# Table of Contents

What Is the AWS Elemental MediaLive API? ................................................................. 1  
Resources .......................................................................................................................... 2  
Channels ........................................................................................................................... 2  
  URI .................................................................................................................................. 2  
  HTTP Methods .............................................................................................................. 2  
  Schemas ....................................................................................................................... 3  
  Properties ..................................................................................................................... 28  
Channels channelId ....................................................................................................... 123  
  URI .............................................................................................................................. 123  
  HTTP Methods .......................................................................................................... 123  
  Schemas ..................................................................................................................... 125  
  Properties .................................................................................................................. 159  
Channels channelId Start ................................................................. 253  
  URI .............................................................................................................................. 253  
  HTTP Methods .......................................................................................................... 253  
  Schemas ..................................................................................................................... 254  
  Properties .................................................................................................................. 266  
Channels channelId Stop .................................................................................. 359  
  URI .............................................................................................................................. 359  
  HTTP Methods .......................................................................................................... 359  
  Schemas ..................................................................................................................... 360  
  Properties .................................................................................................................. 371  
InputSecurityGroups .......................................................................................... 464  
  URI .............................................................................................................................. 464  
  HTTP Methods .......................................................................................................... 464  
  Schemas ..................................................................................................................... 465  
  Properties .................................................................................................................. 467  
InputSecurityGroups inputSecurityGroupId .......................................................... 470  
  URI .............................................................................................................................. 470  
  HTTP Methods .......................................................................................................... 470  
  Schemas ..................................................................................................................... 472  
  Properties .................................................................................................................. 474  
Inputs ............................................................................................................................ 476  
  URI .............................................................................................................................. 476  
  HTTP Methods .......................................................................................................... 477  
  Schemas ..................................................................................................................... 478  
  Properties .................................................................................................................. 480  
Inputs inputId .......................................................................................................... 485  
  URI .............................................................................................................................. 485  
  HTTP Methods .......................................................................................................... 485  
  Schemas ..................................................................................................................... 487  
  Properties .................................................................................................................. 490  
Document History ..................................................................................................... 495  
AWS Glossary ......................................................................................................... 496
What Is the AWS Elemental MediaLive API?

This is the AWS Elemental MediaLive REST API Reference. It provides information on the URL, request contents, and response contents of each AWS Elemental MediaLive REST operation.

We assume that your IAM user credentials have the permissions needed to use AWS Elemental MediaLive via the REST API. We also assume that you are familiar with the features and operations of AWS Elemental MediaLive, as described in the user guide.

For general information on the service, see the AWS Elemental MediaLive User Guide.
Resources

The AWS Elemental MediaLive REST API includes the following resources.

Topics
- Channels (p. 2)
- Channels channelId (p. 123)
- Channels channelId Start (p. 253)
- Channels channelId Stop (p. 359)
- InputSecurityGroups (p. 464)
- InputSecurityGroups inputSecurityGroupId (p. 470)
- Inputs (p. 476)
- Inputs inputId (p. 485)

Channels

URI
/prod/channels

HTTP Methods

GET

Operation ID: ListChannels

Produces list of channels that have been created

Query Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<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>

Responses

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>ListChannelsResultModel</td>
<td>An array of channels</td>
</tr>
<tr>
<td>400</td>
<td>InvalidRequest (p. 27)</td>
<td>This request was invalid.</td>
</tr>
</tbody>
</table>
### Status Code

<table>
<thead>
<tr>
<th>Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>500</td>
<td>InternalServiceError (p. 27)</td>
<td>Unexpected internal service error.</td>
</tr>
<tr>
<td>502</td>
<td>BadGatewayException (p. 28)</td>
<td>Bad Gateway Error</td>
</tr>
<tr>
<td>403</td>
<td>AccessDenied (p. 27)</td>
<td>You do not have permission to list channels.</td>
</tr>
<tr>
<td>504</td>
<td>GatewayTimeoutException (p. 28)</td>
<td>Gateway Timeout Error</td>
</tr>
<tr>
<td>429</td>
<td>LimitExceeded (p. 27)</td>
<td>Request limit exceeded on list channel calls to channel service.</td>
</tr>
</tbody>
</table>

### POST

Operation ID: CreateChannel

Creates a new channel

### Responses

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>201</td>
<td>CreateChannelResultModel (p. 16)</td>
<td>Creation of channel is started.</td>
</tr>
<tr>
<td>400</td>
<td>InvalidRequest (p. 27)</td>
<td>This request was invalid.</td>
</tr>
<tr>
<td>422</td>
<td>ChannelConfigurationValidationError (p. 27)</td>
<td>The Channel failed validation and could not be created.</td>
</tr>
<tr>
<td>500</td>
<td>InternalServiceError (p. 27)</td>
<td>Unexpected internal service error.</td>
</tr>
<tr>
<td>502</td>
<td>BadGatewayException (p. 28)</td>
<td>Bad Gateway Error</td>
</tr>
<tr>
<td>403</td>
<td>AccessDenied (p. 27)</td>
<td>You do not have permission to list channels.</td>
</tr>
<tr>
<td>504</td>
<td>GatewayTimeoutException (p. 28)</td>
<td>Gateway Timeout Error</td>
</tr>
<tr>
<td>429</td>
<td>LimitExceeded (p. 27)</td>
<td>Request limit exceeded on list channel calls to channel service.</td>
</tr>
<tr>
<td>409</td>
<td>ResourceConflict (p. 27)</td>
<td>The channel is unable to create due to an issue with channel resources.</td>
</tr>
</tbody>
</table>

### Schemas

#### Request Bodies

**Example POST**

```json
{
  "inputAttachments (p. 51)": [ 
```


```json
{
    "inputId (p. 86)": "string",
    "inputSettings (p. 86)": {
        "sourceEndBehavior (p. 89)": enum,
        "audioSelectors (p. 89)": [
            {
                "name (p. 38)": "string",
                "selectorSettings (p. 38)": {
                    "audioLanguageSelection (p. 38)": {
                        "languageSelectionPolicy (p. 36)": enum,
                        "languageCode (p. 36)": "string"
                    },
                    "audioPidSelection (p. 38)": {
                        "pid (p. 37)": integer
                    }
                }
            }
        ],
        "deblockFilter (p. 90)": enum,
        "networkInputSettings (p. 90)": {
            "hlsInputSettings (p. 108)": {
                "retries (p. 82)": integer,
                "bandwidth (p. 82)": integer,
                "retryInterval (p. 82)": integer,
                "bufferSegments (p. 82)": integer
            },
            "serverValidation (p. 108)": enum
        },
        "inputFilter (p. 90)": enum,
        "videoSelector (p. 90)": {
            "colorSpace (p. 122)": enum,
            "selectorSettings (p. 122)": {
                "videoSelectorPid (p. 123)": {
                    "pid (p. 122)": integer
                },
                "videoSelectorProgramId (p. 123)": {
                    "programId (p. 123)": integer
                }
            },
            "colorSpaceUsage (p. 122)": enum
        },
        "filterStrength (p. 90)": integer,
        "captionSelectors (p. 90)": [
            {
                "name (p. 47)": "string",
                "languageCode (p. 47)": "string",
                "selectorSettings (p. 47)": {
                    "embeddedSourceSettings (p. 47)": {
                        "scte20Detection (p. 62)": enum,
                        "source608ChannelNumber (p. 62)": integer,
                        "convert608To708 (p. 63)": enum,
                        "source608TrackNumber (p. 63)": integer
                    },
                    "scte20SourceSettings (p. 47)": {
                        "source608ChannelNumber (p. 114)": integer,
                        "convert608To708 (p. 114)": enum
                    },
                    "dvbSubSourceSettings (p. 47)": {
                        "pid (p. 57)": integer
                    },
                    "aribSourceSettings (p. 47)": {
                    },
                    "teletextSourceSettings (p. 47)": {
                        "pageNumber (p. 118)": "string"
                    },
                    "scte27SourceSettings (p. 48)": {
```
"pid (p. 114)": integer
    
],
"denoiseFilter (p. 90)": enum
    
],
"requestId (p. 51)": "string",
"roleArn (p. 51)": "string",
"destinations (p. 51)": [
    
],
"settings (p. 109)": [
    
],
"id (p. 109)": "string"
    
],
"name (p. 52)": "string",
"encoderSettings (p. 52)": { 
    "timecodeConfig (p. 63)": { 
        "syncThreshold (p. 118)": integer,
        "source (p. 118)": enum
    },
    "outputGroups (p. 63)": [ 
        
    ],
"outputs (p. 110)": [ 
        
    ],
"archiveOutputSettings (p. 111)": { 
    "extension (p. 32)": "string",
    "containerSettings (p. 33)": {
        "m2tsSettings (p. 32)": { 
            "audioStreamType (p. 94)": enum,
            "ecmPid (p. 94)": "string",
            "dvbTeletextPid (p. 94)": "string",
            "aribCaptionsPidControl (p. 94)": enum,
            "bitrate (p. 94)": integer,
            "segmentationTime (p. 94)": number,
            "rateMode (p. 95)": enum,
            "audioPids (p. 95)": "string",
            "fragmentTime (p. 95)": number,
            "ebpAudioInterval (p. 95)": enum,
            "ebpLookaheadMs (p. 95)": integer,
            "audioFramesPerPes (p. 95)": integer,
            "scte35Pid (p. 96)": "string",
            "pcrPeriod (p. 96)": integer,
            "scte35HeaderLength (p. 96)": number,
"pmtInterval (p. 96)": integer,
"programNum (p. 96)": integer,
"segmentationStyle (p. 96)": enum,
"ebif (p. 97)": enum,
"audioBufferModel (p. 97)": enum,
"dvbNitSettings (p. 97)": {
  "networkName (p. 52)": "string",
  "networkId (p. 52)": integer,
  "repInterval (p. 52)": integer
},
"absentInputAudioBehavior (p. 97)": enum,
"timedMetadataPid (p. 97)": "string",
"timedMetadataBehavior (p. 97)": enum,
"etvSignalPid (p. 97)": "string",
"pmtPid (p. 98)": "string",
"bufferModel (p. 98)": enum,
"scte35Control (p. 98)": enum,
"ebpPlacement (p. 98)": enum,
"arib (p. 98)": enum,
"nullPacketBitrate (p. 98)": number,
"dvbSdtSettings (p. 98)": {
  "serviceName (p. 53)": "string",
  "serviceProviderName (p. 53)": "string",
  "repInterval (p. 53)": integer,
  "outputSdt (p. 53)": enum
},
"pcrPid (p. 99)": "string",
"transportStreamId (p. 99)": integer,
"pcrControl (p. 99)": enum,
"videoPid (p. 99)": "string",
"esRateInPes (p. 99)": enum,
"segmentationMarkers (p. 99)": enum,
"dvbTdtSettings (p. 100)": {
  "repInterval (p. 57)": integer
},
"klv (p. 100)": enum,
"ccDescriptor (p. 100)": enum,
"patInterval (p. 100)": integer,
"etvPlatformPid (p. 100)": "string",
"dvbSubPids (p. 100)": "string",
"aribCaptionsPid (p. 100)": "string",
"scte27Pids (p. 101)": "string",
"klvDataPids (p. 101)": "string"
},
"nameModifier (p. 33)": "string"
},
"msSmoothOutputSettings (p. 111)": {
  "nameModifier (p. 107)": "string"
},
"udpOutputSettings (p. 111)": {
  "bufferMsec (p. 119)": integer,
  "destination (p. 120)": {
    "destinationRefId (p. 111)": "string"
  }
},
"containerSettings (p. 120)": {
  "m2tsSettings (p. 119)": {
    "audioStreamType (p. 94)": enum,
    "ecmPid (p. 94)": "string",
    "dvbTeletextPid (p. 94)": "string",
    "aribCaptionsPidControl (p. 94)": enum,
    "bitrate (p. 94)": integer,
    "segmentationTime (p. 94)": number,
    "rateMode (p. 95)": enum,
    "audioPids (p. 95)": "string",
    "fragmentTime (p. 95)": number,
"ebpAudioInterval (p. 95)": enum,
"ebpLookaheadMs (p. 95)": integer,
"audioFramesPerPES (p. 95)": integer,
"scte35Pid (p. 96)": "string",
"pcrPeriod (p. 96)": integer,
"pmtInterval (p. 96)": integer,
"programNum (p. 96)": integer,
"segmentationStyle (p. 96)": enum,
"ebif (p. 97)": enum,
"audioBufferModel (p. 97)": enum,
"dvbNitSettings (p. 97)": {
  "networkName (p. 52)": "string",
  "networkId (p. 52)": integer,
  "repInterval (p. 52)": integer
},
"absentInputAudioBehavior (p. 97)": enum,
"timedMetadataPid (p. 97)": "string",
"timedMetadataBehavior (p. 97)": enum,
"etvSignalPid (p. 97)": "string",
"pmtPid (p. 98)": "string",
"bufferModel (p. 98)": enum,
"scte35Control (p. 98)": enum,
"ebpPlacement (p. 98)": enum,
"arib (p. 98)": enum,
"nullPacketBitrate (p. 98)": number,
"dvbSdtSettings (p. 98)": {
  "serviceName (p. 53)": "string",
  "serviceProviderName (p. 53)": "string",
  "repInterval (p. 53)": integer,
  "outputSdt (p. 53)": enum
},
"pcrPid (p. 99)": "string",
"transportStreamId (p. 99)": integer,
"pcrControl (p. 99)": enum,
"videoPid (p. 99)": "string",
"esRateInPES (p. 99)": enum,
"segmentationMarkers (p. 99)": enum,
"dvbTdtSettings (p. 100)": {
  "repInterval (p. 57)": integer
},
"klv (p. 100)": enum,
"ccDescriptor (p. 100)": enum,
"patInterval (p. 100)": integer,
"etvPlatformPid (p. 100)": "string",
"dvbSubPids (p. 100)": "string",
"aribCaptionsPid (p. 100)": "string",
"scte27Pids (p. 101)": "string",
"klvDataPids (p. 101)": "string"
}
},
"fecOutputSettings (p. 120)": {
  "rowLength (p. 64)": integer,
  "columnDepth (p. 64)": integer,
  "includeFec (p. 65)": enum
}
},
"hlsOutputSettings (p. 111)": {
  "segmentModifier (p. 84)": "string",
  "hlsSettings (p. 84)": {
    "audioOnlyHlsSettings (p. 85)": {
      "audioTrackType (p. 37)": enum,
      "audioGroupId (p. 37)": "string",
      "audioOnlyImage (p. 37)": {
        "passwordParam (p. 87)": "string",
        "uri (p. 87)": "string",
        "username (p. 88)": "string"
      }
    }
  }
}
}
"standardHlsSettings (p. 85)": {
  "m3u8Settings (p. 117)": {
    "pmtPid (p. 101)": "string",
    "ecmPid (p. 101)": "string",
    "scte35Behavior (p. 102)": enum,
    "pcrPid (p. 102)": "string",
    "audioPids (p. 102)": "string",
    "audioFramesPerPes (p. 102)": integer,
    "scte35Pid (p. 102)": "string",
    "transportStreamId (p. 102)": integer,
    "videoPid (p. 102)": "string",
    "pcrControl (p. 103)": enum,
    "pmtInterval (p. 103)": integer,
    "pcrPeriod (p. 103)": integer,
    "programNum (p. 103)": integer,
    "patInterval (p. 103)": integer,
    "timedMetadataPid (p. 103)": "string",
    "timedMetadataBehavior (p. 104)": enum
  },
  "audioRenditionSets (p. 117)": "string"
},
"nameModifier (p. 84)": "string"
},
"audioDescriptionNames (p. 109)": [
  "string"
],
"outputGroupSettings (p. 110)": {
  "archiveGroupSettings (p. 110)": {
    "destination (p. 32)": {
      "destinationRefId (p. 111)": "string"
    },
    "rolloverInterval (p. 32)": integer
  },
  "rtmpGroupSettings (p. 110)": {
    "captionData (p. 112)": enum,
    "authenticationScheme (p. 112)": enum,
    "cacheLength (p. 113)": integer,
    "restartDelay (p. 113)": integer,
    "cacheFullBehavior (p. 113)": enum
  },
  "udpGroupSettings (p. 110)": {
    "inputLossAction (p. 119)": enum,
    "timedMetadataId3Frame (p. 119)": enum,
    "timedMetadataId3Period (p. 119)": integer
  },
  "msSmoothGroupSettings (p. 110)": {
    "eventId (p. 104)": "string",
    "fragmentLength (p. 105)": integer,
    "timestampOffset (p. 105)": "string",
    "segmentationMode (p. 105)": enum,
    "numRetries (p. 105)": integer,
    "eventStopBehavior (p. 105)": enum,
    "acquisitionPointId (p. 105)": "string",
    "sparseTrackType (p. 105)": enum,
    "timestampOffsetMode (p. 106)": enum,
    "destination (p. 106)": {
      "destinationRefId (p. 111)": "string"
    },
    "audioOnlyTimecodeControl (p. 106)": enum,
    "connectionRetryInterval (p. 106)": integer,
"filecacheDuration (p. 106)": integer,
"certificateMode (p. 106)": enum,
"inputLossAction (p. 106)": enum,
"sendDelayMs (p. 107)": integer,
"eventIdMode (p. 107)": enum,
"restartDelay (p. 107)": integer,
"streamManifestBehavior (p. 107)": enum
},
"hlsGroupSettings (p. 110)": {
  "segmentsPerSubdirectory (p. 77)": integer,
  "ivInManifest (p. 77)": enum,
  "outputSelection (p. 77)": enum,
  "encryptionType (p. 77)": enum,
  "indexNSegments (p. 77)": integer,
  "destination (p. 77)": {
    "destinationRefId (p. 111)": "string"
  },
  "constantIv (p. 78)": "string",
  "timedMetadataId3Frame (p. 78)": enum,
  "baseUrlManifest (p. 78)": "string",
  "captionLanguageSetting (p. 78)": enum,
  "minSegmentLength (p. 78)": integer,
  "mode (p. 78)": enum,
  "ivSource (p. 79)": enum,
  "manifestCompression (p. 79)": enum,
  "keyProviderSettings (p. 79)": {
    "staticKeySettings (p. 91)": {
      "staticKeyValue (p. 117)": "string",
      "keyProviderServer (p. 117)": {
        "passwordParam (p. 87)": "string",
        "uri (p. 87)": "string",
        "username (p. 88)": "string"
      }
    }
  },
  "tsFileMode (p. 79)": enum,
  "manifestDurationFormat (p. 79)": enum,
  "keyFormatVersions (p. 79)": "string",
  "streamInfResolution (p. 79)": enum,
  "timestampDeltaMilliseconds (p. 80)": integer,
  "segmentationMode (p. 80)": enum,
  "baseUrlContent (p. 80)": "string",
  "clientCache (p. 80)": enum,
  "captionLanguageMappings (p. 80)": [
    {
      "languageDescription (p. 46)": "string",
      "captionChannel (p. 46)": integer,
      "languageCode (p. 46)": "string"
    }
  ],
  "codecSpecification (p. 80)": enum,
  "keepSegments (p. 80)": integer,
  "timedMetadataId3Period (p. 81)": integer,
  "programDateTime (p. 81)": enum,
  "directoryStructure (p. 81)": enum,
  "keyFormat (p. 81)": "string",
  "inputLossAction (p. 81)": enum,
  "adMarkers (p. 81)": [
    enum
  ],
  "programDateTimePeriod (p. 81)": enum,
  "segmentLength (p. 82)": integer,
  "hlsCdnSettings (p. 82)": {
    "hlsAkamaiSettings (p. 76)": {
      "httpTransferMode (p. 74)": enum,
      "salt (p. 74)": "string"
"numRetries (p. 74)": integer,
"restartDelay (p. 74)": integer,
"connectionRetryInterval (p. 75)": integer,
"filecacheDuration (p. 75)": integer,
"token (p. 75)": "string"
},
"hlsWebdavSettings (p. 76)": {
  "httpTransferMode (p. 85)": enum,
  "numRetries (p. 85)": integer,
  "restartDelay (p. 86)": integer,
  "connectionRetryInterval (p. 86)": integer,
  "filecacheDuration (p. 86)": integer
},
"hlsBasicPutSettings (p. 76)": {
  "numRetries (p. 75)": integer,
  "restartDelay (p. 75)": integer,
  "connectionRetryInterval (p. 75)": integer,
  "filecacheDuration (p. 76)": integer
},
"hlsMediaStoreSettings (p. 76)": {
  "mediaStoreStorageClass (p. 83)": enum,
  "numRetries (p. 83)": integer,
  "restartDelay (p. 83)": integer,
  "connectionRetryInterval (p. 83)": integer,
  "filecacheDuration (p. 84)": integer
}
},
"name (p. 110)": "string"
],
"audioDescriptions (p. 63)": [
  {
    "languageCodeControl (p. 34)": enum,
    "audioTypeControl (p. 34)": enum,
    "remixSettings (p. 34)": {
      "channelMappings (p. 111)": {
        "outputChannel (p. 33)": integer,
        "inputChannelLevels (p. 33)": [
          "inputChannel (p. 86)": integer,
          "gain (p. 87)": integer
        ]
      }
    }
  },
  "channelsOut (p. 111)": integer,
  "channelsIn (p. 112)": integer
},
"audioType (p. 34)": enum,
"name (p. 35)": "string",
"languageCode (p. 35)": "string",
"codecSettings (p. 35)": {
  "aacSettings (p. 33)": {
    "vbrQuality (p. 28)": enum,
    "codingMode (p. 29)": enum,
    "profile (p. 29)": enum,
    "bitrate (p. 29)": number,
    "inputType (p. 29)": enum,
    "rawFormat (p. 29)": enum,
    "sampleRate (p. 29)": number,
    "rateControlMode (p. 29)": enum,
    "spec (p. 30)": enum
  }
}
"ac3Settings (p. 33)": {
  "drcProfile (p. 31)": enum,
  "dialnorm (p. 31)": integer,
  "codingMode (p. 31)": enum,
  "metadataControl (p. 31)": enum,
  "bitrate (p. 31)": number,
  "lfeFilter (p. 31)": enum,
  "bitstreamMode (p. 32)": enum
},
"eac3Settings (p. 34)": {
  "dialnorm (p. 59)": integer,
  "passthroughControl (p. 59)": enum,
  "metadataControl (p. 59)": enum,
  "drcLine (p. 59)": enum,
  "bitrate (p. 60)": number,
  "surroundExMode (p. 60)": enum,
  "ltRtSurroundMixLevel (p. 60)": number,
  "lifeControl (p. 60)": enum,
  "codingMode (p. 60)": enum,
  "surroundMode (p. 60)": enum,
  "attenuationControl (p. 60)": enum,
  "lfeFilter (p. 60)": enum,
  "ltRtCenterMixLevel (p. 61)": number,
  "dcFilter (p. 61)": enum,
  "phaseControl (p. 61)": enum,
  "stereoDownmix (p. 61)": enum,
  "bitstreamMode (p. 61)": enum,
  "loRoSurroundMixLevel (p. 61)": number,
  "drRf (p. 61)": enum,
  "loRoCenterMixLevel (p. 62)": number
},
"passThroughSettings (p. 34)": {
},
"mp2Settings (p. 34)": {
  "codingMode (p. 104)": enum,
  "bitrate (p. 104)": number,
  "sampleRate (p. 104)": number
},
"streamName (p. 35)": "string",
"audioNormalizationSettings (p. 35)": {
  "targetLkfs (p. 36)": number,
  "algorithmControl (p. 36)": enum,
  "algorithm (p. 37)": enum
},
"audioSelectorName (p. 35)": "string"
},
"captionDescriptions (p. 63)": [
  {
    "captionSelectorName (p. 44)": "string",
    "languageDescription (p. 44)": "string",
    "name (p. 44)": "string",
    "languageCode (p. 45)": "string",
    "destinationSettings (p. 45)": {
      "scte27DestinationSettings (p. 45)": {
      },
      "burnInDestinationSettings (p. 45)": {
        "xPosition (p. 41)": integer,
        "backgroundColor (p. 41)": enum,
        "yPosition (p. 41)": integer,
        "teletextGridControl (p. 41)": enum,
        "backgroundOpacity (p. 41)": integer,
        "fontOpacity (p. 42)": integer,
        "fontResolution (p. 42)": integer,
        "shadowOpacity (p. 42)": integer,
      }},
      "burnInDestinationSettings (p. 45)": {
        "xPosition (p. 41)": integer,
        "backgroundColor (p. 41)": enum,
        "yPosition (p. 41)": integer,
        "teletextGridControl (p. 41)": enum,
        "backgroundOpacity (p. 41)": integer,
        "fontOpacity (p. 42)": integer,
        "fontResolution (p. 42)": integer,
        "shadowOpacity (p. 42)": integer,
      }},
      "burnInDestinationSettings (p. 45)": {
        "xPosition (p. 41)": integer,
        "backgroundColor (p. 41)": enum,
        "yPosition (p. 41)": integer,
        "teletextGridControl (p. 41)": enum,
        "backgroundOpacity (p. 41)": integer,
        "fontOpacity (p. 42)": integer,
        "fontResolution (p. 42)": integer,
        "shadowOpacity (p. 42)": integer,
      }},
      "burnInDestinationSettings (p. 45)": {
        "xPosition (p. 41)": integer,
        "backgroundColor (p. 41)": enum,
        "yPosition (p. 41)": integer,
        "teletextGridControl (p. 41)": enum,
        "backgroundOpacity (p. 41)": integer,
        "fontOpacity (p. 42)": integer,
        "fontResolution (p. 42)": integer,
        "shadowOpacity (p. 42)": integer,
      }},
      "burnInDestinationSettings (p. 45)": {
        "xPosition (p. 41)": integer,
        "backgroundColor (p. 41)": enum,
        "yPosition (p. 41)": integer,
        "teletextGridControl (p. 41)": enum,
        "backgroundOpacity (p. 41)": integer,
        "fontOpacity (p. 42)": integer,
        "fontResolution (p. 42)": integer,
        "shadowOpacity (p. 42)": integer,
      }},
      "burnInDestinationSettings (p. 45)": {
        "xPosition (p. 41)": integer,
        "backgroundColor (p. 41)": enum,
        "yPosition (p. 41)": integer,
        "teletextGridControl (p. 41)": enum,
        "backgroundOpacity (p. 41)": integer,
        "fontOpacity (p. 42)": integer,
        "fontResolution (p. 42)": integer,
        "shadowOpacity (p. 42)": integer,
      }},
      "burnInDestinationSettings (p. 45)": {
        "xPosition (p. 41)": integer,
        "backgroundColor (p. 41)": enum,
        "yPosition (p. 41)": integer,
        "teletextGridControl (p. 41)": enum,
        "backgroundOpacity (p. 41)": integer,
        "fontOpacity (p. 42)": integer,
        "fontResolution (p. 42)": integer,
        "shadowOpacity (p. 42)": integer,
      }},
      "burnInDestinationSettings (p. 45)": {
        "xPosition (p. 41)": integer,
        "backgroundColor (p. 41)": enum,
        "yPosition (p. 41)": integer,
        "teletextGridControl (p. 41)": enum,
        "backgroundOpacity (p. 41)": integer,
        "fontOpacity (p. 42)": integer,
        "fontResolution (p. 42)": integer,
        "shadowOpacity (p. 42)": integer,
      }},
      "burnInDestinationSettings (p. 45)": {
        "xPosition (p. 41)": integer,
        "backgroundColor (p. 41)": enum,
        "yPosition (p. 41)": integer,
        "teletextGridControl (p. 41)": enum,
        "backgroundOpacity (p. 41)": integer,
        "fontOpacity (p. 42)": integer,
"shadowYOffset (p. 42)": integer,
"outlineSize (p. 42)": integer,
"outlineColor (p. 42)": enum,
"fontSize (p. 43)": "string",
"shadowXOffset (p. 43)": integer,
"alignment (p. 43)": enum,
"shadowColor (p. 43)": enum,
"fontColor (p. 43)": enum,
"font (p. 43)": {
  "passwordParam (p. 87)": "string",
  "uri (p. 87)": "string",
  "username (p. 88)": "string"
}
},
"teletextDestinationSettings (p. 45)": {},
"smpteTtDestinationSettings (p. 45)": {},
"webvttDestinationSettings (p. 45)": {},
"ttmlDestinationSettings (p. 45)": {
  "styleControl (p. 118)": enum
},
"embeddedPlusScte20DestinationSettings (p. 45)": {},
"dvbSubDestinationSettings (p. 46)": {
  "xPosition (p. 54)": integer,
  "backgroundColor (p. 54)": enum,
  "position (p. 54)": integer,
  "teletextGridControl (p. 55)": enum,
  "backgroundColorOpacity (p. 55)": integer,
  "fontOpacity (p. 55)": integer,
  "fontResolution (p. 55)": integer,
  "shadowOpacity (p. 55)": integer,
  "shadowYOffset (p. 55)": integer,
  "outlineSize (p. 56)": integer,
  "outlineColor (p. 56)": enum,
  "fontSize (p. 56)": "string",
  "shadowXOffset (p. 56)": integer,
  "alignment (p. 56)": enum,
  "shadowColor (p. 56)": enum,
  "fontColor (p. 57)": enum,
  "font (p. 57)": {
    "passwordParam (p. 87)": "string",
    "uri (p. 87)": "string",
    "username (p. 88)": "string"
  }
},
"embeddedDestinationSettings (p. 46)": {},
"rtmpCaptionInfoDestinationSettings (p. 46)": {},
"aribDestinationSettings (p. 46)": {},
"scte20PlusEmbeddedDestinationSettings (p. 46)": {}
"scte35SpliceInsert (p. 39)": {
  "adAvailOffset (p. 115)": integer,
  "webDeliveryAllowedFlag (p. 115)": enum,
  "noRegionalBlackoutFlag (p. 115)": enum
}

"globalConfiguration (p. 64)": {
  "inputLossBehavior (p. 65)": {
    "inputLossImageType (p. 88)": enum,
    "inputLossImageColor (p. 88)": "string",
    "inputLossImageSlate (p. 88)": {
      "passwordParam (p. 87)": "string",
      "uri (p. 87)": "string",
      "username (p. 88)": "string"
    },
    "blackFrameMsec (p. 89)": integer,
    "repeatFrameMsec (p. 89)": integer
  },
  "supportLowFramerateInputs (p. 65)": enum,
  "initialAudioGain (p. 65)": integer,
  "inputEndAction (p. 66)": enum,
  "outputTimingSource (p. 66)": enum
},
"videoDescriptions (p. 64)": [
  {
    "respondToAfd (p. 120)": enum,
    "scalingBehavior (p. 121)": enum,
    "name (p. 121)": "string",
    "width (p. 121)": integer,
    "sharpness (p. 121)": integer,
    "codecSettings (p. 121)": {
      "h264Settings (p. 120)": {
        "minIInterval (p. 68)": integer,
        "slices (p. 68)": integer,
        "parNumerator (p. 69)": integer,
        "gopSizeUnits (p. 69)": enum,
        "maxBitrate (p. 69)": integer,
        "bitrate (p. 69)": integer,
        "bufFillPct (p. 69)": integer,
        "temporalAq (p. 69)": enum,
        "afdSignaling (p. 70)": enum,
        "timecodeInsertion (p. 70)": enum,
        "bufSize (p. 70)": integer,
        "softness (p. 70)": integer,
        "framerateControl (p. 70)": enum,
        "fixedAfd (p. 70)": enum,
        "level (p. 70)": enum,
        "lookAheadRateControl (p. 71)": enum,
        "profile (p. 71)": enum,
        "framerateNumerator (p. 71)": integer,
        "gopClosedCadence (p. 71)": integer,
        "framerateDenominator (p. 71)": integer,
        "entropyEncoding (p. 71)": enum,
        "spatialAq (p. 71)": enum,
        "adaptiveQuantization (p. 72)": enum,
        "colorMetadata (p. 72)": enum,
        "gopSize (p. 72)": number,
        "numRefFrames (p. 72)": integer,
        "gopBReference (p. 72)": enum,
        "sceneChangeDetect (p. 72)": enum,
        "parControl (p. 72)": enum,
        "parDenominator (p. 73)": integer,
        "syntax (p. 73)": enum,
        "scanType (p. 73)": enum,
        "gopNumBFrames (p. 73)": integer,
"flickerAq (p. 73)": enum,
"rateControlMode (p. 73)": enum
},

"height (p. 121)": integer
],
"blackoutSlate (p. 64)": {
"networkEndBlackoutImage (p. 39)": {
"passwordParam (p. 87)": "string",
"uri (p. 87)": "string",
"username (p. 88)": "string"
},
"networkEndBlackout (p. 40)": enum,
"networkId (p. 40)": "string",
"state (p. 40)": enum,
"blackoutSlateImage (p. 40)": {
"passwordParam (p. 87)": "string",
"uri (p. 87)": "string",
"username (p. 88)": "string"
},
"availBlanking (p. 64)": {
"state (p. 38)": enum,
"availBlankingImage (p. 39)": {
"passwordParam (p. 87)": "string",
"uri (p. 87)": "string",
"username (p. 88)": "string"
}
},
"inputSpecification (p. 52)": {
"codec (p. 91)": enum,
"resolution (p. 91)": enum,
"maximumBitrate (p. 91)": enum
}
}

Response Bodies

Example ListChannelsResultModel

```json
{
  "channels (p. 92)": [
    {
      "inputAttachments (p. 50)": [
        {
          "inputId (p. 86)": "string",
          "inputSettings (p. 86)": {
            "sourceEndBehavior (p. 89)": enum,
            "audioSelectors (p. 89)": [
              {
                "name (p. 38)": "string",
                "selectorSettings (p. 38)": {
                  "audioLanguageSelection (p. 38)": enum,
                  "languageSelectionPolicy (p. 36)": enum,
                  "languageCode (p. 36)": "string"
                },
                "audioPidSelection (p. 38)": {
                  "pid (p. 37)": integer
                }
              }
            ]
          }
        }
      ]
    }
  ]
}
```
"deblockFilter (p. 90)": enum,
"networkInputSettings (p. 90)": {
  "hlsInputSettings (p. 108)": {
    "retries (p. 82)": integer,
    "bandwidth (p. 82)": integer,
    "retryInterval (p. 82)": integer,
    "bufferSegments (p. 82)": integer,
  },
  "serverValidation (p. 108)": enum,
},
"inputFilter (p. 90)": enum,
"videoSelector (p. 90)": {
  "colorSpace (p. 122)": enum,
  "selectorSettings (p. 122)": {
    "videoSelectorPid (p. 123)": {
      "pid (p. 122)": integer
    },
    "videoSelectorProgramId (p. 123)": {
      "programId (p. 123)": integer
    }
  },
  "colorSpaceUsage (p. 122)": enum,
  "filterStrength (p. 90)": integer,
  "captionSelectors (p. 90)": [
    {
      "name (p. 47)": "string",
      "languageCode (p. 47)": "string",
      "selectorSettings (p. 47)": {
        "embeddedSourceSettings (p. 47)": {
          "scte20Detection (p. 62)": enum,
          "source608ChannelNumber (p. 62)": integer,
          "convert608To708 (p. 63)": enum,
          "source608TrackNumber (p. 63)": integer
        },
        "scte20SourceSettings (p. 47)": {
          "source608ChannelNumber (p. 114)": integer,
          "convert608To708 (p. 114)": enum
        },
        "dvbSubSourceSettings (p. 47)": {
          "pid (p. 57)": integer
        },
        "aribSourceSettings (p. 47)": {
        },
        "teletextSourceSettings (p. 47)": {
          "pageNumber (p. 118)": "string"
        },
        "scte27SourceSettings (p. 48)": {
          "pid (p. 114)": integer
        }
      }
    }
  ],
  "denoiseFilter (p. 90)": enum
},
"roleArn (p. 50)": "string",
"destinations (p. 50)": [
  {
    "settings (p. 109)": [
      {
        "passwordParam (p. 109)": "string",
        "streamName (p. 109)": "string",
        "url (p. 109)": "string",
        "username (p. 109)": "string"
      }
    ]
  }
]
Example CreateChannelResultModel

```
{
  "channel (p. 52)": {
    "inputAttachments (p. 48)": [
      {
        "inputId (p. 86)": "string",
        "inputSettings (p. 86)": {
          "sourceEndBehavior (p. 89)": enum,
          "audioSelectors (p. 89)": [
            {
              "name (p. 38)": "string",
              "selectorSettings (p. 38)": {
                "audioLanguageSelection (p. 38)": {
                  "languageSelectionPolicy (p. 36)": enum,
                  "languageCode (p. 36)": "string"
                },
                "audioPidSelection (p. 38)": {
                  "pid (p. 37)": integer
                }
              }
            }
          ],
          "deblockFilter (p. 90)": enum,
          "networkInputSettings (p. 90)": {
            "hlsInputSettings (p. 108)": {
              "retries (p. 82)": integer,
              "bandwidth (p. 82)": integer,
              "retryInterval (p. 82)": integer,
              "bufferSegments (p. 82)": integer
            },
            "serverValidation (p. 108)": enum
          }
        }
      }
    ],
    "networkInputSettings (p. 90)": {
      "hlsInputSettings (p. 108)": {
        "retries (p. 82)": integer,
        "bandwidth (p. 82)": integer,
        "retryInterval (p. 82)": integer,
        "bufferSegments (p. 82)": integer
      },
      "serverValidation (p. 108)": enum
    },
    "inputFilter (p. 90)": enum,
    "videoSelector (p. 90)": {
      "colorSpace (p. 122)": enum,
      "selectorSettings (p. 122)": {
        "videoSelectorPid (p. 123)": {
          "pid (p. 122)": integer
        }
      }
    }
  }
}
```
"videoSelectorProgramId (p. 123)": {
  "programId (p. 123)": integer
}
,"colorSpaceUsage (p. 122)": enum
,"filterStrength (p. 90)": integer,
"captionSelectors (p. 90)": [
  {
    "name (p. 47)": "string",
    "languageCode (p. 47)": "string",
    "selectorSettings (p. 47)": {
      "embeddedSourceSettings (p. 47)": {
        "scte20Detection (p. 62)": enum,
        "source608ChannelNumber (p. 62)": integer,
        "convert608To708 (p. 63)": enum,
        "source608TrackNumber (p. 63)": integer
      },
      "scte20SourceSettings (p. 47)": {
        "source608ChannelNumber (p. 114)": integer,
        "convert608To708 (p. 114)": enum
      },
      "dvbSubSourceSettings (p. 47)": {
        "pid (p. 57)": integer
      },
      "aribSourceSettings (p. 47)": {
      },
      "teletextSourceSettings (p. 47)": {
        "pageNumber (p. 118)": "string"
      },
      "scte27SourceSettings (p. 48)": {
        "pid (p. 114)": integer
      }
    }
  }
],
"denoiseFilter (p. 90)": enum
,"roleArn (p. 48)": "string",
"destinations (p. 48)": [
  {
    "settings (p. 109)": [
      {
        "passwordParam (p. 109)": "string",
        "streamName (p. 109)": "string",
        "url (p. 109)": "string",
        "username (p. 109)": "string"
      }
    ],
    "id (p. 109)": "string"
  }
],
"name (p. 48)": "string",
"encoderSettings (p. 48)": {
  "timecodeConfig (p. 63)": {
    "syncThreshold (p. 118)": integer,
    "source (p. 118)": enum
  },
  "outputGroups (p. 63)": [
    {
      "outputs (p. 110)": [
        "videoDescriptionName (p. 108)": "string",
        
```
"captionDescriptionNames (p. 108)": [
"string"
],
"outputName (p. 108)": "string",
"outputSettings (p. 108)": {
"rtmpOutputSettings (p. 111)": {
"certificateMode (p. 113)": enum,
"numRetries (p. 113)": integer,
"destination (p. 114)": {
"destinationRefId (p. 111)": "string"
},
"connectionRetryInterval (p. 114)": integer
},
"archiveOutputSettings (p. 111)": {
"extension (p. 32)": "string",
"containerSettings (p. 33)": {
"m2tsSettings (p. 32)": {
"audioStreamType (p. 94)": enum,
"ecmPid (p. 94)": "string",
"dvbTeletextPid (p. 94)": "string",
"aribCaptionsPidControl (p. 94)": enum,
"bitrate (p. 94)": integer,
"segmentationTime (p. 94)": number,
"rateMode (p. 95)": enum,
"audioPids (p. 95)": "string",
"fragmentTime (p. 95)": number,
"ebpAudioInterval (p. 95)": enum,
"ebpLookaheadMs (p. 95)": integer,
"audioFramesPerPes (p. 95)": integer,
"scte35Pid (p. 96)": "string",
"pcrPeriod (p. 96)": integer,
"pmtInterval (p. 96)": integer,
"programNum (p. 96)": integer,
"segmentationStyle (p. 96)": enum,
"ebif (p. 97)": enum,
"audioBufferModel (p. 97)": enum,
"dvbNitSettings (p. 97)": {
"networkName (p. 52)": "string",
"networkId (p. 52)": integer,
"repInterval (p. 52)": integer
},
"absentInputAudioBehavior (p. 97)": enum,
"timedMetadataPid (p. 97)": "string",
"timedMetadataBehavior (p. 97)": enum,
"etvSignalPid (p. 97)": "string",
"pmtPid (p. 98)": "string",
"bufferModel (p. 98)": enum,
"scte35Control (p. 98)": enum,
"ebpPlacement (p. 98)": enum,
"arib (p. 98)": enum,
"nullPacketBitrate (p. 98)": number,
"dvbSdtSettings (p. 98)": {
"serviceName (p. 53)": "string",
"serviceProviderName (p. 53)": "string",
"repInterval (p. 53)": integer,
"outputSdt (p. 53)": enum
},
"pcrPid (p. 99)": "string",
"transportStreamId (p. 99)": integer,
"pcrControl (p. 99)": enum,
"videoPid (p. 99)": "string",
"esRateInPes (p. 99)": enum,
"segmentationMarkers (p. 99)": enum,
"dvbTdtSettings (p. 100)": {
"repInterval (p. 57)": integer
},
"klv (p. 100)": enum,
"ccDescriptor (p. 100)": enum,
"patInterval (p. 100)": integer,
"etvPlatformPid (p. 100)": "string",
"dvbSubPids (p. 100)": "string",
"aribCaptionsPid (p. 100)": "string",
"scte27Pids (p. 101)": "string",
"klvDataPids (p. 101)": "string",
"nameModifier (p. 33)": "string",
"msSmoothOutputSettings (p. 111)": {
  "nameModifier (p. 107)": "string"
},
"udpOutputSettings (p. 111)": {
  "bufferMsec (p. 119)": integer,
  "destination (p. 120)": {
    "destinationRefId (p. 111)": "string"
  },
  "containerSettings (p. 120)": {
    "m2tsSettings (p. 119)": {
      "audioStreamType (p. 94)": enum,
      "ecmPid (p. 94)": "string",
      "dvbTeletextPid (p. 94)": "string",
      "aribCaptionsPidControl (p. 94)": enum,
      "bitrate (p. 94)": integer,
      "segmentationTime (p. 94)": number,
      "rateMode (p. 95)": enum,
      "audioPids (p. 95)": "string",
      "fragmentTime (p. 95)": number,
      "ebpAudioInterval (p. 95)": enum,
      "ebpLookaheadMs (p. 95)": integer,
      "audioFramesPerPES (p. 95)": integer,
      "scte35Pid (p. 96)": "string",
      "pcrPeriod (p. 96)": integer,
      "pmtInterval (p. 96)": integer,
      "programNum (p. 96)": integer,
      "segmentationStyle (p. 96)": enum,
      "ebif (p. 97)": enum,
      "audioBufferModel (p. 97)": enum,
      "dvbSdtSettings (p. 97)": {
        "networkName (p. 52)": "string",
        "networkId (p. 52)": integer,
        "repInterval (p. 52)": integer
      },
      "absentInputAudioBehavior (p. 97)": enum,
      "timeMetadataPid (p. 97)": "string",
      "timeMetadataBehavior (p. 97)": enum,
      "etvSignalPid (p. 97)": "string",
      "pmtPid (p. 98)": "string",
      "bufferModel (p. 98)": enum,
      "scte35Control (p. 98)": enum,
      "ebpPlacement (p. 98)": enum,
      "arib (p. 98)": enum,
      "nullPacketBitrate (p. 98)": number,
      "dvbSdtSettings (p. 98)": {
        "serviceName (p. 53)": "string",
        "serviceProviderName (p. 53)": "string",
        "repInterval (p. 53)": integer,
        "outputSdt (p. 53)": enum
      },
      "pcrPid (p. 99)": "string",
      "transportStreamId (p. 99)": integer,
      "pcrControl (p. 99)": enum,
      "videoPid (p. 99)": "string",
      "nullPacketBitrate (p. 98)": number,
"esRateInPes (p. 99)": enum,
"segmentationMarkers (p. 99)": enum,
"dvbTdtSettings (p. 100)": {
   "repInterval (p. 57)": integer
},
"klv (p. 100)": enum,
"ccDescriptor (p. 100)": enum,
"patInterval (p. 100)": integer,
"etvPlatformPid (p. 100)": "string",
"dvbSubPids (p. 100)": "string",
"aribCaptionsPid (p. 100)": "string",
"scte27Pids (p. 101)": "string",
"klvDataPids (p. 101)": "string"
},
"fecOutputSettings (p. 120)": {
   "rowLength (p. 64)": integer,
   "columnDepth (p. 64)": integer,
   "includeFec (p. 65)": enum
},
"hlsOutputSettings (p. 111)": {
   "segmentModifier (p. 84)": "string",
   "hlsSettings (p. 84)": {
      "audioOnlyHlsSettings (p. 85)": {
         "audioTrackType (p. 37)": enum,
         "audioGroupId (p. 37)": "string",
         "audioOnlyImage (p. 37)": {
            "passwordParam (p. 87)": "string",
            "uri (p. 87)": "string",
            "username (p. 88)": "string"
         }
      },
      "standardHlsSettings (p. 85)": {
         "m3u8Settings (p. 117)": {
            "pmtPid (p. 101)": "string",
            "ecmPid (p. 101)": "string",
            "scte35Behavior (p. 102)": enum,
            "pcrPid (p. 102)": "string",
            "audioPids (p. 102)": "string",
            "audioFramesPerPes (p. 102)": integer,
            "scte35Pid (p. 102)": "string",
            "videoPid (p. 102)": "string",
            "pcrControl (p. 103)": enum,
            "pmtInterval (p. 103)": integer,
            "pcrPeriod (p. 103)": integer,
            "programNum (p. 103)": integer,
            "patInterval (p. 103)": integer,
            "timedMetadataPid (p. 103)": "string",
            "timedMetadataBehavior (p. 104)": enum
         },
         "audioRenditionSets (p. 117)": "string"
      }
   },
   "nameModifier (p. 84)": "string"
},
"audioDescriptionNames (p. 109)": ["string"
],
"outputGroupSettings (p. 110)": {
   "archiveGroupSettings (p. 110)": {
      "destination (p. 32)": 
   },
"destinationRefId (p. 111)": "string"
},
"rolloverInterval (p. 32)
},
"rtmpGroupSettings (p. 110)
"captionData (p. 112)
"authenticationScheme (p. 112)
"cacheLength (p. 113)
"restartDelay (p. 113)
"cacheFullBehavior (p. 113)
},
"udpGroupSettings (p. 110)
"inputLossAction (p. 119)
"timedMetadataId3Frame (p. 119)
"timedMetadataId3Period (p. 119)
"msSmoothGroupSettings (p. 110)
"eventId (p. 104)
"fragmentLength (p. 105)
"timestampOffset (p. 105)
"segmentationMode (p. 105)
"numRetries (p. 105)
"eventStopBehavior (p. 105)
"acquisitionPointId (p. 105)
"sparseTrackType (p. 105)
"cacheFullBehavior (p. 113)
"cacheLength (p. 113)
"restartDelay (p. 113)
"cacheFullBehavior (p. 113)
"destination (p. 106)
"destinationRefId (p. 111)
"audioOnlyTimecodeControl (p. 106)
"connectionRetryInterval (p. 106)
"filecacheDuration (p. 106)
"certificateMode (p. 106)
"inputLossAction (p. 106)
"sendDelayMs (p. 107)
"eventIdMode (p. 107)
"restartDelay (p. 107)
"streamManifestBehavior (p. 107)
"hlsGroupSettings (p. 110)
"segmentsPerSubdirectory (p. 77)
"ivInManifest (p. 77)
"outputSelection (p. 77)
"encryptionType (p. 77)
"indexNSegments (p. 77)
"destination (p. 77)
"destinationRefId (p. 111)
"constantIv (p. 78)
"timedMetadataId3Frame (p. 78)
"baseUrlManifest (p. 78)
"captionLanguageSetting (p. 78)
"minSegmentLength (p. 78)
"mode (p. 78)
"ivSource (p. 79)
"manifestCompression (p. 79)
"keyProviderSettings (p. 79)
"staticKeySettings (p. 91)
"staticKeyValue (p. 117)
"keyProviderServer (p. 117)
"passwordParam (p. 87)
"uri (p. 87)
"username (p. 88)
"tsFileNotFoundException (p. 79)": enum,
"manifestDurationFormat (p. 79)": enum,
"keyFormatVersions (p. 79)": "string",
"streamInfResolution (p. 79)": enum,
"timestampDeltaMilliseconds (p. 80)": integer,
"segmentationMode (p. 80)": enum,
"baseUriContent (p. 80)": "string",
"clientCache (p. 80)": enum,
"captionLanguageMappings (p. 80)": [
  {
    "languageDescription (p. 46)": "string",
    "captionChannel (p. 46)": integer,
    "languageCode (p. 46)": "string"
  }
],
"codecSpecification (p. 80)": enum,
"keepSegments (p. 80)": integer,
"timedMetadataId3Period (p. 81)": integer,
"programDateTime (p. 81)": enum,
"directoryStructure (p. 81)": enum,
"keyFormat (p. 81)": "string",
"inputLossAction (p. 81)": enum,
"adMarkers (p. 81)": [
  enum
],
"programDateTimePeriod (p. 81)": integer,
"segmentLength (p. 82)": integer,
"hlsCdnSettings (p. 82)": {
  "hlsAkamaiSettings (p. 76)": {
    "httpTransferMode (p. 74)": enum,
    "salt (p. 74)": "string",
    "numRetries (p. 74)": integer,
    "restartDelay (p. 74)": integer,
    "connectionRetryInterval (p. 75)": integer,
    "filecacheDuration (p. 75)": integer,
    "token (p. 75)": "string"
  },
  "hlsWebdavSettings (p. 76)": {
    "httpTransferMode (p. 85)": enum,
    "numRetries (p. 85)": integer,
    "restartDelay (p. 86)": integer,
    "connectionRetryInterval (p. 86)": integer,
    "filecacheDuration (p. 86)": integer
  },
  "hlsBasicPutSettings (p. 76)": {
    "numRetries (p. 75)": integer,
    "restartDelay (p. 75)": integer,
    "connectionRetryInterval (p. 75)": integer,
    "filecacheDuration (p. 76)": integer
  },
  "hlsMediaStoreSettings (p. 76)": {
    "mediaStoreStorageClass (p. 83)": enum,
    "numRetries (p. 83)": integer,
    "restartDelay (p. 83)": integer,
    "connectionRetryInterval (p. 83)": integer,
    "filecacheDuration (p. 84)": integer
  }
},
"name (p. 110)": "string"
],
"audioDescriptions (p. 63)": [
  {
    "languageCodeControl (p. 34)": enum,
"audioTypeControl (p. 34)" : enum,
"remixSettings (p. 34)" : {
  "channelMappings (p. 111)" : [
    {
      "outputChannel (p. 33)" : integer,
      "inputChannelLevels (p. 33)" : [
        {
          "inputChannel (p. 86)" : integer,
          "gain (p. 87)" : integer
        }
      ]
    }
  ],
  "channelsOut (p. 111)" : integer,
  "channelsIn (p. 112)" : integer,
},
"audioType (p. 34)" : enum,
"name (p. 35)" : "string",
"languageCode (p. 35)" : "string",
"codecSettings (p. 35)" : {
  "aacSettings (p. 33)" : {
    "vbrQuality (p. 28)" : enum,
    "codingMode (p. 29)" : enum,
    "profile (p. 29)" : enum,
    "bitrate (p. 29)" : number,
    "inputType (p. 29)" : enum,
    "rawFormat (p. 29)" : enum,
    "sampleRate (p. 29)" : number,
    "rateControlMode (p. 29)" : enum,
    "spec (p. 30)" : enum
  },
  "ac3Settings (p. 33)" : {
    "drcProfile (p. 31)" : enum,
    "dialnorm (p. 31)" : integer,
    "codingMode (p. 31)" : enum,
    "metadataControl (p. 31)" : enum,
    "bitrate (p. 31)" : number,
    "lfeFilter (p. 31)" : enum,
    "bitstreamMode (p. 32)" : enum
  },
  "eaac3Settings (p. 34)" : {
    "dialnorm (p. 59)" : integer,
    "passthroughControl (p. 59)" : enum,
    "metadataControl (p. 59)" : enum,
    "drcLine (p. 59)" : enum,
    "bitrate (p. 60)" : number,
    "surroundExMode (p. 60)" : enum,
    "lfeRtSurroundMixLevel (p. 60)" : number,
    "lfeControl (p. 60)" : enum,
    "codingMode (p. 60)" : enum,
    "surroundMode (p. 60)" : enum,
    "attenuationControl (p. 60)" : enum,
    "lfeFilter (p. 60)" : enum,
    "lfeRtCenterMixLevel (p. 61)" : number,
    "dcFilter (p. 61)" : enum,
    "phaseControl (p. 61)" : enum,
    "stereoDownmix (p. 61)" : enum,
    "bitstreamMode (p. 61)" : enum,
    "loRoSurroundMixLevel (p. 61)" : number,
    "drcRf (p. 61)" : enum,
    "loRoCenterMixLevel (p. 62)" : number
  },
  "passThroughSettings (p. 34)" : {
  },
  "mp2Settings (p. 34)" : {
    "codingMode (p. 104)" : enum,
"bitrate (p. 104)" : number,
"sampleRate (p. 104)" : number
},
"streamName (p. 35)" : "string",
"audioNormalizationSettings (p. 35)" : {
  "targetLkfs (p. 36)" : number,
  "algorithmControl (p. 36)" : enum,
  "algorithm (p. 37)" : enum
},
"audioSelectorName (p. 35)" : "string"
},

"captionDescriptions (p. 63)" : [
  {
    "captionSelectorName (p. 44)" : "string",
    "languageDescription (p. 44)" : "string",
    "name (p. 44)" : "string",
    "languageCode (p. 45)" : "string",
    "destinationSettings (p. 45)" : {
      "scte27DestinationSettings (p. 45)" : {
      },
      "burnInDestinationSettings (p. 45)" : {
      
    },
      "teletextDestinationSettings (p. 45)" : {
      },
      "smpteTtDestinationSettings (p. 45)" : {
      },
      "webvttDestinationSettings (p. 45)" : {
      },
      "ttmlDestinationSettings (p. 45)" : {
      "styleControl (p. 118)" : enum
      }
    },
    "embeddedPlusScte20DestinationSettings (p. 45)" : {
    },
    "dvbSubDestinationSettings (p. 46)" : {
    "xPosition (p. 54)" : integer,
    "backgroundColor (p. 54)" : enum,
    "yPosition (p. 54)" : integer,
    "teletextGridControl (p. 55)" : enum,
    "backgroundColor (p. 55)" : integer,
    "fontOpacity (p. 55)" : integer,
    "fontResolution (p. 55)" : integer,
    "shadowOpacity (p. 55)" : integer,
    "shadowYOffset (p. 55)" : integer,
    "outlineSize (p. 55)" : integer,
    "outlineColor (p. 55)" : enum,
    "fontSize (p. 55)" : integer,
    "shadowXOffset (p. 55)" : integer,
    "alignment (p. 55)" : enum,
    "shadowColor (p. 55)" : enum,
    "fontColor (p. 55)" : enum,
    "font (p. 55)" : {
    "passwordParam (p. 87)" : "string",
    "uri (p. 87)" : "string",
    "username (p. 88)" : "string"
    }
  },
  "teletextTtDestinationSettings (p. 45)" : {
  },
  "smpteTtDestinationSettings (p. 45)" : {
  },
  "webvttDestinationSettings (p. 45)" : {
  },
  "ttmlDestinationSettings (p. 45)" : {
  "styleControl (p. 118)" : enum
  }
},
"embeddedPlusScte20DestinationSettings (p. 45)" : {
},
"dvbSubDestinationSettings (p. 46)" : {
"xPosition (p. 54)" : integer,
"backgroundColor (p. 54)" : enum,
"yPosition (p. 54)" : integer,
"teletextGridControl (p. 55)" : enum,
"backgroundColor (p. 55)" : integer,
"fontOpacity (p. 55)" : integer,
"fontResolution (p. 55)" : integer,
"shadowOpacity (p. 55)" : integer,
"shadowYOffset (p. 55)" : integer,
"outlineSize (p. 56)": integer,
"outlineColor (p. 56)": enum,
"fontSize (p. 56)": "string",
"shadowXOffset (p. 56)": integer,
"alignment (p. 56)": enum,
"shadowColor (p. 56)": enum,
"foregroundColor (p. 57)": enum,
"font (p. 57)": {
  "passwordParam (p. 87)": "string",
  "uri (p. 87)": "string",
  "username (p. 88)": "string"
}
},
"embeddedDestinationSettings (p. 46)": {
},
"rtmpCaptionInfoDestinationSettings (p. 46)": {
},
"sribDestinationSettings (p. 46)": {
},
"scte20PlusEmbeddedDestinationSettings (p. 46)": {
}
],
"availConfiguration (p. 63)": {
  "availSettings (p. 39)": {
    "scte35TimeSignalApos (p. 39)": {
      "adAvailOffset (p. 116)": integer,
      "webDeliveryAllowedFlag (p. 116)": enum,
      "noRegionalBlackoutFlag (p. 116)": enum
    },
    "scte35SpliceInsert (p. 39)": {
      "adAvailOffset (p. 115)": integer,
      "webDeliveryAllowedFlag (p. 115)": enum,
      "noRegionalBlackoutFlag (p. 115)": enum
    }
  }
},
"globalConfiguration (p. 64)": {
  "inputLossBehavior (p. 65)": {
    "inputLossImageType (p. 88)": enum,
    "inputLossImageColor (p. 88)": "string",
    "inputLossImageSlate (p. 88)": {
      "passwordParam (p. 87)": "string",
      "uri (p. 87)": "string",
      "username (p. 88)": "string"
    },
    "blackFrameMsec (p. 89)": integer,
    "repeatFrameMsec (p. 89)": integer
  },
  "supportLowFramerateInputs (p. 65)": enum,
  "initialAudioGain (p. 65)": integer,
  "inputEndAction (p. 66)": enum,
  "outputTimingSource (p. 66)": enum
},
"videoDescriptions (p. 64)": [
  {
    "respondToAfd (p. 120)": enum,
    "scalingBehavior (p. 121)": enum,
    "name (p. 121)": "string",
    "width (p. 121)": integer,
    "sharpness (p. 121)": integer,
    "codecSettings (p. 121)": {
      "h264Settings (p. 120)": {
        "minInterval (p. 68)": integer,
        "slices (p. 68)": integer,
        "numRefFrames (p. 68)": integer,
        "maxBitrate (p. 68)": integer,
        "maxFramerate (p. 68)": integer
      }
    }
  }
]
"parNumerator (p. 69)": integer,
"gopSizeUnits (p. 69)": enum,
"maxBitrate (p. 69)": integer,
"bitrate (p. 69)": integer,
"bufFillPct (p. 69)": integer,
"temporalAq (p. 69)": enum,
"afdSignaling (p. 70)": enum,
"timecodeInsertion (p. 70)": enum,
"bufSize (p. 70)": integer,
"softness (p. 70)": integer,
"framerateControl (p. 70)": enum,
"fixedAfd (p. 70)": enum,
"level (p. 70)": enum,
"lookAheadRateControl (p. 71)": enum,
"profile (p. 71)": enum,
"framerateNumerator (p. 71)": integer,
"gopClosedCadence (p. 71)": integer,
"framerateDenominator (p. 71)": integer,
"entropyEncoding (p. 71)": enum,
"spatialAq (p. 71)": enum,
"adaptiveQuantization (p. 72)": enum,
"colorMetadata (p. 72)": enum,
"gopSize (p. 72)": number,
"numRefFrames (p. 72)": integer,
"gopBReference (p. 72)": enum,
"sceneChangeDetect (p. 72)": enum,
"parControl (p. 72)": enum,
"parDenominator (p. 73)": integer,
"syntax (p. 73)": enum,
"scanType (p. 73)": enum,
"gopNumBFrames (p. 73)": integer,
"flickerAq (p. 73)": enum,
"rateControlMode (p. 73)": enum
"height (p. 121)": integer
}
Example InvalidRequest

