.

  Type: ContainerSettings (p. 860)
  Required: False

captionDescriptions

Caption settings for this preset. There can be multiple caption settings in a single output.

  Type: Array of type CaptionDescriptionPreset (p. 856)
  Required: False

ProresCodecProfile

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

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

  INITIALIZE_FROM_SOURCE
  SPECIFIED

ProresFramerateConversionAlgorithm

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

  DUPLICATE_DROP
  INTERPOLATE
**ProresInterlaceMode**

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**

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**

Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value PRORES.

**interlaceMode**

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.

- **Type**: ProresInterlaceMode (p. 939)
- **Required**: False

**parNumerator**

Pixel Aspect Ratio numerator.

- **Type**: integer
  - **Required**: False
  - **Minimum**: 1
  - **Maximum**: 2147483647
framerateDenominator

Frame rate denominator.

*Type:* integer  
*Required:* False  
*Minimum:* 1  
*Maximum:* 2147483647

codecProfile

Use Profile (ProResCodecProfile) to specify the type of Apple ProRes codec to use for this output.

*Type:* ProresCodecProfile (p. 938)  
*Required:* False

slowPal

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

*Type:* ProresSlowPal (p. 941)  
*Required:* False

denominator

Pixel Aspect Ratio denominator.

*Type:* integer  
*Required:* False  
*Minimum:* 1  
*Maximum:* 2147483647

framerateControl

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

*Type:* ProresFramerateControl (p. 938)  
*Required:* False

telecine

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.

*Type:* ProresTelecine (p. 941)  
*Required:* False
framerateNumerator

When you use the API for transcode jobs that use frame rate conversion, specify the frame rate 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

framerateConversionAlgorithm

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

- Type: ProresFramerateConversionAlgorithm (p. 938)
- Required: False

parControl

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).

- Type: ProresParControl (p. 939)
- Required: False

ProresSlowPal

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

- DISABLED
- ENABLED

ProresTelecine

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

Use Rectangle to identify a specific area of the video frame.

height

Height of rectangle in pixels. Specify only even numbers.

- Type: integer
width

Width of rectangle in pixels. Specify only even numbers.

Type: integer
Required: False
Minimum: 2
Maximum: 2147483647

x

The distance, in pixels, between the rectangle and the left edge of the video frame. Specify only even numbers.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

y

The distance, in pixels, between the rectangle and the top edge of the video frame. Specify only even numbers.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

RemixSettings

Use Manual audio remixing (RemixSettings) to adjust audio levels for each audio channel in each output of your job. With audio remixing, you can output more or fewer audio channels than your input audio source provides.

channelMapping

Channel mapping (ChannelMapping) contains the group of fields that hold the remixing value for each channel. Units are in dB. Acceptable values are within the range from -60 (mute) through 6. A setting of 0 passes the input channel unchanged to the output channel (no attenuation or amplification).

Type: ChannelMapping (p. 858)
Required: False

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: False
Minimum: 1
Maximum: 16

**channelsOut**
Specify the number of channels in this output after remixing. Valid values: 1, 2, 4, 6, 8

Type: integer
Required: False
Minimum: 1
Maximum: 8

**RespondToAfd**
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**
Applies only if your input aspect ratio is different from your output aspect ratio. Choose "Stretch to output" to have the service stretch your video image to fit. Keep the setting "Default" 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**
Set Framerate (SccDestinationFramerate) to make sure that the captions and the video are synchronized in the output. Specify a frame rate that matches the frame rate of the associated video. If the video frame rate 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**
Settings for SCC caption output.
framerate

Set framerate (SccDestinationFramerate) to make sure that the captions and the video are synchronized in the output. Specify a frame rate that matches the frame rate of the associated video. If the video frame rate 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).

Type: SccDestinationFramerate (p. 943)
Required: False

TeletextDestinationSettings

Settings for Teletext caption output

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]$
MinLength: 3
MaxLength: 3

TimecodeBurnin

Timecode burn-in (TimecodeBurnIn)--Burns the output timecode and specified prefix into the output.

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

Use Position (Position) under under Timecode burn-in (TimecodeBurnIn) to specify the location the burned-in timecode on output video.

Type: TimecodeBurninPosition (p. 945)
Required: False

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**: `^[ -~]+$`

### TimecodeBurninPosition

Use Position (Position) under 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

Applies only to HLS outputs. Use this setting to specify whether the service inserts the ID3 timed metadata from the input in this output.

- PASSTHROUGH
- NONE

### TtmlDestinationSettings

Settings specific to TTML caption outputs, including Pass style information (TtmlStylePassthrough).

#### stylePassthrough

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.

- **Type**: TtmlStylePassthrough (p. 945)
- **Required**: False

### TtmlStylePassthrough

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**

- SYSTEM
CUSTOM

**VideoCodec**

Type of video codec

- FRAME_CAPTURE
- H_264
- H_265
- MPEG2
- PRORES

**VideoCodecSettings**

Video codec settings, (CodecSettings) under (VideoDescription), contains the group of settings related to video encoding. The settings in this group vary depending on the value you choose for Video codec (Codec). For each codec enum you choose, define the corresponding settings object. The following lists the codec enum, settings object pairs. * H_264, H264Settings * H_265, H265Settings * MPEG2, Mpeg2Settings * PRORES, ProresSettings * FRAME_CAPTURE, FrameCaptureSettings

**codec**

Specifies the video codec. This must be equal to one of the enum values defined by the object VideoCodec.

- **Type:** VideoCodec (p. 946)
- **Required:** False

**frameCaptureSettings**

Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value FRAME_CAPTURE.

- **Type:** FrameCaptureSettings (p. 876)
- **Required:** False

**h264Settings**

Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value H_264.

- **Type:** H264Settings (p. 881)
- **Required:** False

**h265Settings**

Settings for H265 codec

- **Type:** H265Settings (p. 892)
- **Required:** False

**mpeg2Settings**

Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value MPEG2.
Properties

Type: Mpeg2Settings (p. 928)
Required: False

proresSettings

Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value PRORES.

Type: ProresSettings (p. 939)
Required: False

VideoDescription

Settings for video outputs

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

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

scalingBehavior

Applies only if your input aspect ratio is different from your output aspect ratio. Choose “Stretch to output” to have the service stretch your video image to fit. Keep the setting "Default" to allow the service to letterbox your video instead. This setting overrides any positioning value you specify elsewhere in the job.

Type: ScalingBehavior (p. 943)
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. If you crop your input frame size to smaller than your output frame size, make sure to specify the behavior you want in your output setting "Scaling behavior".

Type: Rectangle (p. 941)
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

videoPreprocessors

Find additional transcoding features under Preprocessors (VideoPreprocessors). Enable the features at each output individually. These features are disabled by default.

Type: VideoPreprocessor (p. 950)  
Required: False

timecodeInsertion

Applies only to H.264, H.265, MPEG2, and ProRes outputs. Only enable Timecode insertion when the input frame rate is identical to the output frame rate. To include timecodes in this output, set Timecode insertion (VideoTimecodeInsertion) to PIC_TIMING_SEI. To leave them out, set it to DISABLED. Default is DISABLED. When the service inserts timecodes in an output, by default, it uses any embedded timecodes from the input. If none are present, the service will set the timecode for the first output frame to zero. To change this default behavior, adjust the settings under Timecode configuration (TimecodeConfig). In the console, these settings are located under Job > Job settings > Timecode configuration. Note – Timecode source under input settings (InputTimecodeSource) does not affect the timecodes that are inserted in the output. Source under Job settings > Timecode configuration (TimecodeSource) does.

Type: VideoTimecodeInsertion (p. 950)  
Required: False

antiAlias

The anti-alias filter is automatically applied to all outputs. The service no longer accepts the value DISABLED for AntiAlias. If you specify that in your job, the service will ignore the setting.

Type: AntiAlias (p. 846)  
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. 941)  
Required: False

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. 0 is the softest setting, 100 the sharpest, and 50 recommended for most content.
Properties

- **Type**: integer
  - **Required**: False
  - **Minimum**: 0
  - **Maximum**: 100

**codecSettings**

Video codec settings, (CodecSettings) under (VideoDescription), contains the group of settings related to video encoding. The settings in this group vary depending on the value you choose for Video codec (Codec). For each codec enum you choose, define the corresponding settings object. The following lists the codec enum, settings object pairs. * H_264, H264Settings * H_265, H265Settings * MPEG2, Mpeg2Settings * PRORES, ProresSettings * FRAME_CAPTURE, FrameCaptureSettings

  - **Type**: VideoCodecSettings (p. 946)
  - **Required**: False

**afdSignaling**

This setting only applies to H.264, H.265, and MPEG2 outputs. Use Insert AFD signaling (AfdSignaling) to specify whether the service includes 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.

  - **Type**: AfdSignaling (p. 845)
  - **Required**: False

**dropFrameTimecode**

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.

  - **Type**: DropFrameTimecode (p. 863)
  - **Required**: False

**respondToAfd**

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.

  - **Type**: RespondToAfd (p. 943)
  - **Required**: False

**colorMetadata**

Enable Insert color metadata (ColorMetadata) to include color metadata in this output. This setting is enabled by default.

  - **Type**: ColorMetadata (p. 859)
Required: False

VideoPreprocessor

Find additional transcoding features under Preprocessors (VideoPreprocessors). Enable the features at each output individually. These features are disabled by default.

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. 858)
Required: False

deinterlacer

Use Deinterlacer (Deinterlacer) to produce smoother motion and a clearer picture.

Type: Deinterlacer (p. 862)
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. 903)
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. 934)
Required: False

timecodeBurnin

Timecode burn-in (TimecodeBurnIn)--Burns the output timecode and specified prefix into the output.

Type: TimecodeBurnin (p. 944)
Required: False

VideoTimecodeInsertion

Applies only to H.264, H.265, MPEG2, and ProRes outputs. Only enable Timecode insertion when the input frame rate is identical to the output frame rate. To include timecodes in this output, set Timecode insertion (VideoTimecodeInsertion) to PIC_TIMING_SEI. To leave them out, set it to DISABLED. Default is DISABLED. When the service inserts timecodes in an output, by default, it uses any embedded timecodes from the input. If none are present, the service will set the timecode for the first output frame to zero. To change this default behavior, adjust the settings under Timecode configuration (TimecodeConfig). In the console, these settings are located under Job > Job settings > Timecode configuration. Note - Timecode
source under input settings (InputTimecodeSource) does not affect the timecodes that are inserted in the output. Source under Job settings > Timecode configuration (TimecodeSource) does.

DISABLED
PIC_TIMING_SEI

### WavFormat

The service defaults to using RIFF for WAV outputs. If your output audio is likely to exceed 4 GB in file size, or if you otherwise need the extended support of the RF64 format, set your output WAV file format to RF64.

RIFF
RF64

### WavSettings

Required when you set (Codec) under (AudioDescriptions)> (CodecSettings) to the value WAV.

#### 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

#### 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

#### sampleRate

Sample rate in Hz.

- **Type**: integer
- **Required**: False
- **Minimum**: 8000
- **Maximum**: 192000

#### format

The service defaults to using RIFF for WAV outputs. If your output audio is likely to exceed 4 GB in file size, or if you otherwise need the extended support of the RF64 format, set your output WAV file format to RF64.

- **Type**: WavFormat (p. 951)
Required: False

See Also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

ListPresets

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Go - Pilot
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

CreatePreset

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Go - Pilot
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

Presets name

URI

/2017-08-29/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. 963)</td>
<td>200 response</td>
</tr>
<tr>
<td>400</td>
<td>ExceptionBody (p. 978)</td>
<td>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>403</td>
<td>ExceptionBody (p. 978)</td>
<td>You don't have permissions for this action with the credentials you sent.</td>
</tr>
<tr>
<td>404</td>
<td>ExceptionBody (p. 978)</td>
<td>The resource you requested does not exist.</td>
</tr>
<tr>
<td>409</td>
<td>ExceptionBody (p. 978)</td>
<td>The service could not complete your request because there is a conflict with the current state of the resource.</td>
</tr>
<tr>
<td>429</td>
<td>ExceptionBody (p. 978)</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>
<tr>
<td>500</td>
<td>ExceptionBody (p. 978)</td>
<td>The service encountered an unexpected condition and cannot fulfill your request.</td>
</tr>
</tbody>
</table>

### 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. 970)</td>
<td>200 response</td>
</tr>
</tbody>
</table>

953
### 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>
<tbody>
<tr>
<td>202</td>
<td>DeletePresetResponse (p. 978)</td>
<td>202 response</td>
</tr>
<tr>
<td>400</td>
<td>ExceptionBody (p. 978)</td>
<td>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>403</td>
<td>ExceptionBody (p. 978)</td>
<td>You don't have permissions for this action with the credentials you sent.</td>
</tr>
<tr>
<td>404</td>
<td>ExceptionBody (p. 978)</td>
<td>The resource you requested does not exist.</td>
</tr>
<tr>
<td>409</td>
<td>ExceptionBody (p. 978)</td>
<td>The service could not complete your request because there is a conflict with the current state of the resource.</td>
</tr>
<tr>
<td>429</td>
<td>ExceptionBody (p. 978)</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>
<tr>
<td>500</td>
<td>ExceptionBody (p. 978)</td>
<td>The service encountered an unexpected condition and cannot fulfill your request.</td>
</tr>
</tbody>
</table>
Schemas

Request Bodies

Example GET

```json
{
   "name": "string"
}
```

Example PUT

```json
{
   "description": "string",
   "category": "string",
   "name": "string",
   "settings": {
      "videoDescription": {
         "fixedAfd": integer,
         "width": integer,
         "scalingBehavior": enum,
         "crop": {
            "height": integer,
            "width": integer,
            "x": integer,
            "y": integer
         },
         "height": integer,
         "videoPreprocessors": {
            "colorCorrector": {
               "brightness": integer,
               "colorSpaceConversion": enum,
               "contrast": integer,
               "hue": integer,
               "saturation": integer,
               "hdr10Metadata": {
                  "redPrimaryX": integer,
                  "redPrimaryY": integer,
                  "greenPrimaryX": integer,
                  "greenPrimaryY": integer,
                  "bluePrimaryX": integer,
                  "bluePrimaryY": integer,
                  "whitePointX": integer,
```
"whitePointY": integer,
"maxFrameAverageLightLevel": integer,
"maxContentLightLevel": integer,
"maxLuminance": integer,
"minLuminance": integer
}
,
"deinterlacer": {
"algorithm": enum,
"mode": enum,
"control": enum
},
"imageInserter": {
"insertableImages": [
{
"width": integer,
"height": integer,
"imageX": integer,
"imageY": integer,
"duration": integer,
"fadeIn": integer,
"layer": integer,
"imageInserterInput": "string",
"startTime": "string",
"fadeOut": integer,
"opacity": integer
}
]
},
"noiseReducer": {
"filter": enum,
"filterSettings": {
"strength": integer
},
"spatialFilterSettings": {
"strength": integer,
"speed": integer,
"postFilterSharpenStrength": integer
}
},
"timecodeBurnin": {
"fontSize": integer,
"position": enum,
"prefix": "string"
}
},
"timecodeInsertion": enum,
"antiAlias": enum,
"position": {
"height": integer,
"width": integer,
"x": integer,
"y": integer
},
"sharpness": integer,
"codecSettings": {
"codec": enum,
"frameCaptureSettings": {
"framerateNumerator": integer,
"framerateDenominator": integer,
"maxCaptures": integer,
"quality": integer
},
"h264Settings": {
"interlaceMode": enum,
"parNumerator": integer,
"numberReferenceFrames": integer,
"syntax": enum,
"softness": integer,
"framerateDenominator": integer,
"gopClosedCadence": integer,
"hrdBufferInitialFillPercentage": integer,
"gopSize": number,
"slices": integer,
"gopBReference": enum,
"hrdBufferSize": integer,
"maxBitrate": integer,
"slowPal": enum,
"parDenominator": integer,
"spatialAdaptiveQuantization": enum,
"temporalAdaptiveQuantization": enum,
"entropyEncoding": enum,
"bitrate": integer,
"framerateControl": enum,
"rateControlMode": enum,
"qvbrSettings": {
  "qvbrQualityLevel": integer,
  "maxAverageBitrate": integer
},
"codecProfile": enum,
"telecine": enum,
"framerateNumerator": integer,
"minIInterval": integer,
"adaptiveQuantization": enum,
"codecLevel": enum,
"fieldEncoding": enum,
"sceneChangeDetect": enum,
"qualityTuningLevel": enum,
"framerateConversionAlgorithm": enum,
"unregisteredSeiT imecode": enum,
"gopSizeUnits": enum,
"parControl": enum,
"numberBFramesBetweenReferenceFrames": integer,
"repeatPps": enum,
"dynamicSubGop": enum
},
"h265Settings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "numberReferenceFrames": integer,
  "framerateDenominator": integer,
  "gopClosedCadence": integer,
  "alternateTransferFunctionSei": enum,
  "hrdBufferInitialFillPercentage": integer,
  "gopSize": number,
  "slices": integer,
  "gopBReference": enum,
  "hrdBufferSize": integer,
  "maxBitrate": integer,
  "slowPal": enum,
  "parDenominator": integer,
  "spatialAdaptiveQuantization": enum,
  "temporalAdaptiveQuantization": enum,
  "flickerAdaptiveQuantization": enum,
  "bitrate": integer,
  "framerateControl": enum,
  "rateControlMode": enum,
  "qvbrSettings": {
    "qvbrQualityLevel": integer,
    "maxAverageBitrate": integer
  }
}
"codecProfile": enum,
"tiles": enum,
"telecine": enum,
"framerateNumerator": integer,
"minIInterval": integer,
"adaptiveQuantization": enum,
"codecLevel": enum,
"sceneChangeDetect": enum,
"qualityTuningLevel": enum,
"framerateConversionAlgorithm": enum,
"unregisteredSeiTimecode": enum,
"gopSizeUnits": enum,
"adaptiveQuantization": enum,
"codecLevel": enum,
"sceneChangeDetect": enum,
"framerateNumerator": integer,
"framerateConversionAlgorithm": enum,
"gopSizeUnits": enum,
"numberBFramesBetweenReferenceFrames": integer,
"adaptiveQuantization": enum,
"codecLevel": enum,
"sceneChangeDetect": enum,
"framerateNumerator": integer,
"framerateConversionAlgorithm": enum,
"gopSizeUnits": enum,
"numberBFramesBetweenReferenceFrames": integer,
"adaptiveQuantization": enum,
"codecLevel": enum,
"sceneChangeDetect": enum,
"framerateNumerator": integer,
"adaptiveQuantization": enum,
"codecLevel": enum,
"sceneChangeDetect": enum,
"framerateNumerator": integer,
"adaptiveQuantization": enum,
"codecLevel": enum,
"sceneChangeDetect": enum,
"framerateNumerator": integer,
"framerateConversionAlgorithm": enum,
"gopSizeUnits": enum,
"numberBFramesBetweenReferenceFrames": integer,
"adaptiveQuantization": enum,
"codecLevel": enum,
"sceneChangeDetect": enum,
"framerateNumerator": integer,
"adaptiveQuantization": enum,
"codecLevel": enum,
"parControl": enum,
"afdsignaling": enum,
"droptimecode": enum,
"respondtoafds": enum,
"colormetadata": enum
},
"audiodescriptions": [
{
"audiotypecontrol": enum,
"audiosourcename": "string",
"audionormalizationsettings": {
"algorithm": enum,
"algorithmcontrol": enum,
"correctiongatelevel": integer,
"loudnesslogging": enum,
"targetlkfs": number,
"peakcalculation": enum
},
"coderesettings": {
"codec": enum,
"sacsettings": {
"audiodescriptionbroadcastermix": enum,
"vbrquality": enum,
"bitrate": integer,
"ratecontrolmode": enum,
"codecprofile": enum,
"codingmode": enum,
"rawformat": enum,
"samplingrate": integer,
"specification": enum
},
"ac3settings": {
"bitrate": integer,
"bitstreammode": enum,
"codingmode": enum,
"dialnorm": integer,
"dynamicrangecompressionprofile": enum,
"metadatacontrol": enum,
"lifefilter": enum,
"samplerate": integer
},
"aiffsettings": {
"bitdepth": integer,
"channels": integer,
"samplerate": integer
},
"eac3settings": {
"metadatacontrol": enum,
"surroundexmode": enum,
"lorosurroundmixlevel": number,
"phasecontrol": enum,
"dialnorm": integer,
"lrrtsurroundmixlevel": number,
"bitrate": integer,
"lrrtcentermixlevel": number,
"passthroughcontrol": enum,
"lifecode": enum,
"lorocentermixlevel": number,
"attenuationcontrol": enum,
"codingmode": enum,
"surroundmode": enum,
"bitstreammode": enum,
"lifefilter": enum,
"stereodownmix": enum,
"dynamicrangecompressionrf": enum,
"samplerate": integer,
"dynamicrangecompressionline": enum,
"dcFilter": enum
},
"mp2Settings": {
  "bitrate": integer,
  "channels": integer,
  "sampleRate": integer
},
"wavSettings": {
  "bitDepth": integer,
  "channels": integer,
  "sampleRate": integer,
  "format": enum
},
"remixSettings": {
  "channelMapping": {
    "outputChannels": [,
      {
        "inputChannels": [ integer
      ]
    ]
  },
  "channelsIn": integer,
  "channelsOut": integer
},
"streamName": "string",
"languageCodeControl": enum,
"audioType": integer,
"customLanguageCode": "string",
"languageCode": enum
],
"containerSettings": {
  "container": enum,
  "m3u8Settings": {
    "audioFramesPerPes": integer,
    "pcrControl": enum,
    "pcrPid": integer,
    "pmtPid": integer,
    "privateMetadataPid": integer,
    "programNumber": integer,
    "patInterval": integer,
    "pmtInterval": integer,
    "scte35Source": enum,
    "scte35Pid": integer,
    "nielsenId3": enum,
    "timedMetadata": enum,
    "timedMetadataPid": integer,
    "transportStreamId": integer,
    "videoPid": integer,
    "audioPids": [, integer
  ]
},
"f4vSettings": {
  "moovPlacement": enum
},
"m2tsSettings": {
  "audioBufferModel": enum,
  "minEbpInterval": integer,
  "esRateInPes": enum,
  "patInterval": integer,
  "dvbNitSettings": {
    "nitInterval": integer,
"networkId": integer,
"networkName": "string"
},
"dvbSdtSettings": {
    "outputSdt": enum,
    "sdtInterval": integer,
    "serviceName": "string",
    "serviceProviderName": "string"
},
"scte35Source": enum,
"scte35Pid": integer,
"scte35Esam": {
    "scte35EsamPid": integer
},
"videoPid": integer,
"dvbTdtSettings": {
    "tdtInterval": integer
},
"pmtInterval": integer,
"segmentationStyle": enum,
"segmentationTime": number,
"pmtPid": integer,
"bitrate": integer,
"audioPids": [ integer
],
"privateMetadataPid": integer,
"nielsenId3": enum,
"timedMetadataPid": integer,
"maxPcrInterval": integer,
"运输StreamId": integer,
"dvbSubPids": [ integer
],
"rateMode": enum,
"audioFramesPerPes": integer,
"pcrControl": enum,
"segmentationMarkers": enum,
"ebpAudioInterval": enum,
"forceTsVideoEbpOrder": enum,
"programNumber": integer,
"pcrPid": integer,
"bufferModel": enum,
"dvbTeletextPid": integer,
"fragmentTime": number,
"ebpPlacement": enum,
"nullPacketBitrate": number
},
"movSettings": {
    "clapAtom": enum,
    "cslgAtom": enum,
    "paddingControl": enum,
    "reference": enum,
    "mpeg2FourCCControl": enum
},
"mp4Settings": {
    "cslgAtom": enum,
    "freeSpaceBox": enum,
    "mp4MajorBrand": "string",
    "moovPlacement": enum
}
},
"captionDescriptions": [
    {"destinationSettings": {
        "destinationType": enum,
"burninDestinationSettings": {
  "backgroundOpacity": integer,
  "shadowXOffset": integer,
  "teletextSpacing": enum,
  "alignment": enum,
  "outlineSize": integer,
  "yPosition": integer,
  "shadowColor": enum,
  "fontOpacity": enum,
  "fontSize": integer,
  "fontScript": enum,
  "fontColor": enum,
  "backgroundColor": enum,
  "fontResolution": integer,
  "outlineColor": enum,
  "shadowYOffset": integer,
  "xPosition": integer,
  "shadowOpacity": integer
},
"dvbSubDestinationSettings": {
  "backgroundOpacity": integer,
  "shadowXOffset": integer,
  "teletextSpacing": enum,
  "alignment": enum,
  "outlineSize": integer,
  "yPosition": integer,
  "shadowColor": enum,
  "fontOpacity": enum,
  "fontSize": integer,
  "fontScript": enum,
  "fontColor": enum,
  "backgroundColor": enum,
  "fontResolution": integer,
  "outlineColor": enum,
  "shadowYOffset": integer,
  "xPosition": integer,
  "shadowOpacity": integer
},
"sccDestinationSettings": {
  "framerate": enum
},
"teletextDestinationSettings": {
  "pageNumber": "string"
},
"ttmlDestinationSettings": {
  "stylePassthrough": enum
},
"embeddedDestinationSettings": {
  "destination608ChannelNumber": integer
},
"customLanguageCode": "string",
"languageCode": enum,
"languageDescription": "string"
}
}