```json
{
  "message (p. 91)": "string"
}
```

Example AccessDenied

```json
{
  "message (p. 32)": "string"
}
```

Example ResourceConflict

```json
{
  "message (p. 112)": "string"
}
```

Example ChannelConfigurationValidationError

```json
{
  "validationErrors (p. 49)": [
    {
      "errorMessage (p. 120)": "string",
      "elementPath (p. 120)": "string"
    }
  ],
  "message (p. 49)": "string"
}
```

Example LimitExceeded

```json
{
  "message (p. 92)": "string"
}
```

Example InternalServiceError

```json
{
  "message (p. 91)": "string"
}
```
Example BadGatewayException

```json
{
  "message (p. 39)": "string"
}
```

Example GatewayTimeoutException

```json
{
  "message (p. 65)": "string"
}
```

Properties

**AacCodingMode (enum)**

- AD_RECEIVER_MIX
- CODING_MODE_1_0
- CODING_MODE_1_1
- CODING_MODE_2_0
- CODING_MODE_5_1

**AacInputType (enum)**

- BROADCASTER_MIXED_AD
- NORMAL

**AacProfile (enum)**

- HEV1
- HEV2
- LC

**AacRateControlMode (enum)**

- CBR
- VBR

**AacRawFormat (enum)**

- LATM_LOAS
- NONE

**AacSettings**

**vbrQuality**

VBR Quality Level - Only used if rateControlMode is VBR.

- Type: string
- Required: False
codingMode

Mono, Stereo, or 5.1 channel layout. Valid values depend on rate control mode and profile. The adReceiverMix 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: string
  Required: False

profile

AAC Profile.

  Type: string
  Required: False

bitrate

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

  Type: number
  Required: False

inputType

Set to "broadcasterMixedAd" when input contains pre-mixed main audio + AD (narration) as a stereo pair. The Audio Type field (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. The values in audioTypeControl and audioType (in AudioDescription) are ignored when set to broadcasterMixedAd. Leave set to "normal" when input does not contain pre-mixed audio + AD.

  Type: string
  Required: False

rawFormat

Sets LATM / LOAS AAC output for raw containers.

  Type: string
  Required: False

sampleRate

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

  Type: number
  Required: False

rateControlMode

Rate Control Mode.

  Type: string
Required: False

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

Type: string
Required: False

AacSpec (enum)
MPEG2
MPEG4

AacVbrQuality (enum)
HIGH
LOW
MEDIUM_HIGH
MEDIUM_LOW

Ac3BitstreamMode (enum)
COMMENTARY
COMPLETE_MAIN
DIALOGUE
EMERGENCY
HEARING_IMPAIRED
MUSIC_AND_EFFECTS
VISUALLY_IMPAIRED
VOICE_OVER

Ac3CodingMode (enum)
CODING_MODE_1_0
CODING_MODE_1_1
CODING_MODE_2_0
CODING_MODE_3_2_LFE

Ac3DrcProfile (enum)
FILM_STANDARD
NONE

Ac3LfeFilter (enum)
DISABLED
ENABLED

Ac3MetadataControl (enum)
FOLLOW_INPUT
USE_CONFIGURED

**Ac3Settings**

**drcProfile**

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

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

**dialnorm**

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

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

**codingMode**

Dolby Digital coding mode. Determines number of channels.

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

**metadataControl**

When set to "followInput", 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**: string
- **Required**: False

**bitrate**

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

- **Type**: number
- **Required**: False

**lfeFilter**

When set to enabled, applies a 120Hz lowpass filter to the LFE channel prior to encoding. Only valid in codingMode32Lfe mode.

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

bitstreamMode

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

  Type: string
  Required: False

AccessDenied

message

  Type: string
  Required: False

AfdSignaling (enum)

  AUTO
  FIXED
  NONE

ArchiveContainerSettings

m2tsSettings

  Type: M2tsSettings (p. 94)
  Required: False

ArchiveGroupSettings

destination

A directory and base filename where archive files should be written. If the base filename portion of the URI is left blank, the base filename of the first input will be automatically inserted.

  Type: OutputLocationRef (p. 111)
  Required: True

rolloverInterval

Number of seconds to write to archive file before closing and starting a new one.

  Type: integer
  Required: False
  Minimum: 1

ArchiveOutputSettings

extension

Output file extension. If excluded, this will be auto-selected from the container type.
Properties

Type: string
Required: False

containerSettings
Settings specific to the container type of the file.

Type: ArchiveContainerSettings (p. 32)
Required: True

nameModifier
String concatenated to the end of the destination filename. Required for multiple outputs of the same type.

Type: string
Required: False

AribDestinationSettings

AribSourceSettings

AudioChannelMapping

outputChannel
The index of the output channel being produced.

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

inputChannelLevels
Indices and gain values for each input channel that should be remixed into this output channel.

Type: Array of type InputChannelLevel (p. 86)
Required: True

AudioCodecSettings

aacSettings

Type: AacSettings (p. 28)
Required: False

ac3Settings

Type: Ac3Settings (p. 31)
**Properties**

**Required**: False

**eac3Settings**
- **Type**: Eac3Settings (p. 59)
- **Required**: False

**passThroughSettings**
- **Type**: PassThroughSettings (p. 111)
- **Required**: False

**mp2Settings**
- **Type**: Mp2Settings (p. 104)
- **Required**: False

**AudioDescription**

**languageCodeControl**
Choosing `followInput` will cause the ISO 639 language code of the output to follow the ISO 639 language code of the input. The `languageCode` will be used when `useConfigured` is set, or when `followInput` is selected but there is no ISO 639 language code specified by the input.

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

**audioTypeControl**
Determines how audio type is determined. `followInput`: If the input contains an ISO 639 audioType, then that value is passed through to the output. If the input contains no ISO 639 audioType, the value in Audio Type is included in the output. `useConfigured`: The value in Audio Type is included in the output. Note that this field and audioType are both ignored if `inputType` is `broadcasterMixedAd`.

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

**remixSettings**
Settings that control how input audio channels are remixed into the output audio channels.

- **Type**: RemixSettings (p. 111)
- **Required**: False

**audioType**
Applies only if `audioTypeControl` is `useConfigured`. The values for `audioType` are defined in ISO-IEC 13818-1.

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

The name of this AudioDescription. Outputs will use this name to uniquely identify this AudioDescription. Description names should be unique within this Live Event.

  Type: string
  Required: True

languageCode

Indicates the language of the audio output track. Only used if languageControlMode is useConfigured, or there is no ISO 639 language code specified in the input.

  Type: string
  Required: False

codecSettings

Audio codec settings.

  Type: AudioCodecSettings (p. 33)
  Required: False

streamName

Used for MS Smooth and Apple HLS outputs. Indicates the name displayed by the player (eg. English, or Director Commentary).

  Type: string
  Required: False

audioNormalizationSettings

Advanced audio normalization settings.

  Type: AudioNormalizationSettings (p. 36)
  Required: False

audioSelectorName

The name of the AudioSelector used as the source for this AudioDescription.

  Type: string
  Required: True

AudioDescriptionAudioTypeControl (enum)

  FOLLOW_INPUT
  USE_CONFIGURED

AudioDescriptionLanguageCodeControl (enum)

  FOLLOW_INPUT
  USE_CONFIGURED
AudioLanguageSelection

languageSelectionPolicy
When set to "strict", the transport stream demux strictly identifies audio streams by their language descriptor. If a PMT update occurs such that an audio stream matching the initially selected language is no longer present then mute will be encoded until the language returns. If "loose", then on a PMT update the demux will choose another audio stream in the program with the same stream type if it can't find one with the same language.

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

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

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

AudioLanguageSelectionPolicy (enum)
- LOOSE
- STRICT

AudioNormalizationAlgorithm (enum)
- ITU_1770_1
- ITU_1770_2

AudioNormalizationAlgorithmControl (enum)
- CORRECT_AUDIO

AudioNormalizationSettings

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

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

algorithmControl
When set to correctAudio the output audio is corrected using the chosen algorithm. If set to measureOnly, the audio will be measured but not adjusted.

- **Type**: string
- **Required**: False
algorithm
Audio normalization algorithm to use. itu17701 conforms to the CALM Act specification, itu17702 conforms to the EBU R-128 specification.

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

### AudioOnlyHlsSettings

#### 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, 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**: string  
**Required**: False

#### audioGroupId

Specifies the group to which the audio Rendition belongs.

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

#### audioOnlyImage

For use with an audio only Stream. Must be a .jpg or .png file. If given, this image will be used as the cover-art for the audio only output. Ideally, it should be formatted for an iPhone screen for two reasons. The iPhone does not resize the image, it crops a centered image on the top/bottom and left/right. Additionally, this image file gets saved bit-for-bit into every 10-second segment file, so will increase bandwidth by {image file size} * {segment count} * {user count}.

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

#### AudioOnlyHlsTrackType (enum)

- ALTERNATE_AUDIO_AUTO_SELECT
- ALTERNATE_AUDIO_AUTO_SELECT_DEFAULT
- ALTERNATE_AUDIO_NOT_AUTO_SELECT
- AUDIO_ONLY_VARIANT_STREAM

### AudioPidSelection

#### pid

Selects a specific PID from within a source.
Properties

**Type**: integer
**Required**: True
**Minimum**: 0
**Maximum**: 8191

**AudioSelector**

**name**

The name of this AudioSelector. AudioDescriptions will use this name to uniquely identify this Selector. Selector names should be unique per input.

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

**selectorSettings**

The audio selector settings.

**Type**: AudioSelectorSettings (p. 38)
**Required**: False

**AudioSelectorSettings**

**audioLanguageSelection**

**Type**: AudioLanguageSelection (p. 36)
**Required**: False

**audioPidSelection**

**Type**: AudioPidSelection (p. 37)
**Required**: False

**AudioType (enum)**

- CLEAN_EFFECTS
- HEARING_IMPAIRED
- UNDEFINED
- VISUAL_IMPAIRED_COMMENTARY

**AuthenticationScheme (enum)**

- AKAMAI
- COMMON

**AvailBlanking**

**state**

When set to enabled, causes video, audio and captions to be blanked when insertion metadata is added.
null

Type: string
Required: False

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

Type: InputLocation (p. 87)
Required: False

_AvailBlankingState (enum)_

DISABLED
ENABLED

_AvailConfiguration_

_availSettings_
Ad avail settings.

Type: AvailSettings (p. 39)
Required: False

_AvailSettings_

_scte35TimeSignalApos_

Type: Scte35TimeSignalApos (p. 116)
Required: False

_scte35SpliceInsert_

Type: Scte35SpliceInsert (p. 115)
Required: False

_BadGatewayException_

_message_

Type: string
Required: False

_BlackoutSlate_

_networkEndBlackoutImage_
Path to local file to use as Network End Blackout image. Image will be scaled to fill the entire output raster.

Type: InputLocation (p. 87)
Required: False

**networkEndBlackout**

Setting to enabled causes the encoder to blackout the video, audio, and captions, and raise the "Network Blackout Image" slate when an SCTE104/35 Network End Segmentation Descriptor is encountered. The blackout will be lifted when the Network Start Segmentation Descriptor is encountered. The Network End and Network Start descriptors must contain a network ID that matches the value entered in "Network ID".

Type: string

Required: False

**networkId**

Provides Network ID that matches EIDR ID format (e.g., "10.XXXX/XXXX-XXXX-XXXX-XXXX-XXXX-C").

Type: string

Required: False

**state**

When set to enabled, causes video, audio and captions to be blanked when indicated by program metadata.

Type: string

Required: False

**blackoutSlateImage**

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

Type: InputLocation (p. 87)

Required: False

**BlackoutSlateNetworkEndBlackout (enum)**

DISABLED
ENABLED

**BlackoutSlateState (enum)**

DISABLED
ENABLED

**BurnInAlignment (enum)**

CENTERED
LEFT
SMART

**BurnInBackgroundColor (enum)**

BLACK
**BurnInDestinationSettings**

**xPosition**

Specifies the horizontal position of the caption relative to the left side of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the left of the output. If no explicit xPosition is provided, the horizontal caption position will be determined by the alignment parameter. All burn-in and DVB-Sub font settings must match.

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

**backgroundColor**

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

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

**yPosition**

Specifies the vertical position of the caption relative to the top of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the top of the output. If no explicit yPosition is provided, the caption will be positioned towards the bottom of the output. All burn-in and DVB-Sub font settings must match.

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

**teletextGridControl**

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

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

**backgroundOpacity**

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

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

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

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

fontResolution

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

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

shadowOpacity

Specifies the opacity of the shadow. 255 is opaque; 0 is transparent. Leaving this parameter out 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

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

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

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.
### Properties

- **type**: `string`  
  **required**: `false`

- **fontSize**
  
  When set to 'auto' fontSize will scale depending on the size of the output. Giving a positive integer will specify the exact font size in points. All burn-in and DVB-Sub font settings must match.
  
  - **type**: `string`  
  - **required**: `false`

- **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`

- **alignment**
  
  If no explicit xPosition or yPosition 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. Selecting "smart" justification will left-justify live subtitles and center-justify pre-recorded subtitles. All burn-in and DVB-Sub font settings must match.
  
  - **type**: `string`  
  - **required**: `false`

- **shadowColor**
  
  Specifies the color of the shadow cast by the captions. All burn-in and DVB-Sub font settings must match.
  
  - **type**: `string`  
  - **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**: `string`  
  - **required**: `false`

- **font**
  
  External font file used for caption burn-in. File extension must be 'ttf' or 'tte'. Although the user can select output fonts for many different types of input captions, embedded, STL and teletext sources use a strict grid system. Using external fonts with these caption sources could cause unexpected display of proportional fonts. All burn-in and DVB-Sub font settings must match.
Properties

Type: InputLocation (p. 87)  
Required: False

BurnInFontColor (enum)

BLACK  
BLUE  
GREEN  
RED  
WHITE  
YELLOW

BurnInOutlineColor (enum)

BLACK  
BLUE  
GREEN  
RED  
WHITE  
YELLOW

BurnInShadowColor (enum)

BLACK  
NONE  
WHITE

BurnInTeletextGridControl (enum)

FIXED  
SCALED

CaptionDescription

captionSelectorName

Specifies which input caption selector to use as a caption source when generating output captions. This field should match a captionSelector name.

Type: string  
Required: True

languageDescription

Human readable information to indicate captions available for players (eg. English, or Spanish).

Type: string  
Required: False

name

Name of the caption description. Used to associate a caption description with an output. Names must be unique within an event.
Type: string  
Required: True

### languageCode

Type: string  
Required: False

### destinationSettings
Additional settings for captions destination that depend on the destination type.

Type: CaptionDestinationSettings (p. 45)  
Required: False

#### CaptionDestinationSettings

### scte27DestinationSettings

Type: Scte27DestinationSettings (p. 114)  
Required: False

### burnInDestinationSettings

Type: BurnInDestinationSettings (p. 41)  
Required: False

### teletextDestinationSettings

Type: TeletextDestinationSettings (p. 118)  
Required: False

### smpteTtDestinationSettings

Type: SmpteTtDestinationSettings (p. 117)  
Required: False

### webvttDestinationSettings

Type: WebvttDestinationSettings (p. 123)  
Required: False

### ttmlDestinationSettings

Type: TtmlDestinationSettings (p. 118)  
Required: False

### embeddedPlusScte20DestinationSettings

Type: EmbeddedPlusScte20DestinationSettings (p. 62)
**Required:** False

**dvbSubDestinationSettings**

*Type: DvbSubDestinationSettings (p. 54)*  
*Required: False*

**embeddedDestinationSettings**

*Type: EmbeddedDestinationSettings (p. 62)*  
*Required: False*

**rtmpCaptionInfoDestinationSettings**

*Type: RtmpCaptionInfoDestinationSettings (p. 112)*  
*Required: False*

**aribDestinationSettings**

*Type: AribDestinationSettings (p. 33)*  
*Required: False*

**scte20PlusEmbeddedDestinationSettings**

*Type: Scte20PlusEmbeddedDestinationSettings (p. 114)*  
*Required: False*

### CaptionLanguageMapping

**languageDescription**

Textual description of language  
*Type: string*  
*Required: True*

**captionChannel**

Channel to insert closed captions. Each channel mapping must have a unique channel number (maximum of 4)  
*Type: integer*  
*Required: True*  
*Minimum: 1*  
*Maximum: 4*

**languageCode**

Three character ISO 639-2 language code (see http://www.loc.gov/standards/iso639-2)  
*Type: string*
CaptionSelector

name
Name identifier for a caption selector. This name is used to associate this caption selector with one or more caption descriptions. Names must be unique within an event.

Type: string
Required: True

languageCode
When specified this field indicates the three letter language code of the caption track to extract from the source.

Type: string
Required: False

selectorSettings
Caption selector settings.

Type: CaptionSelectorSettings (p. 47)
Required: False

CaptionSelectorSettings

embeddedSourceSettings

Type: EmbeddedSourceSettings (p. 62)
Required: False

scte20SourceSettings

Type: Scte20SourceSettings (p. 114)
Required: False

dvbSubSourceSettings

Type: DvbSubSourceSettings (p. 57)
Required: False

aribSourceSettings

Type: AribSourceSettings (p. 33)
Required: False

teletextSourceSettings

Type: TeletextSourceSettings (p. 118)
Required: False

scte27SourceSettings

Type: Scte27SourceSettings (p. 114)
Required: False

Channel

inputAttachments

List of input attachments for channel.

Type: Array of type InputAttachment (p. 86)
Required: False

roleArn

The Amazon Resource Name (ARN) of the role assumed when running the Channel.

Type: string
Required: False

destinations

A list of destinations of the channel. For UDP outputs, there is one destination per output. For other types (HLS, for example), there is one destination per packager.

Type: Array of type OutputDestination (p. 109)
Required: False

name

The name of the channel. (user-mutable)

Type: string
Required: False

encoderSettings

Type: EncoderSettings (p. 63)
Required: False

pipelinesRunningCount

The number of currently healthy pipelines.

Type: integer
Required: False

state

Type: string
**Required**: False

**id**

The unique id of the channel.

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

**egressEndpoints**

The endpoints where outgoing connections initiate from

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

**arn**

The unique arn of the channel.

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

**inputSpecification**

- **Type**: [InputSpecification](#)
- **Required**: False

**ChannelConfigurationValidationError**

**validationErrors**

A collection of validation error responses from attempting to create a channel with a bouquet of settings.

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

**message**

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

**ChannelEgressEndpoint**

**sourceIp**

Public IP of where a channel's output comes from

- **Type**: string
- **Required**: False
**ChannelState (enum)**

- CREATING
- CREATE_FAILED
- IDLE
- STARTING
- RUNNING
- RECOVERING
- STOPPING
- DELETING
- DELETED

**ChannelSummary**

**inputAttachments**

List of input attachments for channel.

- **Type**: Array of type `InputAttachment (p. 86)`
- **Required**: False

**roleArn**

The Amazon Resource Name (ARN) of the role assumed when running the Channel.

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

**destinations**

A list of destinations of the channel. For UDP outputs, there is one destination per output. For other types (HLS, for example), there is one destination per packager.

- **Type**: Array of type `OutputDestination (p. 109)`
- **Required**: False

**name**

The name of the channel. (user-mutable)

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

**state**

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

**pipelinesRunningCount**

The number of currently healthy pipelines.

- **Type**: integer
- **Required**: False
id
The unique id of the channel.

  Type: string
  Required: False

egressEndpoints
The endpoints where outgoing connections initiate from

  Type: Array of type ChannelEgressEndpoint (p. 49)
  Required: False

arn
The unique arn of the channel.

  Type: string
  Required: False

inputSpecification

  Type: InputSpecification (p. 91)
  Required: False

CreateChannel
inputAttachments
List of input attachments for channel.

  Type: Array of type InputAttachment (p. 86)
  Required: False

requestId
Unique request ID to be specified. This is needed to prevent retries from creating multiple resources.

  Type: string
  Required: False

roleArn
An optional Amazon Resource Name (ARN) of the role to assume when running the Channel.

  Type: string
  Required: False

destinations

  Type: Array of type OutputDestination (p. 109)
  Required: False
name
Name of channel.

Type: string
Required: False

encoderSettings

Type: EncoderSettings (p. 63)
Required: False

inputSpecification
Specification of input for this channel (max. bitrate, resolution, codec, etc.)

Type: InputSpecification (p. 91)
Required: False

CreateChannelResultModel

channel

Type: Channel (p. 48)
Required: False

DvbNitSettings

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

Type: string
Required: True

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

Type: integer
Required: True
Minimum: 0
Maximum: 65536

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

Type: integer
Required: False
Minimum: 25
Maximum: 10000
DvbSdtOutputSdt (enum)

SDT_FOLLOW
SDT_FOLLOW_IF_PRESENT
SDT_MANUAL
SDT_NONE

DvbSdtSettings

serviceName

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

Type: string
Required: False

serviceProviderName

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

Type: string
Required: False

repInterval

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

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

outputSdt

Selects method of inserting SDT information into output stream. The sdtFollow setting copies SDT information from input stream to output stream. The sdtFollowIfPresent setting 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. The sdtManual setting means user will enter the SDT information. The sdtNone setting means output stream will not contain SDT information.

Type: string
Required: False

DvbSubDestinationAlignment (enum)

CENTERED
LEFT
SMART

DvbSubDestinationBackgroundColor (enum)

BLACK
NONE
WHITE

DvbSubDestinationFontColor (enum)
BLACK
BLUE
GREEN
RED
WHITE
YELLOW

DvbSubDestinationOutlineColor (enum)
BLACK
BLUE
GREEN
RED
WHITE
YELLOW

DvbSubDestinationSettings

xPosition

Specifies the horizontal position of the caption relative to the left side of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the left of the output. If no explicit xPosition 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

backgroundColor

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

Type: string
Required: False

yPosition

Specifies the vertical position of the caption relative to the top of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the top of the output. If no explicit yPosition 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
**teletextGridControl**

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

*Type: string*  
*Required: False*

**backgroundOpacity**

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

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

**fontOpacity**

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

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

**fontResolution**

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

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

**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*

**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*
**Properties**

**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

**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:** string

**Required:** False

**fontSize**
When set to auto fontSize will scale depending on the size of the output. Giving a positive integer will specify the exact font size in points. All burn-in and DVB-Sub font settings must match.

**Type:** string

**Required:** False

**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

**alignment**
If no explicit xPosition or yPosition 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. Selecting "smart" justification will left-justify live subtitles and center-justify pre-recorded subtitles. This option is not valid for source captions that are STL or 608/embedded. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

**Type:** string

**Required:** False

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

**fontColor**

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

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

**font**

External font file used for caption burn-in. File extension must be 'ttf' or 'tte'. Although the user can select output fonts for many different types of input captions, embedded, STL and teletext sources use a strict grid system. Using external fonts with these caption sources could cause unexpected display of proportional fonts. All burn-in and DVB-Sub font settings must match.

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

**DvbSubDestinationShadowColor (enum)**

- BLACK
- NONE
- WHITE

**DvbSubDestinationTeletextGridControl (enum)**

- FIXED
- SCALED

**DvbSubSourceSettings**

**pid**

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

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

**DvbTdtSettings**

**repInterval**

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

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

Eac3AttenuationControl (enum)

ATTENUATE_3_DB
NONE

Eac3BitstreamMode (enum)

COMMENTARY
COMPLETE_MAIN
EMERGENCY
HEARING_IMPAIRED
VISUALLY_IMPAIRED

Eac3CodingMode (enum)

CODING_MODE_1_0
CODING_MODE_2_0
CODING_MODE_3_2

Eac3DcFilter (enum)

DISABLED
ENABLED

Eac3DrcLine (enum)

FILM_LIGHT
FILM_STANDARD
MUSIC_LIGHT
MUSIC_STANDARD
NONE
SPEECH

Eac3DrcRf (enum)

FILM_LIGHT
FILM_STANDARD
MUSIC_LIGHT
MUSIC_STANDARD
NONE
SPEECH

Eac3LfeControl (enum)

LFE
NO_LFE

Eac3LfeFilter (enum)

DISABLED
ENABLED

Eac3MetadataControl (enum)

FOLLOW_INPUT
USE_CONFIGURED

Eac3PassthroughControl (enum)

NO_PASSTHROUGH
WHEN_POSSIBLE

Eac3PhaseControl (enum)

NO_SHIFT
SHIFT_90_DEGREES

Eac3Settings

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

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

passthroughControl

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

Type: string
Required: False

metadataControl

When set to followInput, 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: string
Required: False

drcLine

Sets the Dolby dynamic range compression profile.

Type: string
Required: False
bitrate
Average bitrate in bits/second. Valid bitrates depend on the coding mode.

    Type: number
    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: string
    Required: False

ltRtSurroundMixLevel
Left total/Right total surround mix level. Only used for 3/2 coding mode.

    Type: number
    Required: False

lfeControl
When encoding 3/2 audio, setting to lfe enables the LFE channel

    Type: string
    Required: False

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

    Type: string
    Required: False

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

    Type: string
    Required: False

attenuationControl
When set to attenuate3Db, applies a 3 dB attenuation to the surround channels. Only used for 3/2 coding mode.

    Type: string
    Required: False

lfeFilter
When set to enabled, applies a 120Hz lowpass filter to the LFE channel prior to encoding. Only valid with codingMode32 coding mode.
Type: string  
Required: False

ltRtCenterMixLevel
Left total/Right total center mix level. Only used for 3/2 coding mode.

  Type: number  
  Required: False

dcFilter
When set to enabled, activates a DC highpass filter for all input channels.

  Type: string  
  Required: False

phaseControl
When set to shift90Degrees, applies a 90-degree phase shift to the surround channels. Only used for 3/2 coding mode.

  Type: string  
  Required: False

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

  Type: string  
  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: string  
  Required: False

loRoSurroundMixLevel
Left only/Right only surround mix level. Only used for 3/2 coding mode.

  Type: number  
  Required: False

drcRf
Sets the profile for heavy Dolby dynamic range compression, ensures that the instantaneous signal peaks do not exceed specified levels.

  Type: string  
  Required: False
Properties

LoRoCenterMixLevel
Left only/Right only center mix level. Only used for 3/2 coding mode.

  Type: number
  Required: False

Eac3StereoDownmix (enum)

  DPL2
  LO_RO
  LT_RT
  NOT_INDICATED

Eac3SurroundExMode (enum)

  DISABLED
  ENABLED
  NOT_INDICATED

Eac3SurroundMode (enum)

  DISABLED
  ENABLED
  NOT_INDICATED

EmbeddedConvert608To708 (enum)

  DISABLED
  UPCONVERT

EmbeddedDestinationSettings

EmbeddedPlusScte20DestinationSettings

EmbeddedScte20Detection (enum)

  AUTO
  OFF

EmbeddedSourceSettings

scte20Detection
Set to "auto" to handle streams with intermittent and/or non-aligned SCTE-20 and Embedded captions.

  Type: string
  Required: False

source608ChannelNumber
Specifies the 608/708 channel number within the video track from which to extract captions. Unused for passthrough.
Properties

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

**convert608To708**
If 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**: string
- **Required**: False

**source608TrackNumber**
This field is unused and deprecated.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 5

**EncoderSettings**

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

- **Type**: TimecodeConfig (p. 118)
- **Required**: True

**outputGroups**
- **Type**: Array of type OutputGroup (p. 110)
- **Required**: True

**audioDescriptions**
- **Type**: Array of type AudioDescription (p. 34)
- **Required**: True

**captionDescriptions**
Settings for caption descriptions

- **Type**: Array of type CaptionDescription (p. 44)
- **Required**: False

**availConfiguration**
Event-wide configuration settings for ad avail insertion.

- **Type**: AvailConfiguration (p. 39)
Properties

**Required**: False

**globalConfiguration**

Configuration settings that apply to the event as a whole.

- **Type**: GlobalConfiguration (p. 65)
- **Required**: False

**videoDescriptions**

- **Type**: Array of type VideoDescription (p. 120)
- **Required**: True

**blackoutSlate**

Settings for blackout slate.

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

**availBlanking**

Settings for ad avail blanking.

- **Type**: AvailBlanking (p. 38)
- **Required**: False

**FecOutputIncludeFec (enum)**

- COLUMN
- COLUMN_AND_ROW

**FecOutputSettings**

**rowLength**

Parameter L from SMPTE 2022-1. The width of the FEC protection matrix. Must be between 1 and 20, inclusive. If only Column FEC is used, then larger values increase robustness. If Row FEC is used, then this is the number of transport stream packets per row error correction packet, and the value must be between 4 and 20, inclusive, if includeFec is columnAndRow. If includeFec is column, this value must be 1 to 20, inclusive.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
- **Maximum**: 20

**columnDepth**

Parameter D from SMPTE 2022-1. The height of the FEC protection matrix. The number of transport stream packets per column error correction packet. Must be between 4 and 20, inclusive.

- **Type**: integer
**Properties**

**includeFec**

Enables column only or column and row based FEC

*Type: string*
*Required: False*

**FixedAfd (enum)**

- AFD_0000
- AFD_0010
- AFD_0011
- AFD_0100
- AFD_1000
- AFD_1001
- AFD_1010
- AFD_1011
- AFD_1101
- AFD_1110
- AFD_1111

**GatewayTimeoutException**

**message**

*Type: string*
*Required: False*

**GlobalConfiguration**

**inputLossBehavior**

Settings for system actions when input is lost.

*Type: InputLossBehavior (p. 88)*
*Required: False*

**supportLowFramerateInputs**

Adjusts video input buffer for streams with very low video framerates. This is commonly set to enabled for music channels with less than one video frame per second.

*Type: string*
*Required: False*

**initialAudioGain**

Value to set the initial audio gain for the Live Event.
Properties

**Type**: integer  
**Required**: False  
**Minimum**: -60  
**Maximum**: 60

**inputEndAction**

Indicates the action to take when an input completes (e.g. end-of-file.) Options include immediately switching to the next sequential input (via "switchInput"), switching to the next input and looping back to the first input when last input ends (via "switchAndLoopInputs") or not switching inputs and instead transcoding black / color / slate images per the "Input Loss Behavior" configuration until an activateInput REST command is received (via "none").

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

**outputTimingSource**

Indicates whether the rate of frames emitted by the Live encoder should be paced by its system clock (which optionally may be locked to another source via NTP) or should be locked to the clock of the source that is providing the input stream.

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

**GlobalConfigurationInputEndAction** (enum)

- NONE
- SWITCH_AND_LOOP_INPUTS

**GlobalConfigurationLowFramerateInputs** (enum)

- DISABLED
- ENABLED

**GlobalConfigurationOutputTimingSource** (enum)

- INPUT_CLOCK
- SYSTEM_CLOCK

**H264AdaptiveQuantization** (enum)

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

**H264ColorMetadata** (enum)

- IGNORE
H264EntropyEncoding (enum)

CABAC
CAVLC

H264FlickerAq (enum)

DISABLED
ENABLED

H264FramerateControl (enum)

INITIALIZE_FROM_SOURCE
SPECIFIED

H264GopBReference (enum)

DISABLED
ENABLED

H264GopSizeUnits (enum)

FRAMES
SECONDS

H264Level (enum)

H264_LEVEL_1
H264_LEVEL_1_1
H264_LEVEL_1_2
H264_LEVEL_1_3
H264_LEVEL_2
H264_LEVEL_2_1
H264_LEVEL_2_2
H264_LEVEL_3
H264_LEVEL_3_1
H264_LEVEL_3_2
H264_LEVEL_4
H264_LEVEL_4_1
H264_LEVEL_4_2
H264_LEVEL_5
H264_LEVEL_5_1
H264_LEVEL_5_2
H264_LEVEL_AUTO

H264LookAheadRateControl (enum)

HIGH
LOW
MEDIUM

H264ParControl (enum)

INITIALIZE_FROM_SOURCE
SPECIFIED

H264Profile (enum)

BASELINE
HIGH
HIGH_10BIT
HIGH_422
HIGH_422_10BIT
MAIN

H264RateControlMode (enum)

CBR
VBR

H264ScanType (enum)

INTERLACED
PROGRESSIVE

H264SceneChangeDetect (enum)

DISABLED
ENABLED

H264Settings

minIInterval

Only meaningful if sceneChangeDetect is set to enabled. 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. Note: Maximum GOP stretch = GOP size + Min-I-interval - 1

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

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. This field is optional; when no value is specified the encoder will choose the number of slices based on encode resolution.
**parNumerator**

Pixel Aspect Ratio numerator.

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

**gopSizeUnits**

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

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

**maxBitrate**

Maximum bitrate in bits/second (for VBR mode only).

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

**bitrate**

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

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

**bufFillPct**

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

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

**temporalAq**

If set to enabled, adjust quantization within each frame based on temporal variation of content complexity.

- **Type**: string
- **Required**: False
**afdSignaling**
Indicates that AFD values will be written into the output stream. If afdSignaling is "auto", the system will try to preserve the input AFD value (in cases where multiple AFD values are valid). If set to "fixed", the AFD value will be the value configured in the fixedAfd parameter.

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

**timecodeInsertion**
Determines how timecodes should be inserted into the video elementary stream. - ‘disabled’: Do not include timecodes - 'picTimingSei': Pass through picture timing SEI messages from the source specified in Timecode Config

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

**bufSize**
Size of buffer (HRD buffer model) in bits/second.

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

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

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

**framerateControl**
This field indicates how the output video frame rate is specified. If "specified" is selected then the output video frame rate is determined by framerateNumerator and framerateDenominator, else if "initializeFromSource" is selected then the output video frame rate will be set equal to the input video frame rate of the first input.

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

**fixedAfd**
Four bit AFD value to write on all frames of video in the output stream. Only valid when afdSignaling is set to 'Fixed'.

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

**level**
H.264 Level.
Properties

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

**lookAheadRateControl**
Amount of lookahead. A value of low can decrease latency and memory usage, while high can produce better quality for certain content.

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

**profile**
H.264 Profile.

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

**framerateNumerator**
Framerate numerator - framerate is a fraction, e.g. \(\frac{24000}{1001} = 23.976\) fps.

**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

**framerateDenominator**
Framerate denominator.

**Type**: integer
**Required**: False

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

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

**spatialAq**
If set to enabled, adjust quantization within each frame based on spatial variation of content complexity.

**Type**: string
**Required**: False
**adaptiveQuantization**
Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.

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

**colorMetadata**
Includes colorspace metadata in the output.

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

**gopSize**
GOP size (keyframe interval) in units of either frames or seconds per gopSizeUnits. Must be greater than zero.

- **Type:** number
- **Required:** False
- **Minimum:** 1.0

**numRefFrames**
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

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

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

**sceneChangeDetect**
Scene change detection. Inserts I-frames on scene changes when enabled.

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

**parControl**
This field indicates how the output pixel aspect ratio is specified. If "specified" is selected then the output video pixel aspect ratio is determined by parNumerator and parDenominator, else if "initializeFromSource" is selected then the output pixel aspect ratio will be set equal to the input video pixel aspect ratio of the first input.

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

Pixel Aspect Ratio denominator.

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

**syntax**

Produces a bitstream compliant with SMPTE RP-2027.

*Type:* string  
*Required:* False

**scanType**

Sets the scan type of the output to progressive or top-field-first interlaced.

*Type:* string  
*Required:* False

**gopNumBFrames**

Number of B-frames between reference frames.

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

**flickerAq**

If set to enabled, adjust quantization within each frame to reduce flicker or 'pop' on I-frames.

*Type:* string  
*Required:* False

**rateControlMode**

Rate control mode.

*Type:* string  
*Required:* False

**H264SpatialAq (enum)**

DISABLED  
ENABLED

**H264Syntax (enum)**

DEFAULT  
RP2027
H264TemporalAq (enum)

DISABLED
ENABLED

H264TimecodeInsertionBehavior (enum)

DISABLED
PIC_TIMING_SEI

HlsAdMarkers (enum)

ADOBE
ELEMENTAL
ELEMENTAL_SCTE35

HlsAkamaiHttpTransferMode (enum)

CHUNKED
NON_CHUNKED

HlsAkamaiSettings

httpTransferMode

Specify whether or not to use chunked transfer encoding to Akamai. User should contact Akamai to enable this feature.

Type: string
Required: False

salt

Salt for authenticated Akamai.

Type: string
Required: False

numRetries

Number of retry attempts that will be made before the Live Event is put into an error state.

Type: integer
Required: False
Minimum: 0

restartDelay

If a streaming output fails, number of seconds to wait until a restart is initiated. A value of 0 means never restart.

Type: integer
Required: False
Properties

 Minimum: 0
 Maximum: 15

**connectionRetryInterval**
Number of seconds to wait before retrying connection to the CDN if the connection is lost.

 Type: integer
 Required: False
 Minimum: 0

**filecacheDuration**
Size in seconds of file cache for streaming outputs.

 Type: integer
 Required: False
 Minimum: 0
 Maximum: 600

**token**
Token parameter for authenticated akamai. If not specified, _gda_ is used.

 Type: string
 Required: False

**HlsBasicPutSettings**

**numRetries**
Number of retry attempts that will be made before the Live Event is put into an error state.

 Type: integer
 Required: False
 Minimum: 0

**restartDelay**
If a streaming output fails, number of seconds to wait until a restart is initiated. A value of 0 means never restart.

 Type: integer
 Required: False
 Minimum: 0
 Maximum: 15

**connectionRetryInterval**
Number of seconds to wait before retrying connection to the CDN if the connection is lost.

 Type: integer
 Required: False
 Minimum: 0
filecacheDuration

Size in seconds of file cache for streaming outputs.

Type: integer
Required: False
Minimum: 0
Maximum: 600

HlsCaptionLanguageSetting (enum)

INSERT
NONE
OMIT

HlsCdnSettings

hlsAkamaiSettings

Type: HlsAkamaiSettings (p. 74)
Required: False

hlsWebdavSettings

Type: HlsWebdavSettings (p. 85)
Required: False

HlsBasicPutSettings

Type: HlsBasicPutSettings (p. 75)
Required: False

hlsMediaStoreSettings

Type: HlsMediaStoreSettings (p. 83)
Required: False

HlsClientCache (enum)

DISABLED
ENABLED

HlsCodecSpecification (enum)

RFC_4281
RFC_6381

HlsDirectoryStructure (enum)

SINGLE_DIRECTORY
SUBDIRECTORY_PER_STREAM
HlsEncryptionType (enum)

- AES128
- SAMPLE_AES

HlsGroupSettings

segmentsPerSubdirectory

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

Type: integer
Required: False
Minimum: 1

ivInManifest

For use with encryptionType. The IV (Initialization Vector) is a 128-bit number used in conjunction with the key for encrypting blocks. If set to "include", IV is listed in the manifest, otherwise the IV is not in the manifest.

Type: string
Required: False

outputSelection

Generates the .m3u8 playlist file for this HLS output group. The segmentsOnly option will output segments without the .m3u8 file.

Type: string
Required: False

encryptionType

Encrypts the segments with the given encryption scheme. Exclude this parameter if no encryption is desired.

Type: string
Required: False

indexNSegments

Number of segments to keep in the playlist (.m3u8) file. mode must be "vod" for this setting to have an effect, and this number should be less than or equal to keepSegments.

Type: integer
Required: False
Minimum: 3

destination

A directory or HTTP destination for the HLS segments, manifest files, and encryption keys (if enabled).
**constantIv**

For use with encryptionType. This is a 128-bit, 16-byte hex value represented by a 32-character text string. If ivSource is set to "explicit" then this parameter is required and is used as the IV for encryption.

Type: string  
Required: False

**timedMetadataId3Frame**

Indicates ID3 frame that has the timecode.

Type: string  
Required: False

**baseUrlManifest**

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

**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: string  
Required: False

**minSegmentLength**

When set, minimumSegmentLength 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

**mode**

If set to "vod", keeps and indexes all segments starting with the first segment. If set to "live" segments will age out and only the last keepSegments number of segments will be retained.

Type: string
**Required**: False

### ivSource
For use with encryptionType. The IV (Initialization Vector) is a 128-bit number used in conjunction with the key for encrypting blocks. If this setting is “followsSegmentNumber”, it will cause the IV to change every segment (to match the segment number). If this is set to “explicit”, you must enter a constant IV value.

*Type*: string  
*Required*: False

### manifestCompression
When set to gzip, compresses HLS playlist.

*Type*: string  
*Required*: False

### keyProviderSettings
The key provider settings.

*Type*: KeyProviderSettings (p. 91)  
*Required*: False

### tsFileMode
When set to "singleFile", emits the program as a single media resource (.ts) file, and uses #EXT-X-BYTERANGE tags to index segment for playback. Playback of VOD mode content during event is not guaranteed due to HTTP server caching.

*Type*: string  
*Required*: False

### manifestDurationFormat
Indicates whether the output manifest should use floating point or integer values for segment duration.

*Type*: string  
*Required*: False

### keyFormatVersions
Either a single positive integer version value or a slash delimited list of version values (1/2/3).

*Type*: string  
*Required*: False

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

*Type*: string  
*Required*: False
timestampDeltaMilliseconds
Provides an extra millisecond delta offset to fine tune the timestamps.
- **Type:** integer
- **Required:** False
- **Minimum:** 0

segmentationMode
When set to useInputSegmentation, the output segment or fragment points are set by the RAI markers from the input streams.
- **Type:** string
- **Required:** False

baseUrlContent
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

clientCache
When set to "disabled", sets the #EXT-X-ALLOW-CACHE:no tag in the manifest, which prevents clients from saving media segments for later replay.
- **Type:** string
- **Required:** False

captionLanguageMappings
Mapping of up to 4 caption channels to caption languages. Is only meaningful if captionLanguageSetting is set to "insert".
- **Type:** Array of type CaptionLanguageMapping (p. 46)
- **Required:** False

codecSpecification
Specification to use (RFC-6381 or the default RFC-4281) during m3u8 playlist generation.
- **Type:** string
- **Required:** False

keepSegments
Number of segments to retain in the destination directory. mode must be "live" for this setting to have an effect.
- **Type:** integer
- **Required:** False
- **Minimum:** 1
timedMetadataId3Period

Timed Metadata interval in seconds.

Type: integer
Required: False
Minimum: 0

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 timestampOffset.

Type: string
Required: False

directoryStructure

Place segments in subdirectories.

Type: string
Required: False

keyFormat

The value specifies how the key is represented in the resource identified by the URI. If parameter is absent, an implicit value of "identity" is used. A reverse DNS string can also be given.

Type: string
Required: False

inputLossAction

Parameter that control output group behavior on input loss.

Type: string
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 string
Required: False

programDateTimePeriod

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

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

hlsCdnSettings
Parameters that control interactions with the CDN.

  Type: HlsCdnSettings (p. 76)
  Required: False

HlsInputSettings

retries
The number of consecutive times that attempts to read a manifest or segment must fail before the input is considered unavailable.

  Type: integer
  Required: False
  Minimum: 0

bandwidth
When specified the HLS stream with the m3u8 BANDWIDTH that most closely matches this value will be chosen, otherwise the highest bandwidth stream in the m3u8 will be chosen. The bitrate is specified in bits per second, as in an HLS manifest.

  Type: integer
  Required: False
  Minimum: 0

retryInterval
The number of seconds between retries when an attempt to read a manifest or segment fails.

  Type: integer
  Required: False
  Minimum: 0

bufferSegments
When specified, reading of the HLS input will begin this many buffer segments from the end (most recently written segment). When not specified, the HLS input will begin with the first segment specified in the m3u8.

  Type: integer
  Required: False
  Minimum: 0
HlsIvInManifest (enum)
   EXCLUDE
   INCLUDE

HlsIvSource (enum)
   EXPLICIT
   Follows_SEGMENT_NUMBER

HlsManifestCompression (enum)
   GZIP
   NONE

HlsManifestDurationFormat (enum)
   FLOATING_POINT
   INTEGER

HlsMediaStoreSettings

mediaStoreStorageClass
   When set to temporal, output files are stored in non-persistent memory for faster reading and writing.
   
   Type: string
   Required: False

numRetries
   Number of retry attempts that will be made before the Live Event is put into an error state.
   
   Type: integer
   Required: False
   Minimum: 0

restartDelay
   If a streaming output fails, number of seconds to wait until a restart is initiated. A value of 0 means never restart.
   
   Type: integer
   Required: False
   Minimum: 0
   Maximum: 15

connectionRetryInterval
   Number of seconds to wait before retrying connection to the CDN if the connection is lost.
   
   Type: integer
   Required: False
Minimum: 0

filecacheDuration

Size in seconds of file cache for streaming outputs.

Type: integer
Required: False
Minimum: 0
Maximum: 600

HlsMediaStoreStorageClass (enum)

TEMPORAL

HlsMode (enum)

LIVE
VOD

HlsOutputSelection (enum)

MANIFESTS_AND_SEGMENTS
SEGMENTS_ONLY

HlsOutputSettings

segmentModifier

String concatenated to end of segment filenames.

Type: string
Required: False

hlsSettings

Settings regarding the underlying stream. These settings are different for audio-only outputs.

Type: HlsSettings (p. 85)
Required: True

nameModifier

String concatenated to the end of the destination filename. Accepts "Format Identifiers \\"FormatIdentifierParameters.

Type: string
Required: False

HlsProgramDateTime (enum)

EXCLUDE
INCLUDE
HlsSegmentationMode (enum)
USE_INPUT_SEGMENTATION
USE_SEGMENT_DURATION

HlsSettings

audioOnlyHlsSettings
Type: AudioOnlyHlsSettings (p. 37)
Required: False

standardHlsSettings
Type: StandardHlsSettings (p. 117)
Required: False

HlsStreamInfResolution (enum)
EXCLUDE
INCLUDE

HlsTimedMetadataId3Frame (enum)
NONE
PRIV
TDRL

HlsTsFileMode (enum)
SEGMENTED_FILES
SINGLE_FILE

HlsWebdavHttpTransferMode (enum)
CHUNKED
NON_CHUNKED

HlsWebdavSettings

httpTransferMode
Specify whether or not to use chunked transfer encoding to WebDAV.
Type: string
Required: False

numRetries
Number of retry attempts that will be made before the Live Event is put into an error state.
Type: integer
Required: False
Minimum: 0

**restartDelay**

If a streaming output fails, number of seconds to wait until a restart is initiated. A value of 0 means never restart.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 15

**connectionRetryInterval**

Number of seconds to wait before retrying connection to the CDN if the connection is lost.

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

**filecacheDuration**

Size in seconds of file cache for streaming outputs.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 600

**InputAttachment**

**inputId**

The ID of the input

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

**inputSettings**

Settings of an input (caption selector, etc.)

- **Type**: InputSettings (p. 89)
- **Required**: False

**InputChannelLevel**

**inputChannel**

The index of the input channel used as a source.

- **Type**: integer
**Properties**

**gain**

Remixing value. Units are in dB and acceptable values are within the range from -60 (mute) and 6 dB.

- **Type**: integer
- **Required**: True
- **Minimum**: -60
- **Maximum**: 6

**InputCodec (enum)**

codec in increasing order of complexity

- MPEG2
- AVC
- HEVC

**InputDeblockFilter (enum)**

- DISABLED
- ENABLED

**InputDenoiseFilter (enum)**

- DISABLED
- ENABLED

**InputFilter (enum)**

- AUTO
- DISABLED
- FORCED

**InputLocation**

**passwordParam**

key used to extract the password from EC2 Parameter store

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

**uri**

Uniform Resource Identifier - This should be a path to a file accessible to the Live system (eg. a http:// URI) depending on the output type. For example, a rtmpEndpoint should have a uri similar to: "rtmp:// fmsserver/live".
**username**

Username if credentials are required to access a file or publishing point. This can be either a plaintext username, or a reference to an AWS parameter store name from which the username can be retrieved. *AWS Parameter store format: "ssm://<parameter name>"

**InputLossActionForHlsOut (enum)**

- EMIT_OUTPUT
- PAUSE_OUTPUT

**InputLossActionForMsSmoothOut (enum)**

- EMIT_OUTPUT
- PAUSE_OUTPUT

**InputLossActionForUdpOut (enum)**

- DROP_PROGRAM
- DROP_TS
- EMIT_PROGRAM

**InputLossBehavior**

**inputLossImageType**

Indicates whether to substitute a solid color or a slate into the output after input loss exceeds blackFrameMsec.

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

**inputLossImageColor**

When input loss image type is "color" this field specifies the color to use. Value: 6 hex characters representing the values of RGB.

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

**inputLossImageSlate**

When input loss image type is "slate" these fields specify the parameters for accessing the slate.

**Type:** InputLocation (p. 87)  
**Required:** False
blackFrameMsec
On input loss, the number of milliseconds to substitute black into the output before switching to the frame specified by inputLossImageType. A value x, where 0 <= x <= 1,000,000 and a value of 1,000,000 will be interpreted as infinite.

Type: integer
Required: False
Minimum: 0
Maximum: 1000000

repeatFrameMsec
On input loss, the number of milliseconds to repeat the previous picture before substituting black into the output. A value x, where 0 <= x <= 1,000,000 and a value of 1,000,000 will be interpreted as infinite.

Type: integer
Required: False
Minimum: 0
Maximum: 1000000

InputLossImageType (enum)
COLOR
SLATE

InputMaximumBitrate (enum)
Maximum input bitrate in megabits per second. Bitrates up to 50 Mbps are supported currently.

MAX_10_MBPS
MAX_20_MBPS
MAX_50_MBPS

InputResolution (enum)
Input resolution based on lines of vertical resolution in the input; SD is less than 720 lines, HD is 720 to 1080 lines, UHD is greater than 1080 lines

SD
HD
UHD

InputSettings
sourceEndBehavior
Loop input if it is a file. This allows a file input to be streamed indefinitely.

Type: string
Required: False

audioSelectors
Used to select the audio stream to decode for inputs that have multiple available.
Properties

**Type**: Array of type AudioSelector (p. 38)
**Required**: False

**deblockFilter**
Enable or disable the deblock filter when filtering.

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

**networkInputSettings**
Input settings.

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

**inputFilter**
Turns on the filter for this input. MPEG-2 inputs have the deblocking filter enabled by default. 1) auto - filtering will be applied depending on input type/quality 2) disabled - no filtering will be applied to the input 3) forced - filtering will be applied regardless of input type

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

**videoSelector**
Informs which video elementary stream to decode for input types that have multiple available.

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

**filterStrength**
Adjusts the magnitude of filtering from 1 (minimal) to 5 (strongest).

**Type**: integer
**Required**: False
**Minimum**: 1
**Maximum**: 5

**captionSelectors**
Used to select the caption input to use for inputs that have multiple available.

**Type**: Array of type CaptionSelector (p. 47)
**Required**: False

**denoiseFilter**
Enable or disable the denoise filter when filtering.
Properties

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

**InputDialogEndBehavior (enum)**
- CONTINUE
- LOOP

**InputSpecification**

**codec**
- Input codec
  - Type: string
  - Required: False

**resolution**
- Input resolution, categorized coarsely
  - Type: string
  - Required: False

**maximumBitrate**
- Maximum input bitrate, categorized coarsely
  - Type: string
  - Required: False

**InternalServerError**

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

**InvalidRequest**

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

**KeyProviderSettings**

**staticKeySettings**
  - Type: StaticKeySettings (p. 117)
LimitExceeded

message

Type: string
Required: False

ListChannelsResultModel

channels

Type: Array of type ChannelSummary (p. 50)
Required: False

nextToken

Type: string
Required: False

M2tsAbsentInputAudioBehavior (enum)

DROP
ENCODER_SILENCE

M2tsArib (enum)

DISABLED
ENABLED

M2tsAribCaptionsPidControl (enum)

AUTO
USE_CONFIGURED

M2tsAudioBufferModel (enum)

ATSC
DVB

M2tsAudioInterval (enum)

VIDEO_AND_FIXED_INTERVALS
VIDEO_INTERVAL

M2tsAudioStreamType (enum)

ATSC
DVB
M2tsBufferModel (enum)
   MULTIPLEX
   NONE

M2tsCcDescriptor (enum)
   DISABLED
   ENABLED

M2tsEbifControl (enum)
   NONE
   PASSTHRUGH

M2tsEbpPlacement (enum)
   VIDEO_AND_AUDIO_PIDS
   VIDEO_PID

M2tsEsRateInPes (enum)
   EXCLUDE
   INCLUDE

M2tsKlv (enum)
   NONE
   PASSTHRUGH

M2tsPcrControl (enum)
   CONFIGURED_PCR_PERIOD
   PCR_EVERY_PES_PACKET

M2tsRateMode (enum)
   CBR
   VBR

M2tsScte35Control (enum)
   NONE
   PASSTHRUGH

M2tsSegmentationMarkers (enum)
   EBP
   EBP_LEGACY
   NONE
   PSI_SEGSTART
   RAI_ADAPT
**M2tsSegmentationStyle (enum)**

- MAINTAIN_CADENCE
- RESET_CADENCE

**M2tsSettings**

**audioStreamType**

When set to atsc, uses stream type = 0x81 for AC3 and stream type = 0x87 for EAC3. When set to dvb, uses stream type = 0x06.

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

**ecmPid**

Packet Identifier (PID) for ECM in the transport stream. Only enabled when Simulcrypt is enabled. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20).8182 (or 0x1ff6).

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

**dvbTeletextPid**

Packet Identifier (PID) for input source DVB Teletext data to this output. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20).8182 (or 0x1ff6).

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

**aribCaptionsPidControl**

If set to auto, pid number used for ARIB Captions will be auto-selected from unused pids. If set to useConfigured, ARIB Captions will be on the configured pid number.

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

**bitrate**

The output bitrate of the transport stream in bits per second. Setting to 0 lets the muxer automatically determine the appropriate bitrate.

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

**segmentationTime**

The length in seconds of each segment. Required unless markers is set to None.
Properties

Type: number
Required: False
Minimum: 1.0

rateMode

When vbr, does not insert null packets into transport stream to fill specified bitrate. The bitrate setting acts as the maximum bitrate when vbr is set.

Type: string
Required: False

audioPids

Packet Identifier (PID) of the elementary audio stream(s) in the transport stream. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values. Each PID specified must be in the range of 32 (or 0x20)..8182 (or 0x1ff6).

Type: string
Required: False

fragmentTime

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

Type: number
Required: False
Minimum: 0.0

ebpAudioInterval

When videoAndFixedIntervals is selected, 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. Only available when EBP Cablelabs segmentation markers are selected. Partitions 1 and 2 will always follow the video interval.

Type: string
Required: False

ebpLookaheadMs

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

audioFramesPerPes

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

scte35Pid
Packet Identifier (PID) of the SCTE-35 stream in the transport stream. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20)..8192 (or 0x1ff6).

Type: string
Required: False

pcrPeriod
Maximum time in milliseconds between Program Clock Reference (PCRs) inserted into the transport stream.

Type: integer
Required: False
Minimum: 0
Maximum: 500

pmtInterval
The number of milliseconds between instances of this table in the output transport stream. Valid values are 0, 10..1000.

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

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

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

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 "resetCadence" 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 $segmentationTime seconds. When a segmentation style of "maintainCadence" 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 $segmentationTime seconds. Note that EBP lookahead is a slight exception to this rule.

Type: string
Required: False

**ebif**
If set to passthrough, passes any EBIF data from the input source to this output.

  Type: string
  Required: False

**audioBufferModel**
When set to dvb, uses DVB buffer model for Dolby Digital audio. When set to atsc, the ATSC model is used.

  Type: string
  Required: False

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

  Type: DvbNitSettings (p. 52)
  Required: False

**absentInputAudioBehavior**
When set to drop, output audio streams will be removed from the program if the selected input audio stream is removed from the input. This allows the output audio configuration to dynamically change based on input configuration. If this is set to encodeSilence, all output audio streams will output encoded silence when not connected to an active input stream.

  Type: string
  Required: False

**timedMetadataPid**
Packet Identifier (PID) of the timed metadata stream in the transport stream. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20)..8182 (or 0x1ff6).

  Type: string
  Required: False

**timedMetadataBehavior**
When set to passthrough, timed metadata will be passed through from input to output.

  Type: string
  Required: False

**etvSignalPid**
Packet Identifier (PID) for input source ETV Signal data to this output. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20)..8182 (or 0x1ff6).

  Type: string
### Properties

**Required:** False

**pmtPid**
Packet Identifier (PID) for the Program Map Table (PMT) in the transport stream. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20). 8182 (or 0x1ff6).

*Type:* string  
*Required:* False

**bufferModel**
If set to multiplex, use multiplex buffer model for accurate interleaving. Setting to bufferModel to none can lead to lower latency, but low-memory devices may not be able to play back the stream without interruptions.

*Type:* string  
*Required:* False

**scte35Control**
Optionally pass SCTE-35 signals from the input source to this output.

*Type:* string  
*Required:* False

**ebpPlacement**
Controls placement of EBP on Audio PIDs. If set to videoAndAudioPids, EBP markers will be placed on the video PID and all audio PIDs. If set to videoPid, EBP markers will be placed on only the video PID.

*Type:* string  
*Required:* False

**arib**
When set to enabled, uses ARIB-compliant field muxing and removes video descriptor.

*Type:* string  
*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  
*Minimum:* 0.0

**dvbSdtSettings**
Inserts DVB Service Description Table (SDT) at the specified table repetition interval.

*Type:* [DvbSdtSettings](#)
Properties

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. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20).8182 (or 0x1ff6).

Type: string
Required: False

**transportStreamId**

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

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

**pcrControl**

When set to pcrEveryPesPacket, 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: string
Required: False

**videoPid**

Packet Identifier (PID) of the elementary video stream in the transport stream. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20).8182 (or 0x1ff6).

Type: string
Required: False

**esRateInPes**

Include or exclude the ES Rate field in the PES header.

Type: string
Required: False

**segmentationMarkers**

Inserts segmentation markers at each segmentationTime period. raiSegstart sets the Random Access Indicator bit in the adaptation field. raiAdapt sets the RAI bit and adds the current timecode in the private data bytes. psiSegstart 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. ebpLegacy adds Encoder Boundary Point information to the adaptation field using a legacy proprietary format.

Type: string
Required: False
**Properties**

**dvbTdtSettings**

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

*Type:* DvbTdtSettings (p. 57)

*Required:* False

**klv**

If set to passthrough, passes any KLV data from the input source to this output.

*Type:* string

*Required:* False

**ccDescriptor**

When set to enabled, generates captionServiceDescriptor in PMT.

*Type:* string

*Required:* False

**patInterval**

The number of milliseconds between instances of this table in the output transport stream. Valid values are 0, 10..1000.

*Type:* integer

*Required:* False

*Minimum:* 0

*Maximum:* 1000

**etvPlatformPid**

Packet Identifier (PID) for input source ETV Platform data to this output. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20)..8182 (or 0x1ff6).

*Type:* string

*Required:* False

**dvbSubPids**

Packet Identifier (PID) for input source DVB Subtitle data to this output. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values. Each PID specified must be in the range of 32 (0x20)..8182 (0x1ff6).

*Type:* string

*Required:* False

**aribCaptionsPid**

Packet Identifier (PID) for ARIB Captions in the transport stream. Can be entered as a decimal or hexadecimal value. Valid values are 32 (0x20)..8182 (0x1ff6).

*Type:* string
**Properties**

**scte27Pids**
Packet Identifier (PID) for input source SCTE-27 data to this output. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values. Each PID specified must be in the range of 32 (or 0x20)..<2182 (or 0x1ff6).

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

**klvDataPids**
Packet Identifier (PID) for input source KLV data to this output. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values. Each PID specified must be in the range of 32 (or 0x20)..<2182 (or 0x1ff6).

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

**M2tsTimedMetadataBehavior** (enum)
- NO_PASSTHROUGH
- PASSTHROUGH

**M3u8PcrControl** (enum)
- CONFIGURED_PCR_PERIOD
- PCR_EVERY_PES_PACKET

**M3u8Scte35Behavior** (enum)
- NO_PASSTHROUGH
- PASSTHROUGH

**M3u8Settings**

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

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

**ecmPid**
The Platform-protected transport streams using 'microsoft' as Target Client include an ECM stream. This ECM stream contains the size, IV, and PTS of every sample in the transport stream. This stream PID is specified here. This PID has no effect on non Platform-protected streams.

- **Type**: string
- **Required**: False
**scte35Behavior**

If set to passthrough, passes any SCTE-35 signals from the input source to this output.

- **Type**: string
- **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. Can be entered as a decimal or hexadecimal value.

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

**audioPids**

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

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

**audioFramesPerPes**

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

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

**scte35Pid**

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

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

**transportStreamId**

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

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

**videoPid**

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

crcControl
When set to pcrEveryPesPacket, 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: string
Required: False

pmtInterval
The number of milliseconds between instances of this table in the output transport stream. A value of "0" writes out the PMT once per segment file.
Type: integer
Required: False
Minimum: 0
Maximum: 1000

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

programNum
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. A value of "0" writes out the PMT once per segment file.
Type: integer
Required: False
Minimum: 0
Maximum: 1000

timedMetadataPid
Packet Identifier (PID) of the timed metadata stream in the transport stream. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20)..8182 (or 0x1ff6).
Properties

Type: string
Required: False

timedMetadataBehavior
When set to passthrough, timed metadata is passed through from input to output.

Type: string
Required: False

M3u8TimedMetadataBehavior (enum)

NO_PASSTHROUGH
PASSTHROUGH

Mp2CodingMode (enum)

CODING_MODE_1_0
CODING_MODE_2_0

Mp2Settings

codingMode
The MPEG2 Audio coding mode. Valid values are codingMode10 (for mono) or codingMode20 (for stereo).

Type: string
Required: False

bitrate
Average bitrate in bits/second.

Type: number
Required: False

sampleRate
Sample rate in Hz.

Type: number
Required: False

MsSmoothGroupSettings

eventId
MS Smooth event ID to be sent to the IIS server. Should only be specified if eventIdMode is set to useConfigured.

Type: string
**Properties**

- **Required**: False

**fragmentLength**
Length of mp4 fragments to generate (in seconds). Fragment length must be compatible with GOP size and framerate.

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

**timestampOffset**
Timestamp offset for the event. Only used if timestampOffsetMode is set to useConfiguredOffset.

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

**segmentationMode**
When set to useInputSegmentation, the output segment or fragment points are set by the RAI markers from the input streams.

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

**numRetries**
Number of retry attempts.

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

**eventStopBehavior**
When set to sendEos, send EOS signal to IIS server when stopping the event.

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

**acquisitionPointId**
The value of the "Acquisition Point Identity" element used in each message placed in the sparse track. Only enabled if sparseTrackType is not "none".

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

**sparseTrackType**
If set to scte35, use incoming SCTE-35 messages to generate a sparse track in this group of MS-Smooth outputs.

  - **Type**: string
Properties

Required: False

timestampOffsetMode
Type of timestamp date offset to use. - useEventStartDate: Use the date the event was started as the offset - useConfiguredOffset: Use an explicitly configured date as the offset

Type: string
Required: False

destination
Smooth Streaming publish point on an IIS server. Elemental Live acts as a "Push" encoder to IIS.

Type: OutputLocationRef (p. 111)
Required: True

audioOnlyTimecodeControl
If set to passthrough for an audio-only MS Smooth output, the fragment absolute time will be set to the current timecode. This option does not write timecodes to the audio elementary stream.

Type: string
Required: False

connectionRetryInterval
Number of seconds to wait before retrying connection to the IIS server if the connection is lost. Content will be cached during this time and the cache will be delivered to the IIS server once the connection is re-established.

Type: integer
Required: False
Minimum: 0

filecacheDuration
Size in seconds of file cache for streaming outputs.

Type: integer
Required: False
Minimum: 0

certificateMode
If set to verifyAuthenticity, verify the https certificate chain to a trusted Certificate Authority (CA). This will cause https outputs to self-signed certificates to fail unless those certificates are manually added to the OS trusted keystore.

Type: string
Required: False

inputLossAction
Parameter that control output group behavior on input loss.
Properties

**sendDelayMs**
Outputs that are "output locked" can use this delay. Assign a delay to the output that is "secondary". Do not assign a delay to the "primary" output. The delay means that the primary output will always reach the downstream system before the secondary, which helps ensure that the downstream system always uses the primary output. (If there were no delay, the downstream system might flip-flop between whichever output happens to arrive first.) If the primary fails, the downstream system will switch to the secondary output. When the primary is restarted, the downstream system will switch back to the primary (because once again it is always arriving first).

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

**eventIdMode**
Specifies whether or not to send an event ID to the IIS server. If no event ID is sent and the same Live Event is used without changing the publishing point, clients might see cached video from the previous run. Options: - “useConfigured” - use the value provided in eventId - “useTimestamp” - generate and send an event ID based on the current timestamp - “noEventId” - do not send an event ID to the IIS server.

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

**restartDelay**
Number of seconds before initiating a restart due to output failure, due to exhausting the numRetries on one segment, or exceeding filecacheDuration.

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

**streamManifestBehavior**
When set to send, send stream manifest so publishing point doesn't start until all streams start.

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

**MsSmoothOutputSettings**

**nameModifier**
String concatenated to the end of the destination filename. Required for multiple outputs of the same type.

- **Type**: string
- **Required**: False
NetworkInputServerValidation (enum)

- CHECK_CRYPTOGRAPHY_AND_VALIDATE_NAME
- CHECK_CRYPTOGRAPHY_ONLY

NetworkInputSettings

hlsInputSettings

Specifies HLS input settings when the uri is for a HLS manifest.

- Type: HlsInputSettings (p. 82)
- Required: False

serverValidation

Check HTTPS server certificates. When set to checkCryptographyOnly, cryptography in the certificate will be checked, but not the server's name. Certain subdomains (notably S3 buckets that use dots in the bucket name) do not strictly match the corresponding certificate's wildcard pattern and would otherwise cause the event to error. This setting is ignored for protocols that do not use https.

- Type: string
- Required: False

Output

videoDescriptionName

The name of the VideoDescription used as the source for this output.

- Type: string
- Required: False

captionDescriptionNames

The names of the CaptionDescriptions used as caption sources for this output.

- Type: Array of type string
- Required: False

outputName

The name used to identify an output.

- Type: string
- Required: False

outputSettings

Output type-specific settings.

- Type: OutputSettings (p. 111)
Required: True

**audioDescriptionNames**

The names of the AudioDescriptions used as audio sources for this output.

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

**OutputDestination**

**settings**

Destination settings for output; one for each redundant encoder.

**Type:** Array of type **OutputDestinationSettings** (p. 109)
**Required:** False

**id**

User-specified id. This is used in an output group or an output.

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

**OutputDestinationSettings**

**passwordParam**

key used to extract the password from EC2 Parameter store

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

**streamName**

Stream name for RTMP destinations (URLs of type rtmp://)

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

**url**

A URL specifying a destination

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

**username**

username for destination
Type: string
Required: False

**OutputGroup**

outputs

Type: Array of type Output (p. 108)
Required: True

outputGroupSettings

Settings associated with the output group.

Type: OutputGroupSettings (p. 110)
Required: True

name

Custom output group name optionally defined by the user. Only letters, numbers, and the underscore character allowed; only 32 characters allowed.

Type: string
Required: False

**OutputGroupSettings**

archiveGroupSettings

Type: ArchiveGroupSettings (p. 32)
Required: False

rtmpGroupSettings

Type: RtmpGroupSettings (p. 112)
Required: False

udpGroupSettings

Type: UdpGroupSettings (p. 119)
Required: False

msSmoothGroupSettings

Type: MsSmoothGroupSettings (p. 104)
Required: False

hlsGroupSettings

Type: HlsGroupSettings (p. 77)
Properties

**Required:** False

### OutputLocationRef

**destinationRefId**

*Type: string*

*Required: False*

### OutputSettings

#### rtmpOutputSettings

*Type: RtmpOutputSettings (p. 113)*

*Required: False*

#### archiveOutputSettings

*Type: ArchiveOutputSettings (p. 32)*

*Required: False*

#### msSmoothOutputSettings

*Type: MsSmoothOutputSettings (p. 107)*

*Required: False*

#### udpOutputSettings

*Type: UdpOutputSettings (p. 119)*

*Required: False*

#### hlsOutputSettings

*Type: HlsOutputSettings (p. 84)*

*Required: False*

### PassThroughSettings

### RemixSettings

#### channelMappings

Mapping of input channels to output channels, with appropriate gain adjustments.

*Type: Array of type AudioChannelMapping (p. 33)*

*Required: True*

#### channelsOut

Number of output channels to be produced. Valid values: 1, 2, 4, 6, 8
Type: integer
Required: False
Minimum: 1
Maximum: 8

channelsIn

Number of input channels to be used.

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

ResourceConflict

message

Type: string
Required: False

RtmpCacheFullBehavior (enum)

- DISCONNECT_IMMEDIATELY
- WAIT_FOR_SERVER

RtmpCaptionData (enum)

- ALL
- FIELD1_608
- FIELD1_AND_FIELD2_608

RtmpCaptionInfoDestinationSettings

RtmpGroupSettings

captionData

Controls the types of data that passes to onCaptionInfo outputs. If set to 'all' then 608 and 708 carried DTVCC data will be passed. If set to 'field1AndField2608' then DTVCC data will be stripped out, but 608 data from both fields will be passed. If set to 'field1608' then only the data carried in 608 from field 1 video will be passed.

Type: string
Required: False

authenticationScheme

Authentication scheme to use when connecting with CDN

Type: string
Required: False

**cacheLength**

Cache length, in seconds, is used to calculate buffer size.

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

**restartDelay**

If a streaming output fails, number of seconds to wait until a restart is initiated. A value of 0 means never restart.

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

**cacheFullBehavior**

Controls behavior when content cache fills up. If remote origin server stalls the RTMP connection and does not accept content fast enough the 'Media Cache' will fill up. When the cache reaches the duration specified by cacheLength the cache will stop accepting new content. If set to disconnectImmediately, the RTMP output will force a disconnect. Clear the media cache, and reconnect after restartDelay seconds. If set to waitForServer, the RTMP output will wait up to 5 minutes to allow the origin server to begin accepting data again.

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

**RtmpOutputCertificateMode (enum)**

- **SELF_SIGNED**
- **VERIFY_AUTHENTICITY**

**RtmpOutputSettings**

**certificateMode**

If set to verifyAuthenticity, verify the tls certificate chain to a trusted Certificate Authority (CA). This will cause rtmps outputs with self-signed certificates to fail.

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

**numRetries**

Number of retry attempts.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
destination

The RTMP endpoint excluding the stream name (eg. rtmp://host/appname). For connection to Akamai, a
username and password must be supplied. URI fields accept format identifiers.

  Type: OutputLocationRef (p. 111)
  Required: True

collectionRetryInterval

Number of seconds to wait before retrying a connection to the Flash Media server if the connection is
lost.

  Type: integer
  Required: False
  Minimum: 1

Scte20Convert608To708 (enum)

DISABLED
UPCONVERT

Scte20PlusEmbeddedDestinationSettings

Scte20SourceSettings

source608ChannelNumber

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

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

convert608To708

If 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: string
  Required: False

Scte27DestinationSettings

Scte27SourceSettings

pid

The pid field is used in conjunction with the caption selector languageCode field as follows: - Specify PID
and Language: Extracts captions from that PID; the language is "informational". - Specify PID and omit
Language: Extracts the specified PID. - Omit PID and specify Language: Extracts the specified language,
whichever PID that happens to be. - Omit PID and omit Language: Valid only if source is DVB-Sub that is being passed through; all languages will be passed through.

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

### Scte35AposNoRegionalBlackoutBehavior

- **Type**: enum
  - **Values**: FOLLOW, IGNORE

### Scte35AposWebDeliveryAllowedBehavior

- **Type**: enum
  - **Values**: FOLLOW, IGNORE

### Scte35SpliceInsert

#### adAvailOffset

When specified, this offset (in milliseconds) is added to the input Ad Avail PTS time. This only applies to embedded SCTE 104/35 messages and does not apply to OOB messages.

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

#### webDeliveryAllowedFlag

When set to ignore, Segment Descriptors with webDeliveryAllowedFlag set to 0 will no longer trigger blackouts or Ad Avail slates

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

#### noRegionalBlackoutFlag

When set to ignore, Segment Descriptors with noRegionalBlackoutFlag set to 0 will no longer trigger blackouts or Ad Avail slates

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

### Scte35SpliceInsertNoRegionalBlackoutBehavior

- **Type**: enum
  - **Values**: FOLLOW, IGNORE

### Scte35SpliceInsertWebDeliveryAllowedBehavior

- **Type**: enum
  - **Values**: FOLLOW, IGNORE
**Scte35TimeSignalApos**

**adAvailOffset**
When specified, this offset (in milliseconds) is added to the input Ad Avail PTS time. This only applies to embedded SCTE 104/35 messages and does not apply to OOB messages.

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

**webDeliveryAllowedFlag**
When set to ignore, Segment Descriptors with webDeliveryAllowedFlag set to 0 will no longer trigger blackouts or Ad Avail slates

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

**noRegionalBlackoutFlag**
When set to ignore, Segment Descriptors with noRegionalBlackoutFlag set to 0 will no longer trigger blackouts or Ad Avail slates

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

**SmoothGroupAudioOnlyTimecodeControl (enum)**
- PASSTHROUGH
- USE_CONFIGURED_CLOCK

**SmoothGroupCertificateMode (enum)**
- SELF_SIGNED
- VERIFY_AUTHENTICITY

**SmoothGroupEventIdMode (enum)**
- NO_EVENT_ID
- USE_CONFIGURED
- USE_TIMESTAMP

**SmoothGroupEventStopBehavior (enum)**
- NONE
- SEND_EOS

**SmoothGroupSegmentationMode (enum)**
- USE_INPUT_SEGMENTATION
- USE_SEGMENT_DURATON
**SmoothGroupSparseTrackType (enum)**

- NONE
- SCTE_35

**SmoothGroupStreamManifestBehavior (enum)**

- DO_NOT_SEND
- SEND

**SmoothGroupTimestampOffsetMode (enum)**

- USE_CONFIGURED_OFFSET
- USE_EVENT_START_DATE

**SmpteTtDestinationSettings**

**StandardHlsSettings**

**m3u8Settings**

- **Type:** M3u8Settings (p. 101)
- **Required:** True

**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

**StaticKeySettings**

**staticKeyValue**

Static key value as a 32 character hexadecimal string.

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

**keyProviderServer**

The URL of the license server used for protecting content.

- **Type:** InputLocation (p. 87)
- **Required:** False
TeletextDestinationSettings

TeletextSourceSettings

pageNumber

Specifies the teletext page number within the data stream from which to extract captions. Range of 0x100 (256) to 0x8FF (2303). Unused for passthrough. Should be specified as a hexadecimal string with no "0x" prefix.

Type: string
Required: False

TimecodeConfig

syncThreshold

Threshold in frames beyond which output timecode is resynchronized to the input timecode. Discrepancies below this threshold are permitted to avoid unnecessary discontinuities in the output timecode. No timecode sync when this is not specified.

Type: integer
Required: False
Minimum: 1
Maximum: 1000000

source

Identifies the source for the timecode that will be associated with the events outputs. -Embedded (embedded): Initialize the output timecode with timecode from the the source. If no embedded timecode is detected in the source, the system falls back to using "Start at 0" (zerobased). -System Clock (systemclock): Use the UTC time. -Start at 0 (zerobased): The time of the first frame of the event will be 00:00:00:00.

Type: string
Required: True

TimecodeConfigSource (enum)

EMBEDDED
SYSTEMCLOCK
ZEROBASED

TtmlDestinationSettings

styleControl

When set to passthrough, passes 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: string
Required: False
**TtmlDestinationStyleControl (enum)**

- PASSTHROUGH
- USE_CONFIGURED

**UdpContainerSettings**

**m2tsSettings**

- **Type**: M2tsSettings (p. 94)
- **Required**: False

**UdpGroupSettings**

**inputLossAction**

Specifies behavior of last resort when input video is lost, and no more backup inputs are available. When dropTs is selected the entire transport stream will stop being emitted. When dropProgram is selected the program can be dropped from the transport stream (and replaced with null packets to meet the TS bitrate requirement). Or, when emitProgram is chosen the transport stream will continue to be produced normally with repeat frames, black frames, or slate frames substituted for the absent input video.

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

**timedMetadataId3Frame**

Indicates ID3 frame that has the timecode.

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

**timedMetadataId3Period**

Timed Metadata interval in seconds.

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

**UdpOutputSettings**

**bufferMsec**

UDP output buffering in milliseconds. Larger values increase latency through the transcoder but simultaneously assist the transcoder in maintaining a constant, low-jitter UDP/RTP output while accommodating clock recovery, input switching, input disruptions, picture reordering, etc.

- **Type**: integer
  - **Required**: False
  - **Minimum**: 0
  - **Maximum**: 10000
destination
Destination address and port number for RTP or UDP packets. Can be unicast or multicast RTP or UDP (eg. rtp://239.10.10.5001 or udp://10.100.100.100:5002).

Type: OutputLocationRef (p. 111)
Required: True

containerSettings

Type: UdpContainerSettings (p. 119)
Required: True

fecOutputSettings
Settings for enabling and adjusting Forward Error Correction on UDP outputs.

Type: FecOutputSettings (p. 64)
Required: False

UdpTimedMetadataId3Frame (enum)

NONE
PRIV
TDRL

ValidationError

errorMessage

Type: string
Required: False

elementPath

Type: string
Required: False

VideoCodecSettings

h264Settings

Type: H264Settings (p. 68)
Required: False

VideoDescription

respondToAfd
Indicates how to respond to the AFD values in the input stream. Setting to "respond" causes input video to be clipped, depending on AFD value, input display aspect ratio and output display aspect ratio.

Type: string
Properties

**Required**: False

**scalingBehavior**
When set to "stretchToOutput", automatically configures the output position to stretch the video to the specified output resolution. This option will override any position value.

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

**name**
The name of this VideoDescription. Outputs will use this name to uniquely identify this Description. Description names should be unique within this Live Event.

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

**width**
Output video width (in pixels). Leave out to use source video width. If left out, height must also be left out. Display aspect ratio is always preserved by letterboxing or pillarboxing when necessary.

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

**sharpness**
Changes the width of the anti-alias filter kernel used for scaling. Only applies if scaling is being performed and antiAlias is set to true. 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.

- **Type**: VideoCodecSettings (p. 120)

**height**
Output video height (in pixels). Leave blank to use source video height. If left blank, width must also be unspecified.

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

**VideoDescriptionRespondToAfd (enum)**

NONE
PASSTHROUGH
RESPOND

**VideoDescriptionScalingBehavior (enum)**

- DEFAULT
- STRETCH_TO_OUTPUT

**VideoSelector**

**colorSpace**

Specifies the colorspace of an input. This setting works in tandem with colorSpaceConversion to determine if any conversion will be performed.

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

**selectorSettings**

The video selector settings.

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

**colorSpaceUsage**

Applies only if colorSpace is a value other than follow. This field controls how the value in the colorSpace field will be used. fallback means that when the input does include color space data, that data will be used, but when the input has no color space data, the value in colorSpace will be used. Choose fallback if your input is sometimes missing color space data, but when it does have color space data, that data is correct. force means to always use the value in colorSpace. Choose force if your input usually has no color space data or might have unreliable color space data.

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

**VideoSelectorColorSpace (enum)**

- FOLLOW
- REC_601
- REC_709

**VideoSelectorColorSpaceUsage (enum)**

- FALLOUT
- FORCE

**VideoSelectorPid**

**pid**

Selects a specific PID from within a video source.
Channels channelId

**Type**: integer
**Required**: False
**Minimum**: 0
**Maximum**: 8191

**VideoSelectorProgramId**

**programId**

Selects a specific program from within a multi-program transport stream. If the program doesn't exist, the first program within the transport stream will be selected by default.

**Type**: integer
**Required**: False
**Minimum**: 0
**Maximum**: 65536

**VideoSelectorSettings**

**videoSelectorPid**

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

**videoSelectorProgramId**

**Type**: VideoSelectorProgramId (p. 123)
**Required**: False

**WebvttDestinationSettings**

**Channels channelId**

**URI**

/prod/channels/ channelId

**HTTP Methods**

**GET**

Operation ID: DescribeChannel

Gets details about a channel

**Path Parameters**

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>channelId</td>
<td>String</td>
<td>True</td>
<td>channel ID</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>Channel (p. 136)</td>
<td>Channel details</td>
</tr>
<tr>
<td>400</td>
<td>InvalidRequest (p. 158)</td>
<td>This request was invalid.</td>
</tr>
<tr>
<td>500</td>
<td>InternalServiceError (p. 158)</td>
<td>Unexpected internal service error.</td>
</tr>
<tr>
<td>502</td>
<td>BadGatewayException (p. 159)</td>
<td>Bad Gateway Error</td>
</tr>
<tr>
<td>403</td>
<td>AccessDenied (p. 158)</td>
<td>You do not have permission to list channels.</td>
</tr>
<tr>
<td>404</td>
<td>ResourceNotFound (p. 158)</td>
<td>The channel you're requesting to describe does not exist.</td>
</tr>
<tr>
<td>504</td>
<td>GatewayTimeoutException (p. 159)</td>
<td>Gateway Timeout Error</td>
</tr>
<tr>
<td>429</td>
<td>LimitExceeded (p. 158)</td>
<td>Request limit exceeded on list channel calls to channel service.</td>
</tr>
</tbody>
</table>

**PUT**

Operation ID: UpdateChannel

Updates a channel.

**Path Parameters**

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>channelId</td>
<td>String</td>
<td>True</td>
<td>channel ID</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>UpdateChannelResultModel (p. 157)</td>
<td>Channel is successfully updated.</td>
</tr>
<tr>
<td>400</td>
<td>InvalidRequest (p. 158)</td>
<td>This request was invalid.</td>
</tr>
<tr>
<td>422</td>
<td>ChannelConfigurationValidationError (p. 158)</td>
<td>The channel failed validation and could not be created.</td>
</tr>
<tr>
<td>500</td>
<td>InternalServiceError (p. 158)</td>
<td>Unexpected internal service error.</td>
</tr>
<tr>
<td>502</td>
<td>BadGatewayException (p. 159)</td>
<td>Bad Gateway Error</td>
</tr>
<tr>
<td>403</td>
<td>AccessDenied (p. 158)</td>
<td>You do not have permission to list channels.</td>
</tr>
<tr>
<td>504</td>
<td>GatewayTimeoutException (p. 159)</td>
<td>Gateway Timeout Error</td>
</tr>
<tr>
<td>409</td>
<td>ResourceConflict (p. 158)</td>
<td>The channel is unable to create due to an issue with channel resources.</td>
</tr>
</tbody>
</table>
DELETE

Operation ID: DeleteChannel

Starts deletion of channel. The associated outputs are also deleted.

Path Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>channelId</td>
<td>String</td>
<td>True</td>
<td>channel ID</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>Channel (p. 136)</td>
<td>Deletion was successfully initiated.</td>
</tr>
<tr>
<td>400</td>
<td>InvalidRequest (p. 158)</td>
<td>This request was invalid.</td>
</tr>
<tr>
<td>500</td>
<td>InternalServiceError (p. 158)</td>
<td>Unexpected internal service error.</td>
</tr>
<tr>
<td>502</td>
<td>BadGatewayException (p. 159)</td>
<td>Bad Gateway Error</td>
</tr>
<tr>
<td>403</td>
<td>AccessDenied (p. 158)</td>
<td>You do not have permission to list channels.</td>
</tr>
<tr>
<td>404</td>
<td>ResourceNotFound (p. 158)</td>
<td>The channel you're requesting to describe does not exist.</td>
</tr>
<tr>
<td>504</td>
<td>GatewayTimeoutException (p. 159)</td>
<td>Gateway Timeout Error</td>
</tr>
<tr>
<td>429</td>
<td>LimitExceeded (p. 158)</td>
<td>Request limit exceeded on list channel calls to channel service.</td>
</tr>
<tr>
<td>409</td>
<td>ResourceConflict (p. 158)</td>
<td>The channel is unable to create due to an issue with channel resources.</td>
</tr>
</tbody>
</table>

Schemas

Request Bodies

Example PUT