Example DELETE

{
  "name": "string"
}
Response Bodies

Example GetPresetResponse

```json
{
  "preset": {
    "arn": "string",
    "createdAt": "string",
    "lastUpdated": "string",
    "description": "string",
    "category": "string",
    "name": "string",
    "type": enum,
    "settings": {
      "videoDescription": {
        "fixedAfd": integer, 
        "width": integer, 
        "scalingBehavior": enum, 
        "crop": {
          "height": integer,
          "width": integer,
          "x": integer,
          "y": integer
        },
        "height": integer,
        "videoPreprocessors": {
          "colorCorrector": {
            "brightness": integer, 
            "colorSpaceConversion": enum, 
            "contrast": integer,
            "hue": integer,
            "saturation": integer,
            "hdr10Metadata": {
              "redPrimaryX": integer,
              "redPrimaryY": integer,
              "greenPrimaryX": integer,
              "greenPrimaryY": integer,
              "bluePrimaryX": integer,
              "bluePrimaryY": integer,
              "whitePointX": integer,
              "whitePointY": integer,
              "maxFrameAverageLightLevel": integer,
              "maxContentLightLevel": integer,
              "maxLuminance": integer,
              "minLuminance": integer
            }
          },
          "deinterlacer": {
            "algorithm": enum, 
            "mode": enum, 
            "control": enum
          },
          "imageInserter": {
            "insertableImages": [
              {
                "width": integer, 
                "height": integer, 
                "imageX": integer,
                "imageY": integer,
                "duration": integer,
                "fadeIn": integer,
                "layer": integer,
                "imageInserterInput": "string",
                "startTime": "string"
              }
```
"fadeOut": integer,
"opacity": integer
}
]
"noiseReducer": {
"filter": enum,
"filterSettings": {
"strength": integer
},
"spatialFilterSettings": {
"strength": integer,
"speed": integer,
"postFilterSharpenStrength": integer
}
"timecodeBurnin": {
"fontSize": integer,
"position": enum,
"prefix": "string"
}
"timecodeInsertion": enum,
"antiAlias": enum,
"position": {
"height": integer,
"width": integer,
"x": integer,
"y": integer
}
"sharpness": integer,
"codecSettings": {
"codec": enum,
"frameCaptureSettings": {
"framerateNumerator": integer,
"framerateDenominator": integer,
"maxCaptures": integer,
"quality": integer
}
"h264Settings": {
"interlaceMode": enum,
"parNumerator": integer,
"numberReferenceFrames": integer,
"syntax": enum,
"softness": integer,
"framerateDenominator": integer,
"gopClosedCadence": integer,
"hrdBufferInitialFillPercentage": integer,
"gopSize": number,
"slices": integer,
"gopBReference": enum,
"hrdBufferSize": integer,
"maxBitrate": integer,
"slowPal": enum,
"parDenominator": integer,
"spatialAdaptiveQuantization": enum,
"temporalAdaptiveQuantization": enum,
"flickerAdaptiveQuantization": enum,
"entropyEncoding": enum,
"bitrate": integer,
"framerateControl": enum,
"rateControlMode": enum,
"qvbrSettings": {
"qvbrQualityLevel": integer,
"maxAverageBitrate": integer
}
"codecProfile": enum,
"telecine": enum,
"framerateNumerator": integer,
"minIInterval": integer,
"adaptiveQuantization": enum,
"codecLevel": enum,
"fieldEncoding": enum,
"sceneChangeDetect": enum,
"qualityTuningLevel": enum,
"framerateConversionAlgorithm": enum,
"unregisteredSeiTimecode": enum,
"gopSizeUnits": enum,
"parControl": enum,
"numberBFramesBetweenReferenceFrames": integer,
"repeatPps": enum,
"dynamicSubGop": enum
},
"h265Settings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "numberReferenceFrames": integer,
  "framerateDenominator": integer,
  "gopClosedCadence": integer,
  "alternateTransferFunctionSei": enum,
  "hrdBufferInitialFillPercentage": integer,
  "gopSize": number,
  "slices": integer,
  "gopBReference": enum,
  "hrdBufferSize": integer,
  "maxBitrate": integer,
  "slowPal": enum,
  "parDenominator": integer,
  "spatialAdaptiveQuantization": enum,
  "flickerAdaptiveQuantization": enum,
  "bitrate": integer,
  "framerateControl": enum,
  "rateControlMode": enum,
  "qvbrSettings": {
    "qvbrQualityLevel": integer,
    "maxAverageBitrate": integer
  },
  "codecProfile": enum,
  "tiles": enum,
  "telecine": enum,
  "framerateNumerator": integer,
  "minIInterval": integer,
  "adaptiveQuantization": enum,
  "codecLevel": enum,
  "sceneChangeDetect": enum,
  "qualityTuningLevel": enum,
  "framerateConversionAlgorithm": enum,
  "unregisteredSeiTimecode": enum,
  "gopSizeUnits": enum,
  "parControl": enum,
  "numberBFramesBetweenReferenceFrames": integer,
  "temporalIds": enum,
  "sampleAdaptiveOffsetFilterMode": enum,
  "writeMp4PackagingType": enum,
  "dynamicSubGop": enum
},
"mpeg2Settings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "syntax": enum,
  "softness": integer,
"framerateDenominator": integer,
"gopClosedCadence": integer,
"hrdBufferInitialFillPercentage": integer,
"gopSize": number,
"hrdBufferSize": integer,
"maxBitrate": integer,
"slowPal": enum,
"parDenominator": integer,
"spatialAdaptiveQuantization": enum,
"temporalAdaptiveQuantization": enum,
"bitrate": integer,
"intraDcPrecision": integer,
"framerateControl": enum,
"rateControlMode": enum,
"codecProfile": enum,
"telecine": enum,
"framerateNumerator": integer,
"minIInterval": integer,
"adaptiveQuantization": enum,
"codecLevel": enum,
"sceneChangeDetect": enum,
"qualityTuningLevel": enum,
"framerateConversionAlgorithm": enum,
"gopSizeUnits": enum,
"numberBFramesBetweenReferenceFrames": integer,
"dynamicSubGop": enum
},
"proresSettings": {
"interlaceMode": enum,
"parNumerator": integer,
"framerateDenominator": integer,
"codecProfile": enum,
"slowPal": enum,
"parDenominator": integer,
"framerateControl": enum,
"telecine": enum,
"framerateNumerator": integer,
"framerateConversionAlgorithm": enum,
"parControl": enum
}
},
"afdSignaling": enum,
"dropFrameTimecode": enum,
"respondToAfd": enum,
"colorMetadata": enum
},
"audioDescriptions": [ 
{
"audioTypeControl": enum,
"audioSourceName": "string",
"audioNormalizationSettings": {
"algorithm": enum,
"algorithmControl": enum,
"correctionGateLevel": integer,
"loudnessLogging": enum,
"targetLkfs": number,
"peakCalculation": enum
},
"codecSettings": {
"codec": enum,
"aacSettings": {
"audioDescriptionBroadcasterMix": enum,
"vbrQuality": enum,
"bitrate": integer,
"rateControlMode": enum
}
"codecProfile": enum,
"codingMode": enum,
"rawFormat": enum,
"sampleRate": integer,
"specification": enum
},
"ac3Settings": {
"bitrate": integer,
"bitstreamMode": enum,
"codingMode": enum,
"dialnorm": integer,
"dynamicRangeCompressionProfile": enum,
"metadataControl": enum,
"lfeFilter": enum,
"sampleRate": integer
},
"aiffSettings": {
"bitDepth": integer,
"channels": integer,
"sampleRate": integer
},
"eac3Settings": {
"metadataControl": enum,
"surroundExMode": enum,
"loRoSurroundMixLevel": number,
"phaseControl": enum,
"dialnorm": integer,
"ltRtSurroundMixLevel": number,
"bitrate": integer,
"ltRtCenterMixLevel": number,
"passThroughControl": enum,
"lfeControl": enum,
"loRoCenterMixLevel": number,
"attenuationControl": enum,
"codingMode": enum,
"surroundMode": enum,
"bitstreamMode": enum,
"lfeFilter": enum,
"stereoDownmix": enum,
"dynamicRangeCompressionRf": enum,
"sampleRate": integer,
"dynamicRangeCompressionLine": enum,
"dcFilter": enum
},
"mp2Settings": {
"bitrate": integer,
"channels": integer,
"sampleRate": integer
},
"wavSettings": {
"bitDepth": integer,
"channels": integer,
"sampleRate": integer,
"format": enum
},
"remixSettings": {
"channelMapping": {
"outputChannels": [ 
{ 
"inputChannels": [ 
integer
] 
}
] 
},
"967"}
"channelsIn": integer,
"channelsOut": integer
},
"streamName": "string",
"languageCodeControl": enum,
"audioType": integer,
"customLanguageCode": "string",
"languageCode": enum
},
"containerSettings": {
"container": enum,
"m3u8Settings": {
"audioFramesPerPes": integer,
"pcrControl": enum,
"pcrPid": integer,
"pmtPid": integer,
"privateMetadataPid": integer,
"programNumber": integer,
"patInterval": integer,
"pmtInterval": integer,
"scte35Source": enum,
"scte35Pid": integer,
"nielsenId3": enum,
"timedMetadata": enum,
"timedMetadataPid": integer,
"transportStreamId": integer,
"videoPid": integer,
"audioPids": [ integer
}
},
"f4vSettings": {
"moovPlacement": enum
},
"m2tsSettings": {
"audioBufferModel": enum,
"minEbpInterval": integer,
"esRateInPes": enum,
"patInterval": integer,
"dvbNitSettings": {
"nitInterval": integer,
"networkId": integer,
"networkName": "string"
},
"dvbSdtSettings": {
"outputSdt": enum,
"sdtInterval": integer,
"serviceName": "string",
"serviceProviderName": "string"
},
"scte35Source": enum,
"scte35Pid": integer,
"scte35Esam": {
"scte35EsamPid": integer
},
"videoPid": integer,
"dvbTdtSettings": {
"tdtInterval": integer
},
"pmtInterval": integer,
"segmentationStyle": enum,
"segmentationTime": number,
"pmtPid": integer,
"bitrate": integer,
"audioPids": [
}
integer
],
"privateMetadataPid": integer,
"nielsenId3": enum,
"timedMetadataPid": integer,
"maxPcrInterval": integer,
"transportStreamId": integer,
"dvbSubPids": [ integer
],
"rateMode": enum,
"audioFramesPerPes": integer,
"pcrControl": enum,
"segmentationMarkers": enum,
"ebpAudioInterval": enum,
"forceTsVideoEbpOrder": enum,
"programNumber": integer,
"pcrPid": integer,
"bufferModel": enum,
"dvbTeletextPid": integer,
"fragmentTime": number,
"ebpPlacement": enum,
"nullPacketBitrate": number
},
"movSettings": {
"clapAtom": enum,
"cslgAtom": enum,
"paddingControl": enum,
"reference": enum,
"mpeg2FourCCControl": enum
},
"mp4Settings": {
"cslgAtom": enum,
"freeSpaceBox": enum,
"mp4MajorBrand": "string",
"moovPlacement": enum
}
},
"captionDescriptions": [
{
"destinationSettings": {
"destinationType": enum,
"burninDestinationSettings": {
"backgroundColor": enum,
"shadowColor": enum,
"fontColor": enum,
"backgroundColor": enum,
"fontResolution": integer,
"outlineColor": enum,
"shadowXOffset": integer,
"xAlignment": enum,
"alignment": enum,
"outlineSize": integer,
"xPosition": integer,
"yPosition": integer,
"shadowOpacity": integer,
"fontOpacity": integer,
"fontSize": integer,
"fontScript": enum,
"fontColor": enum,
"backgroundColor": enum,
"fontResolution": integer,
"outlineColor": enum,
"shadowXOffset": integer,
"xPosition": integer,
"shadowOpacity": integer
}
},
"dvbSubDestinationSettings": {
"backgroundColor": enum,
"shadowColor": enum,
"fontColor": enum,
"backgroundColor": enum,
"fontResolution": integer,
"outlineColor": enum,
"shadowXOffset": integer,
"xPosition": integer,
"shadowOpacity": integer
}
"outlineSize": integer,
"yPosition": integer,
"shadowColor": enum,
"fontOpacity": integer,
"fontSize": integer,
"fontScript": enum,
"fontColor": enum,
"backgroundColor": enum,
"fontResolution": integer,
"outlineColor": enum,
"shadowYOffset": integer,
"xPosition": integer,
"shadowOpacity": integer
},
"sccDestinationSettings": {
"framerate": enum
},
"teletextDestinationSettings": {
"pageNumber": "string"
},
"ttmlDestinationSettings": {
"stylePassthrough": enum
},
"embeddedDestinationSettings": {
"destination608ChannelNumber": integer
},
"customLanguageCode": "string",
"languageCode": enum,
"languageDescription": "string"
]
}
}

Example UpdatePresetResponse

{
"preset": {
"arn": "string",
"createdAt": "string",
"lastUpdated": "string",
"description": "string",
"category": "string",
"name": "string",
"type": enum,
"settings": {
"videoDescription": {
"fixedAfd": integer,
"width": integer,
"scalingBehavior": enum,
"crop": {
"height": integer,
"width": integer,
"x": integer,
"y": integer
},
"height": integer,
"videoPreprocessors": {
"colorCorrector": {
"brightness": integer,
"colorSpaceConversion": enum,
"contrast": integer,
""width": integer,
"hue": integer,
"saturation": integer,
"hdr10Metadata": {
    "redPrimaryX": integer,
    "redPrimaryY": integer,
    "greenPrimaryX": integer,
    "greenPrimaryY": integer,
    "bluePrimaryX": integer,
    "bluePrimaryY": integer,
    "whitePointX": integer,
    "whitePointY": integer,
    "maxFrameAverageLightLevel": integer,
    "maxContentLightLevel": integer,
    "maxLuminance": integer,
    "minLuminance": integer
},
"deinterlacer": {
    "algorithm": enum,
    "mode": enum,
    "control": enum
},
"imageInserter": {
    "insertableImages": [
        {
            "width": integer,
            "height": integer,
            "imageX": integer,
            "imageY": integer,
            "duration": integer,
            "fadeIn": integer,
            "layer": integer,
            "imageInserterInput": "string",
            "startTime": "string",
            "fadeOut": integer,
            "opacity": integer
        }
    ]
},
"noiseReducer": {
    "filter": enum,
    "filterSettings": {
        "strength": integer
    },
    "spatialFilterSettings": {
        "strength": integer,
        "speed": integer,
        "postFilterSharpenStrength": integer
    }
},
"timecodeBurnin": {
    "fontSize": integer,
    "position": enum,
    "prefix": "string"
},
"timecodeInsertion": enum,
"antiAlias": enum,
"position": {
    "height": integer,
    "width": integer,
    "x": integer,
    "y": integer
},
"sharpness": integer,
"codecSettings": {
"codec": enum,
"frameCaptureSettings": {
  "framerateNumerator": integer,
  "framerateDenominator": integer,
  "maxCaptures": integer,
  "quality": integer
},
"h264Settings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "numberReferenceFrames": integer,
  "syntax": enum,
  "softness": integer,
  "framerateDenominator": integer,
  "gopClosedCadence": integer,
  "hrdBufferInitialFillPercentage": integer,
  "gopSize": number,
  "slices": integer,
  "gopBReference": enum,
  "hrdBufferSize": integer,
  "maxBitrate": integer,
  "slowPal": enum,
  "parDenominator": integer,
  "spatialAdaptiveQuantization": enum,
  "temporalAdaptiveQuantization": enum,
  "flickerAdaptiveQuantization": enum,
  "entropyEncoding": enum,
  "bitrate": integer,
  "framerateControl": enum,
  "rateControlMode": enum,
  "qvbrSettings": {
    "qvbrQualityLevel": integer,
    "maxAverageBitrate": integer
  },
  "codecProfile": enum,
  "telecine": enum,
  "framerateNumerator": integer,
  "minIInterval": integer,
  "adaptiveQuantization": enum,
  "codecLevel": enum,
  "fieldEncoding": enum,
  "sceneChangeDetect": enum,
  "qualityTuningLevel": enum,
  "framerateConversionAlgorithm": enum,
  "unregisteredSeiTimecode": enum,
  "gopSizeUnits": enum,
  "parControl": enum,
  "numberBFramesBetweenReferenceFrames": integer,
  "repeatPps": enum,
  "dynamicSubGop": enum
},
"h265Settings": {
  "interlaceMode": enum,
  "parNumerator": integer,
  "numberReferenceFrames": integer,
  "framerateDenominator": integer,
  "gopClosedCadence": integer,
  "alternateTransferFunctionSei": enum,
  "hrdBufferInitialFillPercentage": integer,
  "gopSize": number,
  "slices": integer,
  "gopBReference": enum,
  "hrdBufferSize": integer,
  "maxBitrate": integer,
  "slowPal": enum,
  "parDenominator": integer
}
"spatialAdaptiveQuantization": enum,
"temporalAdaptiveQuantization": enum,
"flickerAdaptiveQuantization": enum,
"bitrate": integer,
"framerateControl": enum,
"rateControlMode": enum,
"qvbrSettings": {
  "qvbrQualityLevel": integer,
  "maxAverageBitrate": integer
},
"codecProfile": enum,
"tiles": enum,
"telecine": enum,
"framerateNumerator": integer,
"minIInterval": integer,
"adaptiveQuantization": enum,
"codecLevel": enum,
"sceneChangeDetect": enum,
"qualityTuningLevel": enum,
"framerateConversionAlgorithm": enum,
"unregisteredSeiTimecode": enum,
"gopsSizeUnits": enum,
"parControl": enum,
"numberBFramesBetweenReferenceFrames": integer,
"temporalIds": enum,
"sampleAdaptiveOffsetFilterMode": enum,
"writeMp4PackagingType": enum,
"dynamicSubGop": enum
},
"mpeg2Settings": {
"interlaceMode": enum,
"parNumerator": integer,
"syntax": enum,
"softness": integer,
"framerateDenominator": integer,
"gopClosedCadence": integer,
"hrdBufferInitialFillPercentage": integer,
"gopsSize": number,
"hrdBufferSize": integer,
"maxBitrate": integer,
"slowPal": enum,
"parDenominator": integer,
"spatialAdaptiveQuantization": enum,
"temporalAdaptiveQuantization": enum,
"bitrate": integer,
"intraDcPrecision": enum,
"framerateControl": enum,
"rateControlMode": enum,
"codecProfile": enum,
"telecine": enum,
"framerateNumerator": integer,
"minIInterval": integer,
"adaptiveQuantization": enum,
"codecLevel": enum,
"sceneChangeDetect": enum,
"qualityTuningLevel": enum,
"framerateConversionAlgorithm": enum,
"gopsSizeUnits": enum,
"parControl": enum,
"numberBFramesBetweenReferenceFrames": integer,
"dynamicSubGop": enum
},
"proresSettings": {
"interlaceMode": enum,
"parNumerator": integer,
"framerateDenominator": integer,
"codecProfile": enum,
"slowPal": enum,
"parDenominator": integer,
"framerateControl": enum,
"telecine": enum,
"framerateNumerator": integer,
"framerateConversionAlgorithm": enum,
"parControl": enum
},
"afdSignaling": enum,
"dropFrameTimecode": enum,
"respondToAfd": enum,
"colorMetadata": enum
},
"audioDescriptions": [
{
"audioTypeControl": enum,
"audioSourceName": "string",
"audioNormalizationSettings": {
"algorithm": enum,
"algorithmControl": enum,
"correctionGateLevel": integer,
"loudnessLogging": enum,
"targetLkfs": number,
"peakCalculation": enum
},
"codecSettings": {
"codec": enum,
"aacSettings": {
"audioDescriptionBroadcasterMix": enum,
"vbrQuality": enum,
"bitrate": integer,
"rateControlMode": enum,
"codecProfile": enum,
"codingMode": enum,
"rawFormat": enum,
"sampleRate": integer,
"specification": enum
},
"ac3Settings": {
"bitrate": integer,
"bitstreamMode": enum,
"codingMode": enum,
"dialnorm": integer,
"dynamicRangeCompressionProfile": enum,
"metadataControl": enum,
"lfeFilter": enum,
"sampleRate": integer
},
"aiffSettings": {
"bitDepth": integer,
"channels": integer,
"sampleRate": integer
},
"eac3Settings": {
"metadataControl": enum,
"surroundExMode": enum,
"loRoSurroundMixLevel": number,
"phaseControl": enum,
"dialnorm": integer,
"ltRtSurroundMixLevel": number,
"bitrate": integer,
"ltRtCenterMixLevel": number,
"passthroughControl": enum,
"lfeControl": enum,
"sampleRate": integer
}
]
"loRoCenterMixLevel": number,
"attenuationControl": enum,
"codingMode": enum,
"surroundMode": enum,
"bitstreamMode": enum,
"lfeFilter": enum,
"stereoDownmix": enum,
"dynamicRangeCompressionRf": enum,
"sampleRate": integer,
"dynamicRangeCompressionLine": enum,
"dcFilter": enum
},
"mp2Settings": {
"bitrate": integer,
"channels": integer,
"sampleRate": integer
},
"wavSettings": {
"bitDepth": integer,
"channels": integer,
"sampleRate": integer,
"format": enum
}
},
"remixSettings": {
"channelMapping": {
"outputChannels": [
{ "inputChannels": [ integer ]
}
],
"channelsIn": integer,
"channelsOut": integer
},
"streamName": "string",
"languageCodeControl": enum,
"audioType": integer,
"customLanguageCode": "string",
"languageCode": enum
},
"containerSettings": {
"container": enum,
"m3u8Settings": {
"audioFramesPerPes": integer,
"pcrControl": enum,
"pcrPid": integer,
"pmtPid": integer,
"privateMetadataPid": integer,
"programNumber": integer,
"patInterval": integer,
"pmtInterval": integer,
"scte35Source": enum,
"scte35Pid": integer,
"nielsenId3": enum,
"timedMetadata": enum,
"timedMetadataPid": integer,
"transportStreamId": integer,
"videoPid": integer,
"audioPids": [ integer ]
}
},
"975"
"f4vSettings": {
    "moovPlacement": enum
},
"m2tsSettings": {
    "audioBufferModel": enum,
    "minEbpInterval": integer,
    "esRateInPes": enum,
    "patInterval": integer,
    "dvbNitSettings": {
        "nitInterval": integer,
        "networkId": integer,
        "networkName": "string"
    },
    "dvbSdtSettings": {
        "outputSdt": enum,
        "sdtInterval": integer,
        "serviceName": "string",
        "serviceProviderName": "string"
    },
    "scte35Source": enum,
    "scte35Pid": integer,
    "scte35Esam": {
        "scte35EsamPid": integer
    },
    "videoPid": integer,
    "dvbTdtSettings": {
        "tdtInterval": integer
    },
    "pmtInterval": integer,
    "segmentationStyle": enum,
    "segmentationTime": number,
    "pmtPid": integer,
    "bitrate": integer,
    "audioPids": [ integer
    ],
    "privateMetadataPid": integer,
    "nielsenId": enum,
    "timedMetadataPid": integer,
    "maxPcrInterval": integer,
    "transportStreamId": integer,
    "dvbSubPids": [ integer
    ],
    "rateMode": enum,
    "audioFramesPerPes": integer,
    "pcrControl": enum,
    "segmentationMarkers": enum,
    "ebpAudioInterval": enum,
    "forcePsVideoEbpOrder": enum,
    "programNumber": integer,
    "pcrPid": integer,
    "bufferModel": enum,
    "dvbTeletextPid": integer,
    "fragmentTime": number,
    "ebpPlacement": enum,
    "nullPacketBitrate": number
},
"movSettings": {
    "clapAtom": enum,
    "cslgAtom": enum,
    "paddingControl": enum,
    "reference": enum,
    "mpeg2FourCCControl": enum
},
"mp4Settings": {
"cslgAtom": enum,
"freeSpaceBox": enum,
"mp4MajorBrand": "string",
"moovPlacement": enum
},
"captionDescriptions": [
{
"destinationSettings": {
"destinationType": enum,
"burninDestinationSettings": {
"backgroundOpacity": integer,
"shadowXOffset": integer,
"teletextSpacing": enum,
"alignment": enum,
"outlineSize": integer,
"yPosition": integer,
"shadowColor": enum,
"fontOpacity": integer,
"fontSize": integer,
"fontScript": enum,
"fontColor": enum,
"backgroundColor": enum,
"fontResolution": integer,
"outlineColor": enum,
"shadowYOffset": integer,
"xPosition": integer,
"shadowOpacity": integer
},
"dvbSubDestinationSettings": {
"backgroundOpacity": integer,
"shadowXOffset": integer,
"teletextSpacing": enum,
"alignment": enum,
"outlineSize": integer,
"yPosition": integer,
"shadowColor": enum,
"fontOpacity": integer,
"fontSize": integer,
"fontScript": enum,
"fontColor": enum,
"backgroundColor": enum,
"fontResolution": integer,
"outlineColor": enum,
"shadowYOffset": integer,
"xPosition": integer,
"shadowOpacity": integer
},
"sccDestinationSettings": {
"framerate": enum
},
"teletextDestinationSettings": {
"pageNumber": "string"
},
"ttmlDestinationSettings": {
"stylePassthrough": enum
},
"embeddedDestinationSettings": {
"destination608ChannelNumber": integer
},
"customLanguageCode": "string",
"languageCode": enum,
"languageDescription": "string"}
Example DeletePresetResponse

```json
{
}
```

Example ExceptionBody

```json
{
  "message": "string"
}
```

Properties

**AacAudioDescriptionBroadcasterMix**

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**

AAC Profile.

- LC
- HEV1
- HEV2

**AacCodingMode**

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**

Rate Control Mode.

- CBR
- VBR

**AacRawFormat**

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**

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value AAC. The service accepts one of two mutually exclusive groups of AAC settings--VBR and CBR. To select one of these modes, set the value of Bitrate control mode (rateControlMode) to "VBR" or "CBR". In VBR mode, you control the audio quality with the setting VBR quality (vbrQuality). In CBR mode, you use the setting Bitrate (bitrate). Defaults and valid values depend on the rate control mode.

**audioDescriptionBroadcasterMix**

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.

- **Type:** AacAudioDescriptionBroadcasterMix (p. 978)
- **Required:** False

**vbrQuality**

VBR Quality Level - Only used if rate_control_mode is VBR.

- **Type:** AacVbrQuality (p. 981)
- **Required:** False

**bitrate**

Average bitrate in bits/second. The set of valid values for this setting is: 6000, 8000, 10000, 12000, 14000, 16000, 20000, 24000, 28000, 32000, 40000, 48000, 56000, 64000, 80000, 96000, 112000, 128000, 160000, 192000, 224000, 256000, 288000, 320000, 384000, 448000, 512000, 576000, 640000, 768000, 896000, 1024000. The value you set is also constrained by the values you choose for Profile (codecProfile), Bitrate control mode (codingMode), and Sample rate (sampleRate). Default values depend on Bitrate control mode and Profile.

- **Type:** integer
- **Required:** False
Properties

Minimum: 6000
Maximum: 1024000

rateControlMode
Rate Control Mode.

  Type: AacRateControlMode (p. 979)
  Required: False

codecProfile
AAC Profile.

  Type: AacCodecProfile (p. 978)
  Required: False

codingMode
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.

  Type: AacCodingMode (p. 978)
  Required: False

rawFormat
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.

  Type: AacRawFormat (p. 979)
  Required: False

sampleRate
Sample rate in Hz. Valid values depend on rate control mode and profile.

  Type: integer
  Required: False
  Minimum: 8000
  Maximum: 96000

specification
Use MPEG-2 AAC instead of MPEG-4 AAC audio for raw or MPEG-2 Transport Stream containers.

  Type: AacSpecification (p. 980)
  Required: False

AacSpecification
Use MPEG-2 AAC instead of MPEG-4 AAC audio for raw or MPEG-2 Transport Stream containers.
MPEG2
MPEG4

**AacVbrQuality**

VBR Quality Level - Only used if rate_control_mode is VBR.

- LOW
- MEDIUM_LOW
- MEDIUM_HIGH
- HIGH

**Ac3BitstreamMode**

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**

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**

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**

Applies a 120Hz lowpass filter to the LFE channel prior to encoding. Only valid with 3_2_LFE coding mode.

- ENABLED
- DISABLED
**Ac3MetadataControl**

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**

Required when you set (Codec) under (AudioDescriptions)->(CodecSettings) to the value AC3.

**bitrate**

Average bitrate in bits/second. Valid bitrates depend on the coding mode.

- **Type**: integer
  - **Required**: False
  - **Minimum**: 64000
  - **Maximum**: 640000

**bitstreamMode**

Specifies the "Bitstream Mode" (bsmod) for the emitted AC-3 stream. See ATSC A/52-2012 for background on these values.

- **Type**: Ac3BitstreamMode (p. 981)
  - **Required**: False

**codingMode**

Dolby Digital coding mode. Determines number of channels.

- **Type**: Ac3CodingMode (p. 981)
  - **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

**dynamicRangeCompressionProfile**

If set to FILM_STANDARD, adds dynamic range compression signaling to the output bitstream as defined in the Dolby Digital specification.

- **Type**: Ac3DynamicRangeCompressionProfile (p. 981)
  - **Required**: False
metadataControl

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.

  Type: Ac3MetadataControl (p. 982)
  Required: False

lfeFilter

Applies a 120Hz lowpass filter to the LFE channel prior to encoding. Only valid with 3_2_LFE coding mode.

  Type: Ac3LfeFilter (p. 981)
  Required: False

sampleRate

Sample rate in hz. Sample rate is always 48000.

  Type: integer
  Required: False
  Minimum: 48000
  Maximum: 48000

AfdSignaling

This setting only applies to H.264, H.265, and MPEG2 outputs. Use Insert AFD signaling (AfdSignaling) to specify whether the service includes 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

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value AIFF.

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

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.
sampleRate
Sample rate in hz.

AntiAlias
The anti-alias filter is automatically applied to all outputs. The service no longer accepts the value DISABLED for AntiAlias. If you specify that in your job, the service will ignore the setting.

DISABLED
ENABLED

AudioCodec
Type of Audio codec.

AAC
MP2
WAV
AIFF
AC3
EAC3
PASSTHROUGH

AudioCodecSettings
Audio codec settings (CodecSettings) under (AudioDescriptions) contains the group of settings related to audio encoding. The settings in this group vary depending on the value you choose for Audio codec (Codec). For each codec enum you choose, define the corresponding settings object. The following lists the codec enum, settings object pairs. * AAC, AacSettings * MP2, Mp2Settings * WAV, WavSettings * AIFF, AiffSettings * AC3, Ac3Settings * EAC3, Eac3Settings

codec
Type of Audio codec.

Type: AudioCodec (p. 984)
Required: False

aacSettings
Required when you set (Codec) under (AudioDescriptions)->(CodecSettings) to the value AAC. The service accepts one of two mutually exclusive groups of AAC settings--VBR and CBR. To select one of these
modes, set the value of Bitrate control mode (rateControlMode) to "VBR" or "CBR". In VBR mode, you
c Control the audio quality with the setting VBR quality (vbrQuality). In CBR mode, you use the setting
Bitrate (bitrate). Defaults and valid values depend on the rate control mode.

Type: AacSettings (p. 979)
Required: False

ac3Settings
Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value AC3.

Type: Ac3Settings (p. 982)
Required: False

aiffSettings
Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value AIFF.

Type: AiffSettings (p. 983)
Required: False

eac3Settings
Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value EAC3.

Type: Eac3Settings (p. 1009)
Required: False

mp2Settings
Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value MP2.

Type: Mp2Settings (p. 1060)
Required: False

wavSettings
Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value WAV.

Type: WavSettings (p. 1088)
Required: False

AudioDescription
Description of audio output

audioTypeControl
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.

Type: AudioTypeControl (p. 989)
Properties

**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

**audioNormalizationSettings**

Advanced audio normalization settings.

Type: AudioNormalizationSettings (p. 988)
Required: False

**codecSettings**

Audio codec settings (CodecSettings) under (AudioDescriptions) contains the group of settings related to audio encoding. The settings in this group vary depending on the value you choose for Audio codec (Codec). For each codec enum you choose, define the corresponding settings object. The following lists the codec enum, settings object pairs. * AAC, AacSettings * MP2, Mp2Settings * WAV, WavSettings * AIFF, AiffSettings * AC3, Ac3Settings * EAC3, Eac3Settings

Type: AudioCodecSettings (p. 984)
Required: False

**remixSettings**

Advanced audio remixing settings.

Type: RemixSettings (p. 1078)
Required: False

**streamName**

Used for MS Smooth 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]*$

**languageCodeControl**

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.

- **Type:** AudioLanguageCodeControl (p. 987)
- **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

### customLanguageCode

Specify the language for this audio output track, using the ISO 639-2 or ISO 639-3 three-letter language code. The language specified 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:** string
- **Required:** False
- **Pattern:** ^[A-Za-z]{3}$
- **MinLength:** 3
- **MaxLength:** 3

### 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. 1043)
- **Required:** False

### AudioLanguageCodeControl

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

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
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
If set to LOG, log each output's audio track loudness to a CSV file.

  LOG
  DONT_LOG

AudioNormalizationPeakCalculation
If set to TRUE_PEAK, calculate and log the TruePeak for each output's audio track loudness.

  TRUE_PEAK
  NONE

AudioNormalizationSettings
Advanced audio normalization settings.

algorithm
Audio normalization algorithm to use. 1770-1 conforms to the CALM Act specification, 1770-2 conforms to the EBU R-128 specification.

  Type: AudioNormalizationAlgorithm (p. 987)
  Required: False

algorithmControl
When enabled the output audio is corrected using the chosen algorithm. If disabled, the audio will be measured but not adjusted.

  Type: AudioNormalizationAlgorithmControl (p. 988)
  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
Properties

loudnessLogging

If set to LOG, log each output's audio track loudness to a CSV file.

Type: AudioNormalizationLoudnessLogging (p. 988)
Required: False

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

peakCalculation

If set to TRUE_PEAK, calculate and log the TruePeak for each output's audio track loudness.

Type: AudioNormalizationPeakCalculation (p. 988)
Required: False

AudioTypeControl

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

Burn-In Destination Settings.

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

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.
Properties

**Type**: integer  
**Required**: False  
**Minimum**: -2147483648  
**Maximum**: 2147483647

**teletextSpacing**

Only applies to jobs with input captions in Teletext or STL formats. Specify whether the spacing between letters in your captions is set by the captions grid or varies depending on letter width. Choose fixed grid to conform to the spacing specified in the captions file more accurately. Choose proportional to make the text easier to read if the captions are closed caption.

**Type**: BurninSubtitleTeletextSpacing (p. 993)  
**Required**: False

**alignment**

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.

**Type**: BurninSubtitleAlignment (p. 992)  
**Required**: False

**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**: False  
**Minimum**: 0  
**Maximum**: 10

**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

**shadowColor**

 Specifies the color of the shadow cast by the captions. All burn-in and DVB-Sub font settings must match.
Properties

**Type**: BurninSubtitleShadowColor (p. 993)
**Required**: False

**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**: False
**Minimum**: 0
**Maximum**: 255

**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

**fontScript**

Provide the font script, using an ISO 15924 script code, if the LanguageCode is not sufficient for determining the script type. Where LanguageCode or CustomLanguageCode is sufficient, use "AUTOMATIC" or leave unset. This is used to help determine the appropriate font for rendering burn-in captions.

**Type**: FontScript (p. 1013)
**Required**: False

**fontColor**

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.

**Type**: BurninSubtitleFontColor (p. 993)
**Required**: False

**backgroundColor**

Specifies the color of the rectangle behind the captions. All burn-in and DVB-Sub font settings must match.

**Type**: BurninSubtitleBackgroundColor (p. 993)
**Required**: False

**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

**outlineColor**

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.

**Type**: `BurninSubtitleOutlineColor` (p. 993)  
**Required**: False

**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

**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

**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

**BurninSubtitleAlignment**

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.
Properties

**BurninSubtitleBackgroundColor**

Specifies the color of the rectangle behind the captions. All burn-in and DVB-Sub font settings must match.

- NONE
- BLACK
- WHITE

**BurninSubtitleFontColor**

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**

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**

Specifies the color of the shadow cast by the captions. All burn-in and DVB-Sub font settings must match.

- NONE
- BLACK
- WHITE

**BurninSubtitleTeletextSpacing**

Only applies to jobs with input captions in Teletext or STL formats. Specify whether the spacing between letters in your captions is set by the captions grid or varies depending on letter width. Choose fixed grid
to conform to the spacing specified in the captions file more accurately. Choose proportional to make the text easier to read if the captions are closed caption.

FIXED_GRID
PROPORTIONAL

**CaptionDescriptionPreset**

Caption Description for preset

**destinationSettings**

Specific settings required by destination type. Note that burnin_destination_settings are not available if the source of the caption data is Embedded or Teletext.

- **Type**: CaptionDestinationSettings (p. 994)
- **Required**: False

**customLanguageCode**

Indicates the language of the caption output track, using the ISO 639-2 or ISO 639-3 three-letter language code. For most captions output formats, the encoder puts this language information in the output captions metadata. If your output captions format is DVB-Sub or Burn in, the encoder uses this language information to choose the font language for rendering the captions text.

- **Type**: string
- **Required**: False
- **Pattern**: ^[A-Za-z]{3}$
- **MinLength**: 3
- **MaxLength**: 3

**languageCode**

Specify the language of this captions output track. For most captions output formats, the encoder puts this language information in the output captions metadata. If your output captions format is DVB-Sub or Burn in, the encoder uses this language information to choose the font language for rendering the captions text.