```json
{
    "inputAttachments (p. 249)": [
        {
            "inputId (p. 215)": "string",
            "inputSettings (p. 215)": {
                "sourceEndBehavior (p. 218)": enum,
                "audioSelectors (p. 218)": [
                    {
                        "name (p. 169)": "string",
                        "selectorSettings (p. 169)": {
                            "selectorId (p. 219)": "string",
                            "selectorName (p. 219)": "string",
                            "value (p. 219)": enum
                        }
                    }
                ]
            }
        }
    ]
}
```
"audioLanguageSelection (p. 169)": {
  "languageSelectionPolicy (p. 167)": enum,
  "languageCode (p. 167)": "string"
},
"audioPidSelection (p. 169)": {
  "pid (p. 168)": integer
}
},
"deblockFilter (p. 219)": enum,
"networkInputSettings (p. 219)": {
  "hlsInputSettings (p. 236)": {
    "retries (p. 211)": integer,
    "bandwidth (p. 211)": integer,
    "retryInterval (p. 211)": integer,
    "bufferSegments (p. 211)": integer
  },
  "serverValidation (p. 237)": enum
},
"inputFilter (p. 219)": enum,
"videoSelector (p. 219)": {
  "colorSpace (p. 252)": enum,
  "selectorSettings (p. 252)": {
    "videoSelectorPid (p. 253)": {
      "pid (p. 252)": integer
    },
    "videoSelectorProgramId (p. 253)": {
      "programId (p. 253)": integer
    }
  },
  "colorSpaceUsage (p. 252)": enum
},
"filterStrength (p. 219)": integer,
"captionSelectors (p. 219)": [
  {
    "name (p. 178)": "string",
    "languageCode (p. 178)": "string",
    "selectorSettings (p. 178)": {
      "embeddedSourceSettings (p. 178)": {
        "scte20Detection (p. 191)": enum,
        "source608ChannelNumber (p. 191)": integer,
        "convert608To708 (p. 191)": enum,
        "source608TrackNumber (p. 192)": integer
      },
      "scte20SourceSettings (p. 178)": {
        "source608ChannelNumber (p. 243)": integer,
        "convert608To708 (p. 243)": enum
      },
      "dvbSubSourceSettings (p. 178)": {
        "pid (p. 186)": integer
      },
      "aribSourceSettings (p. 178)": {
      },
      "teletextSourceSettings (p. 178)": {
        "pageNumber (p. 247)": "string"
      },
      "scte27SourceSettings (p. 179)": {
        "pid (p. 243)": integer
      }
    }
  }
],
"denoiseFilter (p. 219)": enum
"roleArn (p. 249)": "string",
"destinations (p. 249)": [
  {
    "settings (p. 238)": [
      {
        "passwordParam (p. 238)": "string",
        "streamName (p. 238)": "string",
        "url (p. 238)": "string",
        "username (p. 238)": "string"
      },
      "id (p. 238)": "string"
    ],
    "name (p. 249)": "string",
    "encoderSettings (p. 250)": {
      "timecodeConfig (p. 192)": {
        "syncThreshold (p. 247)": integer,
        "source (p. 247)": enum
      },
      "outputGroups (p. 192)": [
        {
          "outputs (p. 238)": [
            {
              "videoDescriptionName (p. 237)": "string",
              "captionDescriptionNames (p. 237)": [
                "string"
              ],
              "outputName (p. 237)": "string",
              "rtmpOutputSettings (p. 240)": {
                "certificateMode (p. 242)": enum,
                "numRetries (p. 242)": integer,
                "destination (p. 242)": {
                  "destinationRefId (p. 239)": "string"
                },
                "connectionRetryInterval (p. 243)": integer
              },
              "archiveOutputSettings (p. 240)": {
                "extension (p. 163)": "string",
                "containerSettings (p. 164)": {
                  "m2tsSettings (p. 163)": {
                    "audioStreamType (p. 222)": enum,
                    "ecmPid (p. 223)": "string",
                    "dvbTeletextPid (p. 223)": "string",
                    "aribCaptionsPidControl (p. 223)": enum,
                    "bitrate (p. 223)": integer,
                    "segmentationTime (p. 223)": number,
                    "rateMode (p. 223)": enum,
                    "audioPids (p. 224)": "string",
                    "fragmentTime (p. 224)": number,
                    "ebpAudioInterval (p. 224)": enum,
                    "ebpLookaheadMs (p. 224)": integer,
                    "audioFramesPerFes (p. 224)": integer,
                    "scte35Pid (p. 224)": "string",
                    "pcrPeriod (p. 225)": integer,
                    "pmtInterval (p. 225)": integer,
                    "programNum (p. 225)": integer,
                    "segmentationStyle (p. 225)": enum,
                    "ebif (p. 225)": enum,
                    "audioBufferModel (p. 225)": enum,
                    "dvbNitSettings (p. 226)": {
                      "networkName (p. 181)": "string",
                      "networkId (p. 181)": integer,
                      "repInterval (p. 181)": integer
                    }
                  }
                }
              }
            }
          }
        }
      ]
    }
  ]
}
"absentInputAudioBehavior (p. 226)": enum,
"timedMetadataPid (p. 226)": "string",
"timedMetadataBehavior (p. 226)": enum,
"etvSignalPid (p. 226)": "string",
"pmrPid (p. 226)": "string",
"bufferModel (p. 227)": enum,
"scte35Control (p. 227)": enum,
"ebpPlacement (p. 227)": enum,
"arib (p. 227)": enum,
"nullPacketBitrate (p. 227)": number,
"dvbSdtSettings (p. 227)": {
  "serviceName (p. 181)": "string",
  "serviceProviderName (p. 182)": "string",
  "repInterval (p. 182)": integer,
  "outputSdt (p. 182)": enum
},
"pcrPid (p. 227)": "string",
"transportStreamId (p. 228)": integer,
"pcrControl (p. 228)": enum,
"videoPid (p. 228)": "string",
"esRateInPes (p. 228)": enum,
"segmentationMarkers (p. 228)": enum,
"dvbTdtSettings (p. 228)": {
  "repInterval (p. 186)": integer
},
"klv (p. 229)": enum,
"ccDescriptor (p. 229)": enum,
"patInterval (p. 229)": integer,
"etvPlatformPid (p. 229)": "string",
"dvbSubPids (p. 229)": "string",
"aribCaptionsPid (p. 229)": "string",
"scte27Pids (p. 229)": "string",
"klvDataPids (p. 230)": "string"
"nameModifier (p. 164)": "string"
},
"msSmoothOutputSettings (p. 240)": {
  "nameModifier (p. 236)": "string"
},
"udpOutputSettings (p. 240)": {
  "bufferMsec (p. 248)": integer,
  "destination (p. 249)": {
    "destinationRefId (p. 239)": "string"
  },
  "containerSettings (p. 249)": {
    "m2tsSettings (p. 248)": {
      "audioStreamType (p. 222)": enum,
      "ecmPid (p. 223)": "string",
      "dvbTeletextPid (p. 223)": "string",
      "aribCaptionsPidControl (p. 223)": enum,
      "bitrate (p. 223)": integer,
      "segmentationTime (p. 223)": number,
      "rateMode (p. 223)": enum,
      "audioPids (p. 224)": "string",
      "fragmentTime (p. 224)": number,
      "ebpAudioInterval (p. 224)": enum,
      "ebpLookaheadMs (p. 224)": integer,
      "audioFramesPerPes (p. 224)": integer,
      "scte35Pid (p. 224)": "string",
      "pcrPeriod (p. 225)": integer,
      "pmtInterval (p. 225)": integer,
      "programNum (p. 225)": integer,
      "segmentationStyle (p. 225)": enum,
      "ebif (p. 225)": enum,
"audioBufferModel (p. 225)": enum,
"dvbNitSettings (p. 226)": {
  "networkName (p. 181)": "string",
  "networkId (p. 181)": integer,
  "repInterval (p. 181)": integer
},
"absentInputAudioBehavior (p. 226)": enum,
"timedMetadataPid (p. 226)": "string",
"timedMetadataBehavior (p. 226)": enum,
"etvSignalPid (p. 226)": "string",
"bufferModel (p. 227)": enum,
"scte35Control (p. 227)": enum,
"etvPlatformPid (p. 227)": enum,
"ebpPlacement (p. 227)": enum,
"arib (p. 227)": enum,
"nullPacketBitrate (p. 227)": number,
"dvbSdtSettings (p. 227)": {
  "serviceName (p. 181)": "string",
  "serviceProviderName (p. 182)": "string",
  "repInterval (p. 182)": integer,
  "outputSdt (p. 182)": enum
},
"pcrPid (p. 227)": "string",
"transportStreamId (p. 228)": integer,
"pcrControl (p. 228)": enum,
"videoPid (p. 228)": "string",
"esRateInPes (p. 228)": enum,
"segmentationMarkers (p. 228)": enum,
"dvbTdtSettings (p. 228)": {
  "repInterval (p. 186)": integer
},
"klv (p. 229)": enum,
"ccDescriptor (p. 229)": enum,
"patInterval (p. 229)": integer,
"etvPlatformPid (p. 229)": "string",
"dvbSubPids (p. 229)": "string",
"aribCaptionsPid (p. 229)": "string",
"scte27Pids (p. 229)": "string",
"klvDataPids (p. 230)": "string"
}
"
"fecOutputSettings (p. 249)": {
  "rowLength (p. 193)": integer,
  "columnDepth (p. 193)": integer,
  "includeFec (p. 193)": enum
},
"hlsOutputSettings (p. 240)": {
  "segmentModifier (p. 213)": "string",
  "hlsSettings (p. 213)": {
    "audioOnlyHlsSettings (p. 214)": {
      "audioTrackType (p. 168)": enum,
      "audioGroupId (p. 168)": "string",
      "audioOnlyImage (p. 168)": {
        "passwordParam (p. 216)": "string",
        "uri (p. 216)": "string",
        "username (p. 217)": "string"
      }
    },
    "standardHlsSettings (p. 214)": {
      "m3u8Settings (p. 246)": {
        "pmtPid (p. 230)": "string",
        "ecmPid (p. 230)": "string",
        "scte35Behavior (p. 230)": enum,
        "pcrPid (p. 231)": "string",
        "audioPids (p. 231)": "string",
        ""
"audioFramesPerPes (p. 231)": integer,
"scte35Pid (p. 231)": "string",
"transportStreamId (p. 231)": integer,
"videoPid (p. 231)": "string",
"pcrControl (p. 231)": enum,
"pmtpInterval (p. 232)": integer,
"pcrPeriod (p. 232)": integer,
"programNum (p. 232)": integer,
"patInterval (p. 232)": integer,
"timedMetadataPid (p. 232)": "string",
"timedMetadataBehavior (p. 232)": enum,
"audioRenditionSets (p. 246)": "string"
]},
"audioDescriptionNames (p. 237)": [ "string" ]
],
"outputGroupSettings (p. 239)": {
"archiveGroupSettings (p. 239)": {
"destination (p. 163)": {
"destinationRefId (p. 239)": "string"
},
"rolloverInterval (p. 163)": integer
},
"rtmpGroupSettings (p. 239)": {
"captionData (p. 241)": enum,
"authenticationScheme (p. 241)": enum,
"cacheLength (p. 241)": integer,
"restartDelay (p. 242)": integer,
"cacheFullBehavior (p. 242)": enum
},
"udpGroupSettings (p. 239)": {
"inputLossAction (p. 248)": enum,
"timedMetadataId3Frame (p. 248)": enum,
"timedMetadataId3Period (p. 248)": integer
},
"msSmoothGroupSettings (p. 239)": {
"eventId (p. 233)": "string",
"fragmentLength (p. 233)": integer,
"timestampOffset (p. 234)": "string",
"segmentationMode (p. 234)": enum,
"numRetries (p. 234)": integer,
"eventStopBehavior (p. 234)": enum,
"acquisitionPointId (p. 234)": "string",
"sparseTrackType (p. 234)": enum,
"timestampOffsetMode (p. 234)": enum,
"destination (p. 235)": {
"destinationRefId (p. 239)": "string"
},
"sendDelayMs (p. 235)": integer,
"eventIdMode (p. 236)": enum,
"restartDelay (p. 236)": integer,
"streamManifestBehavior (p. 236)": enum
},
"hlsGroupSettings (p. 239)": {

"segmentsPerSubdirectory (p. 206)": integer,
"ivInManifest (p. 206)": enum,
"outputSelection (p. 206)": enum,
"encryptionType (p. 206)": enum,
"indexNSegments (p. 206)": integer,
"destination (p. 206)": {
  "destinationRefId (p. 239)": "string"
},
"constantIv (p. 207)": "string",
"timedMetadataId3Frame (p. 207)": enum,
"baseUrlManifest (p. 207)": "string",
"captionLanguageSetting (p. 207)": enum,
"minSegmentLength (p. 207)": integer,
"mode (p. 207)": enum,
"ivSource (p. 208)": enum,
"manifestCompression (p. 208)": enum,
"keyProviderSettings (p. 208)": {
  "staticKeySettings (p. 220)": {
    "staticKeyValue (p. 246)": "string",
    "keyProviderServer (p. 246)": {
      "passwordParam (p. 216)": "string",
      "uri (p. 216)": "string",
      "username (p. 217)": "string"
    }
  },
  "tsFileMode (p. 208)": enum,
  "manifestDurationFormat (p. 208)": enum,
  "streamInfResolution (p. 208)": enum,
  "timestampDeltaMilliseconds (p. 209)": integer,
  "segmentationMode (p. 209)": enum,
  "baseUrlContent (p. 209)": "string",
  "clientCache (p. 209)": enum,
  "captionLanguageMappings (p. 209)": [
    {
      "languageDescription (p. 177)": "string",
      "captionChannel (p. 177)": integer,
      "languageCode (p. 177)": "string"
    }
  ],
  "codecSpecification (p. 209)": enum,
  "keepSegments (p. 209)": integer,
  "timedMetadataId3Period (p. 210)": integer,
  "programDateTime (p. 210)": enum,
  "directoryStructure (p. 210)": enum,
  "keyFormat (p. 210)": "string",
  "inputLossAction (p. 210)": enum,
  "adMarkers (p. 210)": [enum
  ],
  "segmentLength (p. 211)": integer,
  "hlsCdnSettings (p. 211)": {
    "hlsAkamaiSettings (p. 205)": {
      "httpTransferMode (p. 203)": enum,
      "salt (p. 203)": "string",
      "numRetries (p. 203)": integer,
      "restartDelay (p. 203)": integer,
      "connectionRetryInterval (p. 204)": integer,
      "fileCacheDuration (p. 204)": integer,
      "token (p. 204)": "string"
    },
    "hlsWebdavSettings (p. 205)": {
      "httpTransferMode (p. 214)": enum,
      "numRetries (p. 214)": integer,
"restartDelay (p. 215)" : integer,
"connectionRetryInterval (p. 215)" : integer,
"filecacheDuration (p. 215)" : integer,
"hlsBasicPutSettings (p. 205)" : {
  "numRetries (p. 204)" : integer,
  "restartDelay (p. 204)" : integer,
  "connectionRetryInterval (p. 204)" : integer,
  "filecacheDuration (p. 205)" : integer,
"hlsMediaStoreSettings (p. 205)" : {
  "mediaStoreStorageClass (p. 212)" : enum,
  "numRetries (p. 212)" : integer,
  "restartDelay (p. 212)" : integer,
  "connectionRetryInterval (p. 212)" : integer,
  "filecacheDuration (p. 213)" : integer
}}

"name (p. 239)" : "string",
"audioDescriptions (p. 192)" : [
  {
    "languageCodeControl (p. 165)" : enum,
    "audioTypeControl (p. 165)" : enum,
    "remixSettings (p. 165)" : {
      "channelMappings (p. 240)" : [
        {
          "outputChannel (p. 164)" : integer,
          "inputChannelLevels (p. 164)" : [
            {
              "inputChannel (p. 215)" : integer,
              "gain (p. 216)" : integer
            }
          ]
        }
      ]
    }]
  },
  "channelsOut (p. 240)" : integer,
  "channelsIn (p. 240)" : integer,
},
"audioType (p. 165)" : enum,
"name (p. 166)" : "string",
"languageCode (p. 166)" : "string",
"codecSettings (p. 166)" : {
  "aacSettings (p. 164)" : {
    "vbrQuality (p. 159)" : enum,
    "codingMode (p. 160)" : enum,
    "profile (p. 160)" : enum,
    "bitrate (p. 160)" : number,
    "inputType (p. 160)" : enum,
    "rawFormat (p. 160)" : enum,
    "sampleRate (p. 160)" : number,
    "rateControlMode (p. 160)" : enum,
    "spec (p. 161)" : enum
  },
  "ac3Settings (p. 164)" : {
    "drcProfile (p. 162)" : enum,
    "dialnorm (p. 162)" : integer,
    "codingMode (p. 162)" : enum,
    "metadataControl (p. 162)" : enum,
    "bitrate (p. 162)" : number,
    "lfeFilter (p. 162)" : enum,
    "bitstreamMode (p. 163)" : enum
  }
}
"eac3Settings (p. 165)": {
  "dialnorm (p. 188)": integer,
  "passthroughControl (p. 188)": enum,
  "metadataControl (p. 188)": enum,
  "drcLine (p. 188)": enum,
  "bitrate (p. 188)": number,
  "surroundExMode (p. 188)": enum,
  "ltRtSurroundMixLevel (p. 189)": number,
  "lifeControl (p. 189)": enum,
  "codingMode (p. 189)": enum,
  "surroundMode (p. 189)": enum,
  "attenuationControl (p. 189)": enum,
  "lfeFilter (p. 189)": enum,
  "ltRtCenterMixLevel (p. 189)": number,
  "dcFilter (p. 190)": enum,
  "phaseControl (p. 190)": enum,
  "stereoDownmix (p. 190)": enum,
  "bitstreamMode (p. 190)": enum,
  "loRoSurroundMixLevel (p. 190)": number,
  "drcRf (p. 190)": enum,
  "loRoCenterMixLevel (p. 190)": number
},
"passThroughSettings (p. 165)": {
},
"mp2Settings (p. 165)": {
  "codingMode (p. 233)": enum,
  "bitrate (p. 233)": number,
  "sampleRate (p. 233)": number
},
"streamName (p. 166)": "string",
"audioNormalizationSettings (p. 166)": {
  "targetLkfs (p. 167)": number,
  "algorithmControl (p. 167)": enum,
  "algorithm (p. 168)": enum
},
"audioSelectorName (p. 166)": "string"
],
"captionDescriptions (p. 192)": [
  {
    "captionSelectorName (p. 175)": "string",
    "languageDescription (p. 175)": "string",
    "name (p. 175)": "string",
    "languageCode (p. 176)": "string",
    "destinationSettings (p. 176)": {
      "scte27DestinationSettings (p. 176)": {
      },
      "burnInDestinationSettings (p. 176)": {
        "xPosition (p. 172)": integer,
        "backgroundColor (p. 172)": enum,
        "yPosition (p. 172)": integer,
        "teletextGridControl (p. 172)": enum,
        "backgroundOpacity (p. 172)": integer,
        "fontOpacity (p. 173)": integer,
        "fontResolution (p. 173)": integer,
        "shadowOpacity (p. 173)": integer,
        "shadowYOffset (p. 173)": integer,
        "outlineSize (p. 173)": integer,
        "outlineColor (p. 173)": enum,
        "fontSize (p. 174)": "string",
        "shadowXOffset (p. 174)": integer,
        "alignment (p. 174)": enum,
        "shadowColor (p. 174)": enum,
        "fontColor (p. 174)": enum,
        "font (p. 174)": {
      }
  },
"passwordParam (p. 216)": "string",
"uri (p. 216)": "string",
"username (p. 217)": "string"
},
"teletextDestinationSettings (p. 176)": {
},
"smpteTtDestinationSettings (p. 176)": {
},
"webvttDestinationSettings (p. 176)": {
},
"ttmlDestinationSettings (p. 176)": {
"styleControl (p. 247)": enum
},
"embeddedPlusScte20DestinationSettings (p. 176)": {
},
"dvbSubDestinationSettings (p. 177)": {
"xPosition (p. 183)": integer,
"backgroundColor (p. 183)": enum,
"yPosition (p. 183)": integer,
"teletextGridControl (p. 183)": enum,
"backgroundOpacity (p. 183)": integer,
"fontOpacity (p. 184)": integer,
"fontResolution (p. 184)": integer,
"shadowOpacity (p. 184)": integer,
"shadowYOffset (p. 184)": integer,
"outlineSize (p. 184)": integer,
"outlineColor (p. 185)": enum,
"fontSize (p. 185)": "string",
"shadowXOffset (p. 185)": integer,
"alignment (p. 185)": enum,
"shadowColor (p. 185)": enum,
"fontColor (p. 185)": enum,
"font (p. 186)": {
"passwordParam (p. 216)": "string",
"uri (p. 216)": "string",
"username (p. 217)": "string"
}
},
"embeddedDestinationSettings (p. 177)": {
},
"rtmpCaptionInfoDestinationSettings (p. 177)": {
},
"aribDestinationSettings (p. 177)": {
},
"scte20PlusEmbeddedDestinationSettings (p. 177)": {
}
}
],
"availConfiguration (p. 192)": {
"availSettings (p. 170)": {
"scte35TimeSignalApos (p. 170)": {
"adAvailOffset (p. 245)": integer,
"webDeliveryAllowedFlag (p. 245)": enum,
"noRegionalBlackoutFlag (p. 245)": enum
},
"scte35SpliceInsert (p. 170)": {
"adAvailOffset (p. 244)": integer,
"webDeliveryAllowedFlag (p. 244)": enum,
"noRegionalBlackoutFlag (p. 244)": enum
}
},
"globalConfiguration (p. 192)": {
"inputLossBehavior (p. 194)": {
"passwordParam (p. 216)": "string",
"uri (p. 216)": "string",
"username (p. 217)": "string"
}
"inputLossImageType (p. 217)": enum,
"inputLossImageColor (p. 217)": "string",
"inputLossImageSlate (p. 217)": {
    "passwordParam (p. 216)": "string",
    "uri (p. 216)": "string",
    "username (p. 217)": "string"
},
"blackFrameMsec (p. 218)": integer,
"repeatFrameMsec (p. 218)": integer
},
"supportLowFramerateInputs (p. 194)": enum,
"initialAudioGain (p. 194)": integer,
"inputEndAction (p. 194)": enum,
"outputTimingSource (p. 195)": enum
},
"videoDescriptions (p. 193)": [
    {
        "respondToAfd (p. 250)": enum,
        "scalingBehavior (p. 251)": enum,
        "name (p. 251)": "string",
        "width (p. 251)": integer,
        "sharpness (p. 251)": integer,
        "codecSettings (p. 251)": {
            "h264Settings (p. 250)": {
                "minIInterval (p. 197)": integer,
                "slices (p. 197)": integer,
                "parNumerator (p. 197)": integer,
                "gopSizeUnits (p. 198)": enum,
                "maxBitrate (p. 198)": integer,
                "bitrate (p. 198)": integer,
                "bufFillPct (p. 198)": integer,
                "temporalAq (p. 198)": enum,
                "afdSignaling (p. 198)": enum,
                "timecodeInsertion (p. 199)": enum,
                "bufSize (p. 199)": integer,
                "softness (p. 199)": integer,
                "framerateControl (p. 199)": enum,
                "fixedAfd (p. 199)": enum,
                "level (p. 199)": enum,
                "lookAheadRateControl (p. 199)": enum,
                "profile (p. 200)": enum,
                "framerateNumerator (p. 200)": integer,
                "gopClosedCadence (p. 200)": integer,
                "framerateDenominator (p. 200)": integer,
                "entropyEncoding (p. 200)": enum,
                ".spatialAq (p. 200)": enum,
                "adaptiveQuantization (p. 200)": enum,
                "colorMetadata (p. 201)": enum,
                "gopSize (p. 201)": number,
                "numRefFrames (p. 201)": integer,
                "gopBReference (p. 201)": enum,
                "sceneChangeDetect (p. 201)": enum,
                "parControl (p. 201)": enum,
                "parDenominator (p. 201)": integer,
                "syntax (p. 202)": enum,
                "scanType (p. 202)": enum,
                "gopNumBFrames (p. 202)": integer,
                "flickerAq (p. 202)": enum,
                "rateControlMode (p. 202)": enum
            }
        }
    },
    "height (p. 251)": integer
},
"blackoutSlate (p. 193)": {
    "networkEndBlackoutImage (p. 170)": {
"passwordParam (p. 216)": "string",
"uri (p. 216)": "string",
"username (p. 217)": "string"
},
"networkEndBlackout (p. 171)": enum,
"networkId (p. 171)": "string",
"state (p. 171)": enum,
"blackoutSlateImage (p. 171)": {
  "passwordParam (p. 216)": "string",
  "uri (p. 216)": "string",
  "username (p. 217)": "string"
}
},
"availBlanking (p. 193)": {
  "state (p. 169)": enum,
  "availBlankingImage (p. 170)": {
    "passwordParam (p. 216)": "string",
    "uri (p. 216)": "string",
    "username (p. 217)": "string"
  }
},
"inputSpecification (p. 250)": {
  "codec (p. 220)": enum,
  "resolution (p. 220)": enum,
  "maximumBitrate (p. 220)": enum
}

Response Bodies

Example Channel

{
  "inputAttachments (p. 179)": [
    {
      "inputId (p. 215)": "string",
      "inputSettings (p. 215)": {
        "sourceEndBehavior (p. 218)": enum,
        "audioSelectors (p. 218)": [
          {
            "name (p. 169)": "string",
            "selectorSettings (p. 169)": {
              "audioLanguageSelection (p. 169)": {
                "languageSelectionPolicy (p. 167)": enum,
                "languageCode (p. 167)": "string"
              }
            }
          }
        ],
      "deblockFilter (p. 219)": enum,
      "networkInputSettings (p. 219)": {
        "hlsInputSettings (p. 236)": {
          "retries (p. 211)": integer,
          "bandwidth (p. 211)": integer,
          "retryInterval (p. 211)": integer,
          "bufferSegments (p. 211)": integer
        }
      },
      "serverValidation (p. 237)": enum
    }
  ]
}
"videoSelector (p. 219)": {
  "colorSpace (p. 252)": enum,
  "selectorSettings (p. 252)": {
    "videoSelectorPid (p. 253)": {
      "pid (p. 252)": integer
    },
    "videoSelectorProgramId (p. 253)": {
      "programId (p. 253)": integer
    }
  },
  "colorSpaceUsage (p. 252)": enum,
  "filterStrength (p. 219)": integer,
  "captionSelectors (p. 219)": [
    {
      "name (p. 178)": "string",
      "languageCode (p. 178)": "string",
      "selectorSettings (p. 178)": {
        "embeddedSourceSettings (p. 178)": {
          "scte20Detection (p. 191)": enum,
          "source608ChannelNumber (p. 191)": integer,
          "convert608To708 (p. 191)": enum,
          "source608TrackNumber (p. 192)": integer
        },
        "scte20SourceSettings (p. 178)": {
          "source608ChannelNumber (p. 243)": integer,
          "convert608To708 (p. 243)": enum
        },
        "dvbSubSourceSettings (p. 178)": {
          "pid (p. 186)": integer
        },
        "aribSourceSettings (p. 178)": {
        },
        "teletextSourceSettings (p. 178)": {
          "pageNumber (p. 247)": "string"
        },
        "scte27SourceSettings (p. 179)": {
          "pid (p. 243)": integer
        }
      }
    }
  ],
  "denoiseFilter (p. 219)": enum
}
},
"roleArn (p. 179)": "string",
"destinations (p. 179)": [
  {
    "settings (p. 238)": [
      {
        "passwordParam (p. 238)": "string",
        "streamName (p. 238)": "string",
        "url (p. 238)": "string",
        "username (p. 238)": "string"
      }
    ],
    "id (p. 238)": "string"
  }
],
"name (p. 179)": "string",
"encoderSettings (p. 179)": {
  "timecodeConfig (p. 192)": {
    "syncThreshold (p. 247)": integer,
    "source (p. 247)": enum
  },
  "timecodeSource (p. 192)": enum,
  "timecodeFormat (p. 192)": enum
}
}
"outputGroups (p. 192)": [  
  "outputs (p. 238)": [  
    "videoDescriptionName (p. 237)": "string",  
    "captionDescriptionNames (p. 237)": [  
      "string"  
    ],  
    "outputName (p. 237)": "string",  
    "outputSettings (p. 237)": {  
      "rtmpOutputSettings (p. 240)": {  
        "certificateMode (p. 242)": enum,  
        "numRetries (p. 242)": integer,  
        "destination (p. 242)": {  
          "destinationRefId (p. 239)": "string"  
        },  
        "connectionRetryInterval (p. 243)": integer  
      },  
      "archiveOutputSettings (p. 240)": {  
        "extension (p. 163)": "string",  
        "containerSettings (p. 164)": {  
          "m2tsSettings (p. 163)": {  
            "audioStreamType (p. 222)": enum,  
            "ecmPid (p. 223)": "string",  
            "dvbTeletextPid (p. 223)": "string",  
            "aribCaptionsPidControl (p. 223)": enum,  
            "bitrate (p. 223)": integer,  
            "segmentationTime (p. 223)": number,  
            "rateMode (p. 223)": enum,  
            "audioPids (p. 224)": "string",  
            "fragmentTime (p. 224)": number,  
            "ebpAudioInterval (p. 224)": enum,  
            "ebpLookaheadMs (p. 224)": integer,  
            "audioFramesPerPes (p. 224)": integer,  
            "scte35Pid (p. 224)": "string",  
            "pcrPeriod (p. 225)": integer,  
            "pmtInterval (p. 225)": integer,  
            "programNum (p. 225)": integer,  
            "segmentationStyle (p. 225)": enum,  
            "ebif (p. 225)": enum,  
            "audioBufferModel (p. 225)": enum,  
            "dvbNitSettings (p. 226)": {  
              "networkName (p. 181)": "string",  
              "networkId (p. 181)": integer,  
              "repInterval (p. 181)": integer  
            },  
            "absentInputAudioBehavior (p. 226)": enum,  
            "timedMetadataPid (p. 226)": "string",  
            "timedMetadataBehavior (p. 226)": enum,  
            "etvSignalPid (p. 226)": "string",  
            "pmtPid (p. 226)": "string",  
            "bufferModel (p. 227)": enum,  
            "scte35Control (p. 227)": enum,  
            "ebpPlacement (p. 227)": enum,  
            "arib (p. 227)": enum,  
            "nullPacketBitrate (p. 227)": number,  
            "dvbSdtSettings (p. 227)": {  
              "serviceName (p. 181)": "string",  
              "serviceProviderName (p. 182)": "string",  
              "repInterval (p. 182)": integer,  
              "outputSdt (p. 182)": enum  
            },  
            "pcrPid (p. 227)": "string",  
            "transportStreamId (p. 228)": integer,  
            "pcrControl (p. 228)": enum,  
            "videoPid (p. 228)": "string",  
            "transportStreamPid (p. 228)": "string",  
            "rateControl (p. 228)": enum  
          },  
          "nullPacketBitrate (p. 227)": number,  
          "dvbSdtSettings (p. 227)": {  
            "serviceName (p. 181)": "string",  
            "serviceProviderName (p. 182)": "string",  
            "repInterval (p. 182)": integer,  
            "outputSdt (p. 182)": enum  
          },  
          "pcrPid (p. 227)": "string",  
          "transportStreamId (p. 228)": integer,  
          "pcrControl (p. 228)": enum,  
          "videoPid (p. 228)": "string",  
          "transportStreamPid (p. 228)": "string",  
          "rateControl (p. 228)": enum  
        }  
      }  
    ]  
  ]
}
"esRateInPes (p. 228)" : enum,
"segmentationMarkers (p. 228)" : enum,
"dvbTdtSettings (p. 228)" : {
  "repInterval (p. 186)" : integer
},
"klv (p. 229)" : enum,
"ccDescriptor (p. 229)" : enum,
"patInterval (p. 229)" : integer,
"etvPlatformPid (p. 229)" : "string",
"dvbSubPids (p. 229)" : "string",
"aribCaptionsPid (p. 229)" : "string",
"scte27Pids (p. 229)" : "string",
"klvDataPids (p. 230)" : "string"
}
},
"nameModifier (p. 164)" : "string"
},
"msSmoothOutputSettings (p. 240)" : {
  "nameModifier (p. 236)" : "string"
},
"udpOutputSettings (p. 240)" : {
  "bufferMsec (p. 248)" : integer,
  "destination (p. 249)" : {
    "destinationRefId (p. 239)" : "string"
  }
},
"containerSettings (p. 249)" : {
  "m2tsSettings (p. 248)" : {
    "audioStreamType (p. 222)" : enum,
    "ecmPid (p. 223)" : "string",
    "dvbTeletextPid (p. 223)" : "string",
    "aribCaptionsPidControl (p. 223)" : enum,
    "bitrate (p. 223)" : integer,
    "segmentationTime (p. 223)" : number,
    "rateMode (p. 223)" : enum,
    "audioPids (p. 224)" : "string",
    "fragmentTime (p. 224)" : number,
    "ebpAudioInterval (p. 224)" : enum,
    "ebpLookaheadMs (p. 224)" : integer,
    "audioFramesPerPes (p. 224)" : integer,
    "scte35Pid (p. 224)" : "string",
    "pcrPeriod (p. 225)" : integer,
    "pmtInterval (p. 225)" : integer,
    "programNum (p. 225)" : integer,
    "segmentationStyle (p. 225)" : enum,
    "ebif (p. 225)" : enum,
    "audioBufferModel (p. 225)" : enum,
    "dvbNitSettings (p. 226)" : {
      "networkName (p. 181)" : "string",
      "networkProviderName (p. 182)" : "string",
      "repInterval (p. 182)" : integer,
      "outputSdt (p. 182)" : enum
    }
  }
}
},
  "pcrPid (p. 227)": "string",
  "transportStreamId (p. 228)": integer,
  "pcrControl (p. 228)": enum,
  "videoPid (p. 228)": "string",
  "esRateInPes (p. 228)": enum,
  "segmentationMarkers (p. 228)": enum,
  "dvbTdtSettings (p. 228)": {
    "repInterval (p. 186)": integer
  },
  "klv (p. 229)": enum,
  "ccDescriptor (p. 229)": enum,
  "patInterval (p. 229)": integer,
  "etvPlatformPid (p. 229)": "string",
  "dvbSubPids (p. 229)": "string",
  "aribCaptionsPid (p. 229)": "string",
  "scte27Pids (p. 229)": "string",
  "klvDataPids (p. 230)": "string"
}
},
  "fecOutputSettings (p. 249)": {
    "rowLength (p. 193)": integer,
    "columnDepth (p. 193)": integer,
    "includeFec (p. 193)": enum
  },
  "hlsOutputSettings (p. 240)": {
    "segmentModifier (p. 213)": "string",
    "hlsSettings (p. 213)": {
      "audioOnlyHlsSettings (p. 214)": {
        "audioTrackType (p. 168)": enum,
        "audioGroupId (p. 168)": "string",
        "audioOnlyImage (p. 168)": {
          "passwordParam (p. 216)": "string",
          "uri (p. 216)": "string",
          "username (p. 217)": "string"
        }
      },
      "standardHlsSettings (p. 214)": {
        "m3u8Settings (p. 246)": {
          "pmtPid (p. 230)": "string",
          "ecmPid (p. 230)": "string",
          "scte35Behavior (p. 230)": enum,
          "pcrPid (p. 231)": "string",
          "audioPids (p. 231)": "string",
          "audioFramesPerPes (p. 231)": integer,
          "scte35Pid (p. 231)": "string",
          "transportStreamId (p. 231)": integer,
          "videoPid (p. 231)": "string",
          "pcrControl (p. 231)": enum,
          "pmtInterval (p. 232)": integer,
          "pcrPeriod (p. 232)": integer,
          "programNum (p. 232)": integer,
          "patInterval (p. 232)": integer,
          "timedMetadataPid (p. 232)": "string",
          "timedMetadataBehavior (p. 232)": enum
        },
        "audioRenditionSets (p. 246)": "string"
      }
    },
    "nameModifier (p. 213)": "string"
  }
},
  "audioDescriptionNames (p. 237)": [
    "string"
]
"outputGroupSettings (p. 239)": {
  "archiveGroupSettings (p. 239)": {
    "destination (p. 163)": {
      "destinationRefId (p. 239)": "string"
    },
    "rolloverInterval (p. 163)": integer
  },
  "rtmpGroupSettings (p. 239)": {
    "captionData (p. 241)": enum,
    "authenticationScheme (p. 241)": enum,
    "cacheLength (p. 241)": integer,
    "restartDelay (p. 242)": integer,
    "cacheFullBehavior (p. 242)": enum
  },
  "udpGroupSettings (p. 239)": {
    "inputLossAction (p. 248)": enum,
    "timeMetadataId3Frame (p. 248)": enum,
    "timeMetadataId3Period (p. 248)": integer
  },
  "msSmoothGroupSettings (p. 239)": {
    "eventId (p. 233)": "string",
    "fragmentLength (p. 233)": integer,
    "timestampOffset (p. 234)": "string",
    "segmentationMode (p. 234)": enum,
    "numRetries (p. 234)": integer,
    "eventStopBehavior (p. 234)": enum,
    "acquisitionPointId (p. 234)": "string",
    "sparseTrackType (p. 234)": enum,
    "timestampOffsetMode (p. 234)": enum,
    "destination (p. 235)": {
      "destinationRefId (p. 239)": "string"
    },
    "audioOnlyTimecodeControl (p. 235)": enum,
    "connectionRetryInterval (p. 235)": integer,
    "filecacheDuration (p. 235)": integer,
    "certificateMode (p. 235)": enum,
    "inputLossAction (p. 235)": enum,
    "sendDelayMs (p. 235)": integer,
    "eventIdMode (p. 236)": enum,
    "restartDelay (p. 236)": integer,
    "streamManifestBehavior (p. 236)": enum
  },
  "hlsGroupSettings (p. 239)": {
    "segmentsPerSubdirectory (p. 206)": integer,
    "ivInManifest (p. 206)": enum,
    "outputSelection (p. 206)": enum,
    "encryptionType (p. 206)": enum,
    "indexNSegments (p. 206)": integer,
    "destination (p. 206)": {
      "destinationRefId (p. 239)": "string"
    },
    "constantIV (p. 207)": "string",
    "timeMetadataId3Frame (p. 207)": enum,
    "baseUrlManifest (p. 207)": "string",
    "captionLanguageSetting (p. 207)": enum,
    "minSegmentLength (p. 207)": integer,
    "mode (p. 207)": enum,
    "ivSource (p. 208)": enum,
    "manifestCompression (p. 208)": enum,
    "keyProviderSettings (p. 208)": {
      "staticKeySettings (p. 220)": {
        "staticKeyValue (p. 246)": "string",
        "keyProviderServer (p. 246)": {
          "passwordParam (p. 216)": "string",
          "keyProviderRefId (p. 239)": "string",
          "rolloverInterval (p. 163)": integer
        },
        "rolloverInterval (p. 163)": integer
      },
      "rolloverInterval (p. 163)": integer
    },
    "rolloverInterval (p. 163)": integer
  }
}


"uri (p. 216)": "string",
"username (p. 217)": "string"
}
">
"tsFileMode (p. 208)": enum,
"manifestDurationFormat (p. 208)": enum,
"keyFormatVersions (p. 208)": "string",
"streamInfResolution (p. 208)": enum,
"timestampDeltaMilliseconds (p. 209)": integer,
"segmentationMode (p. 209)": enum,
"baseUrlContent (p. 209)": "string",
"clientCache (p. 209)": enum,
"captionLanguageMappings (p. 209)": [
{
"languageDescription (p. 177)": "string",
"captionChannel (p. 177)": integer,
"languageCode (p. 177)": "string"
}
],
"codecSpecification (p. 209)": enum,
"keepSegments (p. 209)": integer,
"timedMetadataId3Period (p. 210)": integer,
"programDateTime (p. 210)": enum,
"directoryStructure (p. 210)": enum,
"keyFormat (p. 210)": "string",
"inputLossAction (p. 210)": enum,
"adMarkers (p. 210)": [
enum
"programDateTimePeriod (p. 210)": integer,
"segmentLength (p. 211)": integer,
"hlsCdnSettings (p. 211)": {
"hlsAkamaiSettings (p. 205)": {
"httpTransferMode (p. 203)": enum,
"salt (p. 203)": "string",
"numRetries (p. 203)": integer,
"restartDelay (p. 203)": integer,
"connectionRetryInterval (p. 204)": integer,
"filecacheDuration (p. 204)": integer,
"token (p. 204)": "string"
}
"hlsWebdavSettings (p. 205)": {
"httpTransferMode (p. 214)": enum,
"numRetries (p. 214)": integer,
"restartDelay (p. 215)": integer,
"connectionRetryInterval (p. 215)": integer,
"filecacheDuration (p. 215)": integer
}
"hlsBasicPutSettings (p. 205)": {
"numRetries (p. 204)": integer,
"restartDelay (p. 204)": integer,
"connectionRetryInterval (p. 204)": integer,
"filecacheDuration (p. 205)": integer
}
"hlsMediaStoreSettings (p. 205)": {
"mediaStoreStorageClass (p. 212)": enum,
"numRetries (p. 212)": integer,
"restartDelay (p. 212)": integer,
"connectionRetryInterval (p. 212)": integer,
"filecacheDuration (p. 213)": integer
}
}]
"name (p. 239)": "string"
"audioDescriptions (p. 192)": [
  {
    "languageCodeControl (p. 165)": enum,
    "audioTypeControl (p. 165)": enum,
    "remixSettings (p. 165)": {
      "channelMappings (p. 240)": [
        {
          "outputChannel (p. 164)": integer,
          "inputChannelLevels (p. 164)": [
            {
              "inputChannel (p. 215)": integer,
              "gain (p. 216)": integer
            }
          ]
        }
      ]
    }
  },
  "channelsOut (p. 240)": integer,
  "channelsIn (p. 240)": integer
},
"audioType (p. 165)": enum,
"name (p. 166)": "string",
"languageCode (p. 166)": "string",
"codecSettings (p. 166)": {
  "aacSettings (p. 164)": {
    "vbrQuality (p. 159)": enum,
    "codingMode (p. 160)": enum,
    "profile (p. 160)": enum,
    "bitrate (p. 160)": number,
    "inputType (p. 160)": enum,
    "rawFormat (p. 160)": enum,
    "sampleRate (p. 160)": number,
    "rateControlMode (p. 160)": enum,
    "spec (p. 161)": enum
  },
  "ac3Settings (p. 164)": {
    "drcProfile (p. 162)": enum,
    "dialnorm (p. 162)": integer,
    "codingMode (p. 162)": enum,
    "metadataControl (p. 162)": enum,
    "bitrate (p. 162)": number,
    "lfeFilter (p. 162)": enum,
    "bitstreamMode (p. 163)": enum
  },
  "eac3Settings (p. 165)": {
    "dialnorm (p. 188)": integer,
    "passthroughControl (p. 188)": enum,
    "metadataControl (p. 188)": enum,
    "drcLine (p. 188)": enum,
    "bitrate (p. 188)": number,
    "surroundEXMode (p. 188)": enum,
    "ltRtSurroundMixLevel (p. 189)": number,
    "lfeControl (p. 189)": enum,
    "codingMode (p. 189)": enum,
    "surroundMode (p. 189)": enum,
    "attenuationControl (p. 189)": enum,
    "lfeFilter (p. 189)": enum,
    "ltRtCenterMixLevel (p. 189)": number,
    "dcFilter (p. 190)": enum,
    "phaseControl (p. 190)": enum,
    "stereoDownmix (p. 190)": enum,
    "bitstreamMode (p. 190)": enum,
    "loRoSurroundMixLevel (p. 190)": number,
    "drcRf (p. 190)": enum,
    "loRoCenterMixLevel (p. 190)": number
  }
}
"passThroughSettings (p. 165)": {
},
"mp2Settings (p. 165)": {
"codingMode (p. 233)": enum,
"bitrate (p. 233)": number,
"sampleRate (p. 233)": number
},
"streamName (p. 166)": "string",
"audioNormalizationSettings (p. 166)": {
"targetLkfs (p. 167)": number,
"algorithmControl (p. 167)": enum,
"algorithm (p. 168)": enum
},
"audioSelectorName (p. 166)": "string"
},
"captionDescriptions (p. 192)": [
{
"captionSelectorName (p. 175)": "string",
"languageDescription (p. 175)": "string",
"name (p. 175)": "string",
"languageCode (p. 176)": "string",
"destinationSettings (p. 176)": {
"scte27DestinationSettings (p. 176)": {
},
"burnInDestinationSettings (p. 176)": {
"xPosition (p. 172)": integer,
"backgroundColor (p. 172)": enum,
"yPosition (p. 172)": integer,
"teletextGridControl (p. 172)": enum,
"backgroundOpacity (p. 172)": integer,
"fontOpacity (p. 173)": integer,
"fontResolution (p. 173)": integer,
"shadowOpacity (p. 173)": integer,
"shadowYOffset (p. 173)": integer,
"outlineSize (p. 173)": integer,
"outlineColor (p. 173)": enum,
"fontSize (p. 174)": "string",
"shadowXOffset (p. 174)": integer,
"alignment (p. 174)": enum,
"shadowColor (p. 174)": enum,
"fontColor (p. 174)": enum,
"font (p. 174)": {
"passwordParam (p. 216)": "string",
"uri (p. 216)": "string",
"username (p. 217)": "string"
}
},
"teletextDestinationSettings (p. 176)": {
},
"smpteTtDestinationSettings (p. 176)": {
},
"webvttDestinationSettings (p. 176)": {
},
"ttmlDestinationSettings (p. 176)": {
"styleControl (p. 247)": enum
},
"embeddedPlusScte20DestinationSettings (p. 176)": {
},
"dvbSubDestinationSettings (p. 177)": {
"xPosition (p. 183)": integer,
"backgroundColor (p. 183)": enum,
"yPosition (p. 183)": integer,
"teletextGridControl (p. 183)": enum,
"backgroundOpacity (p. 183)": integer,
"fontOpacity (p. 184)": integer,
"fontResolution (p. 184)": integer,
"shadowOpacity (p. 184)": integer,
"shadowYOffset (p. 184)": integer,
"outlineSize (p. 184)": integer,
"outlineColor (p. 185)": enum,
"fontSize (p. 185)": "string",
"shadowXOffset (p. 185)": integer,
"alignment (p. 185)": enum,
"shadowColor (p. 185)": enum,
"outlineColor (p. 185)": enum,
"font (p. 186)": {  
  "passwordParam (p. 216)": "string",
  "uri (p. 216)": "string",
  "username (p. 217)": "string"
}
},
"embeddedDestinationSettings (p. 177)": {  
},
"rtmpCaptionInfoDestinationSettings (p. 177)": {  
},
"aribDestinationSettings (p. 177)": {  
},
"scte20PlusEmbeddedDestinationSettings (p. 177)": {  
}
}
",
"availConfiguration (p. 192)": {  
  "availSettings (p. 170)": {  
    "scte35TimeSignalApos (p. 170)": {  
      "adAvailOffset (p. 245)": integer,
      "webDeliveryAllowedFlag (p. 245)": enum,
      "noRegionalBlackoutFlag (p. 245)": enum
    },
    "scte35SpliceInsert (p. 170)": {  
      "adAvailOffset (p. 244)": integer,
      "webDeliveryAllowedFlag (p. 244)": enum,
      "noRegionalBlackoutFlag (p. 244)": enum
    }
  }
},
"globalConfiguration (p. 192)": {  
  "inputLossBehavior (p. 194)": {  
    "inputLossImageType (p. 217)": enum,
    "inputLossImageColor (p. 217)": "string",
    "inputLossImageSlate (p. 217)": {  
      "passwordParam (p. 216)": "string",
      "uri (p. 216)": "string",
      "username (p. 217)": "string"
    },
    "blackFrameMsec (p. 218)": integer,
    "repeatFrameMsec (p. 218)": integer
  },
  "supportLowFramerateInputs (p. 194)": enum,
  "initialAudioGain (p. 194)": integer,
  "inputEndAction (p. 194)": enum,
  "outputTimingSource (p. 195)": enum
},
"videoDescriptions (p. 193)": [  
  {  
    "respondToAfd (p. 250)": enum,
    "scalingBehavior (p. 251)": enum,
    "name (p. 251)": "string",
    "width (p. 251)": integer,
    "height (p. 251)": integer
  }
]}
"sharpness (p. 251)" : integer,
"codecSettings (p. 251)" : {
  "h264Settings (p. 250)" : {
    "minInterval (p. 197)" : integer,
    "slices (p. 197)" : integer,
    "parNumerator (p. 197)" : integer,
    "gopSizeUnits (p. 198)" : enum,
    "maxBitrate (p. 198)" : integer,
    "bitrate (p. 198)" : integer,
    "bufFillPct (p. 198)" : integer,
    "temporalAq (p. 198)" : enum,
    "afdSignaling (p. 198)" : enum,
    "timecodeInsertion (p. 199)" : enum,
    "bufSize (p. 199)" : integer,
    "softness (p. 199)" : integer,
    "framerateControl (p. 199)" : enum,
    "fixedAfd (p. 199)" : enum,
    "level (p. 199)" : enum,
    "lookAheadRateControl (p. 199)" : enum,
    "profile (p. 200)" : enum,
    "framerateNumerator (p. 200)" : integer,
    "gopClosedCadence (p. 200)" : integer,
    "framerateDenominator (p. 200)" : integer,
    "entropyEncoding (p. 200)" : enum,
    "spatialAq (p. 200)" : enum,
    "adaptiveQuantization (p. 200)" : enum,
    "colorMetadata (p. 201)" : enum,
    "gopSize (p. 201)" : number,
    "numRefFrames (p. 201)" : integer,
    "gopRReference (p. 201)" : enum,
    "sceneChangeDetect (p. 201)" : enum,
    "parControl (p. 201)" : enum,
    "parDenominator (p. 201)" : integer,
    "syntax (p. 202)" : enum,
    "scanType (p. 202)" : enum,
    "gopNumBFrames (p. 202)" : integer,
    "flickerAq (p. 202)" : enum,
    "rateControlMode (p. 202)" : enum
  }
},
"height (p. 251)" : integer
},
"blackoutSlate (p. 193)" : {
  "networkEndBlackoutImage (p. 170)" : {
    "passwordParam (p. 216)" : "string",
    "uri (p. 216)" : "string",
    "username (p. 217)" : "string"
  },
  "networkEndBlackout (p. 171)" : enum,
  "networkId (p. 171)" : "string",
  "state (p. 171)" : enum,
  "blackoutSlateImage (p. 171)" : {
    "passwordParam (p. 216)" : "string",
    "uri (p. 216)" : "string",
    "username (p. 217)" : "string"
  }
},
"availBlanking (p. 193)" : {
  "state (p. 169)" : enum,
  "availBlankingImage (p. 170)" : {
    "passwordParam (p. 216)" : "string",
    "uri (p. 216)" : "string",
    "username (p. 217)" : "string"
  }
}
Example UpdateChannelResultModel