- **Type**: LanguageCode (p. 1043)
- **Required**: False

**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

**CaptionDestinationSettings**

Specific settings required by destination type. Note that burnin_destination_settings are not available if the source of the caption data is Embedded or Teletext.
destinationType

Specify the format for this set of captions on this output. The default format is embedded without SCTE-20. Other options are embedded with SCTE-20, burn-in, DVB-sub, SCC, SRT, teletext, TTML, and web-VTT. If you are using SCTE-20, choose SCTE-20 plus embedded (SCTE20_PLUS_EMBEDDED) to create an output that complies with the SCTE-43 spec. To create a non-compliant output where the embedded captions come first, choose Embedded plus SCTE-20 (EMBEDDED_PLUS_SCTE20).

Type: CaptionDestinationType (p. 996)
Required: False

burninDestinationSettings

Burn-In Destination Settings.

Type: BurninDestinationSettings (p. 989)
Required: False

dvbSubDestinationSettings

DVB-Sub Destination Settings

Type: DvbSubDestinationSettings (p. 1002)
Required: False

c SCCDestinationSettings

Settings for SCC caption output.

Type: SccDestinationSettings (p. 1080)
Required: False

teletextDestinationSettings

Settings for Teletext caption output

Type: TeletextDestinationSettings (p. 1080)
Required: False

ttmlDestinationSettings

Settings specific to TTML caption outputs, including Pass style information (TtmlStylePassthrough).

Type: TtmlDestinationSettings (p. 1081)
Required: False

embeddedDestinationSettings

Settings specific to embedded/ancillary caption outputs, including 608/708 Channel destination number.

Type: EmbeddedDestinationSettings (p. 1012)
Required: False
**CaptionDestinationType**

Specify the format for this set of captions on this output. The default format is embedded without SCTE-20. Other options are embedded with SCTE-20, burn-in, DVB-sub, SCC, SRT, teletext, TTML, and web-VTT. If you are using SCTE-20, choose SCTE-20 plus embedded (SCTE20_PLUS_EMBEDDED) to create an output that complies with the SCTE-43 spec. To create a non-compliant output where the embedded captions come first, choose Embedded plus SCTE-20 (EMBEDDED_PLUS_SCTE20).

- BURN_IN
- DVB_SUB
- EMBEDDED
- EMBEDDED_PLUS_SCTE20
- SCTE20_PLUS_EMBEDDED
- SCC
- SRT
- SMI
- TELETEXT
- TTML
- WEBVTT

**ChannelMapping**

Channel mapping (ChannelMapping) contains the group of fields that hold the remixing value for each channel. Units are in dB. Acceptable values are within the range from -60 (mute) through 6. A setting of 0 passes the input channel unchanged to the output channel (no attenuation or amplification).

**outputChannels**

List of output channels

- **Type**: Array of type OutputChannelMapping (p. 1072)
- **Required**: False

**ColorCorrector**

Settings for color correction.

**brightness**

Brightness level.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 100

**colorSpaceConversion**

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.

- **Type**: ColorSpaceConversion (p. 997)
**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

**saturation**

Saturation level.

**Type:** integer  
**Required:** False  
**Minimum:** 1  
**Maximum:** 100

**hdr10Metadata**

Use the HDR master display (Hdr10Metadata) settings to correct HDR metadata or to provide missing metadata. Note that these settings are not color correction.

**Type:** Hdr10Metadata (p. 1038)  
**Required:** False

**ColorMetadata**

Enable insert color metadata (ColorMetadata) to include color metadata in this output. This setting is enabled by default.

- IGNORE  
- INSERT

**ColorSpaceConversion**

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 colorspace. 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 specific settings.

**container**

Container for this output. Some containers require a container settings object. If not specified, the default object will be created.

*Type: ContainerType (p. 999)*

*Required: False*

**m3u8Settings**

Settings for TS segments in HLS

*Type: M3u8Settings (p. 1056)*

*Required: False*

**f4vSettings**

Settings for F4v container

*Type: F4vSettings (p. 1013)*

*Required: False*

**m2tsSettings**

MPEG-2 TS container settings. These apply to outputs in a File output group when the output's container (ContainerType) is MPEG-2 Transport Stream (M2TS). In these assets, data is organized by the program map table (PMT). Each transport stream program contains subsets of data, including audio, video, and metadata. Each of these subsets of data has a numerical label called a packet identifier (PID). Each transport stream program corresponds to one MediaConvert output. The PMT lists the types of data in a program along with their PID. Downstream systems and players use the program map table to look up the PID for each type of data it accesses and then uses the PIDs to locate specific data within the asset.

*Type: M2tsSettings (p. 1049)*

*Required: False*

**movSettings**

Settings for MOV Container.

*Type: MovSettings (p. 1059)*

*Required: False*

**mp4Settings**

Settings for MP4 Container
Type: Mp4Settings (p. 1061)  
Required: False

**ContainerType**

Container for this output. Some containers require a container settings object. If not specified, the default object will be created.

- F4V
- ISMV
- M2TS
- M3U8
- CMFC
- MOV
- MP4
- MPD
- MXF
- RAW

**DeinterlaceAlgorithm**

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**

Settings for deinterlacer

**algorithm**

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.

Type: DeinterlaceAlgorithm (p. 999)  
Required: False

**mode**

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.

Type: DeinterlacerMode (p. 1000)  
Required: False
**control**

- 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.

  **Type:** DeinterlacerControl (p. 1000)
  **Required:** False

**DeinterlacerControl**

- 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**

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**

**DeletePresetRequest**

Delete a preset by sending a request with the preset name

**name**

The name of the preset to be deleted.

  **Type:** string
  **Required:** False

**DeletePresetResponse**

Delete preset requests will return an OK message or error message with an empty body.

**DropFrameTimecode**

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**

Inserts DVB Network Information Table (NIT) at the specified table repetition interval.

**nitInterval**

The number of milliseconds between instances of this table in the output transport stream.

Type: integer  
Required: False  
Minimum: 25  
Maximum: 10000

**networkId**

The numeric value placed in the Network Information Table (NIT).

Type: integer  
Required: False  
Minimum: 0  
Maximum: 65535

**networkName**

The network name text placed in the network_name_descriptor inside the Network Information Table. Maximum length is 256 characters.

Type: string  
Required: False  
MinLength: 1  
MaxLength: 256

**DvbSdtSettings**

Inserts DVB Service Description Table (NIT) at the specified table repetition interval.

**outputSdt**

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.

Type: OutputSdt (p. 1072)  
Required: False

**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
**MinLength:** 1
**MaxLength:** 256

#### serviceProviderName

The service provider name placed in the service_descriptor in the Service Description Table. Maximum length is 256 characters.

**Type:** string
**Required:** False
**MinLength:** 1
**MaxLength:** 256

---

### DvbSubDestinationSettings

DVB-Sub Destination Settings

#### 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

#### 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

#### teletextSpacing

Only applies to jobs with input captions in Teletext or STL formats. Specify whether the spacing between letters in your captions is set by the captions grid or varies depending on letter width. Choose fixed grid to conform to the spacing specified in the captions file more accurately. Choose proportional to make the text easier to read if the captions are closed caption.
**alignment**

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.

**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.

**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.

**shadowColor**

Specifies the color of the shadow cast by the captions. All burn-in and DVB-Sub font settings must match.

**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.
Maximum: 255

**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

**fontScript**

Provide the font script, using an ISO 15924 script code, if the LanguageCode is not sufficient for determining the script type. Where LanguageCode or CustomLanguageCode is sufficient, use "AUTOMATIC" or leave unset. This is used to help determine the appropriate font for rendering DVB-Sub captions.

- **Type:** FontScript (p. 1013)
- **Required:** False

**fontColor**

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.

- **Type:** DvbSubtitleFontColor (p. 1006)
- **Required:** False

**backgroundColor**

Specifies the color of the rectangle behind the captions. All burn-in and DVB-Sub font settings must match.

- **Type:** DvbSubtitleBackgroundColor (p. 1005)
- **Required:** False

**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

**outlineColor**

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.
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

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

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

DvbSubtitleAlignment

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

Specifies the color of the rectangle behind the captions. All burn-in and DVB-Sub font settings must match.

NONE
DvbSubtitleFontColor

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

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

Specifies the color of the shadow cast by the captions. All burn-in and DVB-Sub font settings must match.

NONE
BLACK
WHITE

DvbSubtitleTeletextSpacing

Only applies to jobs with input captions in Teletext or STL formats. Specify whether the spacing between letters in your captions is set by the captions grid or varies depending on letter width. Choose fixed grid to conform to the spacing specified in the captions file more accurately. Choose proportional to make the text easier to read if the captions are closed caption.

FIXED_GRID
PROPORTIONAL

DvbTdtSettings

Inserts DVB Time and Date Table (TDT) at the specified table repetition interval.
### tdtInterval

The number of milliseconds between instances of this table in the output transport stream.

**Type:** integer  
**Required:** False  
**Minimum:** 1000  
**Maximum:** 30000

### Eac3AttenuationControl

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

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

Dolby Digital Plus coding mode. Determines number of channels.

- CODING_MODE_1_0  
- CODING_MODE_2_0  
- CODING_MODE_3_2

### Eac3DcFilter

Activates a DC highpass filter for all input channels.

- ENABLED  
- DISABLED

### Eac3DynamicRangeCompressionLine

Enables Dynamic Range Compression that restricts the absolute peak level for a signal.

- NONE  
- FILM_STANDARD  
- FILM_LIGHT  
- MUSIC_STANDARD  
- MUSIC_LIGHT  
- SPEECH
Eac3DynamicRangeCompressionRf

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

When encoding 3/2 audio, controls whether the LFE channel is enabled

- LFE
- NO_LFE

Eac3LfeFilter

Applies a 120Hz lowpass filter to the LFE channel prior to encoding. Only valid with 3_2_LFE coding mode.

- ENABLED
- DISABLED

Eac3MetadataControl

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

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

Controls the amount of phase-shift applied to the surround channels. Only used for 3/2 coding mode.

- SHIFT_90_DEGREES
- NO_SHIFT
**Eac3Settings**

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value EAC3.

**metadataControl**

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.

*Type: Eac3MetadataControl (p. 1008)*  
*Required: False*

**surroundExMode**

When encoding 3/2 audio, sets whether an extra center back surround channel is matrix encoded into the left and right surround channels.

*Type: Eac3SurroundExMode (p. 1012)*  
*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*

**phaseControl**

Controls the amount of phase-shift applied to the surround channels. Only used for 3/2 coding mode.

*Type: Eac3PhaseControl (p. 1008)*  
*Required: False*

**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*

**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*
### Properties

**Format**: float
- **Minimum**: -60.0
- **Maximum**: -1.5

**bitrate**
Average bitrate in bits/second. Valid bitrates depend on the coding mode.
- **Type**: integer
- **Required**: False
- **Minimum**: 64000
- **Maximum**: 640000

**leftRtCenterMixLevel**
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

**passthroughControl**
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.
- **Type**: Eac3PassthroughControl (p. 1008)
- **Required**: False

**lfeControl**
When encoding 3/2 audio, controls whether the LFE channel is enabled
- **Type**: Eac3LfeControl (p. 1008)
- **Required**: False

**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

**attenuationControl**
If set to ATTENUATE_3_DB, applies a 3 dB attenuation to the surround channels. Only used for 3/2 coding mode.
**Properties**

- **Type**: Eac3AttenuationControl (p. 1007)
  - **Required**: False

  **codingMode**

  Dolby Digital Plus coding mode. Determines number of channels.
  
  - **Type**: Eac3CodingMode (p. 1007)
  - **Required**: False

- **surroundMode**

  When encoding 2/0 audio, sets whether Dolby Surround is matrix encoded into the two channels.
  
  - **Type**: Eac3SurroundMode (p. 1012)
  - **Required**: False

- **bitstreamMode**

  Specifies the "Bitstream Mode" (bsmod) for the emitted E-AC-3 stream. See ATSC A/52-2012 (Annex E) for background on these values.
  
  - **Type**: Eac3BitstreamMode (p. 1007)
  - **Required**: False

- **lfeFilter**

  Applies a 120Hz lowpass filter to the LFE channel prior to encoding. Only valid with 3_2_LFE coding mode.
  
  - **Type**: Eac3LfeFilter (p. 1008)
  - **Required**: False

- **stereoDownmix**

  Stereo downmix preference. Only used for 3/2 coding mode.
  
  - **Type**: Eac3StereoDownmix (p. 1012)
  - **Required**: False

- **dynamicRangeCompressionRf**

  Enables Heavy Dynamic Range Compression, ensures that the instantaneous signal peaks do not exceed specified levels.
  
  - **Type**: Eac3DynamicRangeCompressionRf (p. 1008)
  - **Required**: False

- **sampleRate**

  Sample rate in hz. Sample rate is always 48000.
  
  - **Type**: integer
  - **Required**: False
Minimum: 48000
Maximum: 48000

dynamicRangeCompressionLine
 Enables Dynamic Range Compression that restricts the absolute peak level for a signal.
  
  Type: Eac3DynamicRangeCompressionLine (p. 1007)
  Required: False

dcFilter
 Activates a DC highpass filter for all input channels.
  
  Type: Eac3DcFilter (p. 1007)
  Required: False

Eac3StereoDownmix
 Stereo downmix preference. Only used for 3/2 coding mode.

   NOT_INDICATED
   LO_RO
   LT_RT
   DPL2

Eac3SurroundExMode
 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
 When encoding 2/0 audio, sets whether Dolby Surround is matrix encoded into the two channels.

   NOT_INDICATED
   ENABLED
   DISABLED

EmbeddedDestinationSettings
 Settings specific to embedded/ancillary caption outputs, including 608/708 Channel destination
number.

destination608ChannelNumber
 Ignore this setting unless your input captions are SCC format and your output container is MXF. With
this combination of input captions format and output container, you can optionally use this setting to
replace the input channel number with the track number that you specify. Specify a different number for each output captions track. If you don’t specify an output track number, the system uses the input channel number for the output channel number. This setting applies to each output individually. You can optionally combine two captions channels in your output. The two output channel numbers can be one of the following pairs: 1,3; 2,4; 1,4; or 2,3.

Type: integer
Required: False
Minimum: 1
Maximum: 4

ExceptionBody
message

Type: string
Required: False

F4vMoovPlacement
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
Settings for F4v container
moovPlacement
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.

Type: F4vMoovPlacement (p. 1013)
Required: False

FontScript
Provide the font script, using an ISO 15924 script code, if the LanguageCode is not sufficient for determining the script type. Where LanguageCode or CustomLanguageCode is sufficient, use "AUTOMATIC" or leave unset.

AUTOMATIC
HANS
HANT

FrameCaptureSettings
Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value FRAME_CAPTURE.
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

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

quality

JPEG Quality - a higher value equals higher quality.

Type: integer
Required: False
Minimum: 1
Maximum: 100

GetPresetRequest

Query a preset by sending a request with the preset name.

name

The name of the preset.

Type: string
Required: False
GetPresetResponse

Successful get preset requests will return an OK message and the preset JSON.

preset

A preset is a collection of preconfigured media conversion settings that you want MediaConvert to apply to the output during the conversion process.

  Type: Preset (p. 1072)
  Required: False

H264AdaptiveQuantization

Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

  OFF
  LOW
  MEDIUM
  HIGH
  HIGHER
  MAX

H264CodecLevel

Specify an H.264 level that is consistent with your output video settings. If you aren't sure what level to specify, choose Auto (AUTO).

  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

H.264 Profile. High 4:2:2 and 10-bit profiles are only available with the AVC-I License.

  BASELINE
Properties

- HIGH
- HIGH_10BIT
- HIGH_422
- HIGH_422_10BIT
- MAIN

**H264DynamicSubGop**

Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting B frames between reference frames (numberBFramesBetweenReferenceFrames).

- ADAPTIVE
- STATIC

**H264EntropyEncoding**

Entropy encoding mode. Use CABAC (must be in Main or High profile) or CAVLC.

- CABAC
- CAVLC

**H264FieldEncoding**

Choosing FORCE_FIELD disables PAFF encoding for interlaced outputs.

- PAFF
- FORCE_FIELD

**H264FlickerAdaptiveQuantization**

Adjust quantization within each frame to reduce flicker or 'pop' on I-frames.

- DISABLED
- ENABLED

**H264FramerateControl**

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

- INITIALIZE_FROM_SOURCE
- SPECIFIED
H264FramerateConversionAlgorithm

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

- DUPLICATE_DROP
- INTERPOLATE

H264GopBReference

If enable, use reference B frames for GOP structures that have B frames > 1.

- DISABLED
- ENABLED

H264GopSizeUnits

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

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, as follows. - 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

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

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

H264QvbrSettings

Settings for quality-defined variable bitrate encoding with the H.264 codec. Required when you set Rate control mode to QVBR. Not valid when you set Rate control mode to a value other than QVBR, or when you don’t define Rate control mode.

qvbrQualityLevel

Required when you use QVBR rate control mode. That is, when you specify qvbrSettings within h264Settings. Specify the target quality level for this output, from 1 to 10. Use higher numbers for greater quality. Level 10 results in nearly lossless compression. The quality level for most broadcast-quality transcodes is between 6 and 9.

Type: integer
Required: False
Minimum: 1
Maximum: 10

maxAverageBitrate

Use this setting only when Rate control mode is QVBR and Quality tuning level is Multi-pass HQ. For Max average bitrate values suited to the complexity of your input video, the service limits the average bitrate of the video part of this output to the value you choose. That is, the total size of the video element is less than or equal to the value you set multiplied by the number of seconds of encoded output.

Type: integer
Required: False
Minimum: 1000
Maximum: 1152000000

H264RateControlMode

Use this setting to specify whether this output has a variable bitrate (VBR), constant bitrate (CBR) or quality-defined variable bitrate (QVBR).

VBR
CBR
QVBR

H264RepeatPps

Places a PPS header on each encoded picture, even if repeated.

DISABLED
ENABLED

H264SceneChangeDetect

Scene change detection (inserts I-frames on scene changes).
DISABLED
ENABLED

**H264Settings**

Required when you set (Codec) under (VideoDescription)->(CodecSettings) to the value H_264.

**interlaceMode**

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, as follows. - 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.

* Type: H264InterlaceMode (p. 1017)
* Required: False

**parNumerator**

Pixel Aspect Ratio numerator.

* Type: integer
* Required: False
* Minimum: 1
* Maximum: 2147483647

**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

**syntax**

Produces a bitstream compliant with SMPTE RP-2027.

* Type: H264Syntax (p. 1025)
* 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

**framerateDenominator**

When you use the API for transcode jobs that use frame rate conversion, specify the frame rate 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 frame rate conversion, provide the value as a decimal number for Framerate. In this example, specify 23.976.

- **Type:** integer
- **Required:** False
- **Minimum:** 1
- **Maximum:** 2147483647

**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

**hrdBufferInitialFillPercentage**

Percentage of the buffer that should initially be filled (HRD buffer model).

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 100

**gopSize**

GOP Length (keyframe interval) in frames or seconds. Must be greater than zero.

- **Type:** number
- **Required:** False
- **Format:** float
- **Minimum:** 0.0

**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
gopBReference
If enable, use reference B frames for GOP structures that have B frames > 1.

Type: H264GopBReference (p. 1017)
Required: False

hrdBufferSize
Size of buffer (HRD buffer model) in bits. For example, enter five megabits as 5000000.

Type: integer
Required: False
Minimum: 0
Maximum: 1152000000

maxBitrate
Maximum bitrate in bits/second. For example, enter five megabits per second as 5000000. Required when Rate control mode is QVBR.

Type: integer
Required: False
Minimum: 1000
Maximum: 1152000000

slowPal
Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

Type: H264SlowPal (p. 1025)
Required: False

parDenominator
Pixel Aspect Ratio denominator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

spatialAdaptiveQuantization
Adjust quantization within each frame based on spatial variation of content complexity.

Type: H264SpatialAdaptiveQuantization (p. 1025)
Required: False

temporalAdaptiveQuantization
Adjust quantization within each frame based on temporal variation of content complexity.

Type: H264TemporalAdaptiveQuantization (p. 1026)
**Properties**

**flickerAdaptiveQuantization**
Adjust quantization within each frame to reduce flicker or 'pop' on I-frames.

Type: H264FlickerAdaptiveQuantization (p. 1016)
Required: False

**entropyEncoding**
Entropy encoding mode. Use CABAC (must be in Main or High profile) or CAVLC.

Type: H264EntropyEncoding (p. 1016)
Required: False

**bitrate**
Average bitrate in bits/second. Required for VBR and CBR. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.

Type: integer
Required: False
Minimum: 1000
Maximum: 115200000

**framerateControl**
If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

Type: H264FramerateControl (p. 1016)
Required: False

**rateControlMode**
Use this setting to specify whether this output has a variable bitrate (VBR), constant bitrate (CBR) or quality-defined variable bitrate (QVBR).

Type: H264RateControlMode (p. 1018)
Required: False

**qvbrSettings**
Settings for quality-defined variable bitrate encoding with the H.264 codec. Required when you set Rate control mode to QVBR. Not valid when you set Rate control mode to a value other than QVBR, or when you don't define Rate control mode.
Properties

Type: H264QvbrSettings (p. 1018)
Required: False

codecProfile

H.264 Profile. High 4:2:2 and 10-bit profiles are only available with the AVC-I License.

Type: H264CodecProfile (p. 1015)
Required: False

telecine

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.

Type: H264Telecine (p. 1025)
Required: False

framerateNumerator

Frame rate numerator - frame rate is a fraction, e.g. 24000 / 1001 = 23.976 fps.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

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

adaptiveQuantization

Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

Type: H264AdaptiveQuantization (p. 1015)
Required: False

codecLevel

Specify an H.264 level that is consistent with your output video settings. If you aren't sure what level to specify, choose Auto (AUTO).
**Properties**

- **Type**: H264CodecLevel (p. 1015)
  - Required: False

**fieldEncoding**

Choosing FORCE_FIELD disables PAFF encoding for interlaced outputs.

- **Type**: H264FieldEncoding (p. 1016)
  - Required: False

**sceneChangeDetect**

Scene change detection (inserts I-frames on scene changes).

- **Type**: H264SceneChangeDetect (p. 1018)
  - Required: False

**qualityTuningLevel**

Use Quality tuning level (H264QualityTuningLevel) to specify whether to use fast single-pass, high-quality singlepass, or high-quality multipass video encoding.

- **Type**: H264QualityTuningLevel (p. 1017)
  - Required: False

**framerateConversionAlgorithm**

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

- **Type**: H264FramerateConversionAlgorithm (p. 1017)
  - Required: False

**unregisteredSeiTimecode**

Inserts timecode for each frame as 4 bytes of an unregistered SEI message.

- **Type**: H264UnregisteredSeiTimecode (p. 1026)
  - Required: False

**gopSizeUnits**

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.

- **Type**: H264GopSizeUnits (p. 1017)
  - Required: False

**parControl**

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.

- **Type**: H264ParControl (p. 1017)
  - Required: False

1024
numberBFramesBetweenReferenceFrames

Number of B-frames between reference frames.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 7

repeatPps

Places a PPS header on each encoded picture, even if repeated.

- **Type**: H264RepeatPps (p. 1018)
- **Required**: False

dynamicSubGop

Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting B frames between reference frames (numberBFramesBetweenReferenceFrames).

- **Type**: H264DynamicSubGop (p. 1016)
- **Required**: False

H264SlowPal

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

- **DISABLED**
- **ENABLED**

H264SpatialAdaptiveQuantization

Adjust quantization within each frame based on spatial variation of content complexity.

- **DISABLED**
- **ENABLED**

H264Syntax

Produces a bitstream compliant with SMPTE RP-2027.

- **DEFAULT**
- **RP2027**

H264Telecine

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
Adjust quantization within each frame based on temporal variation of content complexity.

DISABLED
ENABLED

H264UnregisteredSeiTImecode
Inserts timecode for each frame as 4 bytes of an unregistered SEI message.

DISABLED
ENABLED

H265AdaptiveQuantization
Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

OFF
LOW
MEDIUM
HIGH
HIGHER
MAX

H265AlternateTransferFunctionSei
Enables Alternate Transfer Function SEI message for outputs using Hybrid Log Gamma (HLG) Electro-Optical Transfer Function (EOTF).

DISABLED
ENABLED

H265CodecLevel
H.265 Level.

AUTO
LEVEL_1
LEVEL_2
LEVEL_2_1
LEVEL_3
LEVEL_3_1
H265CodecProfile

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

H265DynamicSubGop

Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting B frames between reference frames (numberBFramesBetweenReferenceFrames).

- ADAPTIVE
- STATIC

H265FlickerAdaptiveQuantization

Adjust quantization within each frame to reduce flicker or 'pop' on I-frames.

- DISABLED
- ENABLED

H265FramerateControl

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input.
SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

INITIALIZE_FROM_SOURCE
SPECIFIED

H265FramerateConversionAlgorithm
When set to INTERPOLATE, produces smoother motion during frame rate conversion.

DUPLICATE_DROP
INTERPOLATE

H265GopBReference
If enable, use reference B frames for GOP structures that have B frames > 1.

DISABLED
ENABLED

H265GopSizeUnits
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
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
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**

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

**H265QvbrSettings**

Settings for quality-defined variable bitrate encoding with the H.265 codec. Required when you set Rate control mode to QVBR. Not valid when you set Rate control mode to a value other than QVBR, or when you don't define Rate control mode.

**qvbrQualityLevel**

Required when you use QVBR rate control mode. That is, when you specify qvbrSettings within h265Settings. Specify the target quality level for this output, from 1 to 10. Use higher numbers for greater quality. Level 10 results in nearly lossless compression. The quality level for most broadcast-quality transcodes is between 6 and 9.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 10

**maxAverageBitrate**

Use this setting only when Rate control mode is QVBR and Quality tuning level is Multi-pass HQ. For Max average bitrate values suited to the complexity of your input video, the service limits the average bitrate of the video part of this output to the value you choose. That is, the total size of the video element is less than or equal to the value you set multiplied by the number of seconds of encoded output.

- **Type**: integer
- **Required**: False
- **Minimum**: 1000
- **Maximum**: 1466400000

**H265RateControlMode**

Use this setting to specify whether this output has a variable bitrate (VBR), constant bitrate (CBR) or quality-defined variable bitrate (QVBR).

- VBR
- CBR
- QVBR

**H265SampleAdaptiveOffsetFilterMode**

Specify Sample Adaptive Offset (SAO) filter strength. Adaptive mode dynamically selects best strength based on content.

- DEFAULT
ADAPTIVE
OFF

H265SceneChangeDetect

Scene change detection (inserts I-frames on scene changes).

DISABLED
ENABLED

H265Settings

Settings for H265 codec

interlaceMode

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.

Type: H265InterlaceMode (p. 1028)
Required: False

parNumerator

Pixel Aspect Ratio numerator.

Type: integer
Required: False
Minimum: 1
Maximum: 2147483647

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

framerateDenominator

Frame rate denominator.

Type: integer
Required: False
**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

**alternateTransferFunctionSei**

Enables Alternate Transfer Function SEI message for outputs using Hybrid Log Gamma (HLG) Electro-Optical Transfer Function (EOTF).

- **Type**: H265AlternateTransferFunctionSei (p. 1026)
- **Required**: False

**hrdBufferInitialFillPercentage**

Percentage of the buffer that should initially be filled (HRD buffer model).

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 100

**gopSize**

GOP Length (keyframe interval) in frames or seconds. Must be greater than zero.

- **Type**: number
- **Required**: False
- **Format**: float
- **Minimum**: 0.0

**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

**gopBReference**

If enable, use reference B frames for GOP structures that have B frames > 1.
**Properties**

**Type**: H265GopBReference (p. 1028)  
**Required**: False

**hrdBufferSize**

Size of buffer (HRD buffer model) in bits. For example, enter five megabits as 5000000.

**Type**: integer  
**Required**: False  
**Minimum**: 0  
**Maximum**: 1466400000

**maxBitrate**

Maximum bitrate in bits/second. For example, enter five megabits per second as 5000000. Required when Rate control mode is QVBR.

**Type**: integer  
**Required**: False  
**Minimum**: 1000  
**Maximum**: 1466400000

**slowPal**

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

**Type**: H265SlowPal (p. 1036)  
**Required**: False

**parDenominator**

Pixel Aspect Ratio denominator.

**Type**: integer  
**Required**: False  
**Minimum**: 1  
**Maximum**: 2147483647

**spatialAdaptiveQuantization**

Adjust quantization within each frame based on spatial variation of content complexity.

**Type**: H265SpatialAdaptiveQuantization (p. 1037)  
**Required**: False

**temporalAdaptiveQuantization**

Adjust quantization within each frame based on temporal variation of content complexity.

**Type**: H265TemporalAdaptiveQuantization (p. 1037)  
**Required**: False
**flickerAdaptiveQuantization**

Adjust quantization within each frame to reduce flicker or 'pop' on I-frames.

- **Type**: H265FlickerAdaptiveQuantization (p. 1027)
- **Required**: False

**bitrate**

Average bitrate in bits/second. Required for VBR and CBR. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.

- **Type**: integer
- **Required**: False
- **Minimum**: 1000
- **Maximum**: 1466400000

**framerateControl**

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job sepcification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

- **Type**: H265FramerateControl (p. 1027)
- **Required**: False

**rateControlMode**

Use this setting to specify whether this output has a variable bitrate (VBR), constant bitrate (CBR) or quality-defined variable bitrate (QVBR).

- **Type**: H265RateControlMode (p. 1029)
- **Required**: False

**qvbrSettings**

Settings for quality-defined variable bitrate encoding with the H.265 codec. Required when you set Rate control mode to QVBR. Not valid when you set Rate control mode to a value other than QVBR, or when you don't define Rate control mode.

- **Type**: H265QvbrSettings (p. 1029)
- **Required**: False

**codecProfile**

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.

- **Type**: H265CodecProfile (p. 1027)
**tiles**

Enable use of tiles, allowing horizontal as well as vertical subdivision of the encoded pictures.

- **Required**: False
- **Type**: H265Tiles (p. 1037)
- **Required**: False

**telecine**

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.

- **Required**: False
- **Type**: H265Telecine (p. 1037)

**framerateNumerator**

Frame rate numerator - frame rate is a fraction, e.g. 24000 / 1001 = 23.976 fps.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 2147483647

**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

**adaptiveQuantization**

Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

- **Type**: H265AdaptiveQuantization (p. 1026)
- **Required**: False

**codecLevel**

H.265 Level.

- **Type**: H265CodecLevel (p. 1026)
**Required**: False

**sceneChangeDetect**

Scene change detection (inserts I-frames on scene changes).

- **Type**: H265SceneChangeDetect (p. 1030)
- **Required**: False

**qualityTuningLevel**

Use Quality tuning level (H265QualityTuningLevel) to specify whether to use fast single-pass, high-quality singlepass, or high-quality multipass video encoding.

- **Type**: H265QualityTuningLevel (p. 1029)
- **Required**: False

**framerateConversionAlgorithm**

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

- **Type**: H265FramerateConversionAlgorithm (p. 1028)
- **Required**: False

**unregisteredSeiTimecode**

Inserts timecode for each frame as 4 bytes of an unregistered SEI message.

- **Type**: H265UnregisteredSeiTimecode (p. 1037)
- **Required**: False

**gopSizeUnits**

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.

- **Type**: H265GopSizeUnits (p. 1028)
- **Required**: False

**parControl**

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.

- **Type**: H265ParControl (p. 1028)
- **Required**: False

**numberBFramesBetweenReferenceFrames**

Number of B-frames between reference frames.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
**Maximum:** 7

**temporalIds**

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.

**Type:** H265TemporalIds (p. 1037)

**Required:** False

**sampleAdaptiveOffsetFilterMode**

Specify Sample Adaptive Offset (SAO) filter strength. Adaptive mode dynamically selects best strength based on content

**Type:** H265SampleAdaptiveOffsetFilterMode (p. 1029)

**Required:** False

**writeMp4PackagingType**

Use this setting only for outputs encoded with H.265 that are in CMAF or DASH output groups. If you include writeMp4PackagingType in your JSON job specification for other outputs, your video might not work properly with downstream systems and video players. If the location of parameter set NAL units don't matter in your workflow, ignore this setting. The service defaults to marking your output as HEV1. Choose HVC1 to mark your output as HVC1. This makes your output compliant with this specification: ISO IECJTC1 SC29 N13798 Text ISO/IEC FDIS 14496-15 3rd Edition. For these outputs, the service stores parameter set NAL units in the sample headers but not in the samples directly. Keep the default HEV1 to mark your output as HEV1. For these outputs, the service writes parameter set NAL units directly into the samples.

**Type:** H265WriteMp4PackagingType (p. 1038)

**Required:** False

**dynamicSubGop**

Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting B frames between reference frames (numberBFramesBetweenReferenceFrames).

**Type:** H265DynamicSubGop (p. 1027)

**Required:** False

**H265SlowPal**

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

DISABLED
**Properties**

** ENABLED **

**H265SpatialAdaptiveQuantization**
Adjust quantization within each frame based on spatial variation of content complexity.

** DISABLED **

** ENABLED **

**H265Telecine**
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:
- Soft: produces 23.976; the player converts this output to 29.97i.

** NONE **

** SOFT **

** HARD **

**H265TemporalAdaptiveQuantization**
Adjust quantization within each frame based on temporal variation of content complexity.

** DISABLED **

** ENABLED **

**H265TemporalIds**
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**
Enable use of tiles, allowing horizontal as well as vertical subdivision of the encoded pictures.

** DISABLED **

** ENABLED **

**H265UnregisteredSeiTimecode**
 Inserts timecode for each frame as 4 bytes of an unregistered SEI message.

** DISABLED **
H265WriteMp4PackagingType

Use this setting only for outputs encoded with H.265 that are in CMAF or DASH output groups. If you include writeMp4PackagingType in your JSON job specification for other outputs, your video might not work properly with downstream systems and video players. If the location of parameter set NAL units don't matter in your workflow, ignore this setting. The service defaults to marking your output as HEV1. Choose HVC1 to mark your output as HVC1. This makes your output compliant with this specification: ISO IECJTC1 SC29 N13798 Text ISO/IEC FDIS 14496-15 3rd Edition. For these outputs, the service stores parameter set NAL units in the sample headers but not in the samples directly. Keep the default HEV1 to mark your output as HEV1. For these outputs, the service writes parameter set NAL units directly into the samples.

HVC1
HEV1

Hdr10Metadata

Use the "HDR master display information" (Hdr10Metadata) settings to correct HDR metadata or to provide missing metadata. These values vary depending on the input video and must be provided by a color grader. Range is 0 to 50,000; each increment represents 0.00002 in CIE1931 color coordinate. Note that these settings are not color correction. Note that if you are creating HDR outputs inside of an HLS CMAF package, to comply with the Apple specification, you must use the following settings. Set "MP4 packaging type" (writeMp4PackagingType) to HVC1 (HVC1). Set "Profile" (H265Settings > codecProfile) to Main10/High (MAIN10_HIGH). Set "Level" (H265Settings > codecLevel) to 5 (LEVEL_5).

redPrimaryX

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

Type: integer
Required: False
Minimum: 0
Maximum: 50000

redPrimaryY

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

Type: integer
Required: False
Minimum: 0
Maximum: 50000

greenPrimaryX

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.
**Properties**

**greenPrimaryY**

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 50000

**bluePrimaryX**

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 50000

**bluePrimaryY**

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 50000

**whitePointX**

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 50000

**whitePointY**

HDR Master Display Information must be provided by a color grader, using color grading tools. Range is 0 to 50,000, each increment represents 0.00002 in CIE1931 color coordinate. Note that this setting is not for color correction.
Properties

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: False
Minimum: 0
Maximum: 65535

**maxContentLightLevel**

Maximum light level among all samples in the coded video sequence, in units of candelas per square meter.

Type: integer
Required: False
Minimum: 0
Maximum: 65535

**maxLuminance**

Nominal maximum mastering display luminance in units of 0.0001 candelas per square meter.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

**minLuminance**

Nominal minimum mastering display luminance in units of 0.0001 candelas per square meter.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

**ImageInserter**

Enable the image inserter feature to include a graphic overlay on your video. Enable or disable this feature for each input or output individually. This setting is disabled by default.

**insertableImages**

Specify the images that you want to overlay on your video. The images must be PNG or TGA files.

Type: Array of type InsertableImage (p. 1041)
InsertableImage

Settings that specify how your still graphic overlay appears.

width

Specify the width of the inserted image in pixels. If you specify a value that's larger than the video resolution width, the service will crop your overlaid image to fit. To use the native width of the image, keep this setting blank.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

height

Specify the height of the inserted image in pixels. If you specify a value that's larger than the video resolution height, the service will crop your overlaid image to fit. To use the native height of the image, keep this setting blank.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

imageX

Specify the distance, in pixels, between the inserted image and the left edge of the video frame. Required for any image overlay that you specify.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

imageY

Specify the distance, in pixels, between the overlaid image and the top edge of the video frame. Required for any image overlay that you specify.

Type: integer
Required: False
Minimum: 0
Maximum: 2147483647

duration

Specify the time, in milliseconds, for the image to remain on the output video. This duration includes fade-in time but not fade-out time.

Type: integer
Properties

**fadeIn**

Specify the length of time, in milliseconds, between the Start time that you specify for the image insertion and the time that the image appears at full opacity. Full opacity is the level that you specify for the opacity setting. If you don't specify a value for Fade-in, the image will appear abruptly at the overlay start time.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 2147483647

**layer**

Specify how overlapping inserted images appear. Images with higher values for Layer appear on top of images with lower values for Layer.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 99

**imageInserterInput**

Specify the Amazon S3 location of the image that you want to overlay on the video. Use a PNG or TGA file.

- **Type**: string
- **Required**: False
- **Pattern**: ^(s3://)(.*?)/(bmp|BMP|png|PNG|tga|TGA)$
- **MinLength**: 14

**startTime**

Specify the timecode of the frame that you want the overlay to first appear on. This must be in timecode (HH:MM:SS:FF or HH:MM:SS;FF) format. Remember to take into account your timecode source settings.

- **Type**: string
- **Required**: False
- **Pattern**: ^(((\[0-1]\d)|(2\[0-3]))((:\[0-5]\d)\(2\)\(\[:;\][0-5]\d\)))$

**fadeOut**

Specify the length of time, in milliseconds, between the end of the time that you have specified for the image overlay Duration and when the overlaid image has faded to total transparency. 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**: 0
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: False  
Minimum: 0  
Maximum: 100

LanguageCode


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
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
1045
M2tsAudioBufferModel

Selects between the DVB and ATSC buffer models for Dolby Digital audio.

DVB
ATSC

**M2tsBufferModel**

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**

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**

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**

Controls whether to include the ES Rate field in the PES header.

INCLUDE
EXCLUDE

**M2tsForceTsVideoEbpOrder**

Keep the default value (DEFAULT) unless you know that your audio EBP markers are incorrectly appearing before your video EBP markers. To correct this problem, set this value to Force (FORCE).

FORCE
DEFAULT

**M2tsNielsenId3**

If INSERT, Nielsen inaudible tones for media tracking will be detected in the input audio and an equivalent ID3 tag will be inserted in the output.

INSERT
NONE

**M2tsPcrControl**

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**

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

**M2tsScte35Esam**

Settings for SCTE-35 signals from ESAM. Include this in your job settings to put SCTE-35 markers in your HLS and transport stream outputs at the insertion points that you specify in an ESAM XML document. Provide the document in the setting SCC XML (sccXml).

**scte35EsamPid**

Packet Identifier (PID) of the SCTE-35 stream in the transport stream generated by ESAM.

- **Type**: integer
- **Required**: False
- **Minimum**: 32
- **Maximum**: 8182

**M2tsScte35Source**

Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output.

- PASSTHROUGH
- NONE

**M2tsSegmentationMarkers**

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

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

MPEG-2 TS container settings. These apply to outputs in a File output group when the output's container (ContainerType) is MPEG-2 Transport Stream (M2TS). In these assets, data is organized by the program map table (PMT). Each transport stream program contains subsets of data, including audio, video, and metadata. Each of these subsets of data has a numerical label called a packet identifier (PID). Each transport stream program corresponds to one MediaConvert output. The PMT lists the types of data in a program along with their PID. Downstream systems and players use the program map table to look up the PID for each type of data it accesses and then uses the PIDs to locate specific data within the asset.

audioBufferModel

Selects between the DVB and ATSC buffer models for Dolby Digital audio.

Type: M2tsAudioBufferModel (p. 1046)
Required: False

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

esRateInPes

Controls whether to include the ES Rate field in the PES header.

Type: M2tsEsRateInPes (p. 1047)
**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

**dvbNitSettings**

Inserts DVB Network Information Table (NIT) at the specified table repetition interval.

- **Type**: DvbNitSettings (p. 1001)
- **Required**: False

**dvbSdtSettings**

Inserts DVB Service Description Table (NIT) at the specified table repetition interval.

- **Type**: DvbSdtSettings (p. 1001)
- **Required**: False

**scte35Source**

Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output.

- **Type**: M2tsScte35Source (p. 1048)
- **Required**: False

**scte35Pid**

Specify the packet identifier (PID) of the SCTE-35 stream in the transport stream.

- **Type**: integer
- **Required**: False
- **Minimum**: 32
- **Maximum**: 8182

**scte35Esam**

Include this in your job settings to put SCTE-35 markers in your HLS and transport stream outputs at the insertion points that you specify in an ESAM XML document. Provide the document in the setting SCC XML (sccXml).

- **Type**: M2tsScte35Esam (p. 1048)
- **Required**: False

**videoPid**

Specify the packet identifier (PID) of the elementary video stream in the transport stream.
Type: integer
Required: False
Minimum: 32
Maximum: 8182

dvbTdtSettings
Inserts DVB Time and Date Table (TDT) at the specified table repetition interval.

Type: DvbTdtSettings (p. 1006)
Required: False

pmtInterval
Specify the number of milliseconds between instances of the program map table (PMT) in the output transport stream.

Type: integer
Required: False
Minimum: 0
Maximum: 1000

segmentationStyle
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.