```json
{
  "channel (p. 250)": {
    "inputAttachments (p. 179)": [
      {
        "inputId (p. 215)": "string",
        "inputSettings (p. 215)": {
          "sourceEndBehavior (p. 218)": enum,
          "audioSelectors (p. 218)": [
            {
              "name (p. 169)": "string",
              "selectorSettings (p. 169)": {
                "audioLanguageSelection (p. 169)": {
                  "languageSelectionPolicy (p. 167)": enum,
                  "languageCode (p. 167)": "string"
                },
                "audioPidSelection (p. 169)": {
                  "pid (p. 168)": integer
                }
              }
            }
          ],
          "deblockFilter (p. 219)": enum,
          "networkInputSettings (p. 219)": {
            "hlsInputSettings (p. 236)": {
              "retries (p. 211)": integer,
              "bandwidth (p. 211)": integer,
              "retryInterval (p. 211)": integer,
              "bufferSegments (p. 211)": integer
            },
            "serverValidation (p. 237)": enum
          }
        }
      }
    ],
    "deblockFilter (p. 219)": enum,
    "networkInputSettings (p. 219)": {
      "hlsInputSettings (p. 236)": {
        "retries (p. 211)": integer,
        "bandwidth (p. 211)": integer,
        "retryInterval (p. 211)": integer,
        "bufferSegments (p. 211)": integer
      },
      "serverValidation (p. 237)": enum
    },
    "inputFilter (p. 219)": enum,
    "videoSelector (p. 219)": {
      "colorSpace (p. 252)": enum,
      "selectorSettings (p. 252)": {
        "videoSelectorPid (p. 253)": {
          "pid (p. 252)": integer
        },
        "videoSelectorProgramId (p. 253)": {
          "programId (p. 253)": integer
        }
      }
    },
    "colorSpaceUsage (p. 252)": enum
  },
  "filterStrength (p. 219)": integer,
  "arn (p. 180)": "string",
  "inputSpecification (p. 180)": {
    "codec (p. 220)": enum,
    "resolution (p. 220)": enum,
    "maximumBitrate (p. 220)": enum
  }
}
```
"captionSelectors (p. 219)": [ 
  {
    "name (p. 178)": "string",
    "languageCode (p. 178)": "string",
    "selectorSettings (p. 178)": {
      "embeddedSourceSettings (p. 178)": {
        "scte20Detection (p. 191)": enum,
        "source608ChannelNumber (p. 191)": integer,
        "convert608To708 (p. 191)": enum,
        "source608TrackNumber (p. 192)": integer
      },
      "scte20SourceSettings (p. 178)": {
        "source608ChannelNumber (p. 243)": integer,
        "convert608To708 (p. 243)": enum
      },
      "dvbSubSourceSettings (p. 178)": {
        "pid (p. 186)": integer
      },
      "aribSourceSettings (p. 178)": {},
      "teletextSourceSettings (p. 178)": {
        "pageNumber (p. 247)": "string"
      },
      "scte27SourceSettings (p. 179)": {
        "pid (p. 243)": integer
      }
    }
  },
  "denoiseFilter (p. 219)": enum
},
"roleArn (p. 179)": "string",
"destinations (p. 179)": [ 
  {
    "settings (p. 238)": [ 
      {
        "passwordParam (p. 238)": "string",
        "streamName (p. 238)": "string",
        "url (p. 238)": "string",
        "username (p. 238)": "string"
      }
    ],
    "id (p. 238)": "string"
  }
],
"name (p. 179)": "string",
"encoderSettings (p. 179)": {
  "timecodeConfig (p. 192)": {
    "syncThreshold (p. 247)": integer,
    "source (p. 247)": enum
  },
  "outputGroups (p. 192)": [ 
    {
      "outputs (p. 238)": [ 
        {
          "videoDescriptionName (p. 237)": "string",
          "captionDescriptionNames (p. 237)": [ "string"
        }
      ]
    },
    "outputName (p. 237)": "string",
    "outputSettings (p. 237)": {
      "rtmpOutputSettings (p. 240)": {
        "certificateMode (p. 242)": enum,
        "numRetries (p. 242)": integer,
        "server (p. 242)": "string"
      }
    }
  ]
}
<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;destinationRefId&quot;</td>
<td>&quot;string&quot;</td>
</tr>
<tr>
<td>&quot;connectionRetryInterval&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;archiveOutputSettings&quot;</td>
<td></td>
</tr>
<tr>
<td>&quot;extension&quot;</td>
<td>&quot;string&quot;</td>
</tr>
<tr>
<td>&quot;containerSettings&quot;</td>
<td></td>
</tr>
<tr>
<td>&quot;m2tsSettings&quot;</td>
<td></td>
</tr>
<tr>
<td>&quot;audioStreamType&quot;</td>
<td>enum</td>
</tr>
<tr>
<td>&quot;ecmPid&quot;</td>
<td>&quot;string&quot;</td>
</tr>
<tr>
<td>&quot;dvbTeletextPid&quot;</td>
<td>&quot;string&quot;</td>
</tr>
<tr>
<td>&quot;aribCaptionsPidControl&quot;</td>
<td>enum</td>
</tr>
<tr>
<td>&quot;bitrate&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;segmentationTime&quot;</td>
<td>number</td>
</tr>
<tr>
<td>&quot;rateMode&quot;</td>
<td>enum</td>
</tr>
<tr>
<td>&quot;audioPids&quot;</td>
<td>&quot;string&quot;</td>
</tr>
<tr>
<td>&quot;fragmentTime&quot;</td>
<td>number</td>
</tr>
<tr>
<td>&quot;ebpAudioInterval&quot;</td>
<td>enum</td>
</tr>
<tr>
<td>&quot;ebpLookaheadMs&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;audioFramesPerPes&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;scte35Pid&quot;</td>
<td>&quot;string&quot;</td>
</tr>
<tr>
<td>&quot;pcrPeriod&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;pmtInterval&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;programNum&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;segmentationStyle&quot;</td>
<td>enum</td>
</tr>
<tr>
<td>&quot;ebpPlacement&quot;</td>
<td>enum</td>
</tr>
<tr>
<td>&quot;ebpLookaheadMs&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;audioFramesPerPes&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;scte35Control&quot;</td>
<td>enum</td>
</tr>
<tr>
<td>&quot;audioBufferModel&quot;</td>
<td>enum</td>
</tr>
<tr>
<td>&quot;scte27Pids&quot;</td>
<td>&quot;string&quot;</td>
</tr>
<tr>
<td>&quot;nullPacketBitrate&quot;</td>
<td>number</td>
</tr>
<tr>
<td>&quot;dvbSdtSettings&quot;</td>
<td></td>
</tr>
<tr>
<td>&quot;serviceName&quot;</td>
<td>&quot;string&quot;</td>
</tr>
<tr>
<td>&quot;serviceProviderName&quot;</td>
<td>&quot;string&quot;</td>
</tr>
<tr>
<td>&quot;repInterval&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;transportStreamId&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;segmentationMarkers&quot;</td>
<td>enum</td>
</tr>
<tr>
<td>&quot;dvbTdtSettings&quot;</td>
<td></td>
</tr>
<tr>
<td>&quot;repInterval&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;klv&quot;</td>
<td>enum</td>
</tr>
<tr>
<td>&quot;ccDescriptor&quot;</td>
<td>enum</td>
</tr>
<tr>
<td>&quot;patInterval&quot;</td>
<td>integer</td>
</tr>
<tr>
<td>&quot;etvPlatformPid&quot;</td>
<td>&quot;string&quot;</td>
</tr>
<tr>
<td>&quot;dvbSubPids&quot;</td>
<td>&quot;string&quot;</td>
</tr>
<tr>
<td>&quot;aribCaptionsPid&quot;</td>
<td>&quot;string&quot;</td>
</tr>
<tr>
<td>&quot;scte27Pids&quot;</td>
<td>&quot;string&quot;</td>
</tr>
<tr>
<td>&quot;klvDataPids&quot;</td>
<td>&quot;string&quot;</td>
</tr>
</tbody>
</table>
"nameModifier (p. 164)": "string",
"msSmoothOutputSettings (p. 240)": {
  "nameModifier (p. 236)": "string"
},
"udpOutputSettings (p. 240)": {
  "bufferMaec (p. 248)": integer,
  "destination (p. 249)": {
    "destinationRefId (p. 239)": "string"
  },
  "containerSettings (p. 249)": {
    "m2tsSettings (p. 248)": {
      "audioStreamType (p. 222)": enum,
      "ecmPid (p. 223)": "string",
      "dvbTeletextPid (p. 223)": "string",
      "aribCaptionsPidControl (p. 223)": enum,
      "bitrate (p. 223)": integer,
      "segmentationTime (p. 223)": number,
      "rateMode (p. 223)": enum,
      "audioPids (p. 224)": "string",
      "fragmentTime (p. 224)": number,
      "ebpAudioInterval (p. 224)": enum,
      "ebpLookaheadMs (p. 224)": integer,
      "audioFramesPerPes (p. 224)": integer,
      "scte35Pid (p. 224)": "string",
      "pcrPeriod (p. 225)": integer,
      "pmtInterval (p. 225)": integer,
      "programNum (p. 225)": integer,
      "segmentationStyle (p. 225)": enum,
      "ebif (p. 225)": enum,
      "audioBufferModel (p. 225)": enum,
      "dvbNitSettings (p. 226)": {
        "networkName (p. 181)": "string",
        "networkId (p. 181)": integer,
        "repInterval (p. 181)": integer
      },
      "absentInputAudioBehavior (p. 226)": enum,
      "timedMetadataPid (p. 226)": "string",
      "timedMetadataBehavior (p. 226)": enum,
      "etvSignalPid (p. 226)": "string",
      "pmtPid (p. 226)": "string",
      "bufferModel (p. 227)": enum,
      "scte35Control (p. 227)": enum,
      "ebpPlacement (p. 227)": enum,
      "arib (p. 227)": enum,
      "nullPacketBitrate (p. 227)": number,
      "dvbSdtSettings (p. 227)": {
        "serviceName (p. 181)": "string",
        "serviceProviderName (p. 182)": "string",
        "repInterval (p. 182)": integer,
        "outputSdt (p. 182)": enum
      },
      "pcrPid (p. 227)": "string",
      "transportStreamId (p. 228)": integer,
      "pcrControl (p. 228)": enum,
      "videoPid (p. 228)": "string",
      "esRateInPes (p. 228)": enum,
      "segmentationMarkers (p. 228)": enum,
      "dvbTdtSettings (p. 228)": {
        "repInterval (p. 186)": integer
      },
      "klv (p. 229)": enum,
      "ccDescriptor (p. 229)": enum,
      "patInterval (p. 229)": integer,
"etvPlatformPid (p. 229)": "string",
"dvbSubPids (p. 229)": "string",
"aribCaptionsPid (p. 229)": "string",
"scte27Pids (p. 229)": "string",
"klvDataPids (p. 230)": "string"
},
"fecOutputSettings (p. 249)": {
"rowLength (p. 193)": integer,
"columnDepth (p. 193)": integer,
"includeFec (p. 193)": enum
},
"hlsOutputSettings (p. 240)": {
"segmentModifier (p. 213)": "string",
"hlsSettings (p. 213)": {
"audioOnlyHlsSettings (p. 214)": {
"audioTrackType (p. 168)": enum,
"audioGroupId (p. 168)": "string",
"audioOnlyImage (p. 168)": {
"passwordParam (p. 216)": "string",
"uri (p. 216)": "string",
"username (p. 217)": "string"
}
},
"standardHlsSettings (p. 214)": {
"m3u8Settings (p. 246)": {
"pmtPid (p. 230)": "string",
"ecmPid (p. 230)": "string",
"scte35Behavior (p. 230)": enum,
"pcrPid (p. 231)": "string",
"audioPids (p. 231)": "string",
"audioFramesPerPes (p. 231)": integer,
"scte35Pid (p. 231)": "string",
"transportStreamId (p. 231)": integer,
"videoPid (p. 231)": "string",
"pcrControl (p. 231)": enum,
"pmtInterval (p. 232)": integer,
"pcrPeriod (p. 232)": integer,
"programNum (p. 232)": integer,
"patInterval (p. 232)": integer,
"timedMetadataPid (p. 232)": "string",
"timedMetadataBehavior (p. 232)": enum
},
"audioRenditionSets (p. 246)": "string"
}
},
"nameModifier (p. 213)": "string"
},
"audioDescriptionNames (p. 237)": ["string"
]
},
"outputGroupSettings (p. 239)": {
"archiveGroupSettings (p. 239)": {
"destination (p. 163)": {
"destinationRefId (p. 239)": "string"
}
},
"rolloverInterval (p. 163)": integer
},
"rtmpGroupSettings (p. 239)": {
"captionData (p. 241)": enum,
"authenticationScheme (p. 241)": enum,
"cacheLength (p. 241)": integer,
"restartDelay (p. 242)": integer,
"cacheFullBehavior (p. 242)": enum
},
"udpGroupSettings (p. 239)": {
  "inputLossAction (p. 248)": enum,
  "timedMetadataId3Frame (p. 248)": enum,
  "timedMetadataId3Period (p. 248)": integer
},
"msSmoothGroupSettings (p. 239)": {
  "eventId (p. 233)": "string",
  "fragmentLength (p. 233)": integer,
  "timestampOffset (p. 234)": "string",
  "segmentationMode (p. 234)": enum,
  "numRetries (p. 234)": integer,
  "eventStopBehavior (p. 234)": enum,
  "acquisitionPointId (p. 234)": "string",
  "sparseTrackType (p. 234)": enum,
  "timestampOffsetMode (p. 234)": enum,
  "destination (p. 235)": {
    "destinationRefId (p. 239)": "string"
  },
  "audioOnlyTimecodeControl (p. 235)": enum,
  "connectionRetryInterval (p. 235)": integer,
  "filecacheDuration (p. 235)": integer,
  "certificateMode (p. 235)": enum,
  "inputLossAction (p. 235)": enum,
  "sendDelayMs (p. 235)": integer,
  "eventIdMode (p. 236)": enum,
  "restartDelay (p. 236)": integer,
  "streamManifestBehavior (p. 236)": enum
},
"hlsGroupSettings (p. 239)": {
  "segmentsPerSubdirectory (p. 206)": integer,
  "ivInManifest (p. 206)": enum,
  "outputSelection (p. 206)": enum,
  "encryptionType (p. 206)": enum,
  "indexNSegments (p. 206)": integer,
  "destination (p. 206)": {
    "destinationRefId (p. 239)": "string"
  },
  "constantIv (p. 207)": "string",
  "timedMetadataId3Frame (p. 207)": enum,
  "baseUrlManifest (p. 207)": "string",
  "captionLanguageSetting (p. 207)": enum,
  "minSegmentLength (p. 207)": integer,
  "mode (p. 207)": enum,
  "ivSource (p. 208)": enum,
  "manifestCompress (p. 208)": enum,
  "keyProviderSettings (p. 208)": {
    "staticKeySettings (p. 220)": {
      "staticKeyValue (p. 246)": "string",
      "keyProviderServer (p. 246)": {
        "passwordParam (p. 216)": "string",
        "uri (p. 216)": "string",
        "username (p. 217)": "string"
      }
    }
  },
  "tsFileMode (p. 208)": enum,
  "manifestDurationFormat (p. 208)": enum,
  "keyFormatVersions (p. 208)": "string",
  "streamInfResolution (p. 208)": enum,
  "timestampDeltaMilliseconds (p. 209)": integer,
  "segmentationMode (p. 209)": enum,
  "baseUrlContent (p. 209)": "string",
  "clientCache (p. 209)": enum,
"captionLanguageMappings (p. 209)": [
  {
    "languageDescription (p. 177)": "string",
    "captionChannel (p. 177)": integer,
    "languageCode (p. 177)": "string"
  }
],
"codecSpecification (p. 209)": enum,
"keepSegments (p. 209)": integer,
"timedMetadataId3Period (p. 210)": integer,
"programDateTime (p. 210)": enum,
"directoryStructure (p. 210)": enum,
"keyFormat (p. 210)": "string",
"inputLossAction (p. 210)": enum,
"adMarkers (p. 210)": [
  enum
],
"programDateTimePeriod (p. 210)": integer,
"segmentLength (p. 211)": integer,
"hlsCdnSettings (p. 211)": {
  "hlsAkamaiSettings (p. 205)": {
    "httpTransferMode (p. 203)": enum,
    "salt (p. 203)": "string",
    "numRetries (p. 203)": integer,
    "restartDelay (p. 203)": integer,
    "connectionRetryInterval (p. 204)": integer,
    "filecacheDuration (p. 204)": integer,
    "token (p. 204)": "string"
  },
  "hlsWebdavSettings (p. 205)": {
    "httpTransferMode (p. 214)": enum,
    "numRetries (p. 214)": integer,
    "restartDelay (p. 215)": integer,
    "connectionRetryInterval (p. 215)": integer,
    "filecacheDuration (p. 215)": integer
  },
  "hlsBasicPutSettings (p. 205)": {
    "numRetries (p. 204)": integer,
    "restartDelay (p. 204)": integer,
    "connectionRetryInterval (p. 204)": integer,
    "filecacheDuration (p. 205)": integer
  },
  "hlsMediaStoreSettings (p. 205)": {
    "mediaStoreStorageClass (p. 212)": enum,
    "numRetries (p. 212)": integer,
    "restartDelay (p. 212)": integer,
    "connectionRetryInterval (p. 212)": integer,
    "filecacheDuration (p. 213)": integer
  }
},
"name (p. 239)": "string"
],
"audioDescriptions (p. 192)": [
  {
    "languageCodeControl (p. 165)": enum,
    "audioTypeControl (p. 165)": enum,
    "remixSettings (p. 165)": {
      "channelMappings (p. 240)": [
        {
          "outputChannel (p. 164)": integer,
          "inputChannelLevels (p. 164)": [
            {
              "inputChannel (p. 215)": integer,
```
"gain (p. 216)" : integer
}
]
"channelsOut (p. 240)" : integer,
"channelsIn (p. 240)" : integer
"audioType (p. 165)" : enum,
"name (p. 166)" : "string",
"languageCode (p. 166)" : "string",
"codecSettings (p. 166)" : {
  "aacSettings (p. 164)" : {
    "vbrQuality (p. 159)" : enum,
    "codingMode (p. 160)" : enum,
    "profile (p. 160)" : enum,
    "bitrate (p. 160)" : number,
    "inputType (p. 160)" : enum,
    "rawFormat (p. 160)" : enum,
    "sampleRate (p. 160)" : number,
    "rateControlMode (p. 160)" : enum,
    "spec (p. 161)" : enum
  },
  "ac3Settings (p. 164)" : {
    "drcProfile (p. 162)" : enum,
    "dialnorm (p. 162)" : integer,
    "codingMode (p. 162)" : enum,
    "metadataControl (p. 162)" : enum,
    "bitrate (p. 162)" : number,
    "lfeFilter (p. 162)" : enum,
    "bitstreamMode (p. 163)" : enum
  },
  "eac3Settings (p. 165)" : {
    "dialnorm (p. 188)" : integer,
    "passthroughControl (p. 188)" : enum,
    "metadataControl (p. 188)" : enum,
    "drcLine (p. 188)" : enum,
    "bitrate (p. 188)" : number,
    "surroundExMode (p. 188)" : enum,
    "lRefSurroundMixLevel (p. 189)" : number,
    "lfeControl (p. 189)" : enum,
    "codingMode (p. 189)" : enum,
    "surroundMode (p. 189)" : enum,
    "attenuationControl (p. 189)" : enum,
    "lfeFilter (p. 189)" : enum,
    "lRefCenterMixLevel (p. 189)" : number,
    "dcFilter (p. 190)" : enum,
    "phaseControl (p. 190)" : enum,
    "stereoDownmix (p. 190)" : enum,
    "bitstreamMode (p. 190)" : enum,
    "loRoSurroundMixLevel (p. 190)" : number,
    "drcRf (p. 190)" : enum,
    "loRoCenterMixLevel (p. 190)" : number
  },
  "passThroughSettings (p. 165)" : {
  },
  "mp2Settings (p. 165)" : {
    "codingMode (p. 233)" : enum,
    "bitrate (p. 233)" : number,
    "sampleRate (p. 233)" : number
  }
},
"streamName (p. 166)" : "string",
"audioNormalizationSettings (p. 166)" : {
  "targetLkfs (p. 167)" : number,
  "algorithmControl (p. 167)" : enum,
"algorithm (p. 168)": enum
},
"audioSelectorName (p. 166)": "string"
],
"captionDescriptions (p. 192)": [
{
"captionSelectorName (p. 175)": "string",
"languageDescription (p. 175)": "string",
"name (p. 175)": "string",
"languageCode (p. 176)": "string",
"destinationSettings (p. 176)": {
"scte27DestinationSettings (p. 176)": {
},
"burnInDestinationSettings (p. 176)": {
"xPosition (p. 172)": integer,
"backgroundColor (p. 172)": enum,
"yPosition (p. 172)": integer,
"teletextGridControl (p. 172)": enum,
"backgroundColorOpacity (p. 172)": integer,
"fontOpacity (p. 173)": integer,
"fontResolution (p. 173)": integer,
"shadowOpacity (p. 173)": integer,
"shadowYOffset (p. 173)": integer,
"outlineSize (p. 173)": integer,
"outlineColor (p. 173)": enum,
"fontSize (p. 174)": "string",
"shadowXOffset (p. 174)": integer,
"alignment (p. 174)": enum,
"shadowColor (p. 174)": enum,
"fontColor (p. 174)": enum,
"font (p. 174)": {
"passwordParam (p. 216)": "string",
"uri (p. 216)": "string",
"username (p. 217)": "string"
}
},
"teletextDestinationSettings (p. 176)": {
},
"smpteTtDestinationSettings (p. 176)": {
},
"webvttDestinationSettings (p. 176)": {
},
"ttmlDestinationSettings (p. 176)": {
"styleControl (p. 247)": enum
},
"embeddedPlusScte20DestinationSettings (p. 176)": {
},
"dvbSubDestinationSettings (p. 177)": {
"xPosition (p. 183)": integer,
"backgroundColor (p. 183)": enum,
"yPosition (p. 183)": integer,
"teletextGridControl (p. 183)": enum,
"backgroundColorOpacity (p. 183)": integer,
"fontOpacity (p. 184)": integer,
"fontResolution (p. 184)": integer,
"shadowOpacity (p. 184)": integer,
"shadowYOffset (p. 184)": integer,
"outlineSize (p. 184)": integer,
"outlineColor (p. 185)": enum,
"fontSize (p. 185)": "string",
"shadowXOffset (p. 185)": integer,
"alignment (p. 185)": enum,
"shadowColor (p. 185)": enum,
"fontColor (p. 185)": enum,
"font (p. 186)": {