Type: M2tsSegmentationStyle (p. 1049)
Required: False

segmentationTime
Specify the length, in seconds, of each segment. Required unless markers is set to _none__.

Type: number
Required: False
Format: float
Minimum: 0.0

pmtPid
Specify the packet identifier (PID) for the program map table (PMT) itself. Default is 480.

Type: integer
Required: False
Minimum: 32
Maximum: 8182
Properties

**bitrate**
Specify 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

**audioPids**
Specify the packet identifiers (PIDs) for any elementary audio streams you include in this output. Specify multiple PIDs as a JSON array. Default is the range 482-492.

- **Type**: Array of type integer
- **Required**: False
- **Minimum**: 32
- **Maximum**: 8182

**privateMetadataPid**
Specify the packet identifier (PID) of the private metadata stream. Default is 503.

- **Type**: integer
- **Required**: False
- **Minimum**: 32
- **Maximum**: 8182

**nielsenId3**
If INSERT, Nielsen inaudible tones for media tracking will be detected in the input audio and an equivalent ID3 tag will be inserted in the output.

- **Type**: M2tsNielsenId3 (p. 1047)
- **Required**: False

**timedMetadataPid**
Specify the packet identifier (PID) for timed metadata in this output. Default is 502.

- **Type**: integer
- **Required**: False
- **Minimum**: 32
- **Maximum**: 8182

**maxPcrInterval**
Specify the maximum time, in milliseconds, between Program Clock References (PCRs) inserted into the transport stream.

- **Type**: integer
- **Required**: False
Minimum: 0  
Maximum: 500

**transportStreamId**

Specify the ID for the transport stream itself in the program map table for this output. Transport stream IDs and program map tables are parts of MPEG-2 transport stream containers, used for organizing data.

- **Type**: integer  
- **Required**: False  
- **Minimum**: 0  
- **Maximum**: 65535

**dvbSubPids**

Specify the packet identifiers (PIDs) for DVB subtitle data included in this output. Specify multiple PIDs as a JSON array. Default is the range 460-479.

- **Type**: Array of type integer  
- **Required**: False  
- **Minimum**: 32  
- **Maximum**: 8182

**rateMode**

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.

- **Type**: M2tsRateMode (p. 1048)  
- **Required**: False

**audioFramesPerPes**

The number of audio frames to insert for each PES packet.

- **Type**: integer  
- **Required**: False  
- **Minimum**: 0  
- **Maximum**: 2147483647

**pcrControl**

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.

- **Type**: M2tsPcrControl (p. 1048)  
- **Required**: False

**segmentationMarkers**

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.

- **Type:** M2tsSegmentationMarkers (p. 1048)
- **Required:** False

**ebpAudioInterval**

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).

- **Type:** M2tsEbpAudioInterval (p. 1047)
- **Required:** False

**forceTsVideoEbpOrder**

Keep the default value (DEFAULT) unless you know that your audio EBP markers are incorrectly appearing before your video EBP markers. To correct this problem, set this value to Force (FORCE).

- **Type:** M2tsForceTsVideoEbpOrder (p. 1047)
- **Required:** False

**programNumber**

Use Program number (programNumber) to specify the program number used in the program map table (PMT) for this output. Default is 1. Program numbers and program map tables are parts of MPEG-2 transport stream containers, used for organizing data.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 65535

**pcrPid**

Specify the packet identifier (PID) for the program clock reference (PCR) in this output. If you do not specify a value, the service will use the value for Video PID (VideoPid).

- **Type:** integer
- **Required:** False
- **Minimum:** 32
- **Maximum:** 8182

**bufferModel**

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.

- **Type:** M2tsBufferModel (p. 1047)
**Required**: False

**dvbTeletextPid**
Specify the packet identifier (PID) for DVB teletext data you include in this output. Default is 499.

- **Type**: integer
- **Required**: False
- **Minimum**: 32
- **Maximum**: 8182

**fragmentTime**
The length, in seconds, of each fragment. Only used with EBP markers.

- **Type**: number
- **Required**: False
- **Format**: float
- **Minimum**: 0.0

**ebpPlacement**
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 selected (segmentationMarkers is EBP or EBP_LEGACY).

- **Type**: `M2tsEbpPlacement` (p. 1047)
- **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

**M3u8NielsenId3**
If INSERT, Nielsen inaudible tones for media tracking will be detected in the input audio and an equivalent ID3 tag will be inserted in the output.

- INSERT
- NONE

**M3u8PcrControl**
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**

Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output.

- PASSTHROUGH
- NONE

**M3u8Settings**

Settings for TS segments in HLS

**audioFramesPerPes**

The number of audio frames to insert for each PES packet.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 2147483647

**pcrControl**

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.

- **Type**: M3u8PcrControl (p. 1055)
- **Required**: False

**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.

- **Type**: integer
- **Required**: False
- **Minimum**: 32
- **Maximum**: 8182

**pmtPid**

Packet Identifier (PID) for the Program Map Table (PMT) in the transport stream.

- **Type**: integer
- **Required**: False
- **Minimum**: 32
- **Maximum**: 8182

**privateMetadataPid**

Packet Identifier (PID) of the private metadata stream in the transport stream.
Properties

**Type**: integer
**Required**: False
**Minimum**: 32
**Maximum**: 8182

**programNumber**
The value of the program number field in the Program Map Table.

**Type**: integer
**Required**: False
**Minimum**: 0
**Maximum**: 65535

**patInterval**
The number of milliseconds between instances of this table in the output transport stream.

**Type**: integer
**Required**: False
**Minimum**: 0
**Maximum**: 1000

**pmtInterval**
The number of milliseconds between instances of this table in the output transport stream.

**Type**: integer
**Required**: False
**Minimum**: 0
**Maximum**: 1000

**scte35Source**
Enables SCTE-35 passthrough (scte35Source) to pass any SCTE-35 signals from input to output.

**Type**: M3u8Scte35Source (p. 1056)
**Required**: False

**scte35Pid**
Packet Identifier (PID) of the SCTE-35 stream in the transport stream.

**Type**: integer
**Required**: False
**Minimum**: 32
**Maximum**: 8182

**nielsenId3**
If INSERT, Nielsen inaudible tones for media tracking will be detected in the input audio and an equivalent ID3 tag will be inserted in the output.

**Type**: M3u8NielsenId3 (p. 1055)
**Properties**

**Required:** False

**timedMetadata**

Applies only to HLS outputs. Use this setting to specify whether the service inserts the ID3 timed metadata from the input in this output.

- **Type:** TimedMetadata (p. 1081)
- **Required:** False

**timedMetadataPid**

Packet Identifier (PID) of the timed metadata stream in the transport stream.

- **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.

- **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.

- **Type:** Array of type integer
- **Required:** False
- **Minimum:** 32
- **Maximum:** 8182

**MovClapAtom**

When enabled, include 'clap' atom if appropriate for the video output settings.

- **INCLUDE**
## MovCslgAtom

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.

### MovMpeg2FourCCControl

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

If set to OMNEON, inserts Omneon-compatible padding

- OMNEON
- NONE

### MovReference

Always keep the default value (SELF_CONTAINED) for this setting.

- SELF_CONTAINED
- EXTERNAL

### MovSettings

Settings for MOV Container.

- clapAtom

  When enabled, include 'clap' atom if appropriate for the video output settings.

    **Type:** MovClapAtom (p. 1058)

    **Required:** False

- cslgAtom

  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.

    **Type:** MovCslgAtom (p. 1059)
Required: False

paddingControl
If set to OMNEON, inserts Omneon-compatible padding

Type: MovPaddingControl (p. 1059)
Required: False

reference
Always keep the default value (SELF_CONTAINED) for this setting.

Type: MovReference (p. 1059)
Required: False

mpeg2FourCCControl
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.

Type: MovMpeg2FourCCControl (p. 1059)
Required: False

Mp2Settings
Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value MP2.

bitrate
Average bitrate in bits/second.

Type: integer
Required: False
Minimum: 32000
Maximum: 384000

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

sampleRate
Sample rate in hz.

Type: integer
Required: False
Properties

**Minimum**: 32000
**Maximum**: 48000

**Mp4CslgAtom**

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.

**Type**: Mp4CslgAtom (p. 1061)
**Required**: False

**Mp4FreeSpaceBox**

Inserts a free-space box immediately after the moov box.

**Type**: Mp4FreeSpaceBox (p. 1061)
**Required**: False

**Mp4MoovPlacement**

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.

**Type**: string

**Mp4Settings**

Settings for MP4 Container

**cslgAtom**

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.

**Type**: Mp4CslgAtom (p. 1061)
**Required**: False

**freeSpaceBox**

Inserts a free-space box immediately after the moov box.

**Type**: Mp4FreeSpaceBox (p. 1061)
**Required**: False

**mp4MajorBrand**

Overrides the "Major Brand" field in the output file. Usually not necessary to specify.

**Type**: string
**Required:** False

**moovPlacement**

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.

*Type:* Mp4MoovPlacement (p. 1061)

**Mpeg2AdaptiveQuantization**

Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

OFF
LOW
MEDIUM
HIGH

**Mpeg2CodecLevel**

Use Level (Mpeg2CodecLevel) to set the MPEG-2 level for the video output.

AUTO
LOW
MAIN
HIGH1440
HIGH

**Mpeg2CodecProfile**

Use Profile (Mpeg2CodecProfile) to set the MPEG-2 profile for the video output.

MAIN
PROFILE_422

**Mpeg2DynamicSubGop**

Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting B frames between reference frames (numberBFramesBetweenReferenceFrames).

ADAPTIVE
STATIC

**Mpeg2FramerateControl**

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown
in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

**INITIALIZE_FROM_SOURCE**

**SPECIFIED**

### Mpeg2FramerateConversionAlgorithm

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

**DUPLICATE_DROP**

**INTERPOLATE**

### Mpeg2GopSizeUnits

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

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

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**

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**

Use Quality tuning level (Mpeg2QualityTuningLevel) to specify whether to use single-pass or multipass video encoding.

- SINGLE_PASS
- MULTI_PASS

**Mpeg2RateControlMode**

Use Rate control mode (Mpeg2RateControlMode) to specify whether the bitrate is variable (vbr) or constant (cbr).

- VBR
- CBR

**Mpeg2SceneChangeDetect**

Scene change detection (inserts I-frames on scene changes).

- DISABLED
- ENABLED

**Mpeg2Settings**

Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value MPEG2.

**interlaceMode**

Use Interlace mode (InterlaceMode) to choose the scan line type for the output. * Top Field First (TOP_FIELD) and Bottom Field First (BOTTOMFIELD) 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.

- **Type**: Mpeg2InterlaceMode (p. 1063)
- **Required**: False

**parNumerator**

Pixel Aspect Ratio numerator.
**Properties**

**Type**
- Type: integer
- Required: False
- Minimum: 1
- Maximum: 2147483647

**syntax**

Produces a Type D-10 compatible bitstream (SMPTE 356M-2001).

- Type: Mpeg2Syntax (p. 1070)
- 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

**framerateDenominator**

Frame rate denominator.

- Type: integer
- Required: False
- Minimum: 1
- Maximum: 1001

**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

**hrdBufferInitialFillPercentage**

Percentage of the buffer that should initially be filled (HRD buffer model).

- Type: integer
- Required: False
- Minimum: 0
- Maximum: 100

**gopSize**

GOP Length (keyframe interval) in frames or seconds. Must be greater than zero.
**hrdBufferSize**

Size of buffer (HRD buffer model) in bits. For example, enter five megabits as 5000000.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 47185920

**maxBitrate**

Maximum bitrate in bits/second. For example, enter five megabits per second as 5000000.

- **Type**: integer
- **Required**: False
- **Minimum**: 1000
- **Maximum**: 300000000

**slowPal**

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

- **Type**: `Mpeg2SlowPal` (p. 1069)
- **Required**: False

**parDenominator**

Pixel Aspect Ratio denominator.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 2147483647

**spatialAdaptiveQuantization**

Adjust quantization within each frame based on spatial variation of content complexity.

- **Type**: `Mpeg2SpatialAdaptiveQuantization` (p. 1069)
- **Required**: False

**temporalAdaptiveQuantization**

Adjust quantization within each frame based on temporal variation of content complexity.

- **Type**: `Mpeg2TemporalAdaptiveQuantization` (p. 1070)
- **Required**: False
bitrate

Average bitrate in bits/second. Required for VBR and CBR. For MS Smooth outputs, bitrates must be unique when rounded down to the nearest multiple of 1000.

- **Type**: integer
- **Required**: False
- **Minimum**: 1000
- **Maximum**: 288000000

intraDcPrecision

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.

- **Type**: Mpeg2IntraDcPrecision (p. 1063)
- **Required**: False

framerateControl

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

- **Type**: Mpeg2FramerateControl (p. 1062)
- **Required**: False

rateControlMode

Use Rate control mode (Mpeg2RateControlMode) to specify whether the bitrate is variable (vbr) or constant (cbr).

- **Type**: Mpeg2RateControlMode (p. 1064)
- **Required**: False

codecProfile

Use Profile (Mpeg2CodecProfile) to set the MPEG-2 profile for the video output.

- **Type**: Mpeg2CodecProfile (p. 1062)
- **Required**: False

telecine

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.
**framerateNumerator**

Frame rate numerator - frame rate is a fraction, e.g. $24000 \div 1001 = 23.976$ fps.

- **Type:** integer
- **Required:** False
- **Minimum:** 24
- **Maximum:** 60000

**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

**adaptiveQuantization**

Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

- **Type:** Mpeg2AdaptiveQuantization (p. 1062)
- **Required:** False

**codecLevel**

Use Level (Mpeg2CodecLevel) to set the MPEG-2 level for the video output.

- **Type:** Mpeg2CodecLevel (p. 1062)
- **Required:** False

**sceneChangeDetect**

Scene change detection (inserts I-frames on scene changes).

- **Type:** Mpeg2SceneChangeDetect (p. 1064)
- **Required:** False

**qualityTuningLevel**

Use Quality tuning level (Mpeg2QualityTuningLevel) to specify whether to use single-pass or multipass video encoding.

- **Type:** Mpeg2QualityTuningLevel (p. 1064)
- **Required:** False
framerateConversionAlgorithm

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

Type: Mpeg2FramerateConversionAlgorithm (p. 1063)
Required: False

gopSizeUnits

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.

Type: Mpeg2GopSizeUnits (p. 1063)
Required: False

parControl

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.

Type: Mpeg2ParControl (p. 1064)
Required: False

numberBFramesBetweenReferenceFrames

Number of B-frames between reference frames.

Type: integer
Required: False
Minimum: 0
Maximum: 7

dynamicSubGop

Choose Adaptive to improve subjective video quality for high-motion content. This will cause the service to use fewer B-frames (which infer information based on other frames) for high-motion portions of the video and more B-frames for low-motion portions. The maximum number of B-frames is limited by the value you provide for the setting B frames between reference frames (numberBFramesBetweenReferenceFrames).

Type: Mpeg2DynamicSubGop (p. 1062)
Required: False

Mpeg2SlowPal

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

DISABLED
ENABLED

Mpeg2SpatialAdaptiveQuantization

Adjust quantization within each frame based on spatial variation of content complexity.
**DISABLED**
**ENABLED**

**Mpeg2Syntax**

Produces a Type D-10 compatible bitstream (SMPTE 356M-2001).

- DEFAULT
- D_10

**Mpeg2Telecine**

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**

Adjust quantization within each frame based on temporal variation of content complexity.

- DISABLED
- ENABLED

**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. When you enable Noise reducer (NoiseReducer), you must also select a value for Noise reducer filter (NoiseReducerFilter).

**filter**

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 a frequency-domain filter based on JND principles.

- **Type:** NoiseReducerFilter (p. 1071)
- **Required:** False

**filterSettings**

Settings for a noise reducer filter

- **Type:** NoiseReducerFilterSettings (p. 1071)
- **Required:** False
**spatialFilterSettings**

Noise reducer filter settings for spatial filter.

- **Type:** `NoiseReducerSpatialFilterSettings (p. 1071)`
- **Required:** False

**NoiseReducerFilter**

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 a frequency-domain filter based on JND principles.

- BILATERAL
- MEAN
- GAUSSIAN
- LANCZOS
- SHARPEN
- CONSERVE
- SPATIAL

**NoiseReducerFilterSettings**

Settings for a noise reducer filter

**strength**

Relative strength of noise reducing filter. Higher values produce stronger filtering.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 3

**NoiseReducerSpatialFilterSettings**

Noise reducer filter settings for spatial filter.

**strength**

Relative strength of noise reducing filter. Higher values produce stronger filtering.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 16

**speed**

The speed of the filter, from -2 (lower speed) to 3 (higher speed), with 0 being the nominal value.
Properties

**Type**
- Type: integer
- Required: False
- Minimum: -2
- Maximum: 3

**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

**OutputChannelMapping**
OutputChannel mapping settings.

**inputChannels**
List of input channels
- Type: Array of type integer
- Required: False
- Minimum: -60
- Maximum: 6

**OutputSdt**
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**
A preset is a collection of preconfigured media conversion settings that you want MediaConvert to apply to the output during the conversion process.

**arn**
An identifier for this resource that is unique within all of AWS.
- Type: string
- Required: False
createdAt

The timestamp in epoch seconds for preset creation.

Type: string
Required: False
Format: date-time

lastUpdated

The timestamp in epoch seconds when the preset was last updated.

Type: string
Required: False
Format: date-time

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

name

A name you create for each preset. Each name must be unique within your account.

Type: string
Required: True

type

A preset can be of two types: system or custom. System or built-in preset can't be modified or deleted by the user.

Type: Type (p. 1081)
Required: False

settings

Settings for preset

Type: PresetSettings (p. 1073)
Required: True

PresetSettings

Settings for preset
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. 1084)
- **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. 985)
- **Required:** False

containerSettings

Container specific settings.

- **Type:** ContainerSettings (p. 998)
- **Required:** False

captionDescriptions

Caption settings for this preset. There can be multiple caption settings in a single output.

- **Type:** Array of type CaptionDescriptionPreset (p. 994)
- **Required:** False

ProresCodecProfile

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

If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

- **INITIALIZE_FROM_SOURCE**
**ProresFramerateConversionAlgorithm**

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

- **DUPLICATE_DROP**
- **INTERPOLATE**

**ProresInterlaceMode**

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**

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**

Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value PRORES.

**interlaceMode**

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.

**Type**: ProresInterlaceMode (p. 1075)

**Required**: False
parNumerator
Pixel Aspect Ratio numerator.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 2147483647

framerateDenominator
Frame rate denominator.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 2147483647

codecProfile
Use Profile (ProResCodecProfile) to specify the type of Apple ProRes codec to use for this output.

- **Type**: ProresCodecProfile (p. 1074)
- **Required**: False

slowPal
Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

- **Type**: ProresSlowPal (p. 1077)
- **Required**: False

parDenominator
Pixel Aspect Ratio denominator.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 2147483647

framerateControl
If you are using the console, use the Framerate setting to specify the frame rate for this output. If you want to keep the same frame rate as the input video, choose Follow source. If you want to do frame rate conversion, choose a frame rate from the dropdown list or choose Custom. The framerates shown in the dropdown list are decimal approximations of fractions. If you choose Custom, specify your frame rate as a fraction. If you are creating your transcoding job specification as a JSON file without the console, use FramerateControl to specify which value the service uses for the frame rate for this output. Choose INITIALIZE_FROM_SOURCE if you want the service to use the frame rate from the input. Choose SPECIFIED if you want the service to use the frame rate you specify in the settings FramerateNumerator and FramerateDenominator.

- **Type**: ProresFramerateControl (p. 1074)
- **Required**: False
telecine

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.

Type: ProresTelecine (p. 1077)
Required: False

framerateNumerator

When you use the API for transcode jobs that use frame rate conversion, specify the frame rate 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

framerateConversionAlgorithm

When set to INTERPOLATE, produces smoother motion during frame rate conversion.

Type: ProresFramerateConversionAlgorithm (p. 1075)
Required: False

parControl

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).

Type: ProresParControl (p. 1075)
Required: False

ProresSlowPal

Enables Slow PAL rate conversion. 23.976fps and 24fps input is relabeled as 25fps, and audio is sped up correspondingly.

DISABLED
ENABLED

ProresTelecine

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

Use Rectangle to identify a specific area of the video frame.

**height**

Height of rectangle in pixels. Specify only even numbers.

- **Type:** integer
- **Required:** False
- **Minimum:** 2
- **Maximum:** 2147483647

**width**

Width of rectangle in pixels. Specify only even numbers.

- **Type:** integer
- **Required:** False
- **Minimum:** 2
- **Maximum:** 2147483647

**x**

The distance, in pixels, between the rectangle and the left edge of the video frame. Specify only even numbers.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 2147483647

**y**

The distance, in pixels, between the rectangle and the top edge of the video frame. Specify only even numbers.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 2147483647

RemixSettings

Use Manual audio remixing (RemixSettings) to adjust audio levels for each audio channel in each output of your job. With audio remixing, you can output more or fewer audio channels than your input audio source provides.