"passwordParam (p. 216)": "string",
"uri (p. 216)": "string",
"username (p. 217)": "string"
},
"embeddedDestinationSettings (p. 177)": {
},
"rtmpCaptionInfoDestinationSettings (p. 177)": {
},
"aribDestinationSettings (p. 177)": {
},
"scte20PlusEmbeddedDestinationSettings (p. 177)": {
}
},
"availConfiguration (p. 192)": {
  "availSettings (p. 170)": {
    "scte35TimeSignalApos (p. 170)": {
      "adAvailOffset (p. 245)": integer,
      "webDeliveryAllowedFlag (p. 245)": enum,
      "noRegionalBlackoutFlag (p. 245)": enum
    },
    "scte35SpliceInsert (p. 170)": {
      "adAvailOffset (p. 244)": integer,
      "webDeliveryAllowedFlag (p. 244)": enum,
      "noRegionalBlackoutFlag (p. 244)": enum
    }
  },
  "globalConfiguration (p. 192)": {
    "inputLossBehavior (p. 194)": {
      "inputLossImageType (p. 217)": enum,
      "inputLossImageColor (p. 217)": "string",
      "inputLossImageSlate (p. 217)": {
        "passwordParam (p. 216)": "string",
        "uri (p. 216)": "string",
        "username (p. 217)": "string"
      },
      "blackFrameMsec (p. 218)": integer,
      "repeatFrameMsec (p. 218)": integer
    },
    "supportLowFramerateInputs (p. 194)": enum,
    "initialAudioGain (p. 194)": integer,
    "inputEndAction (p. 194)": enum,
    "outputTimingSource (p. 195)": enum
  },
  "videoDescriptions (p. 193)": [
    {
      "respondToAfd (p. 250)": enum,
      "scalingBehavior (p. 251)": enum,
      "name (p. 251)": "string",
      "width (p. 251)": integer,
      "sharpness (p. 251)": integer,
      "codecSettings (p. 251)": {
        "h264Settings (p. 250)": {
          "minInterval (p. 197)": integer,
          "slices (p. 197)": integer,
          "parNumerator (p. 197)": integer,
          "gopSizeUnits (p. 198)": enum,
          "maxBitrate (p. 198)": integer,
          "bitrate (p. 198)": integer,
          "bufFillPct (p. 198)": integer,
          "temporalAq (p. 198)": enum,
          "afdSignaling (p. 198)": enum,
          "timecodeInsertion (p. 199)": enum,
          "maxExprFactor (p. 199)": integer
        },
"bufSize (p. 199)": integer,
"softness (p. 199)": integer,
"framerateControl (p. 199)": enum,
"fixedAfd (p. 199)": enum,
"level (p. 199)": enum,
"lookAheadRateControl (p. 199)": enum,
"profile (p. 200)": enum,
"framerateNumerator (p. 200)": integer,
"gopClosedCadence (p. 200)": integer,
"framerateDenominator (p. 200)": integer,
"entropyEncoding (p. 200)": enum,
"spatialAq (p. 200)": enum,
"adaptiveQuantization (p. 200)": enum,
"colorMetadata (p. 201)": enum,
"gopSize (p. 201)": number,
"numRefFrames (p. 201)": integer,
"gopSizeReference (p. 201)": integer,
"sceneChangeDetect (p. 201)": enum,
"parControl (p. 201)": enum,
"parDenominator (p. 201)": integer,
"syntax (p. 202)": enum,
"scanType (p. 202)": enum,
"gopNumBFrames (p. 202)": integer,
"flickerAq (p. 202)": integer,
"rateControlMode (p. 202)": enum
"height (p. 251)": integer
"blackoutSlate (p. 193)": {
  "networkEndBlackoutImage (p. 170)": {
    "passwordParam (p. 216)": "string",
    "uri (p. 216)": "string",
    "username (p. 217)": "string"
  },
  "networkEndBlackout (p. 171)": enum,
  "networkId (p. 171)": "string",
  "state (p. 171)": enum,
  "blackoutSlateImage (p. 171)": {
    "passwordParam (p. 216)": "string",
    "uri (p. 216)": "string",
    "username (p. 217)": "string"
  }
},
"availBlanking (p. 193)": {
  "state (p. 169)": enum,
  "availBlankingImage (p. 170)": {
    "passwordParam (p. 216)": "string",
    "uri (p. 216)": "string",
    "username (p. 217)": "string"
  }
},
"pipelinesRunningCount (p. 179)": integer,
"state (p. 179)": enum,
"id (p. 180)": "string",
"egressEndpoints (p. 180)": [ {
  "sourceIp (p. 180)": "string"
} ],
"arn (p. 180)": "string",
"inputSpecification (p. 180)": {
  "codec (p. 220)": enum,
  "resolution (p. 220)": enum,
"maximumBitrate (p. 220)": enum
}
}

Example InvalidRequest

{
  "message (p. 220)": "string"
}

Example AccessDenied

{
  "message (p. 163)": "string"
}

Example ResourceNotFound

{
  "message (p. 241)": "string"
}

Example ResourceConflict

{
  "message (p. 241)": "string"
}

Example ChannelConfigurationValidationError

{
  "validationErrors (p. 180)": [
    {
      "errorMessage (p. 250)": "string",
      "elementPath (p. 250)": "string"
    }
  ],
  "message (p. 180)": "string"
}

Example LimitExceeded

{
  "message (p. 221)": "string"
}

Example InternalServiceError

{
  "message (p. 220)": "string"
}
Example BadGatewayException

```json
{
  "message (p. 170)": "string"
}
```

Example GatewayTimeoutException

```json
{
  "message (p. 194)": "string"
}
```

Properties

**AacCodingMode (enum)***

- AD_RECEIVER_MIX
- CODING_MODE_1_0
- CODING_MODE_1_1
- CODING_MODE_2_0
- CODING_MODE_5_1

**AacInputType (enum)***

- BROADCASTER_MIXED_AD
- NORMAL

**AacProfile (enum)***

- HEV1
- HEV2
- LC

**AacRateControlMode (enum)***

- CBR
- VBR

**AacRawFormat (enum)***

- LATM_LOAS
- NONE

**AacSettings***

**vbrQuality**

VBR Quality Level - Only used if rateControlMode is VBR.

**Type:** string

**Required:** False
Properties

codingMode
Mono, Stereo, or 5.1 channel layout. Valid values depend on rate control mode and profile. The adReceiverMix 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: string
  Required: False

profile
AAC Profile.

  Type: string
  Required: False

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

  Type: number
  Required: False

inputType
Set to "broadcasterMixedAd" when input contains pre-mixed main audio + AD (narration) as a stereo pair. The Audio Type field (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. The values in audioTypeControl and audioType (in AudioDescription) are ignored when set to broadcasterMixedAd. Leave set to "normal" when input does not contain pre-mixed audio + AD.

  Type: string
  Required: False

rawFormat
Sets LATM / LOAS AAC output for raw containers.

  Type: string
  Required: False

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

  Type: number
  Required: False

rateControlMode
Rate Control Mode.

  Type: string

160
**Required**: False

**spec**

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

**Type**: string

**Required**: False

**AacSpec (enum)**

- MPEG2
- MPEG4

**AacVbrQuality (enum)**

- HIGH
- LOW
- MEDIUM_HIGH
- MEDIUM_LOW

**Ac3BitstreamMode (enum)**

- COMMENTARY
- COMPLETE_MAIN
- DIALOGUE
- EMERGENCY
- HEARING_IMPAIRED
- MUSIC_AND_EFFECTS
- VISUALLY_IMPAIRED
- VOICE_OVER

**Ac3CodingMode (enum)**

- CODING_MODE_1_0
- CODING_MODE_1_1
- CODING_MODE_2_0
- CODING_MODE_3_2_LFE

**Ac3DrcProfile (enum)**

- FILM_STANDARD
- NONE

**Ac3LfeFilter (enum)**

- DISABLED
- ENABLED

**Ac3MetadataControl (enum)**

- FOLLOW_INPUT
USE_CONFIGURED

**Ac3Settings**

**drcProfile**

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

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

**dialnorm**

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

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

**codingMode**

Dolby Digital coding mode. Determines number of channels.

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

**metadataControl**

When set to "followInput", 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**: string
  * **Required**: False

**bitrate**

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

  * **Type**: number
  * **Required**: False

**lfeFilter**

When set to enabled, applies a 120Hz lowpass filter to the LFE channel prior to encoding. Only valid in codingMode32Lfe mode.

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

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

  Type: string
  Required: False

AccessDenied

message

  Type: string
  Required: False

AfdSignaling (enum)

  AUTO
  FIXED
  NONE

ArchiveContainerSettings

m2tsSettings

  Type: M2tsSettings (p. 222)
  Required: False

ArchiveGroupSettings

destination

A directory and base filename where archive files should be written. If the base filename portion of the URI is left blank, the base filename of the first input will be automatically inserted.

  Type: OutputLocationRef (p. 239)
  Required: True

rolloverInterval

Number of seconds to write to archive file before closing and starting a new one.

  Type: integer
  Required: False
  Minimum: 1

ArchiveOutputSettings

extension

Output file extension. If excluded, this will be auto-selected from the container type.
**containerSettings**
Settings specific to the container type of the file.

- **Type:** ArchiveContainerSettings (p. 163)
- **Required:** True

**nameModifier**
String concatenated to the end of the destination filename. Required for multiple outputs of the same type.

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

**AribDestinationSettings**

**AribSourceSettings**

**AudioChannelMapping**

**outputChannel**
The index of the output channel being produced.

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

**inputChannelLevels**
Indices and gain values for each input channel that should be remixed into this output channel.

- **Type:** Array of type InputChannelLevel (p. 215)
- **Required:** True

**AudioCodecSettings**

**aacSettings**

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

**ac3Settings**

- **Type:** Ac3Settings (p. 162)
**Properties**

- **Required**: False

**eac3Settings**

  - **Type**: Eac3Settings (p. 188)
  - **Required**: False

**passThroughSettings**

  - **Type**: PassThroughSettings (p. 240)
  - **Required**: False

**mp2Settings**

  - **Type**: Mp2Settings (p. 233)
  - **Required**: False

**AudioDescription**

**languageCodeControl**

Choosing followInput will cause the ISO 639 language code of the output to follow the ISO 639 language code of the input. The languageCode will be used when useConfigured is set, or when followInput is selected but there is no ISO 639 language code specified by the input.

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

**audioTypeControl**

Determines how audio type is determined. followInput: If the input contains an ISO 639 audioType, then that value is passed through to the output. If the input contains no ISO 639 audioType, the value in Audio Type is included in the output. useConfigured: The value in Audio Type is included in the output. Note that this field and audioType are both ignored if inputType is broadcasterMixedAd.

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

**remixSettings**

Settings that control how input audio channels are remixed into the output audio channels.

  - **Type**: RemixSettings (p. 240)
  - **Required**: False

**audioType**

Applies only if audioTypeControl is useConfigured. The values for audioType are defined in ISO-IEC 13818-1.

  - **Type**: string
  - **Required**: False
name
The name of this AudioDescription. Outputs will use this name to uniquely identify this AudioDescription. Description names should be unique within this Live Event.

    Type: string
    Required: True

languageCode
Indicates the language of the audio output track. Only used if languageControlMode is useConfigured, or there is no ISO 639 language code specified in the input.

    Type: string
    Required: False

codecSettings
Audio codec settings.

    Type: AudioCodecSettings (p. 164)
    Required: False

streamName
Used for MS Smooth and Apple HLS outputs. Indicates the name displayed by the player (eg. English, or Director Commentary).

    Type: string
    Required: False

audioNormalizationSettings
Advanced audio normalization settings.

    Type: AudioNormalizationSettings (p. 167)
    Required: False

audioSelectorName
The name of the AudioSelector used as the source for this AudioDescription.

    Type: string
    Required: True

AudioDescriptionAudioTypeControl (enum)

    FOLLOW_INPUT
    USE_CONFIGURED

AudioDescriptionLanguageCodeControl (enum)

    FOLLOW_INPUT
    USE_CONFIGURED
AudioLanguageSelection

languageSelectionPolicy
When set to "strict", the transport stream demux strictly identifies audio streams by their language descriptor. If a PMT update occurs such that an audio stream matching the initially selected language is no longer present then mute will be encoded until the language returns. If "loose", then on a PMT update the demux will choose another audio stream in the program with the same stream type if it can't find one with the same language.

Type: string  
Required: False

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

Type: string  
Required: True

AudioLanguageSelectionPolicy (enum)

- LOOSE
- STRICT

AudioNormalizationAlgorithm (enum)

- ITU_1770_1
- ITU_1770_2

AudioNormalizationAlgorithmControl (enum)

- CORRECT_AUDIO

AudioNormalizationSettings

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

Type: number  
Required: False  
Minimum: -59.0  
Maximum: 0.0

algorithmControl
When set to correctAudio the output audio is corrected using the chosen algorithm. If set to measureOnly, the audio will be measured but not adjusted.

Type: string  
Required: False
algorithm

Audio normalization algorithm to use. itu17701 conforms to the CALM Act specification, itu17702 conforms to the EBU R-128 specification.

    Type: string
    Required: False

AudioOnlyHlsSettings

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: string
    Required: False

audioGroupId

Specifies the group to which the audio Rendition belongs.

    Type: string
    Required: False

audioOnlyImage

For use with an audio only Stream. Must be a .jpg or .png file. If given, this image will be used as the cover-art for the audio only output. Ideally, it should be formatted for an iPhone screen for two reasons. The iPhone does not resize the image, it crops a centered image on the top/bottom and left/right. Additionally, this image file gets saved bit-for-bit into every 10-second segment file, so will increase bandwidth by \( \text{image file size} \times \text{segment count} \times \text{user count} \).

    Type: InputLocation (p. 216)
    Required: False

AudioOnlyHlsTrackType (enum)

    ALTERNATE_AUDIO_AUTO_SELECT
    ALTERNATE_AUDIO_AUTO_SELECT_DEFAULT
    ALTERNATE_AUDIO_NOT_AUTO_SELECT
    AUDIO_ONLY_VARIANT_STREAM

AudioPidSelection

pid

Selects a specific PID from within a source.
Type: integer
Required: True
Minimum: 0
Maximum: 8191

**AudioSelector**

**name**

The name of this AudioSelector. AudioDescriptions will use this name to uniquely identify this Selector. Selector names should be unique per input.

Type: string
Required: True

**selectorSettings**

The audio selector settings.

Type: AudioSelectorSettings (p. 169)
Required: False

**AudioSelectorSettings**

**audioLanguageSelection**

Type: AudioLanguageSelection (p. 167)
Required: False

**audioPidSelection**

Type: AudioPidSelection (p. 168)
Required: False

**AudioType** (enum)

- CLEAN_EFFECTS
- HEARING_IMPAIRED
- UNDEFINED
- VISUAL_IMPAIRED_COMMENTARY

**AuthenticationScheme** (enum)

- AKAMAI
- COMMON

**AvailBlanking**

**state**

When set to enabled, causes video, audio and captions to be blanked when insertion metadata is added.
Properties

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

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

**Type**: `InputLocation (p. 216)`  
**Required**: False

### AvailBlankingState (enum)

- DISABLED
- ENABLED

### AvailConfiguration

#### availSettings
Ad avail settings.

**Type**: `AvailSettings (p. 170)`  
**Required**: False

### AvailSettings

#### scte35TimeSignalApos

**Type**: `Scte35TimeSignalApos (p. 245)`  
**Required**: False

#### scte35SpliceInsert

**Type**: `Scte35SpliceInsert (p. 244)`  
**Required**: False

### BadGatewayException

#### message

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

### BlackoutSlate

#### networkEndBlackoutImage
Path to local file to use as Network End Blackout image. Image will be scaled to fill the entire output raster.

**Type**: `InputLocation (p. 216)`
Required: False

networkEndBlackout
Setting to enabled causes the encoder to blackout the video, audio, and captions, and raise the "Network Blackout Image" slate when an SCTE104/35 Network End Segmentation Descriptor is encountered. The blackout will be lifted when the Network Start Segmentation Descriptor is encountered. The Network End and Network Start descriptors must contain a network ID that matches the value entered in "Network ID".

Type: string
Required: False

networkId
Provides Network ID that matches EIDR ID format (e.g., "10.XXXX/XXXX-XXXX-XXXX-XXXX-XXXX-XXXX-C").

Type: string
Required: False

state
When set to enabled, causes video, audio and captions to be blanked when indicated by program metadata.

Type: string
Required: False

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

Type: InputLocation (p. 216)
Required: False

BlackoutSlateNetworkEndBlackout (enum)

DISABLED  
ENABLED  

BlackoutSlateState (enum)

DISABLED  
ENABLED  

BurnInAlignment (enum)

CENTERED  
LEFT  
SMART  

BurnInBackgroundColor (enum)

BLACK
NONE
WHITE

**BurnInDestinationSettings**

**xPosition**

Specifies the horizontal position of the caption relative to the left side of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the left of the output. If no explicit xPosition is provided, the horizontal caption position will be determined by the alignment parameter. All burn-in and DVB-Sub font settings must match.

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

**backgroundColor**

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

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

**yPosition**

Specifies the vertical position of the caption relative to the top of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the top of the output. If no explicit yPosition is provided, the caption will be positioned towards the bottom of the output. All burn-in and DVB-Sub font settings must match.

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

**teletextGridControl**

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

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

**backgroundOpacity**

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

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

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

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

fontResolution

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

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

shadowOpacity

Specifies the opacity of the shadow. 255 is opaque; 0 is transparent. Leaving this parameter out 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

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

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

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: string
Required: False

**fontSize**

When set to 'auto' fontSize will scale depending on the size of the output. Giving a positive integer will specify the exact font size in points. All burn-in and DVB-Sub font settings must match.

Type: string
Required: False

**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

**alignment**

If no explicit xPosition or yPosition 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. Selecting "smart" justification will left-justify live subtitles and center-justify pre-recorded subtitles. All burn-in and DVB-Sub font settings must match.

Type: string
Required: False

**shadowColor**

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

Type: string
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: string
Required: False

**font**

External font file used for caption burn-in. File extension must be 'ttf' or 'tte'. Although the user can select output fonts for many different types of input captions, embedded, STL and teletext sources use a strict grid system. Using external fonts with these caption sources could cause unexpected display of proportional fonts. All burn-in and DVB-Sub font settings must match.
Type: InputLocation (p. 216)  
Required: False

**BurnInFontColor (enum)**
BLACK  
BLUE  
GREEN  
RED  
WHITE  
YELLOW

**BurnInOutlineColor (enum)**
BLACK  
BLUE  
GREEN  
RED  
WHITE  
YELLOW

**BurnInShadowColor (enum)**
BLACK  
NONE  
WHITE

**BurnInTeletextGridControl (enum)**
FIXED  
SCALED

**CaptionDescription**

**captionSelectorName**
Specifies which input caption selector to use as a caption source when generating output captions. This field should match a captionSelector name.

Type: string  
Required: True

**languageDescription**
Human readable information to indicate captions available for players (eg. English, or Spanish).

Type: string  
Required: False

**name**
Name of the caption description. Used to associate a caption description with an output. Names must be unique within an event.
Properties

Type: string  
Required: True

languageCode

Type: string  
Required: False

destinationSettings

Additional settings for captions destination that depend on the destination type.  
Type: CaptionDestinationSettings (p. 176)  
Required: False

CaptionDestinationSettings

scte27DestinationSettings

Type: Scte27DestinationSettings (p. 243)  
Required: False

burnInDestinationSettings

Type: BurnInDestinationSettings (p. 172)  
Required: False

telextDestinationSettings

Type: TeletextDestinationSettings (p. 247)  
Required: False

smpteTtDestinationSettings

Type: SmpteTtDestinationSettings (p. 246)  
Required: False

webvttDestinationSettings

Type: WebvttDestinationSettings (p. 253)  
Required: False

ttmlDestinationSettings

Type: TtmlDestinationSettings (p. 247)  
Required: False

embeddedPlusScte20DestinationSettings

Type: EmbeddedPlusScte20DestinationSettings (p. 191)
Properties

**Required**: False

**dvbSubDestinationSettings**

*Type*: DvbSubDestinationSettings (p. 183)

*Required*: False

**embeddedDestinationSettings**

*Type*: EmbeddedDestinationSettings (p. 191)

*Required*: False

**rtmpCaptionInfoDestinationSettings**

*Type*: RtmpCaptionInfoDestinationSettings (p. 241)

*Required*: False

**aribDestinationSettings**

*Type*: AribDestinationSettings (p. 164)

*Required*: False

**scte20PlusEmbeddedDestinationSettings**

*Type*: Scte20PlusEmbeddedDestinationSettings (p. 243)

*Required*: False

**CaptionLanguageMapping**

**languageDescription**

Textual description of language

*Type*: string

*Required*: True

**captionChannel**

Channel to insert closed captions. Each channel mapping must have a unique channel number (maximum of 4)

*Type*: integer

*Required*: True

*Minimum*: 1

*Maximum*: 4

**languageCode**

Three character ISO 639-2 language code (see http://www.loc.gov/standards/iso639-2)

*Type*: string

*Required*: True
**CaptionSelector**

*name*

Name identifier for a caption selector. This name is used to associate this caption selector with one or more caption descriptions. Names must be unique within an event.

*Type: string*
*Required: True*

*languageCode*

When specified this field indicates the three letter language code of the caption track to extract from the source.

*Type: string*
*Required: False*

*selectorSettings*

Caption selector settings.

*Type: CaptionSelectorSettings (p. 178)*
*Required: False*

**CaptionSelectorSettings**

*embeddedSourceSettings*

*Type: EmbeddedSourceSettings (p. 191)*
*Required: False*

*scte20SourceSettings*

*Type: Scte20SourceSettings (p. 243)*
*Required: False*

*dvbSubSourceSettings*

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

*aribSourceSettings*

*Type: AribSourceSettings (p. 164)*
*Required: False*

*teletextSourceSettings*

*Type: TeletextSourceSettings (p. 247)*
*Required: False*
scte27SourceSettings
  Type: Scte27SourceSettings (p. 243)
  Required: False

Channel

inputAttachments
List of input attachments for channel.
  Type: Array of type InputAttachment (p. 215)
  Required: False

roleArn
The Amazon Resource Name (ARN) of the role assumed when running the Channel.
  Type: string
  Required: False

destinations
A list of destinations of the channel. For UDP outputs, there is one destination per output. For other types (HLS, for example), there is one destination per packager.
  Type: Array of type OutputDestination (p. 238)
  Required: False

name
The name of the channel. (user-mutable)
  Type: string
  Required: False

encoderSettings
  Type: EncoderSettings (p. 192)
  Required: False

pipelinesRunningCount
The number of currently healthy pipelines.
  Type: integer
  Required: False

state
  Type: string
  Required: False
id
The unique id of the channel.

  Type: string
  Required: False

egressEndpoints
The endpoints where outgoing connections initiate from

  Type: Array of type ChannelEgressEndpoint (p. 180)
  Required: False

arn
The unique arn of the channel.

  Type: string
  Required: False

inputSpecification

  Type: InputSpecification (p. 220)
  Required: False

ChannelConfigurationValidationError

validationErrors
A collection of validation error responses from attempting to create a channel with a bouquet of settings.

  Type: Array of type ValidationError (p. 250)
  Required: False

message

  Type: string
  Required: False

ChannelEgressEndpoint

sourceIp
Public IP of where a channel's output comes from

  Type: string
  Required: False

ChannelState (enum)

  CREATING
CREATE_FAILED
IDLE
STARTING
RUNNING
RECOVERING
STOPPING
DELETING
DELETED

DvbNitSettings

networkName

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

Type: string
Required: True

networkId

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

Type: integer
Required: True
Minimum: 0
Maximum: 65536

repInterval

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

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

DvbSdtOutputSdt (enum)

SDT_FOLLOW
SDT_FOLLOW_IF_PRESENT
SDT_MANUAL
SDT_NONE

DvbSdtSettings

serviceName

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

Type: string
Required: False

**serviceProviderName**

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

*Type: string*

Required: False

**repInterval**

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

*Type: integer*

Required: False

Minimum: 25

Maximum: 2000

**outputSdt**

Selects method of inserting SDT information into output stream. The sdtFollow setting copies SDT information from input stream to output stream. The sdtFollowIfPresent setting 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. The sdtManual setting means user will enter the SDT information. The sdtNone setting means output stream will not contain SDT information.

*Type: string*

Required: False

**DvbSubDestinationAlignment (enum)**

CENTERED
LEFT
SMART

**DvbSubDestinationBackgroundColor (enum)**

BLACK
NONE
WHITE

**DvbSubDestinationFontColor (enum)**

BLACK
BLUE
GREEN
RED
WHITE
YELLOW

**DvbSubDestinationOutlineColor (enum)**

BLACK
DvbSubDestinationSettings

xPosition

Specifies the horizontal position of the caption relative to the left side of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the left of the output. If no explicit xPosition 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

backgroundColor

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

Type: string  
Required: False

yPosition

Specifies the vertical position of the caption relative to the top of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the top of the output. If no explicit yPosition 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

teletextGridControl

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

Type: string  
Required: False

backgroundOpacity

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

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

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

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

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

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

**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

**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

**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
### 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**: string
- **Required**: False

### fontSize

When set to auto fontSize will scale depending on the size of the output. Giving a positive integer will specify the exact font size in points. All burn-in and DVB-Sub font settings must match.

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

### 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

### alignment

If no explicit xPosition or yPosition 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. Selecting "smart" justification will left-justify live subtitles and center-justify pre-recorded subtitles. This option is not valid for source captions that are STL or 608/embedded. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

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

### shadowColor

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

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

External font file used for caption burn-in. File extension must be 'ttf' or 'tte'. Although the user can select output fonts for many different types of input captions, embedded, STL and teletext sources use a strict grid system. Using external fonts with these caption sources could cause unexpected display of proportional fonts. All burn-in and DVB-Sub font settings must match.

  Type: InputLocation (p. 216)
  Required: False

DvbSubDestinationShadowColor (enum)

BLACK
NONE
WHITE

DvbSubDestinationTeletextGridControl (enum)

FIXED
SCALED

DvbSubSourceSettings

pid

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

  Type: integer
  Required: False
  Minimum: 1

DvbTdtSettings

repInterval

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

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

Eac3AttenuationControl (enum)

ATTENUATE_3_DB
NONE

Eac3BitstreamMode (enum)

COMMENTARY
COMPLETE_MAIN
EMERGENCY
HEARING_IMPAIRED
VISUALLY_IMPAIRED

Eac3CodingMode (enum)
CODING_MODE_1_0
CODING_MODE_2_0
CODING_MODE_3_2

Eac3DcFilter (enum)
DISABLED
ENABLED

Eac3DrcLine (enum)
FILM_LIGHT
FILM_STANDARD
MUSIC_LIGHT
MUSIC_STANDARD
NONE
SPEECH

Eac3DrcRf (enum)
FILM_LIGHT
FILM_STANDARD
MUSIC_LIGHT
MUSIC_STANDARD
NONE
SPEECH

Eac3LfeControl (enum)
LFE
NO_LFE

Eac3LfeFilter (enum)
DISABLED
ENABLED

Eac3MetadataControl (enum)
FOLLOW_INPUT
USE_CONFIGURED

Eac3PassthroughControl (enum)
NO_PASSTHROUGH
WHEN_POSSIBLE
**Eac3PhaseControl (enum)**

- NO_SHIFT
- SHIFT_90_DEGREES

**Eac3Settings**

**dialnorm**

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

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

**passthroughControl**

When set to whenPossible, 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:** string
- **Required:** False

**metadataControl**

When set to followInput, 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:** string
- **Required:** False

**drcLine**

Sets the Dolby dynamic range compression profile.

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

**bitrate**

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

- **Type:** number
- **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.
<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>ltRtSurroundMixLevel</td>
<td>string</td>
<td>False</td>
</tr>
<tr>
<td>lfeControl</td>
<td>string</td>
<td>False</td>
</tr>
<tr>
<td>codingMode</td>
<td>string</td>
<td>False</td>
</tr>
<tr>
<td>surroundMode</td>
<td>string</td>
<td>False</td>
</tr>
<tr>
<td>attenuationControl</td>
<td>string</td>
<td>False</td>
</tr>
<tr>
<td>lfeFilter</td>
<td>string</td>
<td>False</td>
</tr>
<tr>
<td>ltRtCenterMixLevel</td>
<td>number</td>
<td>False</td>
</tr>
</tbody>
</table>

**ltRtSurroundMixLevel**

Left total/Right total surround mix level. Only used for 3/2 coding mode.

**lfeControl**

When encoding 3/2 audio, setting to lfe enables the LFE channel.

**codingMode**

Dolby Digital Plus coding mode. Determines number of channels.

**surroundMode**

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

**attenuationControl**

When set to attenuate3Db, applies a 3 dB attenuation to the surround channels. Only used for 3/2 coding mode.

**lfeFilter**

When set to enabled, applies a 120Hz lowpass filter to the LFE channel prior to encoding. Only valid with codingMode32 coding mode.

**ltRtCenterMixLevel**

Left total/Right total center mix level. Only used for 3/2 coding mode.
dcFilter
When set to enabled, activates a DC highpass filter for all input channels.

  Type: string
  Required: False

phaseControl
When set to shift90Degrees, applies a 90-degree phase shift to the surround channels. Only used for 3/2 coding mode.

  Type: string
  Required: False

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

  Type: string
  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: string
  Required: False

loRoSurroundMixLevel
Left only/Right only surround mix level. Only used for 3/2 coding mode.

  Type: number
  Required: False

drcRf
Sets the profile for heavy Dolby dynamic range compression, ensures that the instantaneous signal peaks do not exceed specified levels.

  Type: string
  Required: False

loRoCenterMixLevel
Left only/Right only center mix level. Only used for 3/2 coding mode.

  Type: number
  Required: False

Eac3StereoDownmix (enum)

DPL2
LO_RO
LT_RT
NOT_INDICATED

Eac3SurroundExMode (enum)
  DISABLED
  ENABLED
  NOT_INDICATED

Eac3SurroundMode (enum)
  DISABLED
  ENABLED
  NOT_INDICATED

EmbeddedConvert608To708 (enum)
  DISABLED
  UPCONVERT

EmbeddedDestinationSettings

EmbeddedPlusScte20DestinationSettings

EmbeddedScte20Detection (enum)
  AUTO
  OFF

EmbeddedSourceSettings

scte20Detection
Set to "auto" to handle streams with intermittent and/or non-aligned SCTE-20 and Embedded captions.
  Type: string
  Required: False

source608ChannelNumber
Specifies the 608/708 channel number within the video track from which to extract captions. Unused for passthrough.
  Type: integer
  Required: False
  Minimum: 1
  Maximum: 4

convert608To708
If 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.
**source608TrackNumber**

This field is unused and deprecated.

- **Type:** integer
- **Required:** False
- **Minimum:** 1
- **Maximum:** 5

---

**EncoderSettings**

**timecodeConfig**

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

- **Type:** TimecodeConfig (p. 247)
- **Required:** True

**outputGroups**

- **Type:** Array of type OutputGroup (p. 238)
- **Required:** True

**audioDescriptions**

- **Type:** Array of type AudioDescription (p. 165)
- **Required:** True

**captionDescriptions**

Settings for caption descriptions

- **Type:** Array of type CaptionDescription (p. 175)
- **Required:** False

**availConfiguration**

Event-wide configuration settings for ad avail insertion.

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

**globalConfiguration**

Configuration settings that apply to the event as a whole.

- **Type:** GlobalConfiguration (p. 194)
- **Required:** False
videoDescriptions
Type: Array of type VideoDescription (p. 250)
Required: True

blackoutSlate
Settings for blackout slate.
Type: BlackoutSlate (p. 170)
Required: False

availBlanking
Settings for ad avail blanking.
Type: AvailBlanking (p. 169)
Required: False

FecOutputIncludeFec (enum)
COLUMN
COLUMN_AND_ROW

FecOutputSettings
rowLength
Parameter L from SMPTE 2022-1. The width of the FEC protection matrix. Must be between 1 and 20, inclusive. If only Column FEC is used, then larger values increase robustness. If Row FEC is used, then this is the number of transport stream packets per row error correction packet, and the value must be between 4 and 20, inclusive, if includeFec is columnAndRow. If includeFec is column, this value must be 1 to 20, inclusive.
Type: integer
Required: False
Minimum: 1
Maximum: 20

columnDepth
Parameter D from SMPTE 2022-1. The height of the FEC protection matrix. The number of transport stream packets per column error correction packet. Must be between 4 and 20, inclusive.
Type: integer
Required: False
Minimum: 4
Maximum: 20

includeFec
Enables column only or column and row based FEC
Type: string
**Required**: False

**FixedAfd (enum)**

- `AFD_0000`
- `AFD_0010`
- `AFD_0011`
- `AFD_0100`
- `AFD_1000`
- `AFD_1001`
- `AFD_1010`
- `AFD_1011`
- `AFD_1101`
- `AFD_1110`
- `AFD_1111`

**GatewayTimeoutException**

**message**

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

**GlobalConfiguration**

**inputLossBehavior**

Settings for system actions when input is lost.

- **Type**: `InputLossBehavior` (p. 217)
- **Required**: False

**supportLowFramerateInputs**

Adjusts video input buffer for streams with very low video framerates. This is commonly set to enabled for music channels with less than one video frame per second.

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

**initialAudioGain**

Value to set the initial audio gain for the Live Event.

- **Type**: integer
  - **Required**: False
  - **Minimum**: -60
  - **Maximum**: 60

**inputEndAction**

Indicates the action to take when an input completes (e.g. end-of-file.) Options include immediately switching to the next sequential input (via "switchInput"), switching to the next input and looping.
back to the first input when last input ends (via "switchAndLoopInputs") or not switching inputs and instead transcoding black / color / slate images per the "Input Loss Behavior" configuration until an activateInput REST command is received (via "none").

Type: string
Required: False

outputTimingSource

Indicates whether the rate of frames emitted by the Live encoder should be paced by its system clock (which optionally may be locked to another source via NTP) or should be locked to the clock of the source that is providing the input stream.

Type: string
Required: False

GlobalConfigurationInputEndAction (enum)

NONE
SWITCH_AND_LOOP_INPUTS

GlobalConfigurationLowFramerateInputs (enum)

DISABLED
ENABLED

GlobalConfigurationOutputTimingSource (enum)

INPUT_CLOCK
SYSTEM_CLOCK

H264AdaptiveQuantization (enum)

HIGH
HIGHER
LOW
MAX
MEDIUM
OFF

H264ColorMetadata (enum)

IGNORE
INSERT

H264EntropyEncoding (enum)

CABAC
CAVLC

H264FlickerAq (enum)

DISABLED
**ENABLED**

**H264FramerateControl (enum)**

INITIALIZE_FROM_SOURCE
SPECIFIED

**H264GopBReference (enum)**

DISABLED
ENABLED

**H264GopSizeUnits (enum)**

FRAMES
SECONDS

**H264Level (enum)**

H264_LEVEL_1
H264_LEVEL_1_1
H264_LEVEL_1_2
H264_LEVEL_1_3
H264_LEVEL_2
H264_LEVEL_2_1
H264_LEVEL_2_2
H264_LEVEL_2_3
H264_LEVEL_3
H264_LEVEL_3_1
H264_LEVEL_3_2
H264_LEVEL_4
H264_LEVEL_4_1
H264_LEVEL_4_2
H264_LEVEL_5
H264_LEVEL_5_1
H264_LEVEL_5_2
H264_LEVEL_AUTO

**H264LookAheadRateControl (enum)**

HIGH
LOW
MEDIUM

**H264ParControl (enum)**

INITIALIZE_FROM_SOURCE
SPECIFIED

**H264Profile (enum)**

BASELINE
Properties

HIGH
HIGH_10BIT
HIGH_422
HIGH_422_10BIT
MAIN

H264RateControlMode (enum)

CBR
VBR

H264ScanType (enum)

INTERLACED
PROGRESSIVE

H264SceneChangeDetect (enum)

DISABLED
ENABLED

H264Settings

minIInterval

Only meaningful if sceneChangeDetect is set to enabled. 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. Note: Maximum GOP stretch = GOP size + Min-I-interval - 1

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

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. This field is optional; when no value is specified the encoder will choose the number of slices based on encode resolution.

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

parNumerator

Pixel Aspect Ratio numerator.
Properties

**Type**: integer  
**Required**: False

**gopSizeUnits**
Indicates if the gopSize is specified in frames or seconds. If seconds the system will convert the gopSize into a frame count at run time.

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

**maxBitrate**
Maximum bitrate in bits/second (for VBR mode only).

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

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

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

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

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

**temporalAq**
If set to enabled, adjust quantization within each frame based on temporal variation of content complexity.

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

**afdSignaling**
Indicates that AFD values will be written into the output stream. If afdSignaling is "auto", the system will try to preserve the input AFD value (in cases where multiple AFD values are valid). If set to “fixed”, the AFD value will be the value configured in the fixedAfd parameter.

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

Determines how timecodes should be inserted into the video elementary stream. - 'disabled': Do not include timecodes - 'picTimingSei': Pass through picture timing SEI messages from the source specified in Timecode Config

*Type:* string  
*Required:* False

**bufSize**

Size of buffer (HRD buffer model) in bits/second.

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

**softness**

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

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

**framerateControl**

This field indicates how the output video frame rate is specified. If "specified" is selected then the output video frame rate is determined by framerateNumerator and framerateDenominator, else if "initializeFromSource" is selected then the output video frame rate will be set equal to the input video frame rate of the first input.

*Type:* string  
*Required:* False

**fixedAfD**

Four bit AFD value to write on all frames of video in the output stream. Only valid when afdSignaling is set to 'Fixed'.

*Type:* string  
*Required:* False

**level**

H.264 Level.

*Type:* string  
*Required:* False

**lookAheadRateControl**

Amount of lookahead. A value of low can decrease latency and memory usage, while high can produce better quality for certain content.
Properties

- **profile**
  - Type: string
  - Required: False
  - H.264 Profile.

- **frameratenumerator**
  - Type: integer
  - Required: False
  - Framerate numerator - framerate is a fraction, e.g. 24000 / 1001 = 23.976 fps.

- **gopcloscadence**
  - Type: integer
  - Required: False
  - 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.

- **framerateDenominator**
  - Type: integer
  - Required: False
  - Framerate denominator.

- **entropyEncoding**
  - Type: string
  - Required: False
  - Entropy encoding mode. Use cabac (must be in Main or High profile) or cavlc.

- **spatialAq**
  - Type: string
  - Required: False
  - If set to enabled, adjust quantization within each frame based on spatial variation of content complexity.

- **adaptiveQuantization**
  - Type: string
  - Required: False
  - Adaptive quantization. Allows intra-frame quantizers to vary to improve visual quality.
colorMetadata
Includes colorspace metadata in the output.

  Type: string
  Required: False

gopSize
GOP size (keyframe interval) in units of either frames or seconds per gopSizeUnits. Must be greater than zero.

  Type: number
  Required: False
  Minimum: 1.0

numRefFrames
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

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

  Type: string
  Required: False

sceneChangeDetect
Scene change detection. Inserts I-frames on scene changes when enabled.

  Type: string
  Required: False

parControl
This field indicates how the output pixel aspect ratio is specified. If "specified" is selected then the output video pixel aspect ratio is determined by parNumerator and parDenominator, else if "initializeFromSource" is selected then the output pixel aspect ratio will be set equal to the input video pixel aspect ratio of the first input.

  Type: string
  Required: False

parDenominator
Pixel Aspect Ratio denominator.

  Type: integer
Properties

**syntax**

Produces a bitstream compliant with SMPTE RP-2027.

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

**scanType**

Sets the scan type of the output to progressive or top-field-first interlaced.

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

**gopNumBFrames**

Number of B-frames between reference frames.

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

**flickerAq**

If set to enabled, adjust quantization within each frame to reduce flicker or 'pop' on I-frames.

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

**rateControlMode**

Rate control mode.

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

**H264SpatialAq (enum)**

- DISABLED
- ENABLED

**H264Syntax (enum)**

- DEFAULT
- RP2027

**H264TemporalAq (enum)**

- DISABLED
ENABLED

**H264TimecodeInsertionBehavior (enum)**
- DISABLED
- PIC_TIMING_SEI

**HlsAdMarkers (enum)**
- ADOBE
- ELEMENTAL
- ELEMENTAL_SCTE35

**HlsAkamaiHttpTransferMode (enum)**
- CHUNKED
- NON_CHUNKED

**HlsAkamaiSettings**

**httpTransferMode**
Specify whether or not to use chunked transfer encoding to Akamai. User should contact Akamai to enable this feature.
- **Type**: string
- **Required**: False

**salt**
Salt for authenticated Akamai.
- **Type**: string
- **Required**: False

**numRetries**
Number of retry attempts that will be made before the Live Event is put into an error state.
- **Type**: integer
- **Required**: False
- **Minimum**: 0

**restartDelay**
If a streaming output fails, number of seconds to wait until a restart is initiated. A value of 0 means never restart.
- **Type**: integer
- **Required**: False
- **Minimum**: 0
Maximum: 15

**connectionRetryInterval**

Number of seconds to wait before retrying connection to the CDN if the connection is lost.

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

**filecacheDuration**

Size in seconds of file cache for streaming outputs.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 600

**token**

Token parameter for authenticated akamai. If not specified, _gda_ is used.

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

**HlsBasicPutSettings**

**numRetries**

Number of retry attempts that will be made before the Live Event is put into an error state.

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

**restartDelay**

If a streaming output fails, number of seconds to wait until a restart is initiated. A value of 0 means never restart.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 15

**connectionRetryInterval**

Number of seconds to wait before retrying connection to the CDN if the connection is lost.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
filecacheDuration

Size in seconds of file cache for streaming outputs.

Type: integer
Required: False
Minimum: 0
Maximum: 600

HlsCaptionLanguageSetting (enum)

INSERT
NONE
OMIT

HlsCdnSettings

hlsAkamaiSettings

Type: HlsAkamaiSettings (p. 203)
Required: False

hlsWebdavSettings

Type: HlsWebdavSettings (p. 214)
Required: False

hlsBasicPutSettings

Type: HlsBasicPutSettings (p. 204)
Required: False

hlsMediaStoreSettings

Type: HlsMediaStoreSettings (p. 212)
Required: False

HlsClientCache (enum)

DISABLED
ENABLED

HlsCodecSpecification (enum)

RFC_4281
RFC_6381

HlsDirectoryStructure (enum)

SINGLE_DIRECTORY
SUBDIRECTORY_PER_STREAM
Properties

HlsEncryptionType (enum)

- AES128
- SAMPLE_AES

HlsGroupSettings

segmentsPerSubdirectory

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

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

ivInManifest

For use with encryptionType. The IV (Initialization Vector) is a 128-bit number used in conjunction with the key for encrypting blocks. If set to "include", IV is listed in the manifest, otherwise the IV is not in the manifest.

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

outputSelection

Generates the .m3u8 playlist file for this HLS output group. The segmentsOnly option will output segments without the .m3u8 file.

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

encryptionType

Encrypts the segments with the given encryption scheme. Exclude this parameter if no encryption is desired.

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

indexNSegments

Number of segments to keep in the playlist (.m3u8) file. mode must be "vod" for this setting to have an effect, and this number should be less than or equal to keepSegments.

- **Type**: integer
- **Required**: False
- **Minimum**: 3

destination

A directory or HTTP destination for the HLS segments, manifest files, and encryption keys (if enabled).
**Type**: OutputLocationRef (p. 239)
**Required**: True

**constantIv**

For use with encryptionType. This is a 128-bit, 16-byte hex value represented by a 32-character text string. If ivSource is set to "explicit" then this parameter is required and is used as the IV for encryption.

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

**timedMetadataId3Frame**

Indicates ID3 frame that has the timecode.

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

**baseUrlManifest**

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

**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=NULL line in the manifest. omit: Omit any CLOSED-CAPTIONS line from the manifest.

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

**minSegmentLength**

When set, minimumSegmentLength 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

**mode**

If set to "vod", keeps and indexes all segments starting with the first segment. If set to "live" segments will age out and only the last keepSegments number of segments will be retained.

**Type**: string
Required: False

ivSource
For use with encryptionType. The IV (Initialization Vector) is a 128-bit number used in conjunction with
the key for encrypting blocks. If this setting is “followsSegmentNumber”, it will cause the IV to change
every segment (to match the segment number). If this is set to "explicit", you must enter a constantIV
value.

Type: string
Required: False

manifestCompression
When set to gzip, compresses HLS playlist.

Type: string
Required: False

keyProviderSettings
The key provider settings.

Type: KeyProviderSettings (p. 220)
Required: False

tsFileMode
When set to "singleFile", emits the program as a single media resource (.ts) file, and uses #EXT-X-
BYTE R ANGE tags to index segment for playback. Playback of VOD mode content during event is not
guaranteed due to HTTP server caching.

Type: string
Required: False

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

Type: string
Required: False

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

Type: string
Required: False

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

Type: string
Required: False
timestampDeltaMilliseconds
Provides an extra millisecond delta offset to fine tune the timestamps.

  Type: integer
  Required: False
  Minimum: 0

segmentationMode
When set to useInputSegmentation, the output segment or fragment points are set by the RAI markers from the input streams.

  Type: string
  Required: False

baseUrlContent
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

clientCache
When set to "disabled", sets the #EXT-X-ALLOW-CACHE:no tag in the manifest, which prevents clients from saving media segments for later replay.

  Type: string
  Required: False

captionLanguageMappings
Mapping of up to 4 caption channels to caption languages. Is only meaningful if captionLanguageSetting is set to "insert".

  Type: Array of type CaptionLanguageMapping (p. 177)
  Required: False

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

  Type: string
  Required: False

keepSegments
Number of segments to retain in the destination directory. mode must be "live" for this setting to have an effect.

  Type: integer
  Required: False
  Minimum: 1
timedMetadataId3Period
Timed Metadata interval in seconds.

Type: integer
Required: False
Minimum: 0

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 timestampOffset.

Type: string
Required: False

directoryStructure
Place segments in subdirectories.

Type: string
Required: False

keyFormat
The value specifies how the key is represented in the resource identified by the URI. If parameter is absent, an implicit value of "identity" is used. A reverse DNS string can also be given.

Type: string
Required: False

inputLossAction
Parameter that control output group behavior on input loss.

Type: string
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 string
Required: False

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

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

**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

**hlsCdnSettings**

Parameters that control interactions with the CDN.

- **Type**: HlsCdnSettings (p. 205)
- **Required**: False

**HlsInputSettings**

**retries**

The number of consecutive times that attempts to read a manifest or segment must fail before the input is considered unavailable.

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

**bandwidth**

When specified the HLS stream with the m3u8 BANDWIDTH that most closely matches this value will be chosen, otherwise the highest bandwidth stream in the m3u8 will be chosen. The bitrate is specified in bits per second, as in an HLS manifest.

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

**retryInterval**

The number of seconds between retries when an attempt to read a manifest or segment fails.

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

**bufferSegments**

When specified, reading of the HLS input will begin this many buffer segments from the end (most recently written segment). When not specified, the HLS input will begin with the first segment specified in the m3u8.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
**Properties**

**HlsIvInManifest (enum)**

- EXCLUDE
- INCLUDE

**HlsIvSource (enum)**

- EXPLICIT
--follows-segment-number

**HlsManifestCompression (enum)**

- GZIP
- NONE

**HlsManifestDurationFormat (enum)**

- FLOATING_POINT
- INTEGER

**HlsMediaStoreSettings**

**mediaStoreStorageClass**

When set to temporal, output files are stored in non-persistent memory for faster reading and writing.

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

**numRetries**

Number of retry attempts that will be made before the Live Event is put into an error state.

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

**restartDelay**

If a streaming output fails, number of seconds to wait until a restart is initiated. A value of 0 means never restart.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 15

**connectionRetryInterval**

Number of seconds to wait before retrying connection to the CDN if the connection is lost.

- **Type:** integer
- **Required:** False
filecacheDuration

Size in seconds of file cache for streaming outputs.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 600

HlsMediaStoreStorageClass (enum)

- **VALUE**: TEMPORAL

HlsMode (enum)

- **VALUE**: LIVE
- **VALUE**: VOD

HlsOutputSelection (enum)

- **VALUE**: MANIFESTS_AND_SEGMENTS
- **VALUE**: SEGMENTS_ONLY

HlsOutputSettings

- **segmentModifier**: String concatenated to end of segment filenames.
  - **Type**: string
  - **Required**: False

hlsSettings

Settings regarding the underlying stream. These settings are different for audio-only outputs.

- **Type**: HlsSettings (p. 214)
- **Required**: True

nameModifier

String concatenated to the end of the destination filename. Accepts \"Format Identifiers \":#formatIdentifierParameters.

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

HlsProgramDateTime (enum)

- **VALUE**: EXCLUDE
- **VALUE**: INCLUDE
HlsSegmentationMode (enum)
- USE_INPUT_SEGMENTATION
- USE_SEGMENT_DURATION

HlsSettings

audioOnlyHlsSettings
- **Type:** AudioOnlyHlsSettings (p. 168)
- **Required:** False

standardHlsSettings
- **Type:** StandardHlsSettings (p. 246)
- **Required:** False

HlsStreamInfResolution (enum)
- EXCLUDE
- INCLUDE

HlsTimedMetadataId3Frame (enum)
- NONE
- PRIV
- TDRL

HlsTsFileMode (enum)
- SEGMENTED_FILES
- SINGLE_FILE

HlsWebdavHttpTransferMode (enum)
- CHUNKED
- NON_CHUNKED

HlsWebdavSettings

httpTransferMode
Specify whether or not to use chunked transfer encoding to WebDAV.
- **Type:** string
- **Required:** False

numRetries
Number of retry attempts that will be made before the Live Event is put into an error state.
- **Type:** integer
**Properties**

**Required**: False  
**Minimum**: 0

**restartDelay**

If a streaming output fails, number of seconds to wait until a restart is initiated. A value of 0 means never restart.

- **Type**: integer  
- **Required**: False  
- **Minimum**: 0  
- **Maximum**: 15

**connectionRetryInterval**

Number of seconds to wait before retrying connection to the CDN if the connection is lost.

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

**filecacheDuration**

Size in seconds of file cache for streaming outputs.

- **Type**: integer  
- **Required**: False  
- **Minimum**: 0  
- **Maximum**: 600

**InputAttachment**

**inputId**

The ID of the input

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

**inputSettings**

Settings of an input (caption selector, etc.)

- **Type**: `InputSettings` (p. 218)  
- **Required**: False

**InputChannelLevel**

**inputChannel**

The index of the input channel used as a source.

- **Type**: integer
**Properties**

**Required**: True
**Minimum**: 0
**Maximum**: 15

**gain**
Remixing value. Units are in dB and acceptable values are within the range from -60 (mute) and 6 dB.

**Type**: integer
**Required**: True
**Minimum**: -60
**Maximum**: 6

**InputCodec (enum)**
codec in increasing order of complexity

- MPEG2
- AVC
- HEVC

**InputDeblockFilter (enum)**

- DISABLED
- ENABLED

**InputDenoiseFilter (enum)**

- DISABLED
- ENABLED

**InputFilter (enum)**

- AUTO
- DISABLED
- FORCED

**InputLocation**

**passwordParam**
key used to extract the password from EC2 Parameter store

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

**uri**
Uniform Resource Identifier - This should be a path to a file accessible to the Live system (eg. a http:// URI) depending on the output type. For example, a rtmpEndpoint should have a uri similar to: "rtmp:// fmsserver/live".
Type: string  
Required: True

**username**

Username if credentials are required to access a file or publishing point. This can be either a plaintext username, or a reference to an AWS parameter store name from which the username can be retrieved. AWS Parameter store format: "ssm://<parameter name>"

Type: string  
Required: False

**InputLossActionForHlsOut (enum)**

- EMIT_OUTPUT
- PAUSE_OUTPUT

**InputLossActionForMsSmoothOut (enum)**

- EMIT_OUTPUT
- PAUSE_OUTPUT

**InputLossActionForUdpOut (enum)**

- DROP_PROGRAM
- DROP_TS
- EMIT_PROGRAM

**InputLossBehavior**

**inputLossImageType**

Indicates whether to substitute a solid color or a slate into the output after input loss exceeds blackFrameMsec.

Type: string  
Required: False

**inputLossImageColor**

When input loss image type is "color" this field specifies the color to use. Value: 6 hex characters representing the values of RGB.

Type: string  
Required: False

**inputLossImageSlate**

When input loss image type is "slate" these fields specify the parameters for accessing the slate.

Type: InputLocation (p. 216)  
Required: False
**blackFrameMsec**
On input loss, the number of milliseconds to substitute black into the output before switching to the frame specified by inputLossImageType. A value \( x \), where \( 0 \leq x \leq 1,000,000 \) and a value of 1,000,000 will be interpreted as infinite.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 1000000

**repeatFrameMsec**
On input loss, the number of milliseconds to repeat the previous picture before substituting black into the output. A value \( x \), where \( 0 \leq x \leq 1,000,000 \) and a value of 1,000,000 will be interpreted as infinite.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 1000000

**InputLossImageType (enum)**
- COLOR
- SLATE

**InputMaximumBitrate (enum)**
Maximum input bitrate in megabits per second. Bitrates up to 50 Mbps are supported currently.
- MAX_10_MBPS
- MAX_20_MBPS
- MAX_50_MBPS

**InputResolution (enum)**
Input resolution based on lines of vertical resolution in the input; SD is less than 720 lines, HD is 720 to 1080 lines, UHD is greater than 1080 lines
- SD
- HD
- UHD

**InputSettings**

**sourceEndBehavior**
Loop input if it is a file. This allows a file input to be streamed indefinitely.

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

**audioSelectors**
Used to select the audio stream to decode for inputs that have multiple available.
**deblockFilter**
Enable or disable the deblock filter when filtering.

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

**networkInputSettings**
Input settings.

- **Type:** NetworkInputSettings (p. 236)
- **Required:** False

**inputFilter**
Turns on the filter for this input. MPEG-2 inputs have the deblocking filter enabled by default. 1) auto - filtering will be applied depending on input type/quality 2) disabled - no filtering will be applied to the input 3) forced - filtering will be applied regardless of input type

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

**videoSelector**
Informs which video elementary stream to decode for input types that have multiple available.

- **Type:** VideoSelector (p. 252)
- **Required:** False

**filterStrength**
Adjusts the magnitude of filtering from 1 (minimal) to 5 (strongest).

- **Type:** integer
- **Required:** False
  - **Minimum:** 1
  - **Maximum:** 5

**captionSelectors**
Used to select the caption input to use for inputs that have multiple available.

- **Type:** Array of type CaptionSelector (p. 178)
- **Required:** False

**denoiseFilter**
Enable or disable the denoise filter when filtering.

- **Type:** string
Required: False

**InputSourceEndBehavior (enum)**

CONTINUE
LOOP

**InputSpecification**

codec
Input codec
- **Type:** string
- **Required:** False

resolution
Input resolution, categorized coarsely
- **Type:** string
- **Required:** False

maximumBitrate
Maximum input bitrate, categorized coarsely
- **Type:** string
- **Required:** False

**InternalServiceError**

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

**InvalidRequest**

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

**KeyProviderSettings**

staticKeySettings
- **Type:** StaticKeySettings (p. 246)
- **Required:** False
LimitExceeded

message

 Type: string
 Required: False

M2tsAbsentInputAudioBehavior (enum)

 DROP
 ENCODE_SILENCE

M2tsArib (enum)

 DISABLED
 ENABLED

M2tsAribCaptionsPidControl (enum)

 AUTO
 USE_CONFIGURED

M2tsAudioBufferModel (enum)

 ATSC
 DVB

M2tsAudioInterval (enum)

 VIDEO_AND_FIXED_INTERVALS
 VIDEO_INTERVAL

M2tsAudioStreamType (enum)

 ATSC
 DVB

M2tsBufferModel (enum)

 MULTIPLEX
 NONE

M2tsCcDescriptor (enum)

 DISABLED
 ENABLED

M2tsEbifControl (enum)

 NONE
 PASSTHROUGH
M2tsEbpPlacement (enum)

- VIDEO_AND_AUDIO_PIDS
- VIDEO_PID

M2tsEsRateInPes (enum)

- EXCLUDE
- INCLUDE

M2tsKlv (enum)

- NONE
- PASSTHROUGH

M2tsPcrControl (enum)

- CONFIGURED_PCR_PERIOD
- PCR_EVERY_PES_PACKET

M2tsRateMode (enum)

- CBR
- VBR

M2tsScte35Control (enum)

- NONE
- PASSTHROUGH

M2tsSegmentationMarkers (enum)

- EBP
- EBP_LEGACY
- NONE
- PSI_SEGSTART
- RAI_ADAPT
- RAI_SEGSTART

M2tsSegmentationStyle (enum)

- MAINTAIN_CADENCE
- RESET_CADENCE

M2tsSettings

**audioStreamType**

When set to atsc, uses stream type = 0x81 for AC3 and stream type = 0x87 for EAC3. When set to dvb, uses stream type = 0x06.

**Type**: string
Properties

**Required**: False

**ecmPid**
Packet Identifier (PID) for ECM in the transport stream. Only enabled when Simulcrypt is enabled. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20).8182 (or 0x1ff6).

  *Type*: string
  *Required*: False

**dvbTeletextPid**
Packet Identifier (PID) for input source DVB Teletext data to this output. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20).8182 (or 0x1ff6).

  *Type*: string
  *Required*: False

**aribCaptionsPidControl**
If set to auto, pid number used for ARIB Captions will be auto-selected from unused pids. If set to useConfigured, ARIB Captions will be on the configured pid number.

  *Type*: string
  *Required*: False

**bitrate**
The output bitrate of the transport stream in bits per second. Setting to 0 lets the muxer automatically determine the appropriate bitrate.

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

**segmentationTime**
The length in seconds of each segment. Required unless markers is set to None.

  *Type*: number
  *Required*: False
  *Minimum*: 1.0

**rateMode**
When vbr, does not insert null packets into transport stream to fill specified bitrate. The bitrate setting acts as the maximum bitrate when vbr is set.

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

Packet Identifier (PID) of the elementary audio stream(s) in the transport stream. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values. Each PID specified must be in the range of 32 (or 0x20)..8182 (or 0x1ff6).

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

**fragmentTime**

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

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

**ebpAudioInterval**

When `videoAndFixedIntervals` is selected, 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. Only available when EBP Cablelabs segmentation markers are selected. Partitions 1 and 2 will always follow the video interval.

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

**ebpLookaheadMs**

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

**audioFramesPerPes**

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

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

**scte35Pid**

Packet Identifier (PID) of the SCTE-35 stream in the transport stream. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20)..8182 (or 0x1ff6).

- **Type:** string
- **Required:** False
**pcrPeriod**
Maximum time in milliseconds between Program Clock Reference (PCRs) inserted into the transport stream.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 500

**pmtInterval**
The number of milliseconds between instances of this table in the output transport stream. Valid values are 0, 10..1000.

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

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

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

**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 "resetCadence" 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 $segmentationTime seconds. When a segmentation style of "maintainCadence" 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 $segmentationTime seconds. Note that EBP lookahead is a slight exception to this rule.

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

**ebif**
If set to passthrough, passes any EBIF data from the input source to this output.

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

**audioBufferModel**
When set to dvb, uses DVB buffer model for Dolby Digital audio. When set to atsc, the ATSC model is used.
**Properties**

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

**dvbNitSettings**

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

**Type**: DvbNitSettings (p. 181)  
**Required**: False

**absentInputAudioBehavior**

When set to drop, output audio streams will be removed from the program if the selected input audio stream is removed from the input. This allows the output audio configuration to dynamically change based on input configuration. If this is set to encodeSilence, all output audio streams will output encoded silence when not connected to an active input stream.

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

**timedMetadataPid**

Packet Identifier (PID) of the timed metadata stream in the transport stream. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20)..8182 (or 0x1ff6).

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

**timedMetadataBehavior**

When set to passthrough, timed metadata will be passed through from input to output.

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

**etvSignalPid**

Packet Identifier (PID) for input source ETV Signal data to this output. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20)..8182 (or 0x1ff6).

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

**pmtPid**

Packet Identifier (PID) for the Program Map Table (PMT) in the transport stream. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20)..8182 (or 0x1ff6).

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

If set to multiplex, use multiplex buffer model for accurate interleaving. Setting to bufferModel to none can lead to lower latency, but low-memory devices may not be able to play back the stream without interruptions.

  Type: string
  Required: False

scte35Control

Optionally pass SCTE-35 signals from the input source to this output.

  Type: string
  Required: False

ebpPlacement

Controls placement of EBP on Audio PIDs. If set to videoAndAudioPids, EBP markers will be placed on the video PID and all audio PIDs. If set to videoPid, EBP markers will be placed on only the video PID.

  Type: string
  Required: False

arib

When set to enabled, uses ARIB-compliant field muxing and removes video descriptor.

  Type: string
  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
  Minimum: 0.0

dvbSdtSettings

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

  Type: DvbSdtSettings (p. 181)
  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. Can be entered as a decimal or hexadecimal value. Valid values are 32 (0x20)..8182 (0x1ff6).

  Type: string
Properties

transportStreamId

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

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

pcrControl

When set to pcrEveryPesPacket, 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**: string
- **Required**: False

videoPid

Packet Identifier (PID) of the elementary video stream in the transport stream. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20)..8182 (or 0x1ff6).

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

esRateInPes

Include or exclude the ES Rate field in the PES header.

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

segmentationMarkers

Inserts segmentation markers at each segmentationTime period. raiSegstart sets the Random Access Indicator bit in the adaptation field. raiAdapt sets the RAI bit and adds the current timecode in the private data bytes. psiSegstart 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. ebpLegacy adds Encoder Boundary Point information to the adaptation field using a legacy proprietary format.

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

dvbTdtSettings

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

- **Type**: DvbTdtSettings (p. 186)
- **Required**: False
**klv**

If set to passthrough, passes any KLV data from the input source to this output.

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

**ccDescriptor**

When set to enabled, generates captionServiceDescriptor in PMT.

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

**patInterval**

The number of milliseconds between instances of this table in the output transport stream. Valid values are 0, 10..1000.

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

**etvPlatformPid**

Packet Identifier (PID) for input source ETV Platform data to this output. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20)..8182 (or 0x1ff6).

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

**dvbSubPids**

Packet Identifier (PID) for input source DVB Subtitle data to this output. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values. Each PID specified must be in the range of 32 (or 0x20)..8182 (or 0x1ff6).

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

**aribCaptionsPid**

Packet Identifier (PID) for ARIB Captions in the transport stream. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20)..8182 (or 0x1ff6).

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

**scte27Pids**

Packet Identifier (PID) for input source SCTE-27 data to this output. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values. Each PID specified must be in the range of 32 (or 0x20)..8182 (or 0x1ff6).
Properties

klvDataPids
Packet Identifier (PID) for input source KLV data to this output. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values. Each PID specified must be in the range of 32 (or 0x20)..8192 (or 0x1ff6).

M2tsTimedMetadataBehavior (enum)

- NO_PASSTHROUGH
- PASSTHROUGH

M3u8PcrControl (enum)

- CONFIGURED_PCR_PERIOD
- PCR_EVERY_PES_PACKET

M3u8Scte35Behavior (enum)

- NO_PASSTHROUGH
- PASSTHROUGH

M3u8Settings

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

- Type: string
- Required: False

ecmPid
ThePlatform-protected transport streams using 'microsoft' as Target Client include an ECM stream. This ECM stream contains the size, IV, and PTS of every sample in the transport stream. This stream PID is specified here. This PID has no effect on non ThePlatform-protected streams.

- Type: string
- Required: False

scte35Behavior
If set to passthrough, passes any SCTE-35 signals from the input source to this output.

- Type: string
- Required: False
Properties

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

  Type: string  
  Required: False

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

  Type: string  
  Required: False

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

  Type: integer  
  Required: False  
  Minimum: 0

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

  Type: string  
  Required: False

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

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

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

  Type: string  
  Required: False

**pcrControl**
When set to pcrEveryPesPacket, 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.
pmtInterval

The number of milliseconds between instances of this table in the output transport stream. A value of "0" writes out the PMT once per segment file.

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

pcrPeriod

Maximum time in milliseconds between Program Clock References (PCRs) inserted into the transport stream.

Type: integer
Required: False
Minimum: 0
Maximum: 500

programNum

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. A value of "0" writes out the PMT once per segment file.

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

timedMetadataPid

Packet Identifier (PID) of the timed metadata stream in the transport stream. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20)..8182 (or 0x1ff6).

Type: string
Required: False

timedMetadataBehavior

When set to passthrough, timed metadata is passed through from input to output.

Type: string
Required: False

M3u8TimedMetadataBehavior (enum)

- NO_PASSTHROUGH
- PASSTHROUGH

Mp2CodingMode (enum)

- CODING_MODE_1_0
- CODING_MODE_2_0

Mp2Settings

codingMode
The MPEG2 Audio coding mode. Valid values are codingMode10 (for mono) or codingMode20 (for stereo).

  Type: string
  Required: False

bitrate
Average bitrate in bits/second.

  Type: number
  Required: False

sampleRate
Sample rate in Hz.

  Type: number
  Required: False

MsSmoothGroupSettings

eventId
MS Smooth event ID to be sent to the IIS server. Should only be specified if eventIdMode is set to useConfigured.

  Type: string
  Required: False

fragmentLength
Length of mp4 fragments to generate (in seconds). Fragment length must be compatible with GOP size and framerate.

  Type: integer
  Required: False
Minimum: 1

timestampOffset
Timestamp offset for the event. Only used if timestampOffsetMode is set to useConfiguredOffset.

  Type: string
  Required: False

segmentationMode
When set to useInputSegmentation, the output segment or fragment points are set by the RAI markers from the input streams.

  Type: string
  Required: False

numRetries
Number of retry attempts.

  Type: integer
  Required: False
  Minimum: 0

eventStopBehavior
When set to sendEos, send EOS signal to IIS server when stopping the event

  Type: string
  Required: False

acquisitionPointId
The value of the "Acquisition Point Identity" element used in each message placed in the sparse track. Only enabled if sparseTrackType is not "none".

  Type: string
  Required: False

sparseTrackType
If set to scte35, use incoming SCTE-35 messages to generate a sparse track in this group of MS-Smooth outputs.

  Type: string
  Required: False

timestampOffsetMode
Type of timestamp date offset to use. - useEventStartDate: Use the date the event was started as the offset - useConfiguredOffset: Use an explicitly configured date as the offset

  Type: string
  Required: False
destination

Smooth Streaming publish point on an IIS server. Elemental Live acts as a "Push" encoder to IIS.

Type: OutputLocationRef (p. 239)
Required: True

audioOnlyTimecodeControl

If set to passthrough for an audio-only MS Smooth output, the fragment absolute time will be set to the current timecode. This option does not write timecodes to the audio elementary stream.

Type: string
Required: False

connectionRetryInterval

Number of seconds to wait before retrying connection to the IIS server if the connection is lost. Content will be cached during this time and the cache will be delivered to the IIS server once the connection is re-established.

Type: integer
Required: False
Minimum: 0

filecacheDuration

Size in seconds of file cache for streaming outputs.

Type: integer
Required: False
Minimum: 0

certificateMode

If set to verifyAuthenticity, verify the https certificate chain to a trusted Certificate Authority (CA). This will cause https outputs to self-signed certificates to fail unless those certificates are manually added to the OS trusted keystore.

Type: string
Required: False

inputLossAction

Parameter that control output group behavior on input loss.

Type: string
Required: False

sendDelayMs

Outputs that are "output locked" can use this delay. Assign a delay to the output that is "secondary". Do not assign a delay to the "primary" output. The delay means that the primary output will always reach the downstream system before the secondary, which helps ensure that the downstream system always uses the primary output. (If there were no delay, the downstream system might flip-flop between whichever output happens to arrive first.) If the primary fails, the downstream system will switch to the
secondary output. When the primary is restarted, the downstream system will switch back to the primary (because once again it is always arriving first)

**eventIdMode**

Specifies whether or not to send an event ID to the IIS server. If no event ID is sent and the same Live Event is used without changing the publishing point, clients might see cached video from the previous run. Options: - "useConfigured" - use the value provided in eventId - "useTimestamp" - generate and send an event ID based on the current timestamp - "noEventId" - do not send an event ID to the IIS server.

**streamManifestBehavior**

When set to send, send stream manifest so publishing point doesn't start until all streams start.

**MsSmoothOutputSettings**

**nameModifier**

String concatenated to the end of the destination filename. Required for multiple outputs of the same type.

**NetworkInputServerValidation (enum)**

- CHECK_CRYPTOGRAPHY_AND_VALIDATE_NAME
- CHECK_CRYPTOGRAPHY_ONLY

**NetworkInputSettings**

**hlsInputSettings**

Specifies HLS input settings when the uri is for a HLS manifest.
Properties

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

**serverValidation**

Check HTTPS server certificates. When set to checkCryptographyOnly, cryptography in the certificate will be checked, but not the server's name. Certain subdomains (notably S3 buckets that use dots in the bucket name) do not strictly match the corresponding certificate's wildcard pattern and would otherwise cause the event to error. This setting is ignored for protocols that do not use https.

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

**Output**

**videoDescriptionName**

The name of the VideoDescription used as the source for this output.

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

**captionDescriptionNames**

The names of the CaptionDescriptions used as caption sources for this output.

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

**outputName**

The name used to identify an output.

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

**outputSettings**

Output type-specific settings.

**Type:** OutputSettings (p. 240)
**Required:** True

**audioDescriptionNames**

The names of the AudioDescriptions used as audio sources for this output.

**Type:** Array of type string
**Required:** False
OutputDestination

settings
Destination settings for output; one for each redundant encoder.
  Type: Array of type OutputDestinationSettings (p. 238)
  Required: False

id
User-specified id. This is used in an output group or an output.
  Type: string
  Required: False

OutputDestinationSettings

passwordParam
key used to extract the password from EC2 Parameter store
  Type: string
  Required: False

streamName
Stream name for RTMP destinations (URLs of type rtmp://)
  Type: string
  Required: False

url
A URL specifying a destination
  Type: string
  Required: False

username
username for destination
  Type: string
  Required: False

OutputGroup

outputs
  Type: Array of type Output (p. 237)
Required: True

outputGroupSettings

Settings associated with the output group.

Type: OutputGroupSettings (p. 239)
Required: True

name

Custom output group name optionally defined by the user. Only letters, numbers, and the underscore character allowed; only 32 characters allowed.

Type: string
Required: False

OutputGroupSettings

archiveGroupSettings

Type: ArchiveGroupSettings (p. 163)
Required: False

rtmpGroupSettings

Type: RtmpGroupSettings (p. 241)
Required: False

udpGroupSettings

Type: UdpGroupSettings (p. 248)
Required: False

msSmoothGroupSettings

Type: MsSmoothGroupSettings (p. 233)
Required: False

hlsGroupSettings

Type: HlsGroupSettings (p. 206)
Required: False

OutputLocationRef

destinationRefId

Type: string
Required: False
OutputSettings

rtmpOutputSettings

Type: RtmpOutputSettings (p. 242)  
Required: False

archiveOutputSettings

Type: ArchiveOutputSettings (p. 163)  
Required: False

msSmoothOutputSettings

Type: MsSmoothOutputSettings (p. 236)  
Required: False

udpOutputSettings

Type: UdpOutputSettings (p. 248)  
Required: False

hlsOutputSettings

Type: HlsOutputSettings (p. 213)  
Required: False

PassThroughSettings

RemixSettings

channelMappings

Mapping of input channels to output channels, with appropriate gain adjustments.

Type: Array of type AudioChannelMapping (p. 164)  
Required: True

channelsOut

Number of output channels to be produced. Valid values: 1, 2, 4, 6, 8

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

channelsIn

Number of input channels to be used.

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

**ResourceConflict**

*message*

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

**ResourceNotFound**

*message*

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

**RtmpCacheFullBehavior (enum)**

- DISCONNECT_IMMEDIATELY
- WAIT_FOR_SERVER

**RtmpCaptionData (enum)**

- ALL
- FIELD1_608
- FIELD1_AND_FIELD2_608

**RtmpCaptionInfoDestinationSettings**

**RtmpGroupSettings**

*captionData*

Controls the types of data that passes to onCaptionInfo outputs. If set to 'all' then 608 and 708 carried DTVCC data will be passed. If set to 'field1AndField2608' then DTVCC data will be stripped out, but 608 data from both fields will be passed. If set to 'field1608' then only the data carried in 608 from field 1 video will be passed.

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

*authenticationScheme*

Authentication scheme to use when connecting with CDN

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

*cacheLength*

Cache length, in seconds, is used to calculate buffer size.
### Properties

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

#### restartDelay

If a streaming output fails, number of seconds to wait until a restart is initiated. A value of 0 means never restart.

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

#### cacheFullBehavior

Controls behavior when content cache fills up. If remote origin server stalls the RTMP connection and does not accept content fast enough the 'Media Cache' will fill up. When the cache reaches the duration specified by cacheLength the cache will stop accepting new content. If set to disconnectImmediately, the RTMP output will force a disconnect. Clear the media cache, and reconnect after restartDelay seconds. If set to waitForServer, the RTMP output will wait up to 5 minutes to allow the origin server to begin accepting data again.

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

#### RtmpOutputCertificateMode (enum)

- SELF_SIGNED
- VERIFY_AUTHENTICITY

#### RtmpOutputSettings

**certificateMode**

If set to verifyAuthenticity, verify the tls certificate chain to a trusted Certificate Authority (CA). This will cause rtmps outputs with self-signed certificates to fail.

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

#### numRetries

Number of retry attempts.

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

#### destination

The RTMP endpoint excluding the stream name (eg. rtmp://host/appname). For connection to Akamai, a username and password must be supplied. URI fields accept format identifiers.

**Type**: `OutputLocationRef (p. 239)`
**Properties**

**Required**: True

**connectionRetryInterval**

Number of seconds to wait before retrying a connection to the Flash Media server if the connection is lost.

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

**Scte20Convert608To708 (enum)**

- DISABLED
- UPCONVERT

**Scte20PlusEmbeddedDestinationSettings**

**Scte20SourceSettings**

**source608ChannelNumber**

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

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

**convert608To708**

If 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**: string
- **Required**: False

**Scte27DestinationSettings**

**Scte27SourceSettings**

**pid**

The pid field is used in conjunction with the caption selector languageCode field as follows: - Specify PID and Language: Extracts captions from that PID; the language is "informational". - Specify PID and omit Language: Extracts the specified PID. - Omit PID and specify Language: Extracts the specified language, whichever PID that happens to be. - Omit PID and omit Language: Valid only if source is DVB-Sub that is being passed through; all languages will be passed through.

- **Type**: integer
- **Required**: False
- **Minimum**: 1
Scte35AposNoRegionalBlackoutBehavior (enum)

FOLLOW
IGNORE

Scte35AposWebDeliveryAllowedBehavior (enum)

FOLLOW
IGNORE

Scte35SpliceInsert

adAvailOffset

When specified, this offset (in milliseconds) is added to the input Ad Avail PTS time. This only applies to embedded SCTE 104/35 messages and does not apply to OOB messages.

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

webDeliveryAllowedFlag

When set to ignore, Segment Descriptors with webDeliveryAllowedFlag set to 0 will no longer trigger blackouts or Ad Avail slates

Type: string
Required: False

noRegionalBlackoutFlag

When set to ignore, Segment Descriptors with noRegionalBlackoutFlag set to 0 will no longer trigger blackouts or Ad Avail slates

Type: string
Required: False

Scte35SpliceInsertNoRegionalBlackoutBehavior (enum)

FOLLOW
IGNORE

Scte35SpliceInsertWebDeliveryAllowedBehavior (enum)

FOLLOW
IGNORE
**Scte35TimeSignalApos**

**adAvailOffset**
When specified, this offset (in milliseconds) is added to the input Ad Avail PTS time. This only applies to embedded SCTE 104/35 messages and does not apply to OOB messages.

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

**webDeliveryAllowedFlag**
When set to ignore, Segment Descriptors with webDeliveryAllowedFlag set to 0 will no longer trigger blackouts or Ad Avail slates

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

**noRegionalBlackoutFlag**
When set to ignore, Segment Descriptors with noRegionalBlackoutFlag set to 0 will no longer trigger blackouts or Ad Avail slates

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

**SmoothGroupAudioOnlyTimecodeControl (enum)**
- PASSTHROUGH
- USE_CONFIGURED_CLOCK

**SmoothGroupCertificateMode (enum)**
- SELF_SIGNED
- VERIFY_AUTHENTICITY

**SmoothGroupEventIdMode (enum)**
- NO_EVENT_ID
- USE_CONFIGURED
- USE_TIMESTAMP

**SmoothGroupEventStopBehavior (enum)**
- NONE
- SEND_EOS

**SmoothGroupSegmentationMode (enum)**
- USE_INPUT_SEGMENTATION
- USE_SEGMENT_DURATION

245
**SmoothGroupSparseTrackType (enum)**

- NONE
- SCTE_35

**SmoothGroupStreamManifestBehavior (enum)**

- DO_NOT_SEND
- SEND

**SmoothGroupTimestampOffsetMode (enum)**

- USE_CONFIGURED_OFFSET
- USE_EVENT_START_DATE

**SmpteTtDestinationSettings**

**StandardHlsSettings**

**m3u8Settings**

*Type:* M3u8Settings (p. 230)

*Required:* True

**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

**StaticKeySettings**

**staticKeyValue**

Static key value as a 32 character hexadecimal string.  

*Type:* string  

*Required:* True

**keyProviderServer**

The URL of the license server used for protecting content.  

*Type:* InputLocation (p. 216)  

*Required:* False
**TeletextDestinationSettings**

**TeletextSourceSettings**

**pageNumber**

Specifies the teletext page number within the data stream from which to extract captions. Range of 0x100 (256) to 0x8FF (2303). Unused for passthrough. Should be specified as a hexadecimal string with no "0x" prefix.

*Type:* string  
*Required:* False

**TimecodeConfig**

**syncThreshold**

Threshold in frames beyond which output timecode is resynchronized to the input timecode. Discrepancies below this threshold are permitted to avoid unnecessary discontinuities in the output timecode. No timecode sync when this is not specified.

*Type:* integer  
*Required:* False  
*Minimum:* 1  
*Maximum:* 1000000

**source**

Identifies the source for the timecode that will be associated with the events outputs. -Embedded (embedded): Initialize the output timecode with timecode from the the source. If no embedded timecode is detected in the source, the system falls back to using "Start at 0" (zerobased). -System Clock (systemclock): Use the UTC time. -Start at 0 (zerobased): The time of the first frame of the event will be 00:00:00:00.

*Type:* string  
*Required:* True

**TimecodeConfigSource (enum)**

- EMBEDDED  
- SYSTEMCLOCK  
- ZEROBASED

**TtmlDestinationSettings**

**styleControl**

When set to passthrough, passes 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:* string  
*Required:* False
```
TtmlDestinationStyleControl (enum)
   PASSTHROUGH
   USE_CONFIGURED

UdpContainerSettings

m2tsSettings
   Type: M2tsSettings (p. 222)
   Required: False

UdpGroupSettings

inputLossAction
   Specifies behavior of last resort when input video is lost, and no more backup inputs are available. When dropTs is selected the entire transport stream will stop being emitted. When dropProgram is selected the program can be dropped from the transport stream (and replaced with null packets to meet the TS bitrate requirement). Or, when emitProgram is chosen the transport stream will continue to be produced normally with repeat frames, black frames, or slate frames substituted for the absent input video.
   Type: string
   Required: False

timedMetadataId3Frame
   Indicates ID3 frame that has the timecode.
   Type: string
   Required: False

timedMetadataId3Period
   Timed Metadata interval in seconds.
   Type: integer
   Required: False
   Minimum: 0

UdpOutputSettings

bufferMsec
   UDP output buffering in milliseconds. Larger values increase latency through the transcoder but simultaneously assist the transcoder in maintaining a constant, low-jitter UDP/RTP output while accommodating clock recovery, input switching, input disruptions, picture reordering, etc.
   Type: integer
   Required: False
   Minimum: 0
   Maximum: 10000
```
Properties

destination

Destination address and port number for RTP or UDP packets. Can be unicast or multicast RTP or UDP (eg. rtp://239.10.10.10:5001 or udp://10.100.100.100:5002).

  Type: OutputLocationRef (p. 239)
  Required: True

containerSettings

  Type: UdpContainerSettings (p. 248)
  Required: True

fecOutputSettings

Settings for enabling and adjusting Forward Error Correction on UDP outputs.

  Type: FecOutputSettings (p. 193)
  Required: False

UdpTimedMetadataId3Frame (enum)

  NONE
  PRIV
  TDRL

UpdateChannel

inputAttachments

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

roleArn

An optional Amazon Resource Name (ARN) of the role to assume when running the Channel. If you do not specify this on an update call but the role was previously set that role will be removed.

  Type: string
  Required: False

destinations

A list of output destinations for this channel.

  Type: Array of type OutputDestination (p. 238)
  Required: False

name

The name of the channel.

  Type: string
**Properties**

**encoderSettings**

The encoder settings for this channel.

- **Type:** EncoderSettings (p. 192)
- **Required:** False

**inputSpecification**

Specification of input for this channel (max. bitrate, resolution, codec, etc.)

- **Type:** InputSpecification (p. 220)
- **Required:** False

**UpdateChannelResultModel**

**channel**

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

**ValidationError**

**errorMessage**

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

**elementPath**

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

**VideoCodecSettings**

**h264Settings**

- **Type:** H264Settings (p. 197)
- **Required:** False

**VideoDescription**

**respondToAfd**

Indicates how to respond to the AFD values in the input stream. Setting to "respond" causes input video to be clipped, depending on AFD value, input display aspect ratio and output display aspect ratio.

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

When set to "stretchToOutput", automatically configures the output position to stretch the video to the specified output resolution. This option will override any position value.

Type: string
Required: False

name

The name of this VideoDescription. Outputs will use this name to uniquely identify this Description. Description names should be unique within this Live Event.

Type: string
Required: True

width

Output video width (in pixels). Leave out to use source video width. If left out, height must also be left out. Display aspect ratio is always preserved by letterboxing or pillarboxing when necessary.

Type: integer
Required: False

sharpness

Changes the width of the anti-alias filter kernel used for scaling. Only applies if scaling is being performed and antiAlias is set to true. 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.

Type: VideoCodecSettings (p. 250)
Required: False

height

Output video height (in pixels). Leave blank to use source video height. If left blank, width must also be unspecified.

Type: integer
Required: False

VideoDescriptionRespondToAfd (enum)

NONE
PASSTHROUGH
RESPOND

**VideoDescriptionScalingBehavior (enum)**

- DEFAULT
- STRETCH_TO_OUTPUT

**VideoSelector**

**colorSpace**

Specifies the colorspace of an input. This setting works in tandem with colorSpaceConversion to determine if any conversion will be performed.

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

**selectorSettings**

The video selector settings.

- **Type:** VideoSelectorSettings (p. 253)
- **Required:** False

**colorSpaceUsage**

Applies only if colorSpace is a value other than follow. This field controls how the value in the colorSpace field will be used. fallback means that when the input does include color space data, that data will be used, but when the input has no color space data, the value in colorSpace will be used. Choose fallback if your input is sometimes missing color space data, but when it does have color space data, that data is correct. force means to always use the value in colorSpace. Choose force if your input usually has no color space data or might have unreliable color space data.

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

**VideoSelectorColorSpace (enum)**

- FOLLOW
- REC_601
- REC_709

**VideoSelectorColorSpaceUsage (enum)**

- FALLBACK
- FORCE

**VideoSelectorPid**

**pid**

Selects a specific PID from within a video source.
Type: integer
Required: False
Minimum: 0
Maximum: 8191

**VideoSelectorProgramId**

**programId**

Selects a specific program from within a multi-program transport stream. If the program doesn't exist, the first program within the transport stream will be selected by default.

Type: integer
Required: False
Minimum: 0
Maximum: 65536

**VideoSelectorSettings**

**videoSelectorPid**

Type: VideoSelectorPid (p. 252)
Required: False

**videoSelectorProgramId**

Type: VideoSelectorProgramId (p. 253)
Required: False

**WebvttDestinationSettings**

**Channels channelId Start**

**URI**

/prod/channels/ channelID /start

**HTTP Methods**

**POST**

Operation ID: StartChannel

Starts an existing channel

**Path Parameters**

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>channelId</td>
<td>String</td>
<td>True</td>
<td>channel ID</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>Channel (p. 254)</td>
<td>Successfully initiated start of the channel.</td>
</tr>
<tr>
<td>400</td>
<td>InvalidRequest (p. 265)</td>
<td>This request was invalid.</td>
</tr>
<tr>
<td>500</td>
<td>InternalServiceError (p. 265)</td>
<td>Unexpected internal service error.</td>
</tr>
<tr>
<td>502</td>
<td>BadGatewayException (p. 266)</td>
<td>Bad Gateway Error</td>
</tr>
<tr>
<td>403</td>
<td>AccessDenied (p. 265)</td>
<td>You do not have permission to list channels.</td>
</tr>
<tr>
<td>404</td>
<td>ResourceNotFound (p. 265)</td>
<td>The channel you’re requesting to describe does not exist.</td>
</tr>
<tr>
<td>504</td>
<td>GatewayTimeoutException (p. 266)</td>
<td>Gateway Timeout Error</td>
</tr>
<tr>
<td>429</td>
<td>LimitExceeded (p. 265)</td>
<td>Request limit exceeded on list channel calls to channel service.</td>
</tr>
<tr>
<td>409</td>
<td>ResourceConflict (p. 265)</td>
<td>The channel is unable to create due to an issue with channel resources.</td>
</tr>
</tbody>
</table>

## Schemas

### Response Bodies

#### Example Channel

```json
{
  "inputAttachments (p. 286)": [ {
    "inputId (p. 322)": "string",
    "inputSettings (p. 322)": { 
      "sourceEndBehavior (p. 325)": enum,
      "audioSelectors (p. 325)": [ 
        { 
          "name (p. 276)": "string",
          "selectorSettings (p. 276)": { 
            "audioLanguageSelection (p. 276)": enum,
            "languageSelectionPolicy (p. 274)": enum,
            "languageCode (p. 274)": "string"
          },
          "audioPidSelection (p. 276)": { 
            "pid (p. 275)": integer
          }
        }
      ],
      "deblockFilter (p. 325)": enum,
      "networkInputSettings (p. 326)": { 
        "hlsInputSettings (p. 343)": { 
          "retries (p. 318)": integer,
          "bandwidth (p. 318)": integer
        }
      }
    }
  }
}
```

254
"retryInterval (p. 318)" : integer,
"bufferSegments (p. 318)" : integer
},
"serverValidation (p. 344)" : enum
},
"inputFilter (p. 326)" : enum,
"videoSelector (p. 326)" : {
"colorSpace (p. 357)" : enum,
"selectorSettings (p. 357)" : {
"videoSelectorPid (p. 358)" : {
"pid (p. 358)" : integer
},
"videoSelectorProgramId (p. 358)" : {
"programId (p. 358)" : integer
}
},
"colorSpaceUsage (p. 357)" : enum
},
"filterStrength (p. 326)" : integer,
"captionSelectors (p. 326)" : [
{
"name (p. 285)" : "string",
"languageCode (p. 285)" : "string",
"selectorSettings (p. 285)" : {
"embeddedSourceSettings (p. 285)" : {
"scte20Detection (p. 298)" : enum,
"source608ChannelNumber (p. 298)" : integer,
"convert608To708 (p. 298)" : enum,
"source608TrackNumber (p. 298)" : integer
},
"scte20SourceSettings (p. 285)" : {
"source608ChannelNumber (p. 350)" : integer,
"convert608To708 (p. 350)" : enum
},
"dvbSubSourceSettings (p. 285)" : {
"pid (p. 293)" : integer
},
"aribSourceSettings (p. 285)" : {
},
"teletextSourceSettings (p. 285)" : {
"pageNumber (p. 353)" : "string"
},
"scte27SourceSettings (p. 286)" : {
"pid (p. 350)" : integer
}
}
},
"denoiseFilter (p. 326)" : enum
}]
},
"roleArn (p. 286)" : "string",
"destinations (p. 286)" : [
{
"settings (p. 344)" : [
{
"passwordParam (p. 345)" : "string",
"streamName (p. 345)" : "string",
"url (p. 345)" : "string",
"username (p. 345)" : "string"
}
],
"id (p. 345)" : "string"
}]}
"name (p. 286)" : "string",
"encoderSettings (p. 286)" : {
  "timecodeConfig (p. 299)" : {
    "syncThreshold (p. 353)" : integer,
    "source (p. 354)" : enum
  },
  "outputGroups (p. 299)" : [
    {
      "outputs (p. 345)" : [
        {
          "videoDescriptionName (p. 344)" : "string",
          "captionDescriptionNames (p. 344)" : [
            "string"
          ],
          "outputName (p. 344)" : "string",
          "outputSettings (p. 344)" : {
            "rtmpOutputSettings (p. 346)" : {
              "certificateMode (p. 349)" : enum,
              "numRetries (p. 349)" : integer,
              "destination (p. 349)" : {
                "destinationRefId (p. 346)" : "string"
              },
              "connectionRetryInterval (p. 349)" : integer
            },
            "archiveOutputSettings (p. 346)" : {
              "extension (p. 270)" : "string",
              "containerSettings (p. 271)" : {
                "m3u8Settings (p. 270)" : {
                  "audioStreamType (p. 329)" : enum,
                  "ecmPid (p. 330)" : "string",
                  "dvbTeletextPid (p. 330)" : "string",
                  "aribCaptionsPidControl (p. 330)" : enum,
                  "bitrate (p. 330)" : integer,
                  "segmentationTime (p. 330)" : number,
                  "rateMode (p. 330)" : enum,
                  "audioPids (p. 330)" : "string",
                  "fragmentTime (p. 331)" : number,
                  "ebpAudioInterval (p. 331)" : integer,
                  "ebpLookaheadMs (p. 331)" : integer,
                  "audioFramesPerPes (p. 331)" : integer,
                  "scte35Pid (p. 331)" : "string",
                  "pcrPeriod (p. 331)" : integer,
                  "pmtInterval (p. 332)" : integer,
                  "programNum (p. 332)" : integer,
                  "segmentationStyle (p. 332)" : enum,
                  "ebif (p. 332)" : enum,
                  "audioBufferModel (p. 332)" : enum,
                  "dvbNitSettings (p. 333)" : {
                    "networkName (p. 287)" : "string",
                    "networkId (p. 288)" : integer,
                    "repInterval (p. 288)" : integer
                  },
                  "absentInputAudioBehavior (p. 333)" : enum,
                  "timedMetadataPId (p. 333)" : "string",
                  "timedMetadataBehavior (p. 333)" : enum,
                  "etvSignalPId (p. 333)" : "string",
                  "pmtPId (p. 333)" : "string",
                  "bufferModel (p. 333)" : enum,
                  "scte35Control (p. 334)" : enum,
                  "ebpPlacement (p. 334)" : enum,
                  "arib (p. 334)" : enum,
                  "nullPacketBitrate (p. 334)" : number,
                  "dvbSdtSettings (p. 334)" : {
                    "serviceName (p. 288)" : "string",
                    "serviceProviderName (p. 288)" : "string",
                    "repInterval (p. 288)" : integer,
                  }
                }
              }
            }
          }]
        }
      ]
    }
  ]}
"outputSdt (p. 289)" : enum,

"pcrPid (p. 334)" : "string",
"transportStreamId (p. 334)" : integer,
"pcrControl (p. 335)" : enum,
"videoPid (p. 335)" : "string",
"esRateInPes (p. 335)" : enum,
"segmentationMarkars (p. 335)" : enum,
"dvbTdtSettings (p. 335)" : {
  "repInterval (p. 293)" : integer
},
"klv (p. 335)" : enum,
"ccDescriptor (p. 336)" : enum,
"patInterval (p. 336)" : integer,
"etvPlatformPid (p. 336)" : "string",
"dvbSubPids (p. 336)" : "string",
"aribCaptionsPId (p. 336)" : "string",
"scte27Pids (p. 336)" : "string",
"klvDataPids (p. 336)" : "string"
}
],
"nameModifier (p. 271)" : "string"
},
"msSmoothOutputSettings (p. 347)" : {
  "nameModifier (p. 343)" : "string"
},
"udpOutputSettings (p. 347)" : {
  "bufferMsec (p. 355)" : integer,
  "destination (p. 355)" : {
    "destinationRefId (p. 346)" : "string"
  },
  "containerSettings (p. 355)" : {
    "m2tsSettings (p. 354)" : {
      "audioStreamType (p. 329)" : enum,
      "ecmPid (p. 330)" : "string",
      "dvbTeletextPId (p. 330)" : "string",
      "aribCaptionsPIdControl (p. 330)" : enum,
      "bitrate (p. 330)" : integer,
      "segmentationTime (p. 330)" : number,
      "rateMode (p. 330)" : enum,
      "audioPids (p. 330)" : "string",
      "fragmentTime (p. 331)" : number,
      "ebpAudioInterval (p. 331)" : enum,
      "ebpLookaheadMs (p. 331)" : integer,
      "audioFramesPerPes (p. 331)" : integer,
      "scte35Pid (p. 331)" : "string",
      "pcrPeriod (p. 331)" : integer,
      "pmtInterval (p. 332)" : integer,
      "programNum (p. 332)" : integer,
      "segmentationStyle (p. 332)" : enum,
      "ebif (p. 332)" : enum,
      "audioBufferModel (p. 332)" : enum,
      "dvbNitSettings (p. 333)" : {
        "networkName (p. 287)" : "string",
        "networkId (p. 288)" : integer,
        "repInterval (p. 288)" : integer
      },
      "absentInputAudioBehavior (p. 333)" : enum,
      "timedMetadataPid (p. 333)" : "string",
      "timedMetadataBehavior (p. 333)" : enum,
      "etvSignalPid (p. 333)" : "string",
      "pmtPid (p. 333)" : "string",
      "bufferModel (p. 333)" : enum,
      "scte35Control (p. 334)" : enum,
      "ebpPlacement (p. 334)" : enum,
      "arib (p. 334)" : enum,

257
nullPacketBitrate (p. 334): number,
dvbSdtSettings (p. 334): {
  "serviceName (p. 288): "string",
  "serviceProviderName (p. 288): "string",
  "repInterval (p. 288): integer,
  "outputSdt (p. 289): enum
},
"pcrPid (p. 334): "string",
"transportStreamId (p. 334): integer,
"pcrControl (p. 335): enum,
"videoPid (p. 335): "string",
"esRateInPes (p. 335): enum,
"segmentationMarkers (p. 335): enum,
"dvbTdtSettings (p. 335): {
  "repInterval (p. 293): integer
},
klv (p. 335): enum,
"ccDescriptor (p. 336): enum,
"patInterval (p. 336): integer,
"etvPlatformPid (p. 336): "string",
"dvbSubPids (p. 336): "string",
"aribCaptionsPid (p. 336): "string",
"scte27Pids (p. 336): "string",
"klvDataPids (p. 336): "string"
},
"fecOutputSettings (p. 355): {
  "rowLength (p. 300): integer,
  "columnDepth (p. 300): integer,
  "includeFec (p. 300): enum
},
hlsOutputSettings (p. 347): {
  "segmentModifier (p. 320): "string",
  "hlsSettings (p. 320): {
    "audioOnlyHlsSettings (p. 320): {
      "audioTrackType (p. 275): enum,
      "audioGroupId (p. 275): "string",
      "passwordParam (p. 323): "string",
      "uri (p. 323): "string",
      "username (p. 323): "string"
    },
    "standardHlsSettings (p. 321): {
      "m3u8Settings (p. 353): {
        "pmtPid (p. 337): "string",
        "ecmPid (p. 337): "string",
        "scte35Behavior (p. 337): enum,
        "pcrPid (p. 337): "string",
        "audioPids (p. 338): "string",
        "audioFramesPerPes (p. 338): integer,
        "scte35Pid (p. 338): "string",
        "transportStreamId (p. 338): integer,
        "videoPid (p. 338): "string",
        "pcrControl (p. 338): enum,
        "pcrPid (p. 339): integer,
        "pcrPeriod (p. 339): integer,
        "programNum (p. 339): integer,
        "patInterval (p. 339): integer,
        "timedMetadataPid (p. 339): "string",
        "timedMetadataBehavior (p. 339): enum
      },
      "audioRenditionSets (p. 353): "string"
    }
  }
}
"nameModifier (p. 320)": "string",
],
"audioDescriptionNames (p. 344)": [
  "string"
],
outputGroupSettings (p. 345): {
  archiveGroupSettings (p. 346): {
    destination (p. 270): {
      destinationRefId (p. 346): "string",
    },
    rolloverInterval (p. 270): integer,
  }
},
rtmpGroupSettings (p. 346): {
  captionData (p. 348): enum,
  authenticationScheme (p. 348): enum,
  cacheLength (p. 348): integer,
  restartDelay (p. 348): integer,
  cacheFullBehavior (p. 349): enum,
},
udpGroupSettings (p. 346): {
  inputLossAction (p. 354): enum,
  timedMetadataId3Frame (p. 355): enum,
  timedMetadataId3Period (p. 355): integer,
},
msSmoothGroupSettings (p. 346): {
  eventId (p. 340): "string",
  fragmentLength (p. 340): integer,
  timestampOffset (p. 340): "string",
  segmentationMode (p. 341): enum,
  numRetries (p. 341): integer,
  eventStopBehavior (p. 341): enum,
  acquisitionPointId (p. 341): "string",
  sparseTrackType (p. 341): enum,
  timestampOffsetMode (p. 341): enum,
  destination (p. 341): {
    destinationRefId (p. 346): "string",
  },
  audioOnlyTimecodeControl (p. 342): enum,
  connectionRetryInterval (p. 342): integer,
  filecacheDuration (p. 342): integer,
  certificateMode (p. 342): enum,
  inputLossAction (p. 342): enum,
  sendDelayMs (p. 342): integer,
  eventIdMode (p. 343): enum,
  restartDelay (p. 343): integer,
  streamManifestBehavior (p. 343): enum,
},
hlsGroupSettings (p. 346): {
  segmentsPerSubdirectory (p. 312): integer,
  ivInManifest (p. 313): enum,
  outputSelection (p. 313): enum,
  encryptionType (p. 313): enum,
  indexNSegments (p. 313): integer,
  destination (p. 313): {
    destinationRefId (p. 346): "string",
  },
  constantIv (p. 313): "string",
  timedMetadataId3Frame (p. 313): enum,
  baseUrlManifest (p. 314): "string",
  captionLanguageSetting (p. 314): enum,
  minSegmentLength (p. 314): integer,
  mode (p. 314): enum,
  ivSource (p. 314): enum,
"manifestCompression (p. 314)": enum,
"keyProviderSettings (p. 315)": {
  "staticKeySettings (p. 327)": {
    "staticKeyValue (p. 353)": "string",
    "KeyProviderServer (p. 353)": {
      "passwordParam (p. 323)": "string",
      "url (p. 323)": "string",
      "username (p. 323)": "string"
    }
  }
},
"tsFileMode (p. 315)": enum,
"manifestDurationFormat (p. 315)": enum,
"keyFormatVersions (p. 315)": "string",
"streamInfResolution (p. 315)": enum,
"timestampDeltaMilliseconds (p. 315)": integer,
"manifestDurationFormat (p. 315)": enum,
"keyProviderSettings (p. 315)": {
  "staticKeySettings (p. 327)": {
    "staticKeyValue (p. 353)": "string",
    "KeyProviderServer (p. 353)": {
      "passwordParam (p. 323)": "string",
      "url (p. 323)": "string",
      "username (p. 323)": "string"
    }
  }
},
"manifestCompression (p. 314)": enum,
"manifestDurationFormat (p. 315)": enum,
"keyFormatVersions (p. 315)": "string",
"streamInfResolution (p. 315)": enum,
"timestampDeltaMilliseconds (p. 315)": integer,
"manifestDurationFormat (p. 315)": enum,
"keyProviderSettings (p. 315)": {
  "staticKeySettings (p. 327)": {
    "staticKeyValue (p. 353)": "string",
    "KeyProviderServer (p. 353)": {
      "passwordParam (p. 323)": "string",
      "url (p. 323)": "string",
      "username (p. 323)": "string"
    }
  }
},
"tsFileMode (p. 315)": enum,
"manifestDurationFormat (p. 315)": enum,
"keyFormatVersions (p. 315)": "string",
"streamInfResolution (p. 315)": enum,
"timestampDeltaMilliseconds (p. 315)": integer,
"manifestDurationFormat (p. 315)": enum,
"keyProviderSettings (p. 315)": {
  "staticKeySettings (p. 327)": {
    "staticKeyValue (p. 353)": "string",
    "KeyProviderServer (p. 353)": {
      "passwordParam (p. 323)": "string",
      "url (p. 323)": "string",
      "username (p. 323)": "string"
    }
  }
},
"tsFileMode (p. 315)": enum,
<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>filecacheDuration (p. 319)</code></td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td><code>audioDescriptions (p. 299)</code></td>
<td>array</td>
<td></td>
</tr>
<tr>
<td><code>languageCodeControl (p. 272)</code></td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>audioTypeControl (p. 272)</code></td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>remixSettings (p. 272)</code></td>
<td>array</td>
<td></td>
</tr>
<tr>
<td><code>outputChannel (p. 271)</code></td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td><code>inputChannelLevels (p. 271)</code></td>
<td>array</td>
<td></td>
</tr>
<tr>
<td><code>channelsOut (p. 347)</code></td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td><code>channelsIn (p. 347)</code></td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td><code>audioType (p. 272)</code></td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>name (p. 273)</code></td>
<td>string</td>
<td></td>
</tr>
<tr>
<td><code>languageCode (p. 273)</code></td>
<td>string</td>
<td></td>
</tr>
<tr>
<td><code>codecSettings (p. 273)</code></td>
<td>array</td>
<td></td>
</tr>
<tr>
<td><code>aacSettings (p. 271)</code></td>
<td>array</td>
<td></td>
</tr>
<tr>
<td><code>vbrQuality (p. 266)</code></td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>codingMode (p. 267)</code></td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>profile (p. 267)</code></td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>bitrate (p. 267)</code></td>
<td>number</td>
<td></td>
</tr>
<tr>
<td><code>inputType (p. 267)</code></td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>rawFormat (p. 267)</code></td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>sampleRate (p. 267)</code></td>
<td>number</td>
<td></td>
</tr>
<tr>
<td><code>rateControlMode (p. 267)</code></td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>spec (p. 268)</code></td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>ac3Settings (p. 271)</code></td>
<td>array</td>
<td></td>
</tr>
<tr>
<td><code>drcProfile (p. 269)</code></td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>codingMode (p. 269)</code></td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>metadataControl (p. 269)</code></td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>bitrate (p. 269)</code></td>
<td>number</td>
<td></td>
</tr>
<tr>
<td><code>lfeFilter (p. 269)</code></td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>bitstreamMode (p. 270)</code></td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>eac3Settings (p. 272)</code></td>
<td>array</td>
<td></td>
</tr>
<tr>
<td><code>dialnorm (p. 294)</code></td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td><code>metadataControl (p. 295)</code></td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>bitrate (p. 295)</code></td>
<td>number</td>
<td></td>
</tr>
<tr>
<td><code>surroundEXMode (p. 295)</code></td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>ltRtSurroundMixLevel (p. 295)</code></td>
<td>number</td>
<td></td>
</tr>
<tr>
<td><code>lfeControl (p. 295)</code></td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>codingMode (p. 296)</code></td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>surroundMode (p. 296)</code></td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>attenuationControl (p. 296)</code></td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>lfeFilter (p. 296)</code></td>
<td>enum</td>
<td></td>
</tr>
<tr>
<td><code>ltRtCenterMixLevel (p. 296)</code></td>
<td>number</td>
<td></td>
</tr>
<tr>
<td><code>dcFilter (p. 296)</code></td>
<td>enum</td>
<td></td>
</tr>
</tbody>
</table>
"phaseControl (p. 296)": enum,
"stereoDownmix (p. 297)": enum,
"bitstreamMode (p. 297)": enum,
"loRoSurroundMixLevel (p. 297)": number,
"drcRf (p. 297)": enum,
"loRoCenterMixLevel (p. 297)": number
},
"passThroughSettings (p. 272)": {
},
"mp2Settings (p. 272)": {
"codingMode (p. 340)": enum,
"bitrate (p. 340)": number,
"sampleRate (p. 340)": number
},
"streamName (p. 273)": "string",
"audioNormalizationSettings (p. 273)": {
"targetLkfs (p. 274)": number,
"algorithmControl (p. 274)": enum,
"algorithm (p. 275)": enum
},
"audioSelectorName (p. 273)": "string"
},
"captionDescriptions (p. 299)": [
{
"captionSelectorName (p. 282)": "string",
"languageDescription (p. 282)": "string",
"name (p. 282)": "string",
"languageCode (p. 283)": "string",
"destinationSettings (p. 283)": {
"scte27DestinationSettings (p. 283)": {
},
"burnInDestinationSettings (p. 283)": {
"xPosition (p. 279)": integer,
"backgroundColor (p. 279)": enum,
"yPosition (p. 279)": integer,
"teletextGridControl (p. 279)": enum,
"backgroundOpacity (p. 279)": integer,
"fontOpacity (p. 280)": integer,
"fontResolution (p. 280)": integer,
"shadowOpacity (p. 280)": integer,
"shadowYOffset (p. 280)": integer,
"outlineSize (p. 280)": integer,
"outlineColor (p. 280)": enum,
"fontColor (p. 280)": enum,
"font (p. 280)": {
"passwordParam (p. 323)": "string",
"uri (p. 323)": "string",
"username (p. 323)": "string"
}
},
"teletextDestinationSettings (p. 283)": {
},
"smpteTtDestinationSettings (p. 283)": {
},
"webvttDestinationSettings (p. 283)": {
},
"ttmlDestinationSettings (p. 283)": {
"styleControl (p. 354)": enum
},
"embeddedPlusScte20DestinationSettings (p. 283)": {
}
},
"dvbSubDestinationSettings (p. 284)": {
"xPosition (p. 289)": integer,
"backgroundColor (p. 290)": enum,
"yPosition (p. 290)": integer,
"teletextGridControl (p. 290)": enum,
"backgroundOpacity (p. 290)": integer,
"fontOpacity (p. 290)": integer,
"fontResolution (p. 291)": integer,
"shadowOpacity (p. 291)": integer,
"shadowYOffset (p. 291)": integer,
"outlineSize (p. 291)": integer,
"outlineColor (p. 291)": enum,
"fontSize (p. 291)": "string",
"shadowXOffset (p. 292)": integer,
"alignment (p. 292)": enum,
"shadowColor (p. 292)": enum,
"fontColor (p. 292)": enum,
"font (p. 292)": {
  "passwordParam (p. 323)": "string",
  "uri (p. 323)": "string",
  "username (p. 323)": "string"
}
},
"embeddedDestinationSettings (p. 284)": {
},
"rtmpCaptionInfoDestinationSettings (p. 284)": {
},
"aribDestinationSettings (p. 284)": {
},
"scte20PlusEmbeddedDestinationSettings (p. 284)": {
}
},