**channelMapping**

Channel mapping (ChannelMapping) contains the group of fields that hold the remixing value for each channel. Units are in dB. Acceptable values are within the range from -60 (mute) through 6. A setting of 0 passes the input channel unchanged to the output channel (no attenuation or amplification).

- **Type:** ChannelMapping (p. 996)
- **Required:** False
**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**: False
- **Minimum**: 1
- **Maximum**: 16

**channelsOut**

Specify the number of channels in this output after remixing. Valid values: 1, 2, 4, 6, 8

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 8

**RespondToAfd**

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**

Applies only if your input aspect ratio is different from your output aspect ratio. Choose "Stretch to output" to have the service stretch your video image to fit. Keep the setting "Default" 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**

Set Framerate (SccDestinationFramerate) to make sure that the captions and the video are synchronized in the output. Specify a frame rate that matches the frame rate of the associated video. If the video frame rate 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
Settings for SCC caption output.

framerate
Set framerate (SccDestinationFramerate) to make sure that the captions and the video are synchronized in the output. Specify a frame rate that matches the frame rate of the associated video. If the video frame rate 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).

Type: SccDestinationFramerate (p. 1079)
Required: False

TeletextDestinationSettings
Settings for Teletext caption output

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]$
MinLength: 3
MaxLength: 3

TimecodeBurnin
Timecode burn-in (TimecodeBurnIn)--Burns the output timecode and specified prefix into the output.

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
Use Position (Position) under under Timecode burn-in (TimecodeBurnIn) to specify the location the burned-in timecode on output video.

Type: TimecodeBurninPosition (p. 1081)
Required: False

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**: ^[ -~]+$  

**TimecodeBurninPosition**

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**

Applies only to HLS outputs. Use this setting to specify whether the service inserts the ID3 timed metadata from the input in this output.

- PASSTHROUGH
- NONE

**TtmlDestinationSettings**

Settings specific to TTML caption outputs, including Pass style information (TtmlStylePassthrough).

**stylePassthrough**

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.

- **Type**: TtmlStylePassthrough (p. 1081)
- **Required**: False

**TtmlStylePassthrough**

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**

- SYSTEM
**UpdatePresetRequest**

Modify a preset by sending a request with the preset name and any of the following that you wish to change: description, category, and transcoding settings.

**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

**name**

The name of the preset you are modifying.

Type: string  
Required: False

**settings**

Settings for preset

Type: PresetSettings (p. 1073)  
Required: False

**UpdatePresetResponse**

Successful update preset requests will return the new preset JSON.

**preset**

A preset is a collection of preconfigured media conversion settings that you want MediaConvert to apply to the output during the conversion process.

Type: Preset (p. 1072)  
Required: False

**VideoCodec**

Type of video codec

FRAME_CAPTURE  
H_264
VideoCodecSettings

Video codec settings, (CodecSettings) under (VideoDescription), contains the group of settings related to video encoding. The settings in this group vary depending on the value you choose for Video codec (Codec). For each codec enum you choose, define the corresponding settings object. The following lists the codec enum, settings object pairs. * H_264, H264Settings * H_265, H265Settings * MPEG2, Mpeg2Settings * PRORES, ProresSettings * FRAME_CAPTURE, FrameCaptureSettings

codec

Specifies the video codec. This must be equal to one of the enum values defined by the object VideoCodec.

Type: VideoCodec (p. 1082)
Required: False

frameCaptureSettings

Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value FRAME_CAPTURE.

Type: FrameCaptureSettings (p. 1013)
Required: False

h264Settings

Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value H_264.

Type: H264Settings (p. 1019)
Required: False

h265Settings

Settings for H265 codec

Type: H265Settings (p. 1030)
Required: False

mpeg2Settings

Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value MPEG2.

Type: Mpeg2Settings (p. 1064)
Required: False

proresSettings

Required when you set (Codec) under (VideoDescription)>(CodecSettings) to the value PRORES.

Type: ProresSettings (p. 1075)
Required: False
**VideoDescription**

Settings for video outputs

**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

**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

**scalingBehavior**

Applies only if your input aspect ratio is different from your output aspect ratio. Choose "Stretch to output" to have the service stretch your video image to fit. Keep the setting "Default" to allow the service to letterbox your video instead. This setting overrides any positioning value you specify elsewhere in the job.

- **Type:** ScalingBehavior (p. 1079)
- **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. If you crop your input frame size to smaller than your output frame size, make sure to specify the behavior you want in your output setting "Scaling behavior".

- **Type:** Rectangle (p. 1078)
- **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
videoPreprocessors

Find additional transcoding features under Preprocessors (VideoPreprocessors). Enable the features at each output individually. These features are disabled by default.

Type: VideoPreprocessor (p. 1086)
Required: False

timecodeInsertion

Applies only to H.264, H.265, MPEG2, and ProRes outputs. Only enable Timecode insertion when the input frame rate is identical to the output frame rate. To include timecodes in this output, set Timecode insertion (VideoTimecodeInsertion) to PIC_TIMING_SEI. To leave them out, set it to DISABLED. Default is DISABLED. When the service inserts timecodes in an output, by default, it uses any embedded timecodes from the input. If none are present, the service will set the timecode for the first output frame to zero. To change this default behavior, adjust the settings under Timecode configuration (TimecodeConfig). In the console, these settings are located under Job > Job settings > Timecode configuration. Note - Timecode source under input settings (InputTimecodeSource) does not affect the timecodes that are inserted in the output. Source under Job settings > Timecode configuration (TimecodeSource) does.

Type: VideoTimecodeInsertion (p. 1087)
Required: False

antiAlias

The anti-alias filter is automatically applied to all outputs. The service no longer accepts the value DISABLED for AntiAlias. If you specify that in your job, the service will ignore the setting.

Type: AntiAlias (p. 984)
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. 1078)
Required: False

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. 0 is the softest setting, 100 the sharpest, and 50 recommended for most content.

Type: integer
Required: False
Minimum: 0
Maximum: 100

codecSettings

Video codec settings, (CodecSettings) under (VideoDescription), contains the group of settings related to video encoding. The settings in this group vary depending on the value you choose for Video codec (Codec). For each codec enum you choose, define the corresponding settings object. The following
lists the codec enum, settings object pairs. * H_264, H264Settings * H_265, H265Settings * MPEG2, Mpeg2Settings * PRORES, ProresSettings * FRAME_CAPTURE, FrameCaptureSettings

  Type: VideoCodecSettings (p. 1083)
  Required: False

**afdSignaling**

This setting only applies to H.264, H.265, and MPEG2 outputs. Use Insert AFD signaling (Afdsignaling) to specify whether the service includes 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.

  Type: AfdSignaling (p. 983)
  Required: False

**dropFrameTimecode**

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.

  Type: DropFrameTimecode (p. 1000)
  Required: False

**respondToAfd**

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.

  Type: RespondToAfd (p. 1079)
  Required: False

**colorMetadata**

Enable Insert color metadata (ColorMetadata) to include color metadata in this output. This setting is enabled by default.

  Type: ColorMetadata (p. 997)
  Required: False

**VideoPreprocessor**

Find additional transcoding features under Preprocessors (VideoPreprocessors). Enable the features at each output individually. These features are disabled by default.

**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. 996)
Required: False

deinterlacer
Use Deinterlacer (Deinterlacer) to produce smoother motion and a clearer picture.

Type: Deinterlacer (p. 999)
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. 1040)
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. 1070)
Required: False

timecodeBurnin
Timecode burn-in (TimecodeBurnIn)–Burns the output timecode and specified prefix into the output.

Type: TimecodeBurnin (p. 1080)
Required: False

VideoTimecodeInsertion
Applies only to H.264, H.265, MPEG2, and ProRes outputs. Only enable Timecode insertion when the input frame rate is identical to the output frame rate. To include timecodes in this output, set Timecode insertion (VideoTimecodeInsertion) to PIC_TIMING_SEI. To leave them out, set it to DISABLED. Default is DISABLED. When the service inserts timecodes in an output, by default, it uses any embedded timecodes from the input. If none are present, the service will set the timecode for the first output frame to zero. To change this default behavior, adjust the settings under Timecode configuration (TimecodeConfig). In the console, these settings are located under Job > Job settings > Timecode configuration. Note - Timecode source under input settings (InputTimecodeSource) does not affect the timecodes that are inserted in the output. Source under Job settings > Timecode configuration (TimecodeSource) does.

DISABLED
PIC_TIMING_SEI

WavFormat
The service defaults to using RIFF for WAV outputs. If your output audio is likely to exceed 4 GB in file size, or if you otherwise need the extended support of the RF64 format, set your output WAV file format to RF64.
RIFF
RF64

**WavSettings**

Required when you set (Codec) under (AudioDescriptions)>(CodecSettings) to the value WAV.

**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

**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

**sampleRate**

Sample rate in Hz.

- **Type:** integer
- **Required:** False
- **Minimum:** 8000
- **Maximum:** 192000

**format**

The service defaults to using RIFF for WAV outputs. If your output audio is likely to exceed 4 GB in file size, or if you otherwise need the extended support of the RF64 format, set your output WAV file format to RF64.

- **Type:** WavFormat (p. 1087)
- **Required:** False

**See Also**

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

**GetPreset**

- AWS Command Line Interface
Queues

URI

/2017-08-29/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>
<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. 1092)</td>
<td>200 response</td>
</tr>
<tr>
<td>400</td>
<td>ExceptionBody (p. 1093)</td>
<td>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>403</td>
<td>ExceptionBody (p. 1093)</td>
<td>You don't have permissions for this action with the credentials you sent.</td>
</tr>
<tr>
<td>404</td>
<td>ExceptionBody (p. 1093)</td>
<td>The resource you requested does not exist.</td>
</tr>
<tr>
<td>409</td>
<td>ExceptionBody (p. 1093)</td>
<td>The service could not complete your request because there is a conflict with the current state of the resource.</td>
</tr>
<tr>
<td>429</td>
<td>ExceptionBody (p. 1093)</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>
<tr>
<td>500</td>
<td>ExceptionBody (p. 1093)</td>
<td>The service encountered an unexpected condition and cannot fulfill your request.</td>
</tr>
</tbody>
</table>

POST

Operation ID: CreateQueue
Create a new transcoding queue. For information about queues, see Working With Queues in the User Guide at https://docs.aws.amazon.com/mediaconvert/latest/ug/working-with-queues.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. 1092)</td>
<td>201 response</td>
</tr>
<tr>
<td>400</td>
<td>ExceptionBody (p. 1093)</td>
<td>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>403</td>
<td>ExceptionBody (p. 1093)</td>
<td>You don't have permissions for this action with the credentials you sent.</td>
</tr>
<tr>
<td>404</td>
<td>ExceptionBody (p. 1093)</td>
<td>The resource you requested does not exist.</td>
</tr>
<tr>
<td>409</td>
<td>ExceptionBody (p. 1093)</td>
<td>The service could not complete your request because there is a conflict with the current state of the resource.</td>
</tr>
<tr>
<td>429</td>
<td>ExceptionBody (p. 1093)</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>
<tr>
<td>500</td>
<td>ExceptionBody (p. 1093)</td>
<td>The service encountered an unexpected condition and cannot fulfill your request.</td>
</tr>
</tbody>
</table>

### Schemas

#### Request Bodies

**Example GET**

```json
{
   "listBy": enum,
   "order": enum,
   "nextToken": "string",
   "maxResults": integer
}
```

**Example POST**

```json
{
   "description": "string",
   "tags": {
   },
   "name": "string",
   "pricingPlan": enum,
}
```
"reservationPlanSettings": {
    "reservedSlots": integer,
    "renewalType": enum,
    "commitment": enum
}

### Response Bodies

#### Example ListQueuesResponse

```json
{
    "queues": [
        {
            "arn": "string",
            "createdAt": "string",
            "lastUpdated": "string",
            "type": enum,
            "pricingPlan": enum,
            "status": enum,
            "description": "string",
            "name": "string",
            "submittedJobsCount": integer,
            "progressingJobsCount": integer,
            "reservationPlan": {
                "reservedSlots": integer,
                "renewalType": enum,
                "commitment": enum,
                "purchasedAt": "string",
                "expiresAt": "string",
                "status": enum
            }
        },
    ]
}
```

#### Example CreateQueueResponse

```json
{
    "queue": {
        "arn": "string",
        "createdAt": "string",
        "lastUpdated": "string",
        "type": enum,
        "pricingPlan": enum,
        "status": enum,
        "description": "string",
        "name": "string",
        "submittedJobsCount": integer,
        "progressingJobsCount": integer,
        "reservationPlan": {
            "reservedSlots": integer,
            "renewalType": enum,
            "commitment": enum,
            "purchasedAt": "string",
            "expiresAt": "string",
            "status": enum
        }
    }
}
```
Example ExceptionBody

```json
{
  "message": "string"
}
```

Properties

Commitment

The length of the term of your reserved queue pricing plan commitment.

ONE_YEAR

CreateQueueRequest

Create an on-demand queue by sending a CreateQueue request with the name of the queue. Create a reserved queue by sending a CreateQueue request with the pricing plan set to RESERVED and with values specified for the settings under reservationPlanSettings. When you create a reserved queue, you enter into a 12-month commitment to purchase the RTS that you specify. You can't cancel this commitment.

description

Optional. A description of the queue that you are creating.

Type: string  
Required: False

tags

The tags that you want to add to the resource. You can tag resources with a key-value pair or with only a key.

Type: object  
Required: False

name

The name of the queue that you are creating.

Type: string  
Required: True

pricingPlan

Specifies whether the pricing plan for the queue is on-demand or reserved. For on-demand, you pay per minute, billed in increments of .01 minute. For reserved, you pay for the transcoding capacity of the entire queue, regardless of how much or how little you use it. Reserved pricing requires a 12-month commitment. When you use the API to create a queue, the default is on-demand.

Type: PricingPlan (p. 1095)
**Required**: False

**reservationPlanSettings**

Details about the pricing plan for your reserved queue. Required for reserved queues and not applicable to on-demand queues.

**Type**: ReservationPlanSettings (p. 1099)
**Required**: False

**CreateQueueResponse**

Successful create queue requests return the name of the queue that you just created and information about it.

**queue**

You can use queues to manage the resources that are available to your AWS account for running multiple transcoding jobs at the same time. If you don't specify a queue, the service sends all jobs through the default queue. For more information, see https://docs.aws.amazon.com/mediaconvert/latest/ug/working-with-queues.html.

**Type**: Queue (p. 1096)
**Required**: False

**ExceptionBody**

**message**

**Type**: string
**Required**: False

**ListQueuesRequest**

You can send list queues requests with an empty body. You can optionally specify the maximum number, up to twenty, of queues to be returned.

**listBy**

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.

**Type**: QueueListBy (p. 1097)
**Required**: False

**order**

When you request lists of resources, you can optionally specify whether they are sorted in ASCENDING or DESCENDING order. Default varies by resource.

**Type**: Order (p. 1095)
**Required**: False
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
Minimum: 1
Maximum: 20

ListQueuesResponse

Successful list queues requests return a JSON array of queues. If you don't specify how they are ordered, you will receive them alphabetically by name.

queues

List of queues.

Type: Array of type Queue (p. 1096)
Required: False

nextToken

Use this string to request the next batch of queues.

Type: string
Required: False

Order

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

PricingPlan

Specifies whether the pricing plan for the queue is on-demand or reserved. For on-demand, you pay per minute, billed in increments of .01 minute. For reserved, you pay for the transcoding capacity of the entire queue, regardless of how much or how little you use it. Reserved pricing requires a 12-month commitment.

ON_DEMAND
RESERVED
Queue

You can use queues to manage the resources that are available to your AWS account for running multiple transcoding jobs at the same time. If you don't specify a queue, the service sends all jobs through the default queue. For more information, see https://docs.aws.amazon.com/mediaconvert/latest/ug/working-with-queues.html.

arn

An identifier for this resource that is unique within all of AWS.

Type: string
Required: False

createdAt

The timestamp in epoch seconds for when you created the queue.

Type: string
Required: False
Format: date-time

lastUpdated

The timestamp in epoch seconds for when you most recently updated the queue.

Type: string
Required: False
Format: date-time

type

Specifies whether this on-demand queue is system or custom. System queues are built in. You can't modify or delete system queues. You can create and modify custom queues.

Type: Type (p. 1099)
Required: False

pricingPlan

Specifies whether the pricing plan for the queue is on-demand or reserved. For on-demand, you pay per minute, billed in increments of .01 minute. For reserved, you pay for the transcoding capacity of the entire queue, regardless of how much or how little you use it. Reserved pricing requires a 12-month commitment.

Type: PricingPlan (p. 1095)
Required: False

status

Queues can be ACTIVE or PAUSED. If you pause a queue, the service won't begin processing jobs in that queue. Jobs that are running when you pause the queue continue to run until they finish or result in an error.

Type: QueueStatus (p. 1097)
Required: False
description

An optional description that you create for each queue.

  Type: string
  Required: False

name

A name that you create for each queue. Each name must be unique within your account.

  Type: string
  Required: True

submittedJobsCount

The estimated number of jobs with a SUBMITTED status.

  Type: integer
  Required: False
  Format: int64

progressingJobsCount

The estimated number of jobs with a PROGRESSING status.

  Type: integer
  Required: False
  Format: int64

reservationPlan

Details about the pricing plan for your reserved queue. Required for reserved queues and not applicable to on-demand queues.

  Type: ReservationPlan (p. 1098)
  Required: False

QueueListBy

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.

  NAME
  CREATION_DATE

QueueStatus

Queues can be ACTIVE or PAUSED. If you pause a queue, jobs in that queue won't begin. Jobs that are running when you pause a queue continue to run until they finish or result in an error.

  ACTIVE
  PAUSED
RenewalType

Specifies whether the term of your reserved queue pricing plan is automatically extended (AUTO_RENEW) or expires (EXPIRE) at the end of the term.

AUTO_RENEW
EXPIRE

ReservationPlan

Details about the pricing plan for your reserved queue. Required for reserved queues and not applicable to on-demand queues.

reservedSlots

Specifies the number of reserved transcode slots (RTS) for this queue. The number of RTS determines how many jobs the queue can process in parallel; each RTS can process one job at a time. When you increase this number, you extend your existing commitment with a new 12-month commitment for a larger number of RTS. The new commitment begins when you purchase the additional capacity. You can't decrease the number of RTS in your reserved queue.

Type: integer
Required: False
Format: int32

renewalType

Specifies whether the term of your reserved queue pricing plan is automatically extended (AUTO_RENEW) or expires (EXPIRE) at the end of the term.

Type: RenewalType (p. 1098)
Required: False

commitment

The length of the term of your reserved queue pricing plan commitment.

Type: Commitment (p. 1093)
Required: False

purchasedAt

The timestamp in epoch seconds for when you set up the current pricing plan for this reserved queue.

Type: string
Required: False
Format: date-time

expiresAt

The timestamp in epoch seconds for when the current pricing plan term for this reserved queue expires.

Type: string
Required: False
**status**

Specifies whether the pricing plan for your reserved queue is ACTIVE or EXPIRED.

- **Type**: ReservationPlanStatus (p. 1099)
- **Required**: False

**ReservationPlanSettings**

Details about the pricing plan for your reserved queue. Required for reserved queues and not applicable to on-demand queues.

**reservedSlots**

Specifies the number of reserved transcode slots (RTS) for this queue. The number of RTS determines how many jobs the queue can process in parallel; each RTS can process one job at a time. You can't decrease the number of RTS in your reserved queue. You can increase the number of RTS by extending your existing commitment with a new 12-month commitment for the larger number. The new commitment begins when you purchase the additional capacity. You can't cancel your commitment or revert to your original commitment after you increase the capacity.

- **Type**: integer
- **Required**: True
- **Format**: int32

**renewalType**

Specifies whether the term of your reserved queue pricing plan is automatically extended (AUTO_RENEW) or expires (EXPIRE) at the end of the term. When your term is auto renewed, you extend your commitment by 12 months from the auto renew date. You can cancel this commitment.

- **Type**: RenewalType (p. 1098)
- **Required**: True

**commitment**

The length of the term of your reserved queue pricing plan commitment.

- **Type**: Commitment (p. 1093)
- **Required**: True

**ReservationPlanStatus**

Specifies whether the pricing plan for your reserved queue is ACTIVE or EXPIRED.