"availConfiguration (p. 299)": {
"availSettings (p. 277)": {
  "scte35TimeSignalApos (p. 277)": {
    "adAvailOffset (p. 351)": integer,
    "webDeliveryAllowedFlag (p. 351)": enum,
    "noRegionalBlackoutFlag (p. 352)": enum
  },
  "scte35SpliceInsert (p. 277)": {
    "adAvailOffset (p. 351)": integer,
    "webDeliveryAllowedFlag (p. 351)": enum,
    "noRegionalBlackoutFlag (p. 351)": enum
  }
},
"globalConfiguration (p. 299)": {
  "inputLossBehavior (p. 301)": {
    "inputLossImageType (p. 324)": enum,
    "inputLossImageColor (p. 324)": "string",
    "inputLossImageSlate (p. 324)": {
      "passwordParam (p. 323)": "string",
      "uri (p. 323)": "string",
      "username (p. 323)": "string"
    },
    "blackFrameMsec (p. 324)": integer,
    "repeatFrameMsec (p. 325)": integer
  },
  "supportLowFramerateInputs (p. 301)": enum,
  "initialAudioGain (p. 301)": integer,
  "inputEndAction (p. 301)": enum,
  "outputTimingSource (p. 301)": enum
},

263
"videoDescriptions (p. 299)"[:
    {
        "respondToAfd (p. 356)" : enum,
        "scalingBehavior (p. 356)" : enum,
        "name (p. 356)" : "string",
        "width (p. 356)" : integer,
        "sharpness (p. 356)" : integer,
        "codecSettings (p. 357)" : {
            "h264Settings (p. 356)" : {
                "minIInterval (p. 304)" : integer,
                "slices (p. 304)" : integer,
                "parNumerator (p. 304)" : integer,
                "gopSizeUnits (p. 304)" : enum,
                "maxBitrate (p. 304)" : integer,
                "bitrate (p. 305)" : integer,
                "bufFillPct (p. 305)" : integer,
                "temporalAq (p. 305)" : enum,
                "sfdSignaling (p. 305)" : enum,
                "timecodeInsertion (p. 305)" : enum,
                "bufSize (p. 305)" : integer,
                "softness (p. 306)" : integer,
                "framerateControl (p. 306)" : enum,
                "fixedAfd (p. 306)" : enum,
                "level (p. 306)" : enum,
                "lookAheadRateControl (p. 306)" : enum,
                "profile (p. 306)" : enum,
                "framerateNumerator (p. 306)" : integer,
                "gopClosedCadence (p. 307)" : integer,
                "framerateDenominator (p. 307)" : integer,
                "entropyEncoding (p. 307)" : enum,
                "spatialAq (p. 307)" : enum,
                "adaptiveQuantization (p. 307)" : enum,
                "colorMetadata (p. 307)" : enum,
                "gopSize (p. 307)" : number,
                "numRefFrames (p. 308)" : integer,
                "gopSReference (p. 308)" : enum,
                "sceneChangeDetect (p. 308)" : enum,
                "parControl (p. 308)" : enum,
                "parDenominator (p. 308)" : integer,
                "syntax (p. 308)" : enum,
                "scanType (p. 308)" : enum,
                "gopNumBFrames (p. 309)" : integer,
                "flickerAq (p. 309)" : enum,
                "rateControlMode (p. 309)" : enum
            }
        },
        "height (p. 357)" : integer
    }
],
"blackoutSlate (p. 299)" : {
    "networkEndBlackoutImage (p. 277)" : {
        "passwordParam (p. 323)" : "string",
        "uri (p. 323)" : "string",
        "username (p. 323)" : "string"
    },
    "networkEndBlackout (p. 278)" : enum,
    "networkId (p. 278)" : "string",
    "state (p. 278)" : enum,
    "blackoutSlateImage (p. 278)" : {
        "passwordParam (p. 323)" : "string",
        "uri (p. 323)" : "string",
        "username (p. 323)" : "string"
    }
},
"availBlanking (p. 300)" : {
    "state (p. 276)" : enum,
"availBlankingImage (p. 277)": {
  "passwordParam (p. 323)": "string",
  "uri (p. 323)": "string",
  "username (p. 323)": "string"
}
},
"pipelinesRunningCount (p. 286)": integer,
"state (p. 286)": enum,
"id (p. 287)": "string",
"egressEndpoints (p. 287)": [
  {
    "sourceIp (p. 287)": "string"
  }
],
"arn (p. 287)": "string",
"inputSpecification (p. 287)": {
  "codec (p. 327)": enum,
  "resolution (p. 327)": enum,
  "maximumBitrate (p. 327)": enum
}
}

Example InvalidRequest
{
  "message (p. 327)": "string"
}

Example AccessDenied
{
  "message (p. 270)": "string"
}

Example ResourceNotFound
{
  "message (p. 348)": "string"
}

Example ResourceConflict
{
  "message (p. 347)": "string"
}

Example LimitExceeded
{
  "message (p. 327)": "string"
}

Example InternalServiceError
{
  "message (p. 327)": "string"
}
Example BadGatewayException

```json
{
  "message (p. 277)": "string"
}
```

Example GatewayTimeoutException

```json
{
  "message (p. 301)": "string"
}
```

Properties

**AacCodingMode** (enum)

- AD_RECEIVER_MIX
- CODING_MODE_1_0
- CODING_MODE_1_1
- CODING_MODE_2_0
- CODING_MODE_5_1

**AacInputType** (enum)

- BROADCASTER_MIXED_AD
- NORMAL

**AacProfile** (enum)

- HEV1
- HEV2
- LC

**AacRateControlMode** (enum)

- CBR
- VBR

**AacRawFormat** (enum)

- LATM_LOAS
- NONE

**AacSettings**

**vbrQuality**

VBR Quality Level - Only used if rateControlMode is VBR.

**Type:** string
Properties

Required: False

codingMode

Mono, Stereo, or 5.1 channel layout. Valid values depend on rate control mode and profile. The adReceiverMix 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: string
    Required: False

profile

AAC Profile.

    Type: string
    Required: False

bitrate

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

    Type: number
    Required: False

inputType

Set to "broadcasterMixedAd" when input contains pre-mixed main audio + AD (narration) as a stereo pair. The Audio Type field (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. The values in audioTypeControl and audioType (in AudioDescription) are ignored when set to broadcasterMixedAd. Leave set to "normal" when input does not contain pre-mixed audio + AD.

    Type: string
    Required: False

rawFormat

Sets LATM / LOAS AAC output for raw containers.

    Type: string
    Required: False

sampleRate

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

    Type: number
    Required: False

rateControlMode

Rate Control Mode.
**Properties**

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

**spec**

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

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

**AacSpec (enum)**

- MPEG2
- MPEG4

**AacVbrQuality (enum)**

- HIGH
- LOW
- MEDIUM_HIGH
- MEDIUM_LOW

**Ac3BitstreamMode (enum)**

- COMMENTARY
- COMPLETE_MAIN
- DIALOGUE
- EMERGENCY
- HEARING_IMPAIRED
- MUSIC_AND_EFFECTS
- VISUALLY_IMPAIRED
- VOICE_OVER

**Ac3CodingMode (enum)**

- CODING_MODE_1_0
- CODING_MODE_1_1
- CODING_MODE_2_0
- CODING_MODE_3_2_LFE

**Ac3DrcProfile (enum)**

- FILM_STANDARD
- NONE

**Ac3LfeFilter (enum)**

- DISABLED
- ENABLED

**Ac3MetadataControl (enum)**

- FOLLOW_INPUT
USE_CONFIGURED

**Ac3Settings**

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

Type: string  
Required: False

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

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

codingMode
Dolby Digital coding mode. Determines number of channels.

Type: string  
Required: False

metadataControl
When set to "followInput", 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: string  
Required: False

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

Type: number  
Required: False

lfeFilter
When set to enabled, applies a 120Hz lowpass filter to the LFE channel prior to encoding. Only valid in codingMode32Lfe mode.

Type: string  
Required: False
**bitstreamMode**

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

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

**AccessDenied**

**message**

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

**AfdSignaling (enum)**

- AUTO
- FIXED
- NONE

**ArchiveContainerSettings**

**m2tsSettings**

- **Type**: M2tsSettings (p. 329)
- **Required**: False

**ArchiveGroupSettings**

**destination**

A directory and base filename where archive files should be written. If the base filename portion of the URI is left blank, the base filename of the first input will be automatically inserted.

- **Type**: OutputLocationRef (p. 346)
- **Required**: True

**rolloverInterval**

Number of seconds to write to archive file before closing and starting a new one.

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

**ArchiveOutputSettings**

**extension**

Output file extension. If excluded, this will be auto-selected from the container type.
Properties

Type: string
Required: False

containerSettings

Settings specific to the container type of the file.

Type: ArchiveContainerSettings (p. 270)
Required: True

nameModifier

String concatenated to the end of the destination filename. Required for multiple outputs of the same type.

Type: string
Required: False

AribDestinationSettings

AribSourceSettings

AudioChannelMapping

outputChannel

The index of the output channel being produced.

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

inputChannelLevels

Indices and gain values for each input channel that should be remixed into this output channel.

Type: Array of type InputChannelLevel (p. 322)
Required: True

AudioCodecSettings

aacSettings

Type: AacSettings (p. 266)
Required: False

ac3Settings

Type: Ac3Settings (p. 269)
**Properties**

**Required:** False

**eac3Settings**

*Type:* Eac3Settings (p. 294)

*Required:* False

**passThroughSettings**

*Type:* PassThroughSettings (p. 347)

*Required:* False

**mp2Settings**

*Type:* Mp2Settings (p. 340)

*Required:* False

### AudioDescription

**languageCodeControl**

Choosing followInput will cause the ISO 639 language code of the output to follow the ISO 639 language code of the input. The languageCode will be used when useConfigured is set, or when followInput is selected but there is no ISO 639 language code specified by the input.

*Type:* string

*Required:* False

**audioTypeControl**

Determines how audio type is determined. followInput: If the input contains an ISO 639 audioType, then that value is passed through to the output. If the input contains no ISO 639 audioType, the value in Audio Type is included in the output. useConfigured: The value in Audio Type is included in the output. Note that this field and audioType are both ignored if inputType is broadcasterMixedAd.

*Type:* string

*Required:* False

**remixSettings**

Settings that control how input audio channels are remixed into the output audio channels.

*Type:* RemixSettings (p. 347)

*Required:* False

**audioType**

Applies only if audioTypeControl is useConfigured. The values for audioType are defined in ISO-IEC 13818-1.

*Type:* string

*Required:* False
Properties

name
The name of this AudioDescription. Outputs will use this name to uniquely identify this AudioDescription. Description names should be unique within this Live Event.

Type: string
Required: True

languageCode
Indicates the language of the audio output track. Only used if languageControlMode is useConfigured, or there is no ISO 639 language code specified in the input.

Type: string
Required: False

codecSettings
Audio codec settings.

Type: AudioCodecSettings (p. 271)
Required: False

streamName
Used for MS Smooth and Apple HLS outputs. Indicates the name displayed by the player (eg. English, or Director Commentary).

Type: string
Required: False

audioNormalizationSettings
Advanced audio normalization settings.

Type: AudioNormalizationSettings (p. 274)
Required: False

audioSelectorName
The name of the AudioSelector used as the source for this AudioDescription.

Type: string
Required: True

AudioDescriptionAudioTypeControl (enum)

FOLLOW_INPUT
USE_CONFIGURED

AudioDescriptionLanguageCodeControl (enum)

FOLLOW_INPUT
USE_CONFIGURED
**AudioLanguageSelection**

**languageSelectionPolicy**
When set to "strict", the transport stream demux strictly identifies audio streams by their language descriptor. If a PMT update occurs such that an audio stream matching the initially selected language is no longer present then mute will be encoded until the language returns. If "loose", then on a PMT update the demux will choose another audio stream in the program with the same stream type if it can't find one with the same language.

*Type: string*
*Required: False*

**languageCode**
Selects a specific three-letter language code from within an audio source.

*Type: string*
*Required: True*

**AudioLanguageSelectionPolicy** (enum)
- LOOSE
- STRICT

**AudioNormalizationAlgorithm** (enum)
- ITU_1770_1
- ITU_1770_2

**AudioNormalizationAlgorithmControl** (enum)
- CORRECT_AUDIO

**AudioNormalizationSettings**

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

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

**algorithmControl**
When set to correctAudio the output audio is corrected using the chosen algorithm. If set to measureOnly, the audio will be measured but not adjusted.

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

Audio normalization algorithm to use. itu17701 conforms to the CALM Act specification, itu17702 conforms to the EBU R-128 specification.

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

**AudioOnlyHlsSettings**

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

**audioGroupId**

Specifies the group to which the audio Rendition belongs.

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

**audioOnlyImage**

For use with an audio only Stream. Must be a .jpg or .png file. If given, this image will be used as the cover-art for the audio only output. Ideally, it should be formatted for an iPhone screen for two reasons. The iPhone does not resize the image, it crops a centered image on the top/bottom and left/right. Additionally, this image file gets saved bit-for-bit into every 10-second segment file, so will increase bandwidth by {image file size} * {segment count} * {user count}..

- **Type:** InputLocation (p. 323)
- **Required:** False

**AudioOnlyHlsTrackType (enum)**

- ALTERNATE_AUDIO_AUTO_SELECT
- ALTERNATE_AUDIO_AUTO_SELECT_DEFAULT
- ALTERNATE_AUDIO_NOT_AUTO_SELECT
- AUDIO_ONLY_VARIANT_STREAM

**AudioPidSelection**

**pid**

Selects a specific PID from within a source.
Type: integer  
Required: True  
Minimum: 0  
Maximum: 8191  

**AudioSelector**

**name**

The name of this AudioSelector. AudioDescriptions will use this name to uniquely identify this Selector. Selector names should be unique per input.

Type: string  
Required: True  

**selectorSettings**

The audio selector settings.

Type: AudioSelectorSettings (p. 276)  
Required: False  

**AudioSelectorSettings**

**audioLanguageSelection**

Type: AudioLanguageSelection (p. 274)  
Required: False  

**audioPidSelection**

Type: AudioPidSelection (p. 275)  
Required: False  

**AudioType (enum)**

- CLEAN_EFFECTS  
- HEARING_IMPAIRED  
- UNDEFINED  
- VISUAL_IMPAIRED_COMMENTARY  

**AuthenticationScheme (enum)**

- AKAMAI  
- COMMON  

**AvailBlanking**

**state**

When set to enabled, causes video, audio and captions to be blanked when insertion metadata is added.
Type: string  
Required: False

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

  Type: InputLocation (p. 323)  
  Required: False

**AvailBlankingState (enum)**

- DISABLED
- ENABLED

**AvailConfiguration**

**availSettings**
Ad avail settings.

  Type: AvailSettings (p. 277)  
  Required: False

**AvailSettings**

**scte35TimeSignalApos**

  Type: Scte35TimeSignalApos (p. 351)  
  Required: False

**scte35SpliceInsert**

  Type: Scte35SpliceInsert (p. 351)  
  Required: False

**BadGatewayException**

**message**

  Type: string  
  Required: False

**BlackoutSlate**

**networkEndBlackoutImage**
Path to local file to use as Network End Blackout image. Image will be scaled to fill the entire output raster.

  Type: InputLocation (p. 323)
**网络黑屏**
设置为启用时，会将视频、音频和字幕黑屏，并在遇到SCTE104/35网络结束分割描述符时，触发“网络黑屏图像”画片。黑屏将会在遇到网络开始分割描述符时解除。网络结束和网络开始描述符必须包含一个与在"网络ID"中输入的值匹配的网络ID。

*Type:* string  
*Required:* False

**networkId**
提供匹配EIDR ID格式的网络ID（例如，"10.XXXX-XXXX-XXXX-XXXX-XXXX-XXXX-C"）。  

*Type:* string  
*Required:* False

**state**
当设置为启用时，会导致视频、音频和字幕被空白显示，根据节目元数据指示。

*Type:* string  
*Required:* False

**blackoutSlateImage**
黑屏画片用于显示。留空为纯黑色。只支持bmp和png图像。

*Type:* InputLocation (p. 323)  
*Required:* False

**BlackoutSlateNetworkEndBlackout (enum)**

- DISABLED  
- ENABLED

**BlackoutSlateState (enum)**

- DISABLED  
- ENABLED

**BurnInAlignment (enum)**

- CENTERED  
- LEFT  
- SMART

**BurnInBackgroundColor (enum)**

- BLACK
NONE
WHITE

BurnInDestinationSettings

xPosition

Specifies the horizontal position of the caption relative to the left side of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the left of the output. If no explicit xPosition is provided, the horizontal caption position will be determined by the alignment parameter. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: 0

backgroundColor

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

Type: string
Required: False

yPosition

Specifies the vertical position of the caption relative to the top of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the top of the output. If no explicit yPosition is provided, the caption will be positioned towards the bottom of the output. All burn-in and DVB-Sub font settings must match.

Type: integer
Required: False
Minimum: 0

teletextGridControl

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

Type: string
Required: False

backgroundOpacity

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

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

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

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

fontResolution

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

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

shadowOpacity

Specifies the opacity of the shadow. 255 is opaque; 0 is transparent. Leaving this parameter out 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

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

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

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: string
Required: False

textSize
When set to 'auto' textSize will scale depending on the size of the output. Giving a positive integer will specify the exact font size in points. All burn-in and DVB-Sub font settings must match.

Type: string
Required: False

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

alignment
If no explicit xPosition or yPosition 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. Selecting "smart" justification will left-justify live subtitles and center-justify pre-recorded subtitles. All burn-in and DVB-Sub font settings must match.

Type: string
Required: False

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

Type: string
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: string
Required: False

font
External font file used for caption burn-in. File extension must be 'ttf' or 'tte'. Although the user can select output fonts for many different types of input captions, embedded, STL and teletext sources use a strict grid system. Using external fonts with these caption sources could cause unexpected display of proportional fonts. All burn-in and DVB-Sub font settings must match.
Properties

Type: InputLocation (p. 323)
Required: False

BurnInFontColor (enum)
BLACK
BLUE
GREEN
RED
WHITE
YELLOW

BurnInOutlineColor (enum)
BLACK
BLUE
GREEN
RED
WHITE
YELLOW

BurnInShadowColor (enum)
BLACK
NONE
WHITE

BurnInTeletextGridControl (enum)
FIXED
SCALED

CaptionDescription

captionSelectorName
Specifies which input caption selector to use as a caption source when generating output captions. This field should match a captionSelector name.

Type: string
Required: True

languageDescription
Human readable information to indicate captions available for players (e.g., English, or Spanish).

Type: string
Required: False

name
Name of the caption description. Used to associate a caption description with an output. Names must be unique within an event.
Properties

**Type**: string

**Required**: True

**languageCode**

**Type**: string

**Required**: False

**destinationSettings**
Additional settings for captions destination that depend on the destination type.

**Type**: CaptionDestinationSettings (p. 283)

**Required**: False

**CaptionDestinationSettings**

**scte27DestinationSettings**

**Type**: Scte27DestinationSettings (p. 350)

**Required**: False

**burnInDestinationSettings**

**Type**: BurnInDestinationSettings (p. 279)

**Required**: False

**teletextDestinationSettings**

**Type**: TeletextDestinationSettings (p. 353)

**Required**: False

**smpteTtDestinationSettings**

**Type**: SmpteTtDestinationSettings (p. 353)

**Required**: False

**webvttDestinationSettings**

**Type**: WebvttDestinationSettings (p. 359)

**Required**: False

**ttm1DestinationSettings**

**Type**: TtmlDestinationSettings (p. 354)

**Required**: False

**embeddedPlusScte20DestinationSettings**

**Type**: EmbeddedPlusScte20DestinationSettings (p. 298)
Properties

Required: False

dvbSubDestinationSettings

Type: DvbSubDestinationSettings (p. 289)
Required: False

embeddedDestinationSettings

Type: EmbeddedDestinationSettings (p. 298)
Required: False

rtmpCaptionInfoDestinationSettings

Type: RtmpCaptionInfoDestinationSettings (p. 348)
Required: False

aribDestinationSettings

Type: AribDestinationSettings (p. 271)
Required: False

scte20PlusEmbeddedDestinationSettings

Type: Scte20PlusEmbeddedDestinationSettings (p. 350)
Required: False

CaptionLanguageMapping

languageDescription

Textual description of language

Type: string
Required: True

captionChannel

Channel to insert closed captions. Each channel mapping must have a unique channel number (maximum of 4)

Type: integer
Required: True
Minimum: 1
Maximum: 4

languageCode

Three character ISO 639-2 language code (see http://www.loc.gov/standards/iso639-2)

Type: string
Required: True
CaptionSelector

name

Name identifier for a caption selector. This name is used to associate this caption selector with one or more caption descriptions. Names must be unique within an event.

  Type: string
  Required: True

languageCode

When specified this field indicates the three letter language code of the caption track to extract from the source.

  Type: string
  Required: False

selectorSettings

Caption selector settings.

  Type: CaptionSelectorSettings (p. 285)
  Required: False

CaptionSelectorSettings

embeddedSourceSettings

  Type: EmbeddedSourceSettings (p. 298)
  Required: False

scte20SourceSettings

  Type: Scte20SourceSettings (p. 350)
  Required: False

dvbSubSourceSettings

  Type: DvbSubSourceSettings (p. 293)
  Required: False

aribSourceSettings

  Type: AribSourceSettings (p. 271)
  Required: False

teletextSourceSettings

  Type: TeletextSourceSettings (p. 353)
  Required: False
scte27SourceSettings

Type: Scte27SourceSettings (p. 350)  
Required: False

Channel

inputAttachments

List of input attachments for channel.

Type: Array of type InputAttachment (p. 322)  
Required: False

roleArn

The Amazon Resource Name (ARN) of the role assumed when running the Channel.

Type: string  
Required: False

destinations

A list of destinations of the channel. For UDP outputs, there is one destination per output. For other types (HLS, for example), there is one destination per packager.

Type: Array of type OutputDestination (p. 344)  
Required: False

name

The name of the channel. (user-mutable)

Type: string  
Required: False

encoderSettings

Type: EncoderSettings (p. 299)  
Required: False

pipelinesRunningCount

The number of currently healthy pipelines.

Type: integer  
Required: False

state

Type: string  
Required: False
### id

The unique id of the channel.

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

### egressEndpoints

The endpoints where outgoing connections initiate from

- **Type:** Array of type `ChannelEgressEndpoint (p. 287)`
- **Required:** False

### arn

The unique arn of the channel.

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

### inputSpecification

- **Type:** InputSpecification (p. 327)
- **Required:** False

### ChannelEgressEndpoint

#### sourceIp

Public IP of where a channel's output comes from

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

### ChannelState (enum)

- CREATING
- CREATE_FAILED
- IDLE
- STARTING
- RUNNING
- RECOVERING
- STOPPING
- DELETING
- DELETED

### DvbNitSettings

#### networkName

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

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

- **Type:** integer
- **Required:** True
- **Minimum:** 0
- **Maximum:** 65536

**repInterval**

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

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

**DvbSdtOutputSdt (enum)**

- SDT_FOLLOW
- SDT_FOLLOW_IF_PRESENT
- SDT_MANUAL
- SDT_NONE

**DvbSdtSettings**

**serviceName**

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

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

**serviceProviderName**

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

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

**repInterval**

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

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

**Minimum**: 25  
**Maximum**: 2000

### outputSdt

Selects method of inserting SDT information into output stream. The `sdtdFollow` setting copies SDT information from input stream to output stream. The `sdtdFollowIfPresent` setting 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. The `sdtdManual` setting means user will enter the SDT information. The `sdtdNone` setting means output stream will not contain SDT information.

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

### DvbSubDestinationAlignment (enum)

- CENTERED
- LEFT
- SMART

### DvbSubDestinationBackgroundColor (enum)

- BLACK
- NONE
- WHITE

### DvbSubDestinationFontColor (enum)

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

### DvbSubDestinationOutlineColor (enum)

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

### DvbSubDestinationSettings

**xPosition**

Specifies the horizontal position of the caption relative to the left side of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the left of the output. If no explicit `xPosition` 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.

**backgroundColor**

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

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

**yPosition**

Specifies the vertical position of the caption relative to the top of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the top of the output. If no explicit yPosition 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

**teletextGridControl**

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

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

**backgroundOpacity**

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

**Type:** integer  
**Required:** False

**Minimum:** 0

**Maximum:** 255

**fontOpacity**

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

**Type:** integer  
**Required:** False

**Minimum:** 0

**Maximum:** 255
**fontResolution**

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

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

**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

**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

**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

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

**fontSize**

When set to auto fontSize will scale depending on the size of the output. Giving a positive integer will specify the exact font size in points. All burn-in and DVB-Sub font settings must match.

- **Type**: string
Properties

**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

**alignment**

If no explicit xPosition or yPosition 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. Selecting "smart" justification will left-justify live subtitles and center-justify pre-recorded subtitles. This option is not valid for source captions that are STL or 608/embedded. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

*Type:* string  
*Required:* False

**shadowColor**

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

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

**font**

External font file used for caption burn-in. File extension must be 'ttf' or 'tte'. Although the user can select output fonts for many different types of input captions, embedded, STL and teletext sources use a strict grid system. Using external fonts with these caption sources could cause unexpected display of proportional fonts. All burn-in and DVB-Sub font settings must match.

*Type:* InputLocation (p. 323)  
*Required:* False

**DvbSubDestinationShadowColor (enum)**

BLACK
NONE
DvbSubDestinationTeletextGridControl (enum)

- FIXED
- SCALED

DvbSubSourceSettings

pid

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

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

DvbTdtSettings

repInterval

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

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

Eac3AttenuationControl (enum)

- ATTENUATE_3_DB
- NONE

Eac3BitstreamMode (enum)

- COMMENTARY
- COMPLETE_MAIN
- EMERGENCY
- HEARING_IMPAIRED
- VISUALLY_IMPAIRED

Eac3CodingMode (enum)

- CODING_MODE_1_0
- CODING_MODE_2_0
- CODING_MODE_3_2

Eac3DcFilter (enum)

- DISABLED
- ENABLED
Eac3DrcLine (enum)

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

Eac3DrcRf (enum)

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

Eac3LfeControl (enum)

- LFE
- NO_LFE

Eac3LfeFilter (enum)

- DISABLED
- ENABLED

Eac3MetadataControl (enum)

- FOLLOW_INPUT
- USE_CONFIGURED

Eac3PassthroughControl (enum)

- NO_PASSTHROUGH
- WHEN_POSSIBLE

Eac3PhaseControl (enum)

- NO_SHIFT
- SHIFT_90_DEGREES

Eac3Settings

dialnorm

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

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

**passthroughControl**

When set to whenPossible, 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:** string
- **Required:** False

**metadataControl**

When set to followInput, 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:** string
- **Required:** False

**drcLine**

Sets the Dolby dynamic range compression profile.

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

**bitrate**

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

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

**ltRtSurroundMixLevel**

Left total/Right total surround mix level. Only used for 3/2 coding mode.

- **Type:** number
- **Required:** False

**lfeControl**

When encoding 3/2 audio, setting to lfe enables the LFE channel
**Properties**

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

- **codingMode**
  Dolby Digital Plus coding mode. Determines number of channels.

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

- **attenuationControl**
  When set to attenuate3Db, applies a 3 dB attenuation to the surround channels. Only used for 3/2 coding mode.

- **lfeFilter**
  When set to enabled, applies a 120Hz lowpass filter to the LFE channel prior to encoding. Only valid with codingMode32 coding mode.

- **ltRtCenterMixLevel**
  Left total/Right total center mix level. Only used for 3/2 coding mode.

- **dcFilter**
  When set to enabled, activates a DC highpass filter for all input channels.

- **phaseControl**
  When set to shift90Degrees, applies a 90-degree phase shift to the surround channels. Only used for 3/2 coding mode.
**stereoDownmix**

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

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

**loRoSurroundMixLevel**

Left only/Right only surround mix level. Only used for 3/2 coding mode.

*Type*: number  
*Required*: False

**drcRf**

Sets the profile for heavy Dolby dynamic range compression, ensures that the instantaneous signal peaks do not exceed specified levels.

*Type*: string  
*Required*: False

**loRoCenterMixLevel**

Left only/Right only center mix level. Only used for 3/2 coding mode.

*Type*: number  
*Required*: False

**Eac3StereoDownmix** (enum)

DPL2  
LO_RO  
LT_RT  
NOT_INDICATED

**Eac3SurroundExMode** (enum)

DISABLED  
ENABLED  
NOT_INDICATED

**Eac3SurroundMode** (enum)

DISABLED
### Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ENABLED</strong></td>
<td></td>
</tr>
<tr>
<td><strong>NOT_INDICATED</strong></td>
<td></td>
</tr>
</tbody>
</table>

#### EmbeddedConvert608To708 (enum)

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DISabled</td>
<td></td>
</tr>
<tr>
<td>UPconvert</td>
<td></td>
</tr>
</tbody>
</table>

#### EmbeddedDestinationSettings

#### EmbeddedPlusScte20DestinationSettings

#### EmbeddedScte20Detection (enum)

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUTO</td>
<td></td>
</tr>
<tr>
<td>OFF</td>
<td></td>
</tr>
</tbody>
</table>

#### EmbeddedSourceSettings

**scte20Detection**

Set to "auto" to handle streams with intermittent and/or non-aligned SCTE-20 and Embedded captions.

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>string</td>
<td></td>
</tr>
<tr>
<td>Required</td>
<td>False</td>
</tr>
</tbody>
</table>

**source608ChannelNumber**

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

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>integer</td>
<td></td>
</tr>
<tr>
<td>Required</td>
<td>False</td>
</tr>
<tr>
<td>Minimum</td>
<td>1</td>
</tr>
<tr>
<td>Maximum</td>
<td>4</td>
</tr>
</tbody>
</table>

**convert608To708**

If 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.

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>string</td>
<td></td>
</tr>
<tr>
<td>Required</td>
<td>False</td>
</tr>
</tbody>
</table>

**source608TrackNumber**

This field is unused and deprecated.

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>integer</td>
<td></td>
</tr>
<tr>
<td>Required</td>
<td>False</td>
</tr>
<tr>
<td>Minimum</td>
<td>1</td>
</tr>
</tbody>
</table>
Maximum: 5

**EncoderSettings**

**timecodeConfig**

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

- **Type:** TimecodeConfig (p. 353)
- **Required:** True

**outputGroups**

- **Type:** Array of type OutputGroup (p. 345)
- **Required:** True

**audioDescriptions**

- **Type:** Array of type AudioDescription (p. 272)
- **Required:** True

**captionDescriptions**

Settings for caption descriptions

- **Type:** Array of type CaptionDescription (p. 282)
- **Required:** False

**availConfiguration**

Event-wide configuration settings for ad avail insertion.

- **Type:** AvailConfiguration (p. 277)
- **Required:** False

**globalConfiguration**

Configuration settings that apply to the event as a whole.

- **Type:** GlobalConfiguration (p. 301)
- **Required:** False

**videoDescriptions**

- **Type:** Array of type VideoDescription (p. 356)
- **Required:** True

**blackoutSlate**

Settings for blackout slate.

- **Type:** BlackoutSlate (p. 277)
- **Required:** False
**availBlanking**

Settings for ad avail blanking.

- **Type:** AvailBlanking (p. 276)
- **Required:** False

**FecOutputIncludeFec (enum)**

- COLUMN
- COLUMN_AND_ROW

**FecOutputSettings**

**rowLength**

Parameter L from SMPTE 2022-1. The width of the FEC protection matrix. Must be between 1 and 20, inclusive. If only Column FEC is used, then larger values increase robustness. If Row FEC is used, then this is the number of transport stream packets per row error correction packet, and the value must be between 4 and 20, inclusive, if includeFec is columnAndRow. If includeFec is column, this value must be 1 to 20, inclusive.

- **Type:** integer
- **Required:** False
- **Minimum:** 1
- **Maximum:** 20

**columnDepth**

Parameter D from SMPTE 2022-1. The height of the FEC protection matrix. The number of transport stream packets per column error correction packet. Must be between 4 and 20, inclusive.

- **Type:** integer
- **Required:** False
- **Minimum:** 4
- **Maximum:** 20

**includeFec**

Enables column only or column and row based FEC

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

**FixedAfd (enum)**

- AFD_0000
- AFD_0010
- AFD_0011
- AFD_0100
- AFD_1000
- AFD_1001
- AFD_1010
- AFD_1011
AFD_1101
AFD_1110
AFD_1111

**GatewayTimeoutException**

**message**

*Type:* string  
*Required:* False

**GlobalConfiguration**

**inputLossBehavior**

Settings for system actions when input is lost.

*Type:* [InputLossBehavior](#)  
*Required:* False

**supportLowFramerateInputs**

Adjusts video input buffer for streams with very low video framerates. This is commonly set to enabled for music channels with less than one video frame per second.

*Type:* string  
*Required:* False

**initialAudioGain**

Value to set the initial audio gain for the Live Event.

*Type:* integer  
*Required:* False  
*Minimum:* -60  
*Maximum:* 60

**inputEndAction**

Indicates the action to take when an input completes (e.g. end-of-file.) Options include immediately switching to the next sequential input (via "switchInput"), switching to the next input and looping back to the first input when last input ends (via "switchAndLoopInputs") or not switching inputs and instead transcoding black / color / slate images per the "Input Loss Behavior" configuration until an activateInput REST command is received (via "none").

*Type:* string  
*Required:* False

**outputTimingSource**

Indicates whether the rate of frames emitted by the Live encoder should be paced by its system clock (which optionally may be locked to another source via NTP) or should be locked to the clock of the source that is providing the input stream.
Type: string
Required: False

GlobalConfigurationInputEndAction (enum)
NONE
SWITCH_AND_LOOP_INPUTS

GlobalConfigurationLowFramerateInputs (enum)
DISABLED
ENABLED

GlobalConfigurationOutputTimingSource (enum)
INPUT_CLOCK
SYSTEM_CLOCK

H264AdaptiveQuantization (enum)
HIGH
HIGHER
LOW
MAX
MEDIUM
OFF

H264ColorMetadata (enum)
IGNORE
INSERT

H264EntropyEncoding (enum)
CABAC
CAVLC

H264FlickerAq (enum)
DISABLED
ENABLED

H264FramerateControl (enum)
INITIALIZE_FROM_SOURCE
SPECIFIED

H264GopBReference (enum)
DISABLED
ENABLED
H264GopSizeUnits (enum)

FRAMES
SECONDS

H264Level (enum)

H264_LEVEL_1
H264_LEVEL_1_1
H264_LEVEL_1_2
H264_LEVEL_1_3
H264_LEVEL_2
H264_LEVEL_2_1
H264_LEVEL_2_2
H264_LEVEL_3
H264_LEVEL_3_1
H264_LEVEL_3_2
H264_LEVEL_4
H264_LEVEL_4_1
H264_LEVEL_4_2
H264_LEVEL_5
H264_LEVEL_5_1
H264_LEVEL_5_2
H264_LEVEL_AUTO

H264LookAheadRateControl (enum)

HIGH
LOW
MEDIUM

H264ParControl (enum)

INITIALIZE_FROM_SOURCE
SPECIFIED

H264Profile (enum)

BASELINE
HIGH
HIGH_10BIT
HIGH_422
HIGH_422_10BIT
MAIN

H264RateControlMode (enum)

CBR
VBR

H264ScanType (enum)

INTERLACED
PROGRESSIVE

H264SceneChangeDetect (enum)

DISABLED  ENABLED

H264Settings

minIInterval

Only meaningful if sceneChangeDetect is set to enabled. 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. Note: Maximum GOP stretch = GOP size + Min-I-interval - 1

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

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. This field is optional; when no value is specified the encoder will choose the number of slices based on encode resolution.

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

parNumerator

Pixel Aspect Ratio numerator.

Type: integer
Required: False

gopSizeUnits

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

Type: string
Required: False

maxBitrate

Maximum bitrate in bits/second (for VBR mode only).

Type: integer
Required: False
Minimum: 1000

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

Type: integer
Required: False
Minimum: 1000

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

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

temporalAq
If set to enabled, adjust quantization within each frame based on temporal variation of content complexity.

Type: string
Required: False

afdSignaling
Indicates that AFD values will be written into the output stream. If afdSignaling is "auto", the system will try to preserve the input AFD value (in cases where multiple AFD values are valid). If set to "fixed", the AFD value will be the value configured in the fixedAfd parameter.

Type: string
Required: False

timecodeInsertion
Determines how timecodes should be inserted into the video elementary stream. - 'disabled': Do not include timecodes - 'picTimingSel': Pass through picture timing SEI messages from the source specified in Timecode Config

Type: string
Required: False

bufSize
Size of buffer (HRD buffer model) in bits/second.

Type: integer
Required: False
Minimum: 0
**softness**

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

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

**framerateControl**

This field indicates how the output video frame rate is specified. If "specified" is selected then the output video frame rate is determined by framerateNumerator and framerateDenominator, else if "initializeFromSource" is selected then the output video frame rate will be set equal to the input video frame rate of the first input.

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

**fixedAfd**

Four bit AFD value to write on all frames of video in the output stream. Only valid when afdSignaling is set to 'Fixed'.

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

**level**

H.264 Level.

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

**lookAheadRateControl**

Amount of lookahead. A value of low can decrease latency and memory usage, while high can produce better quality for certain content.

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

**profile**

H.264 Profile.

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

**framerateNumerator**

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

- **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

**framerateDenominator**
Framerate denominator.

*Type*: integer  
*Required*: False

**entropyEncoding**
Enterity encoding mode. Use cabac (must be in Main or High profile) or cavlc.

*Type*: string  
*Required*: False

**spatialAq**
If set to enabled, adjust quantization within each frame based on spatial variation of content complexity.

*Type*: string  
*Required*: False

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

*Type*: string  
*Required*: False

**colorMetadata**
Includes colorspace metadata in the output.

*Type*: string  
*Required*: False

**gopSize**
GOP size (keyframe interval) in units of either frames or seconds per gopSizeUnits. Must be greater than zero.

*Type*: number  
*Required*: False
Minimum: 1.0

**numRefFrames**
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

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

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

**sceneChangeDetect**
Scene change detection. Inserts I-frames on scene changes when enabled.

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

**parControl**
This field indicates how the output pixel aspect ratio is specified. If "specified" is selected then the output video pixel aspect ratio is determined by parNumerator and parDenominator, else if "initializeFromSource" is selected then the output pixel aspect ratio will be set equal to the input video pixel aspect ratio of the first input.

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

**parDenominator**
Pixel Aspect Ratio denominator.

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

**syntax**
Produces a bitstream compliant with SMPTE RP-2027.

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

**scanType**
Sets the scan type of the output to progressive or top-field-first interlaced.
**Type:** string  
**Required:** False

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

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

**flickerAq**
If set to enabled, adjust quantization within each frame to reduce flicker or 'pop' on I-frames.

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

**rateControlMode**
Rate control mode.

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

**H264SpatialAq (enum)**

DISABLED  
ENABLED

**H264Syntax (enum)**

DEFAULT  
RP2027

**H264TemporalAq (enum)**

DISABLED  
ENABLED

**H264TimecodeInsertionBehavior (enum)**

DISABLED  
PIC_TIMING_SEI

**HlsAdMarkers (enum)**

ADOBE  
ELEMENTAL  
ELEMENTAL_SCTE35
HlsAkamaiHttpTransferMode (enum)

- CHUNKED
- NON_CHUNKED

HlsAkamaiSettings

httpTransferMode

Specify whether or not to use chunked transfer encoding to Akamai. User should contact Akamai to enable this feature.

- Type: string
- Required: False

salt

Salt for authenticated Akamai.

- Type: string
- Required: False

numRetries

Number of retry attempts that will be made before the Live Event is put into an error state.

- Type: integer
- Required: False
- Minimum: 0

restartDelay

If a streaming output fails, number of seconds to wait until a restart is initiated. A value of 0 means never restart.

- Type: integer
- Required: False
- Minimum: 0
- Maximum: 15

connectionRetryInterval

Number of seconds to wait before retrying connection to the CDN if the connection is lost.

- Type: integer
- Required: False
- Minimum: 0

filecacheDuration

Size in seconds of file cache for streaming outputs.

- Type: integer
token

Token parameter for authenticated akamai. If not specified, _gda_ is used.

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

**HlsBasicPutSettings**

**numRetries**

Number of retry attempts that will be made before the Live Event is put into an error state.

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

**restartDelay**

If a streaming output fails, number of seconds to wait until a restart is initiated. A value of 0 means never restart.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 15

**connectionRetryInterval**

Number of seconds to wait before retrying connection to the CDN if the connection is lost.

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

**filecacheDuration**

Size in seconds of file cache for streaming outputs.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 600

**HlsCaptionLanguageSetting (enum)**

- INSERT
- NONE
- OMIT
HlsCdnSettings

hlsAkamaiSettings
  Type: HlsAkamaiSettings (p. 310)
  Required: False

hlsWebdavSettings
  Type: HlsWebdavSettings (p. 321)
  Required: False

hlsBasicPutSettings
  Type: HlsBasicPutSettings (p. 311)
  Required: False

hlsMediaStoreSettings
  Type: HlsMediaStoreSettings (p. 319)
  Required: False

HlsClientCache (enum)
  DISABLED
  ENABLED

HlsCodecSpecification (enum)
  RFC_4281
  RFC_6381

HlsDirectoryStructure (enum)
  SINGLE_DIRECTORY
  SUBDIRECTORY_PER_STREAM

HlsEncryptionType (enum)
  AES128
  SAMPLE_AES

HlsGroupSettings

segmentsPerSubdirectory
  Number of segments to write to a subdirectory before starting a new one. directoryStructure must be subdirectoryPerStream for this setting to have an effect.
  Type: integer
  Required: False
  Minimum: 1
Properties

ivInManifest

For use with encryptionType. The IV (Initialization Vector) is a 128-bit number used in conjunction with the key for encrypting blocks. If set to "include", IV is listed in the manifest, otherwise the IV is not in the manifest.

Type: string
Required: False

outputSelection

Generates the .m3u8 playlist file for this HLS output group. The segmentsOnly option will output segments without the .m3u8 file.

Type: string
Required: False

encryptionType

Encrypts the segments with the given encryption scheme. Exclude this parameter if no encryption is desired.

Type: string
Required: False

indexNSegments

Number of segments to keep in the playlist (.m3u8) file. mode must be "vod" for this setting to have an effect, and this number should be less than or equal to keepSegments.

Type: integer
Required: False
Minimum: 3

destination

A directory or HTTP destination for the HLS segments, manifest files, and encryption keys (if enabled).

Type: OutputLocationRef (p. 346)
Required: True

constantIv

For use with encryptionType. This is a 128-bit, 16-byte hex value represented by a 32-character text string. If ivSource is set to "explicit" then this parameter is required and is used as the IV for encryption.

Type: string
Required: False

timedMetadataId3Frame

Indicates ID3 frame that has the timecode.

Type: string
**Properties**

**baseUrlManifest**

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

**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: string  
Required: False

**minSegmentLength**

When set, minimumSegmentLength 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

**mode**

If set to "vod", keeps and indexes all segments starting with the first segment. If set to "live" segments will age out and only the last keepSegments number of segments will be retained.

Type: string  
Required: False

**ivSource**

For use with encryptionType. The IV (Initialization Vector) is a 128-bit number used in conjunction with the key for encrypting blocks. If this setting is "followsSegmentNumber", it will cause the IV to change every segment (to match the segment number). If this is set to "explicit", you must enter a constant IV value.

Type: string  
Required: False

**manifestCompression**

When set to gzip, compresses HLS playlist.

Type: string
**keyProviderSettings**

The key provider settings.

*Type: KeyProviderSettings (p. 327)*

*Required: False*

**tsFileMode**

When set to "singleFile", emits the program as a single media resource (.ts) file, and uses #EXT-X-BYTERANGE tags to index segment for playback. Playback of VOD mode content during event is not guaranteed due to HTTP server caching.

*Type: string*

*Required: False*

**manifestDurationFormat**

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

*Type: string*

*Required: False*

**keyFormatVersions**

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

*Type: string*

*Required: False*

**streamInfResolution**

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

*Type: string*

*Required: False*

**timestampDeltaMilliseconds**

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

*Type: integer*

*Required: False*

*Minimum: 0*

**segmentationMode**

When set to useInputSegmentation, the output segment or fragment points are set by the RAI markers from the input streams.

*Type: string*

*Required: False*
**baseUrlContent**

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

**clientCache**

When set to "disabled", sets the #EXT-X-ALLOW-CACHE:no tag in the manifest, which prevents clients from saving media segments for later replay.

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

**captionLanguageMappings**

Mapping of up to 4 caption channels to caption languages. Is only meaningful if captionLanguageSetting is set to "insert".

- **Type**: Array of type CaptionLanguageMapping (p. 284)
- **Required**: False

**codecSpecification**

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

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

**keepSegments**

Number of segments to retain in the destination directory. mode must be "live" for this setting to have an effect.

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

**timedMetadataId3Period**

Timed Metadata interval in seconds.

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

**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 timestampOffset.

- **Type**: string
**directoryStructure**
Place segments in subdirectories.

*Type:* string  
*Required:* False

**keyFormat**
The value specifies how the key is represented in the resource identified by the URI. If parameter is absent, an implicit value of "identity" is used. A reverse DNS string can also be given.

*Type:* string  
*Required:* False

**inputLossAction**
Parameter that control output group behavior on input loss.

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

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

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

**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

**hlsCdnSettings**
Parameters that control interactions with the CDN.

*Type:* HlsCdnSettings (p. 312)
Required: False

**HlsInputSettings**

**retries**

The number of consecutive times that attempts to read a manifest or segment must fail before the input is considered unavailable.

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

**bandwidth**

When specified the HLS stream with the m3u8 BANDWIDTH that most closely matches this value will be chosen, otherwise the highest bandwidth stream in the m3u8 will be chosen. The bitrate is specified in bits per second, as in an HLS manifest.

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

**retryInterval**

The number of seconds between retries when an attempt to read a manifest or segment fails.

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

**bufferSegments**

When specified, reading of the HLS input will begin this many buffer segments from the end (most recently written segment). When not specified, the HLS input will begin with the first segment specified in the m3u8.

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

**HlsIvInManifest (enum)**

EXCLUDE
INCLUDE

**HlsIvSource (enum)**

EXPLICIT
FOLLOWS_SEGMENT_NUMBER

**HlsManifestCompression (enum)**

GZIP
NONE

**HlsManifestDurationFormat (enum)**

- FLOATING_POINT
- INTEGER

**HlsMediaStoreSettings**

**mediaStoreStorageClass**

When set to temporal, output files are stored in non-persistent memory for faster reading and writing.

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

**numRetries**

Number of retry attempts that will be made before the Live Event is put into an error state.

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

**restartDelay**

If a streaming output fails, number of seconds to wait until a restart is initiated. A value of 0 means never restart.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 15

**connectionRetryInterval**

Number of seconds to wait before retrying connection to the CDN if the connection is lost.

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

**filecacheDuration**

Size in seconds of file cache for streaming outputs.

- **Type:** integer
- **Required:** False
- **Minimum:** 0
- **Maximum:** 600

**HlsMediaStoreStorageClass (enum)**

- TEMPORAL
HlsMode (enum)

- LIVE
- VOD

HlsOutputSelection (enum)

- MANIFESTS_AND_SEGMENTS
- SEGMENTS_ONLY

HlsOutputSettings

segmentModifier

String concatenated to end of segment filenames.