- ACTIVE
- EXPIRED

**Type**

- SYSTEM
See Also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

**ListQueues**

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Go - Pilot
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

**CreateQueue**

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Go - Pilot
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

**Queues name**

**URI**

/2017-08-29/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. 1103)</td>
<td>200 response</td>
</tr>
<tr>
<td>400</td>
<td>ExceptionBody (p. 1104)</td>
<td>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>403</td>
<td>ExceptionBody (p. 1104)</td>
<td>You don't have permissions for this action with the credentials you sent.</td>
</tr>
<tr>
<td>404</td>
<td>ExceptionBody (p. 1104)</td>
<td>The resource you requested does not exist.</td>
</tr>
<tr>
<td>409</td>
<td>ExceptionBody (p. 1104)</td>
<td>The service could not complete your request because there is a conflict with the current state of the resource.</td>
</tr>
<tr>
<td>429</td>
<td>ExceptionBody (p. 1104)</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>
<tr>
<td>500</td>
<td>ExceptionBody (p. 1104)</td>
<td>The service encountered an unexpected condition and cannot fulfill your request.</td>
</tr>
</tbody>
</table>

**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. 1103)</td>
<td>200 response</td>
</tr>
</tbody>
</table>
### HTTP Methods

#### 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>202</td>
<td>DeleteQueueResponse (p. 1104)</td>
<td>202 response</td>
</tr>
<tr>
<td>400</td>
<td>ExceptionBody (p. 1104)</td>
<td>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>403</td>
<td>ExceptionBody (p. 1104)</td>
<td>You don't have permissions for this action with the credentials you sent.</td>
</tr>
<tr>
<td>404</td>
<td>ExceptionBody (p. 1104)</td>
<td>The resource you requested does not exist.</td>
</tr>
<tr>
<td>409</td>
<td>ExceptionBody (p. 1104)</td>
<td>The service could not complete your request because there is a conflict with the current state of the resource.</td>
</tr>
<tr>
<td>429</td>
<td>ExceptionBody (p. 1104)</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>
<tr>
<td>500</td>
<td>ExceptionBody (p. 1104)</td>
<td>The service encountered an unexpected condition and cannot fulfill your request.</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>409</td>
<td>ExceptionBody (p. 1104)</td>
<td>The service could not complete your request because there is a conflict with the current state of the resource.</td>
</tr>
<tr>
<td>429</td>
<td>ExceptionBody (p. 1104)</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>
<tr>
<td>500</td>
<td>ExceptionBody (p. 1104)</td>
<td>The service encountered an unexpected condition and cannot fulfill your request.</td>
</tr>
</tbody>
</table>

**Schemas**

**Request Bodies**

**Example GET**

```json
{
  "name": "string"
}
```

**Example PUT**

```json
{
  "description": "string",
  "status": enum,
  "name": "string",
  "reservationPlanSettings": {
    "reservedSlots": integer,
    "renewalType": enum,
    "commitment": enum
  }
}
```

**Example DELETE**

```json
{
  "name": "string"
}
```

**Response Bodies**

**Example GetQueueResponse**

```json
{
  "queue": {
    "arn": "string",
    "createdAt": "string",
    "lastUpdated": "string",
  }
}
```
"type": enum,
"pricingPlan": enum,
"status": enum,
"description": "string",
"name": "string",
"submittedJobsCount": integer,
"progressingJobsCount": integer,
"reservationPlan": {
    "reservedSlots": integer,
    "renewalType": enum,
    "commitment": enum,
    "purchasedAt": "string",
    "expiresAt": "string",
    "status": enum
}
}

Example UpdateQueueResponse
{
    "queue": {
        "arn": "string",
        "createdAt": "string",
        "lastUpdated": "string",
        "type": enum,
        "pricingPlan": enum,
        "status": enum,
        "description": "string",
        "name": "string",
        "submittedJobsCount": integer,
        "progressingJobsCount": integer,
        "reservationPlan": {
            "reservedSlots": integer,
            "renewalType": enum,
            "commitment": enum,
            "purchasedAt": "string",
            "expiresAt": "string",
            "status": enum
        }
    }
}

Example DeleteQueueResponse
{

}

Example ExceptionBody
{
    "message": "string"
}

Properties

Commitment

The length of the term of your reserved queue pricing plan commitment.
ONE_YEAR

DeleteQueueRequest
Delete a queue by sending a request with the queue name. You can't delete a queue with an active pricing plan or one that has unprocessed jobs in it.

name
The name of the queue that you want to delete.

Type: string  
Required: False

DeleteQueueResponse
Delete queue requests return an OK message or error message with an empty body.

ExceptionBody

message

Type: string  
Required: False

GetQueueRequest
Get information about a queue by sending a request with the queue name.

name
The name of the queue that you want information about.

Type: string  
Required: False

GetQueueResponse
Successful get queue requests return an OK message and information about the queue in JSON.

queue
You can use queues to manage the resources that are available to your AWS account for running multiple transcoding jobs at the same time. If you don't specify a queue, the service sends all jobs through the default queue. For more information, see https://docs.aws.amazon.com/mediaconvert/latest/ug/working-with-queues.html.

Type: Queue (p. 1106)  
Required: False

PricingPlan
Specifies whether the pricing plan for the queue is on-demand or reserved. For on-demand, you pay per minute, billed in increments of .01 minute. For reserved, you pay for the transcoding capacity of
the entire queue, regardless of how much or how little you use it. Reserved pricing requires a 12-month commitment.

```
ON_DEMAND
RESERVED
```

Queue

You can use queues to manage the resources that are available to your AWS account for running multiple transcoding jobs at the same time. If you don't specify a queue, the service sends all jobs through the default queue. For more information, see https://docs.aws.amazon.com/mediaconvert/latest/ug/working-with-queues.html.

**arn**

An identifier for this resource that is unique within all of AWS.

```
Type: string
Required: False
```

**createdAt**

The timestamp in epoch seconds for when you created the queue.

```
Type: string
Required: False
Format: date-time
```

**lastUpdated**

The timestamp in epoch seconds for when you most recently updated the queue.

```
Type: string
Required: False
Format: date-time
```

**type**

Specifies whether this on-demand queue is system or custom. System queues are built in. You can't modify or delete system queues. You can create and modify custom queues.

```
Type: Type (p. 1109)
Required: False
```

**pricingPlan**

Specifies whether the pricing plan for the queue is on-demand or reserved. For on-demand, you pay per minute, billed in increments of .01 minute. For reserved, you pay for the transcoding capacity of the entire queue, regardless of how much or how little you use it. Reserved pricing requires a 12-month commitment.

```
Type: PricingPlan (p. 1105)
Required: False
```
status
Queues can be ACTIVE or PAUSED. If you pause a queue, the service won't begin processing jobs in that queue. Jobs that are running when you pause the queue continue to run until they finish or result in an error.

Type: QueueStatus (p. 1107)
Required: False

description
An optional description that you create for each queue.

Type: string
Required: False

name
A name that you create for each queue. Each name must be unique within your account.

Type: string
Required: True

submittedJobsCount
The estimated number of jobs with a SUBMITTED status.

Type: integer
Required: False
Format: int64

progressingJobsCount
The estimated number of jobs with a PROGRESSING status.

Type: integer
Required: False
Format: int64

reservationPlan
Details about the pricing plan for your reserved queue. Required for reserved queues and not applicable to on-demand queues.

Type: ReservationPlan (p. 1108)
Required: False

QueueStatus
Queues can be ACTIVE or PAUSED. If you pause a queue, jobs in that queue won't begin. Jobs that are running when you pause a queue continue to run until they finish or result in an error.

ACTIVE
PAUSED
RenewalType

Specifies whether the term of your reserved queue pricing plan is automatically extended (AUTO_RENEW) or expires (EXPIRE) at the end of the term.

- AUTO_RENEW
- EXPIRE

ReservationPlan

Details about the pricing plan for your reserved queue. Required for reserved queues and not applicable to on-demand queues.

reservedSlots

Specifies the number of reserved transcode slots (RTS) for this queue. The number of RTS determines how many jobs the queue can process in parallel; each RTS can process one job at a time. When you increase this number, you extend your existing commitment with a new 12-month commitment for a larger number of RTS. The new commitment begins when you purchase the additional capacity. You can’t decrease the number of RTS in your reserved queue.

- Type: integer
- Required: False
- Format: int32

renewalType

Specifies whether the term of your reserved queue pricing plan is automatically extended (AUTO_RENEW) or expires (EXPIRE) at the end of the term.

- Type: RenewalType (p. 1108)
- Required: False

commitment

The length of the term of your reserved queue pricing plan commitment.

- Type: Commitment (p. 1104)
- Required: False

purchasedAt

The timestamp in epoch seconds for when you set up the current pricing plan for this reserved queue.

- Type: string
- Required: False
- Format: date-time

expiresAt

The timestamp in epoch seconds for when the current pricing plan term for this reserved queue expires.

- Type: string
- Required: False
**Properties**

**Format**: date-time

**status**

Specifies whether the pricing plan for your reserved queue is ACTIVE or EXPIRED.

*Type*: ReservationPlanStatus (p. 1109)

*Required*: False

**ReservationPlanSettings**

Details about the pricing plan for your reserved queue. Required for reserved queues and not applicable to on-demand queues.

**reservedSlots**

Specifies the number of reserved transcode slots (RTS) for this queue. The number of RTS determines how many jobs the queue can process in parallel; each RTS can process one job at a time. You can't decrease the number of RTS in your reserved queue. You can increase the number of RTS by extending your existing commitment with a new 12-month commitment for the larger number. The new commitment begins when you purchase the additional capacity. You can't cancel your commitment or revert to your original commitment after you increase the capacity.

*Type*: integer

*Required*: True

*Format*: int32

**renewalType**

Specifies whether the term of your reserved queue pricing plan is automatically extended (AUTO_RENEW) or expires (EXPIRE) at the end of the term. When your term is auto renewed, you extend your commitment by 12 months from the auto renew date. You can cancel this commitment.

*Type*: RenewalType (p. 1108)

*Required*: True

**commitment**

The length of the term of your reserved queue pricing plan commitment.

*Type*: Commitment (p. 1104)

*Required*: True

**ReservationPlanStatus**

Specifies whether the pricing plan for your reserved queue is ACTIVE or EXPIRED.

ACTIVE

EXPIRED

**Type**

SYSTEM

CUSTOM
UpdateQueueRequest

Modify a queue by sending a request with the queue name and any changes to the queue.

description
The new description for the queue, if you are changing it.

Type: string
Required: False

status
Pause or activate a queue by changing its status between ACTIVE and PAUSED. If you pause a queue, jobs in that queue won't begin. Jobs that are running when you pause the queue continue to run until they finish or result in an error.

Type: QueueStatus (p. 1107)
Required: False

name
The name of the queue that you are modifying.

Type: string
Required: False

reservationPlanSettings
The new details of your pricing plan for your reserved queue. When you set up a new pricing plan to replace an expired one, you enter into another 12-month commitment. When you add capacity to your queue by increasing the number of RTS, you extend the term of your commitment to 12 months from when you add capacity. After you make these commitments, you can't cancel them.

Type: ReservationPlanSettings (p. 1109)
Required: False

UpdateQueueResponse

Successful update queue requests return the new queue information in JSON format.

queue
You can use queues to manage the resources that are available to your AWS account for running multiple transcoding jobs at the same time. If you don't specify a queue, the service sends all jobs through the default queue. For more information, see https://docs.aws.amazon.com/mediaconvert/latest/ug/working-with-queues.html.

Type: Queue (p. 1106)
Required: False

See Also
For more information about using this API in one of the language-specific AWS SDKs and references, see the following:
GetQueue

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Go - Pilot
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

UpdateQueue

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Go - Pilot
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

DeleteQueue

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Go - Pilot
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

Tags

URI

/2017-08-29/tags
## HTTP Methods

### POST

**Operation ID:** TagResource

Add tags to a MediaConvert queue, preset, or job template. For information about tagging, see the User Guide at [https://docs.aws.amazon.com/mediaconvert/latest/ug/tagging-resources.html](https://docs.aws.amazon.com/mediaconvert/latest/ug/tagging-resources.html)

**Responses**

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>TagResourceResponse (p. 1112)</td>
<td>200 response</td>
</tr>
<tr>
<td>400</td>
<td>ExceptionBody (p. 1114)</td>
<td>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>403</td>
<td>ExceptionBody (p. 1114)</td>
<td>You don't have permissions for this action with the credentials you sent.</td>
</tr>
<tr>
<td>404</td>
<td>ExceptionBody (p. 1114)</td>
<td>The resource you requested does not exist.</td>
</tr>
<tr>
<td>409</td>
<td>ExceptionBody (p. 1114)</td>
<td>The service could not complete your request because there is a conflict with the current state of the resource.</td>
</tr>
<tr>
<td>429</td>
<td>ExceptionBody (p. 1114)</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>
<tr>
<td>500</td>
<td>ExceptionBody (p. 1114)</td>
<td>The service encountered an unexpected condition and cannot fulfill your request.</td>
</tr>
</tbody>
</table>

### DELETE

**Operation ID:** UntagResourceByDelete

Deprecated: Use PUT operation instead. Remove tags from a MediaConvert queue, preset, or job template.

**Responses**

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>UnTagResourceResponse (p. 1112)</td>
<td>200 response</td>
</tr>
<tr>
<td>400</td>
<td>ExceptionBody (p. 1114)</td>
<td>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>Status Code</td>
<td>Response Model</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>----------------</td>
<td>-------------</td>
</tr>
<tr>
<td>403</td>
<td>ExceptionBody (p. 1114)</td>
<td>You don't have permissions for this action with the credentials you sent.</td>
</tr>
<tr>
<td>404</td>
<td>ExceptionBody (p. 1114)</td>
<td>The resource you requested does not exist.</td>
</tr>
<tr>
<td>409</td>
<td>ExceptionBody (p. 1114)</td>
<td>The service could not complete your request because there is a conflict with the current state of the resource.</td>
</tr>
<tr>
<td>429</td>
<td>ExceptionBody (p. 1114)</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>
<tr>
<td>500</td>
<td>ExceptionBody (p. 1114)</td>
<td>The service encountered an unexpected condition and cannot fulfill your request.</td>
</tr>
</tbody>
</table>

## Schemas

### Request Bodies

**Example POST**

```json
{
   "arn": "string",
   "tags": {
   
   }
}
```

**Example DELETE**

```json
{
   "arn": "string",
   "tagKeys": [
   "string"
   ]
}
```

### Response Bodies

**Example TagResourceResponse**

```json
{
}
```

**Example UntagResourceResponse**

```json
{
}
```
Example ExceptionBody

```json
{
  "message": "string"
}
```

## Properties

### ExceptionBody

**message**

- **Type**: string
- **Required**: False

### TagResourceRequest

To add tags to a queue, preset, or job template, send a request with the Amazon Resource Name (ARN) of the resource and the tags that you want to add.

**arn**

The Amazon Resource Name (ARN) of the resource that you want to tag. To get the ARN, send a GET request with the resource name.

- **Type**: string
- **Required**: True

**tags**

The tags that you want to add to the resource. You can tag resources with a key-value pair or with only a key.

- **Type**: object
- **Required**: True

### TagResourceResponse

A successful request to add tags to a resource returns an OK message.

### UntagResourceRequest

To remove tags from a resource, send a request with the Amazon Resource Name (ARN) of the resource and the keys of the tags that you want to remove.

**arn**

The Amazon Resource Name (ARN) of the resource that you want to remove tags from. To get the ARN, send a GET request with the resource name.

- **Type**: string
- **Required**: False
**tagKeys**

The keys of the tags that you want to remove from the resource.

- **Type:** Array of type string
- **Required:** False

**UntagResourceResponse**

A successful request to remove tags from a resource returns an OK message.

**See Also**

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

**TagResource**

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Go - Pilot
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

**UntagResourceByDelete**

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Go - Pilot
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

**Tags arn URI**

/2017-08-29/tags/arn
HTTP Methods

GET

Operation ID: ListTagsForResource

Retrieve the tags for a MediaConvert resource.

Path Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>arn</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>ListTagsForResourceResponse</td>
<td>200 response</td>
</tr>
<tr>
<td>400</td>
<td>ExceptionBody (p. 1118)</td>
<td>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>403</td>
<td>ExceptionBody (p. 1118)</td>
<td>You don’t have permissions for this action with the credentials you sent.</td>
</tr>
<tr>
<td>404</td>
<td>ExceptionBody (p. 1118)</td>
<td>The resource you requested does not exist.</td>
</tr>
<tr>
<td>409</td>
<td>ExceptionBody (p. 1118)</td>
<td>The service could not complete your request because there is a conflict with the current state of the resource.</td>
</tr>
<tr>
<td>429</td>
<td>ExceptionBody (p. 1118)</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>
<tr>
<td>500</td>
<td>ExceptionBody (p. 1118)</td>
<td>The service encountered an unexpected condition and cannot fulfill your request.</td>
</tr>
</tbody>
</table>

PUT

Operation ID: UntagResource

Remove tags from a MediaConvert queue, preset, or job template. For information about tagging, see the User Guide at https://docs.aws.amazon.com/mediaconvert/latest/ug/tagging-resources.html
Path Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>arn</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>UntagResourceResponse (p. 1117)</td>
<td>200 response</td>
</tr>
<tr>
<td>400</td>
<td>ExceptionBody (p. 1118)</td>
<td>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>403</td>
<td>ExceptionBody (p. 1118)</td>
<td>You don’t have permissions for this action with the credentials you sent.</td>
</tr>
<tr>
<td>404</td>
<td>ExceptionBody (p. 1118)</td>
<td>The resource you requested does not exist.</td>
</tr>
<tr>
<td>409</td>
<td>ExceptionBody (p. 1118)</td>
<td>The service could not complete your request because there is a conflict with the current state of the resource.</td>
</tr>
<tr>
<td>429</td>
<td>ExceptionBody (p. 1118)</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>
<tr>
<td>500</td>
<td>ExceptionBody (p. 1118)</td>
<td>The service encountered an unexpected condition and cannot fulfill your request.</td>
</tr>
</tbody>
</table>

Schemas

Request Bodies

Example GET

```json
{
   "arn": "string"
}
```

Example PUT

```json
{
   "arn": "string",
   "tagKeys": [
      "string"
   ]
}
Response Bodies

Example ListTagsForResourceResponse

```json
{
    "resourceTags": {
        "arn": "string",
        "tags": {
        }
    }
}
```

Example UntagResourceResponse

```json
{}
```

Example ExceptionBody

```json
{
    "message": "string"
}
```

Properties

ExceptionBody

message

- **Type**: string
- **Required**: False

ListTagsForResourceRequest

List the tags for your AWS Elemental MediaConvert resource by sending a request with the Amazon Resource Name (ARN) of the resource. To get the ARN, send a GET request with the resource name.

arn

The Amazon Resource Name (ARN) of the resource that you want to list tags for. To get the ARN, send a GET request with the resource name.

- **Type**: string
- **Required**: False

ListTagsForResourceResponse

A successful request to list the tags for a resource returns a JSON map of tags.

resourceTags

The Amazon Resource Name (ARN) and tags for an AWS Elemental MediaConvert resource.
ResourceTags

The Amazon Resource Name (ARN) and tags for an AWS Elemental MediaConvert resource.

arn

The Amazon Resource Name (ARN) of the resource.

Type: string
Required: False

tags

The tags for the resource.

Type: object
Required: False

UntagResourceRequest

To remove tags from a resource, send a request with the Amazon Resource Name (ARN) of the resource and the keys of the tags that you want to remove.

arn

The Amazon Resource Name (ARN) of the resource that you want to remove tags from. To get the ARN, send a GET request with the resource name.

Type: string
Required: False

tagKeys

The keys of the tags that you want to remove from the resource.

Type: Array of type string
Required: False

UntagResourceResponse

A successful request to remove tags from a resource returns an OK message.

See Also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

ListTagsForResource

- AWS Command Line Interface
• AWS SDK for .NET
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Go - Pilot
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V2

UntagResource

• AWS Command Line Interface
• AWS SDK for .NET
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Go - Pilot
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V2
Document History

The following table describes important changes to this documentation.

- **API version:** latest
- **Latest documentation update:** June 12, 2018

<table>
<thead>
<tr>
<th>Change</th>
<th>Description</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Additional captions formats</td>
<td>Added input caption format support for SCTE-20 and SMI. Added output caption format support for SCTE-20 + Embedded, Embedded + SCTE-20, SMI, and SMPTE-TT. You can now specify graphic overlays (inserted images) on individual inputs. You can now overlay motion graphics over your video.</td>
<td>Nov 23, 2018</td>
</tr>
<tr>
<td>Per-input graphic image overlay</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Motion graphic image overlay</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reserved transcoding queues</td>
<td>Reserved transcoding allows customers who have consistent video volume to benefit from lower rates in exchange for longer term commitments.</td>
<td>Sept 27, 2018</td>
</tr>
<tr>
<td>Quality-Defined Variable Bitrate (QVBR) Encoding</td>
<td>Introduces an encoding mode where the service automatically determines how many bits to use for each portion of the video to maintain constant quality.</td>
<td>Aug 13, 2018</td>
</tr>
<tr>
<td>Tagging and cost allocation</td>
<td>Add support for tagging and integration with cost allocation on AWS</td>
<td>July 2, 2018</td>
</tr>
<tr>
<td>New CMAF output group</td>
<td>cmafGroupSettings and its children added to schema, under OutputGroupSettings.</td>
<td>June 12, 2018</td>
</tr>
<tr>
<td>New Getting Started Using SDKs or CLI</td>
<td>Added chapter that shows how to get your custom endpoint and send MediaConvert requests to it. Includes examples in various programming languages.</td>
<td>May 17, 2018</td>
</tr>
<tr>
<td>New AWS Elemental MediaConvert service release</td>
<td>Initial documentation for the AWS Elemental MediaConvert service.</td>
<td>November 27, 2017</td>
</tr>
</tbody>
</table>
AWS Glossary

For the latest AWS terminology, see the AWS Glossary in the AWS General Reference.