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

hlsSettings

Settings regarding the underlying stream. These settings are different for audio-only outputs.

- **Type**: HlsSettings (p. 320)
- **Required**: True

nameModifier

String concatenated to the end of the destination filename. Accepts "Format Identifiers \"\"Format Identifier Parameters."

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

HlsProgramDateTime (enum)

- EXCLUDE
- INCLUDE

HlsSegmentationMode (enum)

- USE_INPUT_SEGMENTATION
- USE_SEGMENT_DURATION

HlsSettings

audioOnlyHlsSettings

- **Type**: AudioOnlyHlsSettings (p. 275)
- **Required**: False
standardHlsSettings

- **Type**: StandardHlsSettings (p. 353)
- **Required**: False

HlsStreamInfResolution (enum)

- EXCLUDE
- INCLUDE

HlsTimedMetadataId3Frame (enum)

- NONE
- PRIV
- TDRL

HlsTsFileMode (enum)

- SEGMENTED_FILES
- SINGLE_FILE

HlsWebdavHttpTransferMode (enum)

- CHUNKED
- NON_CHUNKED

**HlsWebdavSettings**

**httpTransferMode**

Specify whether or not to use chunked transfer encoding to WebDAV.

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

**numRetries**

Number of retry attempts that will be made before the Live Event is put into an error state.

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

**restartDelay**

If a streaming output fails, number of seconds to wait until a restart is initiated. A value of 0 means never restart.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
Maximum: 15

**connectionRetryInterval**

Number of seconds to wait before retrying connection to the CDN if the connection is lost.

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

**filecacheDuration**

Size in seconds of file cache for streaming outputs.

  - **Type:** integer
  - **Required:** False
  - **Minimum:** 0
  - **Maximum:** 600

**InputAttachment**

**inputId**

The ID of the input

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

**inputSettings**

Settings of an input (caption selector, etc.)

  - **Type:** InputSettings (p. 325)
  - **Required:** False

**InputChannelLevel**

**inputChannel**

The index of the input channel used as a source.

  - **Type:** integer
  - **Required:** True
  - **Minimum:** 0
  - **Maximum:** 15

**gain**

Remixing value. Units are in dB and acceptable values are within the range from -60 (mute) and 6 dB.

  - **Type:** integer
  - **Required:** True
Minimum: -60
Maximum: 6

**InputCodec (enum)**

codec in increasing order of complexity

- MPEG2
- AVC
- HEVC

**InputDeblockFilter (enum)**

- DISABLED
- ENABLED

**InputDenoiseFilter (enum)**

- DISABLED
- ENABLED

**InputFilter (enum)**

- AUTO
- DISABLED
- FORCED

**InputLocation**

**passwordParam**

key used to extract the password from EC2 Parameter store

- Type: string
- Required: False

**uri**

Uniform Resource Identifier - This should be a path to a file accessible to the Live system (eg. a http:// URI) depending on the output type. For example, a rtmpEndpoint should have a uri similar to: "rtmp:// fmsserver/live".

- Type: string
- Required: True

**username**

Username if credentials are required to access a file or publishing point. This can be either a plaintext username, or a reference to an AWS parameter store name from which the username can be retrieved. AWS Parameter store format: "ssm://<parameter name>"

- Type: string
**Properties**

**Required**: False

**InputLossActionForHlsOut** (enum)

- EMIT_OUTPUT
- PAUSE_OUTPUT

**InputLossActionForMsSmoothOut** (enum)

- EMIT_OUTPUT
- PAUSE_OUTPUT

**InputLossActionForUdpOut** (enum)

- DROP_PROGRAM
- DROP_TS
- EMIT_PROGRAM

**InputLossBehavior**

**inputLossImageType**

Indicates whether to substitute a solid color or a slate into the output after input loss exceeds blackFrameMsec.

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

**inputLossImageColor**

When input loss image type is "color" this field specifies the color to use. Value: 6 hex characters representing the values of RGB.

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

**inputLossImageSlate**

When input loss image type is "slate" these fields specify the parameters for accessing the slate.

- **Type**: InputLocation (p. 323)
- **Required**: False

**blackFrameMsec**

On input loss, the number of milliseconds to substitute black into the output before switching to the frame specified by inputLossImageType. A value x, where 0 <= x <= 1,000,000 and a value of 1,000,000 will be interpreted as infinite.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
Maximum: 1000000

repeatFrameMsec

On input loss, the number of milliseconds to repeat the previous picture before substituting black into the output. A value x, where 0 <= x <= 1,000,000 and a value of 1,000,000 will be interpreted as infinite.

Type: integer
Required: False
Minimum: 0
Maximum: 1000000

InputLossImageType (enum)

COLOR
SLATE

InputMaximumBitrate (enum)

Maximum input bitrate in megabits per second. Bitrates up to 50 Mbps are supported currently.

MAX_10_MBPS
MAX_20_MBPS
MAX_50_MBPS

InputResolution (enum)

Input resolution based on lines of vertical resolution in the input; SD is less than 720 lines, HD is 720 to 1080 lines, UHD is greater than 1080 lines

SD
HD
UHD

InputSettings

sourceEndBehavior

Loop input if it is a file. This allows a file input to be streamed indefinitely.

Type: string
Required: False

audioSelectors

Used to select the audio stream to decode for inputs that have multiple available.

Type: Array of type AudioSelector (p. 276)
Required: False

deblockFilter

Enable or disable the deblock filter when filtering.
Properties

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

**networkInputSettings**
Input settings.

**Type:** NetworkInputSettings (p. 343)  
**Required:** False

**inputFilter**
Turns on the filter for this input. MPEG-2 inputs have the deblocking filter enabled by default. 1) auto - filtering will be applied depending on input type/quality 2) disabled - no filtering will be applied to the input 3) forced - filtering will be applied regardless of input type

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

**videoSelector**
Informs which video elementary stream to decode for input types that have multiple available.

**Type:** VideoSelector (p. 357)  
**Required:** False

**filterStrength**
Adjusts the magnitude of filtering from 1 (minimal) to 5 (strongest).

**Type:** integer  
**Required:** False  
**Minimum:** 1  
**Maximum:** 5

**captionSelectors**
Used to select the caption input to use for inputs that have multiple available.

**Type:** Array of type CaptionSelector (p. 285)  
**Required:** False

**denoiseFilter**
Enable or disable the denoise filter when filtering.

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

**InputSourceEndBehavior (enum)**
CONTINUE
InputSpecification

codec
Input codec
  Type: string
  Required: False

resolution
Input resolution, categorized coarsely
  Type: string
  Required: False

maximumBitrate
Maximum input bitrate, categorized coarsely
  Type: string
  Required: False

InternalServerError

message
  Type: string
  Required: False

InvalidRequest

message
  Type: string
  Required: False

KeyProviderSettings

staticKeySettings
  Type: StaticKeySettings (p. 353)
  Required: False

LimitExceeded

message
  Type: string
**Required**: False

**M2tsAbsentInputAudioBehavior (enum)**

- DROP
- ENCODE_SILENCE

**M2tsArib (enum)**

- DISABLED
- ENABLED

**M2tsAribCaptionsPidControl (enum)**

- AUTO
- USE_CONFIGURED

**M2tsAudioBufferModel (enum)**

- ATSC
- DVB

**M2tsAudioInterval (enum)**

- VIDEO_AND_FIXED_INTERVALS
- VIDEO_INTERVAL

**M2tsAudioStreamType (enum)**

- ATSC
- DVB

**M2tsBufferModel (enum)**

- MULTIPLEX
- NONE

**M2tsCcDescriptor (enum)**

- DISABLED
- ENABLED

**M2tsEbifControl (enum)**

- NONE
- PASSTHROUGH

**M2tsEbpPlacement (enum)**

- VIDEO_AND_AUDIO_PIDS
- VIDEO_PID
M2tsEsRateInPes (enum)
EXCLUDE
INCLUDE

M2tsKlv (enum)
NONE
PASSTHROUGH

M2tsPcrControl (enum)
CONFIGURED_PCR_PERIOD
PCR_EVERY_PES_PACKET

M2tsRateMode (enum)
CBR
VBR

M2tsScte35Control (enum)
NONE
PASSTHROUGH

M2tsSegmentationMarkers (enum)
EBP
EBP_LEGACY
NONE
PSI_SEGSTART
RAI_ADAPT
RAI_SEGSTART

M2tsSegmentationStyle (enum)
MAINTAIN_CADENCE
RESET_CADENCE

M2tsSettings

audioStreamType
When set to atsc, uses stream type = 0x81 for AC3 and stream type = 0x87 for EAC3. When set to dvb, uses stream type = 0x06.

Type: string
Required: False
**ecmPid**

Packet Identifier (PID) for ECM in the transport stream. Only enabled when Simulcrypt is enabled. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20)..8182 (or 0x1ff6).

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

**dvbTeletextPid**

Packet Identifier (PID) for input source DVB Teletext data to this output. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20)..8182 (or 0x1ff6).

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

**aribCaptionsPidControl**

If set to auto, pid number used for ARIB Captions will be auto-selected from unused pids. If set to useConfigured, ARIB Captions will be on the configured pid number.

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

**bitrate**

The output bitrate of the transport stream in bits per second. Setting to 0 lets the muxer automatically determine the appropriate bitrate.

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

**segmentationTime**

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

- **Type**: number
- **Required**: False
- **Minimum**: 1.0

**rateMode**

When vbr, does not insert null packets into transport stream to fill specified bitrate. The bitrate setting acts as the maximum bitrate when vbr is set.

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

**audioPids**

Packet Identifier (PID) of the elementary audio stream(s) in the transport stream. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values. Each PID specified must be in the range of 32 (or 0x20)..8182 (or 0x1ff6).
**Properties**

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

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

**Type**: number  
**Required**: False  
**Minimum**: 0.0

**ebpAudioInterval**
When `videoAndFixedIntervals` is selected, 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. Only available when EBP Cablelabs segmentation markers are selected. Partitions 1 and 2 will always follow the video interval.

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

**ebpLookaheadMs**
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

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

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

**scte35Pid**
Packet Identifier (PID) of the SCTE-35 stream in the transport stream. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20)..8182 (or 0x1ff6).

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

**pcrPeriod**
Maximum time in milliseconds between Program Clock Reference (PCRs) inserted into the transport stream.
**Properties**

**Type**: integer  
**Required**: False  
**Minimum**: 0  
**Maximum**: 500

**pmtInterval**

The number of milliseconds between instances of this table in the output transport stream. Valid values are 0, 10..1000.

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

**programNum**

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

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

**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 "resetCadence" 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 $segmentationTime seconds. When a segmentation style of "maintainCadence" 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 $segmentationTime seconds. Note that EBP lookahead is a slight exception to this rule.

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

**ebif**

If set to passthrough, passes any EBIF data from the input source to this output.

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

**audioBufferModel**

When set to dvb, uses DVB buffer model for Dolby Digital audio. When set to atsc, the ATSC model is used.

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

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

*Type: DvbNitSettings (p. 287)*
*Required: False*

**absentInputAudioBehavior**
When set to drop, output audio streams will be removed from the program if the selected input audio stream is removed from the input. This allows the output audio configuration to dynamically change based on input configuration. If this is set to encodeSilence, all output audio streams will output encoded silence when not connected to an active input stream.

*Type: string*
*Required: False*

**timedMetadataPid**
Packet Identifier (PID) of the timed metadata stream in the transport stream. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20)..8182 (or 0x1ff6).

*Type: string*
*Required: False*

**timedMetadataBehavior**
When set to passthrough, timed metadata will be passed through from input to output.

*Type: string*
*Required: False*

**etvSignalPid**
Packet Identifier (PID) for input source ETV Signal data to this output. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20)..8182 (or 0x1ff6).

*Type: string*
*Required: False*

**pmtPid**
Packet Identifier (PID) for the Program Map Table (PMT) in the transport stream. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20)..8182 (or 0x1ff6).

*Type: string*
*Required: False*

**bufferModel**
If set to multiplex, use multiplex buffer model for accurate interleaving. Setting to bufferModel to none can lead to lower latency, but low-memory devices may not be able to play back the stream without interruptions.

*Type: string*
**scte35Control**

Optionally pass SCTE-35 signals from the input source to this output.

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

**ebpPlacement**

Controls placement of EBP on Audio PIDs. If set to videoAndAudioPids, EBP markers will be placed on the video PID and all audio PIDs. If set to videoPid, EBP markers will be placed on only the video PID.

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

**arib**

When set to enabled, uses ARIB-compliant field muxing and removes video descriptor.

- **Type:** string
- **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
- **Minimum:** 0.0

**dvbSdtSettings**

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

- **Type:** DvbSdtSettings (p. 288)
- **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. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20).8182 (or 0x1ff6).

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

**transportStreamId**

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

- **Type:** integer
Properties

Required: False
Minimum: 0
Maximum: 65535

`pcrControl`

When set to `pcrEveryPesPacket`, 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: string
Required: False

`videoPid`

Packet Identifier (PID) of the elementary video stream in the transport stream. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20)..8182 (or 0x1ff6).

Type: string
Required: False

`esRateInPes`

Include or exclude the ES Rate field in the PES header.

Type: string
Required: False

`segmentationMarkers`

Inserts segmentation markers at each `segmentationTime` period. `raiSegstart` sets the Random Access Indicator bit in the adaptation field. `raiAdapt` sets the RAI bit and adds the current timecode in the private data bytes. `psiSegstart` 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. `ebpLegacy` adds Encoder Boundary Point information to the adaptation field using a legacy proprietary format.

Type: string
Required: False

`dvbTdtSettings`

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

Type: `DvbTdtSettings` (p. 293)
Required: False

`klv`

If set to `passthrough`, passes any KLV data from the input source to this output.

Type: string
Required: False
ccDescriptor
When set to enabled, generates captionServiceDescriptor in PMT.

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

patInterval
The number of milliseconds between instances of this table in the output transport stream. Valid values are 0, 10..1000.

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

etvPlatformPid
Packet Identifier (PID) for input source ETV Platform data to this output. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20)..8182 (or 0x1ff6).

  - **Type**: string
  - **Required**: False
dvbSubPids
Packet Identifier (PID) for input source DVB Subtitle data to this output. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values. Each PID specified must be in the range of 32 (or 0x20)..8182 (or 0x1ff6).

  - **Type**: string
  - **Required**: False
aribCaptionsPid
Packet Identifier (PID) for ARIB Captions in the transport stream. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20)..8182 (or 0x1ff6).

  - **Type**: string
  - **Required**: False
scte27Pids
Packet Identifier (PID) for input source SCTE-27 data to this output. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values. Each PID specified must be in the range of 32 (or 0x20)..8182 (or 0x1ff6).

  - **Type**: string
  - **Required**: False
klvDataPids
Packet Identifier (PID) for input source KLV data to this output. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values. Each PID specified must be in the range of 32 (or 0x20)..8182 (or 0x1ff6).
**Type**: string  
**Required**: False

**M2tsTimedMetadataBehavior (enum)**

- NO_PASSTHROUGH  
- PASSTHROUGH

**M3u8PcrControl (enum)**

- CONFIGURED_PCR_PERIOD  
- PCR_EVERY_PES_PACKET

**M3u8Scte35Behavior (enum)**

- NO_PASSTHROUGH  
- PASSTHROUGH

**M3u8Settings**

**pmtPid**

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

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

**ecmPid**

ThePlatform-protected transport streams using 'microsoft' as Target Client include an ECM stream. This ECM stream contains the size, IV, and PTS of every sample in the transport stream. This stream PID is specified here. This PID has no effect on non ThePlatform-protected streams.

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

**scte35Behavior**

If set to passthrough, passes any SCTE-35 signals from the input source to this output.

- **Type**: string  
- **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. Can be entered as a decimal or hexadecimal value.

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

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

* Type: string
* Required: False

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

* Type: integer
* Required: False

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

* Type: string
* Required: False

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

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

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

* Type: string
* Required: False

**pcrControl**
When set to pcrEveryPesPacket, 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: string
* Required: False

**pmtInterval**
The number of milliseconds between instances of this table in the output transport stream. A value of "0" writes out the PMT once per segment file.
aws element media live api reference api reference

properties

type: integer
required: false
minimum: 0
maximum: 1000

pcrPeriod

maximum time in milliseconds between program clock references (PCRs) inserted into the transport stream.

type: integer
required: false
minimum: 0
maximum: 500

programNum

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. A value of 0 writes out the PMT once per segment file.

type: integer
required: false
minimum: 0
maximum: 1000

timedMetadataPid

packet identifier (PID) of the timed metadata stream in the transport stream. Can be entered as a decimal or hexadecimal value. Valid values are 32 (0x20)..8182 (0x1ff6).

type: string
required: false

M3u8TimedMetadataBehavior (enum)

NO_PASSTHROUGH
PASSTHROUGH
Mp2CodingMode (enum)

- CODING_MODE_1_0
- CODING_MODE_2_0

Mp2Settings

codingMode

The MPEG2 Audio coding mode. Valid values are codingMode10 (for mono) or codingMode20 (for stereo).

- Type: string
- Required: False

bitrate

Average bitrate in bits/second.

- Type: number
- Required: False

sampleRate

Sample rate in Hz.

- Type: number
- Required: False

MsSmoothGroupSettings

eventId

MS Smooth event ID to be sent to the IIS server. Should only be specified if eventIdMode is set to useConfigured.

- Type: string
- Required: False

fragmentLength

Length of mp4 fragments to generate (in seconds). Fragment length must be compatible with GOP size and framerate.

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

timestampOffset

Timestamp offset for the event. Only used if timestampOffsetMode is set to useConfiguredOffset.

- Type: string
Required: False

**segmentationMode**

When set to useInputSegmentation, the output segment or fragment points are set by the RAI markers from the input streams.

Type: string
Required: False

**numRetries**

Number of retry attempts.

Type: integer
Required: False
Minimum: 0

**eventStopBehavior**

When set to sendEos, send EOS signal to IIS server when stopping the event

Type: string
Required: False

**acquisitionPointId**

The value of the "Acquisition Point Identity" element used in each message placed in the sparse track. Only enabled if sparseTrackType is not "none".

Type: string
Required: False

**sparseTrackType**

If set to scte35, use incoming SCTE-35 messages to generate a sparse track in this group of MS-Smooth outputs.

Type: string
Required: False

**timestampOffsetMode**

Type of timestamp date offset to use. - useEventStartDate: Use the date the event was started as the offset - useConfiguredOffset: Use an explicitly configured date as the offset

Type: string
Required: False

**destination**

Smooth Streaming publish point on an IIS server. Elemental Live acts as a "Push" encoder to IIS.

Type: OutputLocationRef (p. 346)
Required: True

**audioOnlyTimecodeControl**

If set to passthrough for an audio-only MS Smooth output, the fragment absolute time will be set to the current timecode. This option does not write timecodes to the audio elementary stream.

Type: string  
Required: False

**connectionRetryInterval**

Number of seconds to wait before retrying connection to the IIS server if the connection is lost. Content will be cached during this time and the cache will be delivered to the IIS server once the connection is re-established.

Type: integer  
Required: False  
Minimum: 0

**filecacheDuration**

Size in seconds of file cache for streaming outputs.

Type: integer  
Required: False  
Minimum: 0

**certificateMode**

If set to verifyAuthenticity, verify the https certificate chain to a trusted Certificate Authority (CA). This will cause https outputs to self-signed certificates to fail unless those certificates are manually added to the OS trusted keystore.

Type: string  
Required: False

**inputLossAction**

Parameter that control output group behavior on input loss.

Type: string  
Required: False

**sendDelayMs**

Outputs that are "output locked" can use this delay. Assign a delay to the output that is "secondary". Do not assign a delay to the "primary" output. The delay means that the primary output will always reach the downstream system before the secondary, which helps ensure that the downstream system always uses the primary output. (If there were no delay, the downstream system might flip-flop between whichever output happens to arrive first.) If the primary fails, the downstream system will switch to the secondary output. When the primary is restarted, the downstream system will switch back to the primary (because once again it is always arriving first).

Type: integer
Properties

Required: False
Minimum: 0
Maximum: 10000

eventIdMode

Specifies whether or not to send an event ID to the IIS server. If no event ID is sent and the same Live Event is used without changing the publishing point, clients might see cached video from the previous run. Options: - "useConfigured" - use the value provided in eventId - "useTimestamp" - generate and send an event ID based on the current timestamp - "noEventId" - do not send an event ID to the IIS server.

Type: string
Required: False

restartDelay

Number of seconds before initiating a restart due to output failure, due to exhausting the numRetries on one segment, or exceeding filecacheDuration.

Type: integer
Required: False
Minimum: 0

streamManifestBehavior

When set to send, send stream manifest so publishing point doesn’t start until all streams start.

Type: string
Required: False

MsSmoothOutputSettings

nameModifier

String concatenated to the end of the destination filename. Required for multiple outputs of the same type.

Type: string
Required: False

NetworkInputServerValidation (enum)

CHECK_CRYPTOGRAPHY_AND_VALIDATE_NAME
CHECK_CRYPTOGRAPHY_ONLY

NetworkInputSettings

hlsInputSettings

Specifies HLS input settings when the uri is for a HLS manifest.

Type: HlsInputSettings (p. 318)
Required: False
**serverValidation**

Check HTTPS server certificates. When set to checkCryptographyOnly, cryptography in the certificate will be checked, but not the server's name. Certain subdomains (notably S3 buckets that use dots in the bucket name) do not strictly match the corresponding certificate's wildcard pattern and would otherwise cause the event to error. This setting is ignored for protocols that do not use https.

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

**Output**

**videoDescriptionName**

The name of the VideoDescription used as the source for this output.

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

**captionDescriptionNames**

The names of the CaptionDescriptions used as caption sources for this output.

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

**outputName**

The name used to identify an output.

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

**outputSettings**

Output type-specific settings.

- **Type:** [OutputSettings](p. 346)
- **Required:** True

**audioDescriptionNames**

The names of the AudioDescriptions used as audio sources for this output.

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

**OutputDestination**

**settings**

Destination settings for output; one for each redundant encoder.

- **Type:** Array of type [OutputDestinationSettings](p. 345)
**Properties**

Required: False

id
User-specified id. This is used in an output group or an output.

  Type: string  
  Required: False

**OutputDestinationSettings**

passwordParam
key used to extract the password from EC2 Parameter store

  Type: string  
  Required: False

streamName
Stream name for RTMP destinations (URLs of type rtmp://)

  Type: string  
  Required: False

url
A URL specifying a destination

  Type: string  
  Required: False

username
username for destination

  Type: string  
  Required: False

**OutputGroup**

outputs

  Type: Array of type Output (p. 344)  
  Required: True

outputGroupSettings
Settings associated with the output group.

  Type: OutputGroupSettings (p. 346)  
  Required: True
name
Custom output group name optionally defined by the user. Only letters, numbers, and the underscore character allowed; only 32 characters allowed.

Type: string
Required: False

OutputGroupSettings

archiveGroupSettings

Type: ArchiveGroupSettings (p. 270)
Required: False

rtmpGroupSettings

Type: RtmpGroupSettings (p. 348)
Required: False

udpGroupSettings

Type: UdpGroupSettings (p. 354)
Required: False

msSmoothGroupSettings

Type: MsSmoothGroupSettings (p. 340)
Required: False

hlsGroupSettings

Type: HlsGroupSettings (p. 312)
Required: False

OutputLocationRef

destinationRefId

Type: string
Required: False

OutputSettings

rtmpOutputSettings

Type: RtmpOutputSettings (p. 349)
Required: False

archiveOutputSettings

Type: ArchiveOutputSettings (p. 270)
Required: False

**msSmoothOutputSettings**
- Type: MsSmoothOutputSettings (p. 343)
- Required: False

**udpOutputSettings**
- Type: UdpOutputSettings (p. 355)
- Required: False

**hlsOutputSettings**
- Type: HlsOutputSettings (p. 320)
- Required: False

**PassThroughSettings**

**RemixSettings**

**channelMappings**
Mapping of input channels to output channels, with appropriate gain adjustments.
- Type: Array of type AudioChannelMapping (p. 271)
- Required: True

**channelsOut**
Number of output channels to be produced. Valid values: 1, 2, 4, 6, 8
- Type: integer
- Required: False
- Minimum: 1
- Maximum: 8

**channelsIn**
Number of input channels to be used.
- Type: integer
- Required: False
- Minimum: 1
- Maximum: 16

**ResourceConflict**

**message**
- Type: string
- Required: False
ResourceNotFound

message

Type: string
Required: False

RtmpCacheFullBehavior (enum)

DISCONNECT_IMMEDIATELY
WAIT_FOR_SERVER

RtmpCaptionData (enum)

ALL
FIELD1_608
FIELD1_AND_FIELD2_608

RtmpCaptionInfoDestinationSettings

RtmpGroupSettings

captionData

Controls the types of data that passes to onCaptionInfo outputs. If set to 'all' then 608 and 708 carried DTVCC data will be passed. If set to 'field1AndField2608' then DTVCC data will be stripped out, but 608 data from both fields will be passed. If set to 'field1608' then only the data carried in 608 from field 1 video will be passed.

Type: string
Required: False

authenticationScheme

Authentication scheme to use when connecting with CDN

Type: string
Required: False

cacheLength

Cache length, in seconds, is used to calculate buffer size.

Type: integer
Required: False
Minimum: 30

restartDelay

If a streaming output fails, number of seconds to wait until a restart is initiated. A value of 0 means never restart.

Type: integer
Properties

**Required:** False

**cacheFullBehavior**

Controls behavior when content cache fills up. If remote origin server stalls the RTMP connection and does not accept content fast enough the 'Media Cache' will fill up. When the cache reaches the duration specified by cacheLength the cache will stop accepting new content. If set to disconnectImmediately, the RTMP output will force a disconnect. Clear the media cache, and reconnect after restartDelay seconds. If set to waitForServer, the RTMP output will wait up to 5 minutes to allow the origin server to begin accepting data again.

**Type:** string

**Required:** False

**RtmpOutputCertificateMode (enum)**

SELF_SIGNED

VERIFY_AUTHENTICITY

**RtmpOutputSettings**

**certificateMode**

If set to verifyAuthenticity, verify the tls certificate chain to a trusted Certificate Authority (CA). This will cause rtmps outputs with self-signed certificates to fail.

**Type:** string

**Required:** False

**numRetries**

Number of retry attempts.

**Type:** integer

**Required:** False

**Minimum:** 0

**destination**

The RTMP endpoint excluding the stream name (eg. rtmp://host/appname). For connection to Akamai, a username and password must be supplied. URI fields accept format identifiers.

**Type:** OutputLocationRef (p. 346)

**Required:** True

**connectionRetryInterval**

Number of seconds to wait before retrying a connection to the Flash Media server if the connection is lost.

**Type:** integer

**Required:** False

**Minimum:** 1
**Scte20Convert608To708 (enum)**

- DISABLED
- UPCONVERT

**Scte20PlusEmbeddedDestinationSettings**

**Scte20SourceSettings**

**source608ChannelNumber**

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

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

**convert608To708**

If 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**: string
- **Required**: False

**Scte27DestinationSettings**

**Scte27SourceSettings**

**pid**

The pid field is used in conjunction with the caption selector languageCode field as follows: - Specify PID and Language: Extracts captions from that PID; the language is "informational". - Specify PID and omit Language: Extracts the specified PID. - Omit PID and specify Language: Extracts the specified language, whichever PID that happens to be. - Omit PID and omit Language: Valid only if source is DVB-Sub that is being passed through; all languages will be passed through.

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

**Scte35AposNoRegionalBlackoutBehavior (enum)**

- FOLLOW
- IGNORE

**Scte35AposWebDeliveryAllowedBehavior (enum)**

- FOLLOW
- IGNORE
**Scte35SpliceInsert**

**adAvailOffset**

When specified, this offset (in milliseconds) is added to the input Ad Avail PTS time. This only applies to embedded SCTE 104/35 messages and does not apply to OOB messages.

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

**webDeliveryAllowedFlag**

When set to ignore, Segment Descriptors with webDeliveryAllowedFlag set to 0 will no longer trigger blackouts or Ad Avail slates

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

**noRegionalBlackoutFlag**

When set to ignore, Segment Descriptors with noRegionalBlackoutFlag set to 0 will no longer trigger blackouts or Ad Avail slates

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

**Scte35SpliceInsertNoRegionalBlackoutBehavior** (enum)

- FOLLOW
- IGNORE

**Scte35SpliceInsertWebDeliveryAllowedBehavior** (enum)

- FOLLOW
- IGNORE

**Scte35TimeSignalApos**

**adAvailOffset**

When specified, this offset (in milliseconds) is added to the input Ad Avail PTS time. This only applies to embedded SCTE 104/35 messages and does not apply to OOB messages.

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

**webDeliveryAllowedFlag**

When set to ignore, Segment Descriptors with webDeliveryAllowedFlag set to 0 will no longer trigger blackouts or Ad Avail slates

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

noRegionalBlackoutFlag

When set to ignore, Segment Descriptors with noRegionalBlackoutFlag set to 0 will no longer trigger blackouts or Ad Avail slates

Type: string
Required: False

SmoothGroupAudioOnlyTimecodeControl (enum)

PASSTHROUGH
USE_CONFIGURED_CLOCK

SmoothGroupCertificateMode (enum)

SELF_SIGNED
VERIFY_AUTHENTICITY

SmoothGroupEventIdMode (enum)

NO_EVENT_ID
USE_CONFIGURED
USE_TIMESTAMP

SmoothGroupEventStopBehavior (enum)

NONE
SEND_EOS

SmoothGroupSegmentationMode (enum)

USE_INPUT_SEGMENTATION
USE_SEGMENT_DURATION

SmoothGroupSparseTrackType (enum)

NONE
SCTE_35

SmoothGroupStreamManifestBehavior (enum)

DO_NOT_SEND
SEND

SmoothGroupTimestampOffsetMode (enum)

USE_CONFIGURED_OFFSET
USE_EVENT_START_DATE
SmpteTtDestinationSettings

StandardHlsSettings

m3u8Settings

  Type: M3u8Settings (p. 337)
  Required: True

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

StaticKeySettings

staticKeyValue

Static key value as a 32 character hexadecimal string.

  Type: string
  Required: True

keyProviderServer

The URL of the license server used for protecting content.

  Type: InputLocation (p. 323)
  Required: False

TeletextDestinationSettings

TeletextSourceSettings

pageNumber

Specifies the teletext page number within the data stream from which to extract captions. Range of 0x100 (256) to 0x8FF (2303). Unused for passthrough. Should be specified as a hexadecimal string with no "0x" prefix.

  Type: string
  Required: False

TimecodeConfig

syncThreshold

Threshold in frames beyond which output timecode is resynchronized to the input timecode. Discrepancies below this threshold are permitted to avoid unnecessary discontinuities in the output timecode. No timecode sync when this is not specified.
source

Identifies the source for the timecode that will be associated with the events outputs. - Embedded (embedded): Initialize the output timecode with timecode from the the source. If no embedded timecode is detected in the source, the system falls back to using "Start at 0" (zerobased). - System Clock (systemclock): Use the UTC time. - Start at 0 (zerobased): The time of the first frame of the event will be 00:00:00:00.

Type: string
Required: True

TimecodeConfigSource (enum)

EMBEDDED
SYSTEMCLOCK
ZEROBASED

TtmlDestinationSettings

styleControl

When set to passthrough, passes 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: string
Required: False

TtmlDestinationStyleControl (enum)

PASSTHROUGH
USE_CONFIGURED

UdpContainerSettings

m2tsSettings

Type: M2tsSettings (p. 329)
Required: False

UdpGroupSettings

inputLossAction

Specifies behavior of last resort when input video is lost, and no more backup inputs are available. When dropTs is selected the entire transport stream will stop being emitted. When dropProgram is selected the program can be dropped from the transport stream (and replaced with null packets to meet the TS
bitrate requirement). Or, when emitProgram is chosen the transport stream will continue to be produced normally with repeat frames, black frames, or slate frames substituted for the absent input video.

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

**timedMetadataId3Frame**

Indicates ID3 frame that has the timecode.

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

**timedMetadataId3Period**

Timed Metadata interval in seconds.

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

**UdpOutputSettings**

**bufferMsec**

UDP output buffering in milliseconds. Larger values increase latency through the transcoder but simultaneously assist the transcoder in maintaining a constant, low-jitter UDP/RTP output while accommodating clock recovery, input switching, input disruptions, picture reordering, etc.

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

**destination**

Destination address and port number for RTP or UDP packets. Can be unicast or multicast RTP or UDP (eg. rtp://239.10.10.10:5001 or udp://10.100.100.100:5002).

  **Type:** `OutputLocationRef (p. 346)`  
  **Required:** True

**containerSettings**

  **Type:** `UdpContainerSettings (p. 354)`  
  **Required:** True

**fecOutputSettings**

Settings for enabling and adjusting Forward Error Correction on UDP outputs.

  **Type:** `FecOutputSettings (p. 300)`  
  **Required:** False
UdpTimedMetadataId3Frame (enum)

- NONE
- PRIV
- TDRL

VideoCodecSettings

h264Settings

Type: H264Settings (p. 304)
Required: False

VideoDescription

respondToAfd

Indicates how to respond to the AFD values in the input stream. Setting to "respond" causes input video to be clipped, depending on AFD value, input display aspect ratio and output display aspect ratio.

Type: string
Required: False

scalingBehavior

When set to "stretchToOutput", automatically configures the output position to stretch the video to the specified output resolution. This option will override any position value.

Type: string
Required: False

name

The name of this VideoDescription. Outputs will use this name to uniquely identify this Description. Description names should be unique within this Live Event.

Type: string
Required: True

width

Output video width (in pixels). Leave out to use source video width. If left out, height must also be left out. Display aspect ratio is always preserved by letterboxing or pillarboxing when necessary.

Type: integer
Required: False

sharpness

Changes the width of the anti-alias filter kernel used for scaling. Only applies if scaling is being performed and antiAlias is set to true. 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.

Type: VideoCodecSettings (p. 356)
Required: False

height
Output video height (in pixels). Leave blank to use source video height. If left blank, width must also be unspecified.

Type: integer
Required: False

VideoDescriptionRespondToAfd (enum)

NONE
PASSTHROUGH
RESPOND

VideoDescriptionScalingBehavior (enum)

DEFAULT
STRETCH_TO_OUTPUT

VideoSelector
colorSpace
Specifies the colorspace of an input. This setting works in tandem with colorSpaceConversion to determine if any conversion will be performed.

Type: string
Required: False

selectorSettings
The video selector settings.

Type: VideoSelectorSettings (p. 358)
Required: False

colorSpaceUsage
Applies only if colorSpace is a value other than follow. This field controls how the value in the colorSpace field will be used. fallback means that when the input does include color space data, that data will be
used, but when the input has no color space data, the value in colorSpace will be used. Choose fallback if your input is sometimes missing color space data, but when it does have color space data, that data is correct. force means to always use the value in colorSpace. Choose force if your input usually has no color space data or might have unreliable color space data.

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

---

**VideoSelectorColorSpace (enum)**

- FOLLOW
- REC_601
- REC_709

---

**VideoSelectorColorSpaceUsage (enum)**

- FALLBACK
- FORCE

---

**VideoSelectorPid**

**pid**

Selects a specific PID from within a video source.

**Type**: integer  
**Required**: False  
**Minimum**: 0  
**Maximum**: 8191

---

**VideoSelectorProgramId**

**programId**

Selects a specific program from within a multi-program transport stream. If the program doesn't exist, the first program within the transport stream will be selected by default.

**Type**: integer  
**Required**: False  
**Minimum**: 0  
**Maximum**: 65536

---

**VideoSelectorSettings**

**videoSelectorPid**

- **Type**: VideoSelectorPid (p. 358)  
- **Required**: False

---

**videoSelectorProgramId**

- **Type**: VideoSelectorProgramId (p. 358)
**Required:** False

**WebvttDestinationSettings**

### Channels channelId Stop

#### URI

/prod/channels/ *channelId* /stop

#### HTTP Methods

**POST**

Operation ID: StopChannel

Stops a running channel

#### Path Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>channelId</td>
<td>String</td>
<td>True</td>
<td>channel ID</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>Channel (p. 360)</td>
<td>Successfully initiated stop of the channel.</td>
</tr>
<tr>
<td>400</td>
<td>InvalidRequest (p. 370)</td>
<td>This request was invalid.</td>
</tr>
<tr>
<td>500</td>
<td>InternalServiceError (p. 371)</td>
<td>Unexpected internal service error.</td>
</tr>
<tr>
<td>502</td>
<td>BadGatewayException (p. 371)</td>
<td>Bad Gateway Error</td>
</tr>
<tr>
<td>403</td>
<td>AccessDenied (p. 371)</td>
<td>You do not have permission to list channels.</td>
</tr>
<tr>
<td>404</td>
<td>ResourceNotFound (p. 371)</td>
<td>The channel you're requesting to describe does not exist.</td>
</tr>
<tr>
<td>504</td>
<td>GatewayTimeoutException (p. 371)</td>
<td>Gateway Timeout Error</td>
</tr>
<tr>
<td>429</td>
<td>LimitExceeded (p. 371)</td>
<td>Request limit exceeded on list channel calls to channel service.</td>
</tr>
<tr>
<td>409</td>
<td>ResourceConflict (p. 371)</td>
<td>The channel is unable to create due to an issue with channel resources.</td>
</tr>
</tbody>
</table>
Schemas

Response Bodies

Example Channel

```json
{
    "inputAttachments (p. 391)" : [
        {
            "inputId (p. 427)" : "string",
            "inputSettings (p. 427)" : {
                "sourceEndBehavior (p. 430)" : enum,
                "audioSelectors (p. 430)" : [
                    {
                        "name (p. 381)" : "string",
                        "selectorSettings (p. 381)" : {
                            "audioLanguageSelection (p. 381)" : {
                                "languageSelectionPolicy (p. 379)" : enum,
                                "languageCode (p. 379)" : "string"
                            },
                            "audioPidSelection (p. 382)" : {
                                "pid (p. 381)" : integer
                            }
                        }
                    }
                ],
                "deblockFilter (p. 431)" : enum,
                "networkInputSettings (p. 431)": {
                    "hlsInputSettings (p. 448)" : {
                        "retries (p. 423)" : integer,
                        "bandwidth (p. 423)" : integer,
                        "retryInterval (p. 423)" : integer,
                        "bufferSegments (p. 423)" : integer
                    },
                    "serverValidation (p. 449)" : enum
                },
                "inputFilter (p. 431)" : enum,
                "videoSelector (p. 431)" : {
                    "colorSpace (p. 463)" : enum,
                    "selectorSettings (p. 463)" : {
                        "videoSelectorPid (p. 464)" : {
                            "pid (p. 463)" : integer
                        },
                        "videoSelectorProgramId (p. 464)" : {
                            "programId (p. 464)" : integer
                        }
                    },
                    "colorSpaceUsage (p. 463)" : enum
                },
                "filterStrength (p. 431)" : integer,
                "captionSelectors (p. 431)" : [
                    {
                        "name (p. 390)" : "string",
                        "languageCode (p. 390)" : "string",
                        "selectorSettings (p. 390)" : {
                            "embeddedSourceSettings (p. 390)" : {
                                "scte20Detection (p. 403)" : enum,
                                "source608ChannelNumber (p. 403)" : integer,
                                "convert608To708 (p. 403)" : enum,
                                "source608TrackNumber (p. 404)" : integer
                            },
                            "scte20SourceSettings (p. 390)" : {
                                "source608ChannelNumber (p. 455)" : integer,
                            }
                        }
                    }
                ]
            }
        }
    ]
}
```
"convert608To708 (p. 455)": enum
,
"dvbSubSourceSettings (p. 391)": {
  "pid (p. 398)": integer
},
"aribSourceSettings (p. 391)": {
},
"teletextSourceSettings (p. 391)": {
  "pageNumber (p. 459)": "string"
},
"scte27SourceSettings (p. 391)": {
  "pid (p. 455)": integer
},
"denoiseFilter (p. 431)": enum
},
"roleArn (p. 391)": "string",
"destinations (p. 391)": [
  {
    "settings (p. 450)": [
    {
      "passwordParam (p. 450)": "string",
      "streamName (p. 450)": "string",
      "url (p. 450)": "string",
      "username (p. 450)": "string"
    },
    "id (p. 450)": "string"
    ],
  },
  "name (p. 391)": "string",
  "encoderSettings (p. 392)": {
    "timecodeConfig (p. 404)": {
      "syncThreshold (p. 459)": integer,
      "source (p. 459)": enum
    },
    "outputGroups (p. 404)": [
    {
      "outputs (p. 450)": [
      {
        "videoDescriptionName (p. 449)": "string",
        "captionDescriptionNames (p. 449)": [
        "string"
      },
      "outputName (p. 449)": "string",
      "outputSettings (p. 449)": {
        "rtmpOutputSettings (p. 452)": {
        "certificateMode (p. 454)": enum,
        "numRetries (p. 454)": integer,
        "destination (p. 454)": {
        "destinationRefId (p. 451)": "string"
        },
        "connectionRetryInterval (p. 455)": integer
      },
      "archiveOutputSettings (p. 452)": {
        "extension (p. 376)": "string",
        "containerSettings (p. 376)": {
        "m2tsSettings (p. 375)": {
        "audioStreamType (p. 434)": enum,
        "ecmPid (p. 435)": "string",
        "dvbTeletextPid (p. 435)": "string",
        "aribCaptionsPidControl (p. 435)": enum
      },
      "archivingSettings (p. 452)": {
        "maxBytes (p. 376)": integer,
        "maxDays (p. 376)": integer
      },
      "backupOutputSettings (p. 455)": {
        "destination (p. 455)": enum,
        "connectionRetryInterval (p. 455)": integer
      }
    },
    "liveOutputSettings (p. 455)": {
      "certificateMode (p. 454)": enum,
      "numRetries (p. 454)": integer,
      "destination (p. 454)": enum,
      "connectionRetryInterval (p. 455)": integer
    },
    "muxingSettings (p. 456)": {
      "type (p. 456)": enum,
      "streamName (p. 456)": "string",
      "settings (p. 456)": enum
    },
    "mediaSourceSettings (p. 457)": {
      "type (p. 457)": enum,
      "streamName (p. 457)": "string",
      "settings (p. 457)": enum
    },
    "encoderSettings (p. 458)": {
      "type (p. 458)": enum,
      "streamName (p. 458)": "string",
      "settings (p. 458)": enum
    },
    "aspectSourceSettings (p. 459)": {
      "type (p. 459)": enum,
      "streamName (p. 459)": "string",
      "settings (p. 459)": enum
    },
    "inputSourceSettings (p. 460)": {
      "type (p. 460)": enum,
      "streamName (p. 460)": "string",
      "settings (p. 460)": enum
    },
    "mixedSourceSettings (p. 461)": {
      "type (p. 461)": enum,
      "streamName (p. 461)": "string",
      "settings (p. 461)": enum
    },
    "sourceSettings (p. 462)": {
      "type (p. 462)": enum,
      "streamName (p. 462)": "string",
      "settings (p. 462)": enum
    },
    "version (p. 463)": "string",
    "arn (p. 463)": "string",
    "name (p. 463)": "string",
    "id (p. 463)": "string",
    "description (p. 463)": "string"
  }
}
"bitrate (p. 435)": integer,
"segmentationTime (p. 435)": number,
"rateMode (p. 435)": enum,
"audioPids (p. 436)": "string",
"fragmentTime (p. 436)": number,
"ebpAudioInterval (p. 436)": enum,
"ebpLookaheadMs (p. 436)": integer,
"audioFramesPerPes (p. 436)": integer,
"scte35Pid (p. 436)": "string",
"pmtInterval (p. 437)": integer,
"programNum (p. 437)": integer,
"segmentationStyle (p. 437)": enum,
"ebif (p. 437)": enum,
"audioBufferModel (p. 437)": enum,
"dvbNitSettings (p. 438)": {
  "networkName (p. 393)": "string",
  "networkId (p. 393)": integer,
  "repInterval (p. 393)": integer
},
"absentInputAudioBehavior (p. 438)": enum,
"timedMetadataPid (p. 438)": "string",
"timedMetadataBehavior (p. 438)": enum,
"etvSignalPid (p. 438)": "string",
"pmtPid (p. 438)": "string",
"bufferModel (p. 439)": enum,
"scte35Control (p. 439)": enum,
"ebpPlacement (p. 439)": enum,
"arib (p. 439)": enum,
"nullPacketBitrate (p. 439)": number,
"dvsdtSettings (p. 439)": {
  "serviceName (p. 393)": "string",
  "serviceProviderName (p. 394)": "string",
  "repInterval (p. 394)": integer,
  "outputSdt (p. 394)": enum
},
"pcrPid (p. 439)": "string",
"transportStreamId (p. 440)": integer,
"pcrControl (p. 440)": enum,
"videoPid (p. 440)": "string",
"esRateInPes (p. 440)": enum,
"segmentationMarkers (p. 440)": enum,
"dvbtSettings (p. 440)": {
  "repInterval (p. 398)": integer
},
"klv (p. 441)": enum,
"ccDescriptor (p. 441)": enum,
"patInterval (p. 441)": integer,
"etvPlatformPid (p. 441)": "string",
"dvbsubPids (p. 441)": "string",
"aribCaptionsPid (p. 441)": "string",
"scte27Pids (p. 441)": "string",
"klvDataPids (p. 442)": "string"
}
"m2tsSettings (p. 460)": {
  "audioStreamType (p. 434)": enum,
  "ecmPid (p. 435)": "string",
  "dvbTeletextPid (p. 435)": "string",
  "aribCaptionsPidControl (p. 435)": enum,
  "bitrate (p. 435)": integer,
  "segmentationTime (p. 435)": number,
  "rateMode (p. 435)": enum,
  "audioPids (p. 436)": "string",
  "fragmentTime (p. 436)": number,
  "ebpAudioInterval (p. 436)": enum,
  "ebpLookaheadMs (p. 436)": integer,
  "audioFramesPerPes (p. 436)": integer,
  "scte35Pid (p. 436)": "string",
  "pmtInterval (p. 437)": integer,
  "programNum (p. 437)": integer,
  "segmentationStyle (p. 437)": enum,
  "ebpPlacement (p. 437)": enum,
  "nullPacketBitrate (p. 439)": number,
  "dvbSdtSettings (p. 439)": {
    "serviceName (p. 393)": "string",
    "serviceProviderName (p. 394)": "string",
    "repInterval (p. 394)": integer,
    "outputSdt (p. 394)": enum
  },
  "pcrPid (p. 439)": "string",
  "transportStreamId (p. 440)": integer,
  "pcrControl (p. 440)": enum,
  "videoPid (p. 440)": "string",
  "esRateInPes (p. 440)": enum,
  "segmentationMarkers (p. 440)": enum,
  "dvbTdtSettings (p. 440)": {
    "repInterval (p. 398)": integer
  },
  "klv (p. 441)": enum,
  "ccDescriptor (p. 441)": enum,
  "patInterval (p. 441)": integer,
  "etvPlatformPid (p. 441)": "string",
  "dvbSubPids (p. 441)": "string",
  "aribCaptionsPid (p. 441)": "string",
  "scte27Pids (p. 441)": "string",
  "klvDataPids (p. 442)": "string"
},
"fecOutputSettings (p. 461)": {
  "rowLength (p. 405)": integer,
  "columnDepth (p. 405)": integer,
  "includeFec (p. 405)": enum
},
"dvbNitSettings (p. 438)": {
  "networkName (p. 393)": "string",
  "networkId (p. 393)": integer,
  "repInterval (p. 393)": integer
},
"absentInputAudioBehavior (p. 438)": enum,
"timedMetadataPid (p. 438)": "string",
"timedMetadataBehavior (p. 438)": enum,
"etvSignalPid (p. 439)": "string",
"scte35Control (p. 439)": enum,
"ebpPlacement (p. 439)": enum,
"arib (p. 439)": enum,
"bufferModel (p. 439)": enum,
"pmtPid (p. 438)": "string",
"bufferModel (p. 439)": enum,
"scte35Control (p. 439)": enum,
"ebpPlacement (p. 439)": enum,
"arib (p. 439)": enum,
"nullPacketBitrate (p. 439)": number,
"dvbSdtSettings (p. 439)": {
  "serviceName (p. 393)": "string",
  "serviceProviderName (p. 394)": "string",
  "repInterval (p. 394)": integer,
  "outputSdt (p. 394)": enum
},
"pcrPid (p. 439)": "string",
"transportStreamId (p. 440)": integer,
"pcrControl (p. 440)": enum,
"videoPid (p. 440)": "string",
"esRateInPes (p. 440)": enum,
"segmentationMarkers (p. 440)": enum,
"dvbTdtSettings (p. 440)": {
  "repInterval (p. 398)": integer
},
"klv (p. 441)": enum,
"ccDescriptor (p. 441)": enum,
"patInterval (p. 441)": integer,
"etvPlatformPid (p. 441)": "string",
"dvbSubPids (p. 441)": "string",
"aribCaptionsPid (p. 441)": "string",
"scte27Pids (p. 441)": "string",
"klvDataPids (p. 442)": "string"}
"hlsOutputSettings (p. 452)": {
    "segmentModifier (p. 425)": "string",
    "hlsSettings (p. 425)": {
        "audioOnlyHlsSettings (p. 426)": {
            "audioTrackType (p. 380)": enum,
            "audioGroupId (p. 380)": "string",
            "audioOnlyImage (p. 381)": {
                "passwordParam (p. 428)": "string",
                "uri (p. 428)": "string",
                "username (p. 429)": "string"
            }
        },
        "standardHlsSettings (p. 426)": {
            "m3u8Settings (p. 458)": {
                "pmtPid (p. 442)": "string",
                "ecmPid (p. 442)": "string",
                "scte35Behavior (p. 442)": enum,
                "pcrPid (p. 443)": "string",
                "audioPids (p. 443)": "string",
                "audioFramesPerPes (p. 443)": integer,
                "scte35Pid (p. 443)": "string",
                "transportStreamId (p. 443)": integer,
                "videoPid (p. 443)": "string",
                "pcrControl (p. 443)": enum,
                "pmtInterval (p. 444)": integer,
                "pcrPeriod (p. 444)": integer,
                "programNum (p. 444)": integer,
                "patInterval (p. 444)": integer,
                "timedMetadataPid (p. 444)": "string",
                "timedMetadataBehavior (p. 444)": enum
            },
            "audioRenditionSets (p. 458)": "string"
        }
    },
    "nameModifier (p. 425)": "string"
},
"audioDescriptionNames (p. 449)": [
    "string"
],
"outputGroupSettings (p. 451)": {
    "archiveGroupSettings (p. 451)": {
        "destination (p. 376)": {
            "destinationRefId (p. 451)": "string"
        },
        "rolloverInterval (p. 376)": integer
    },
    "rtmpGroupSettings (p. 451)": {
        "captionData (p. 453)": enum,
        "authenticationScheme (p. 453)": enum,
        "cacheLength (p. 453)": integer,
        "restartDelay (p. 454)": integer,
        "cacheFullBehavior (p. 454)": enum
    },
    "udpGroupSettings (p. 451)": {
        "inputLossAction (p. 460)": enum,
        "timedMetadataId3Frame (p. 460)": enum,
        "timedMetadataId3Period (p. 460)": integer
    },
    "msSmoothGroupSettings (p. 451)": {
        "eventId (p. 445)": "string",
        "fragmentLength (p. 445)": integer,
        "timestampOffset (p. 446)": "string",
        "segmentationMode (p. 446)": enum
"numRetries (p. 446)": integer,
"eventStopBehavior (p. 446)": enum,
"acquisitionPointId (p. 446)": "string",
"sparseTrackType (p. 446)": enum,
"timestampOffsetMode (p. 446)": enum,
"destination (p. 447)": {
    "destinationRefId (p. 451)": "string"
},
"audioOnlyTimecodeControl (p. 447)": enum,
"connectionRetryInterval (p. 447)": integer,
"filecacheDuration (p. 447)": integer,
"certificateMode (p. 447)": enum,
"inputLossAction (p. 447)": enum,
"sendDelayMs (p. 447)": integer,
"eventIdMode (p. 448)": enum,
"restartDelay (p. 448)": integer,
"streamManifestBehavior (p. 448)": enum,
"hlsGroupSettings (p. 451)": {
    "segmentsPerSubdirectory (p. 418)": integer,
    "ivInManifest (p. 418)": enum,
    "outputSelection (p. 418)": enum,
    "encryptionType (p. 418)": enum,
    "indexNSegments (p. 418)": integer,
    "destination (p. 418)": {
        "destinationRefId (p. 451)": "string"
    },
    "constantIv (p. 419)": "string",
    "timedMetadataId3Frame (p. 419)": enum,
    "baseUrlManifest (p. 419)": "string",
    "captionLanguageSetting (p. 419)": enum,
    "minSegmentLength (p. 419)": integer,
    "mode (p. 419)": enum,
    "ivSource (p. 420)": enum,
    "manifestCompression (p. 420)": enum,
    "keyProviderSettings (p. 420)": {
        "staticKeySettings (p. 432)": {
            "staticKeyValue (p. 458)": "string",
            "keyProviderServer (p. 458)": {
                "passwordParam (p. 428)": "string",
                "uri (p. 428)": "string",
                "username (p. 429)": "string"
            }
        }
    },
    "tsFileMode (p. 420)": enum,
    "manifestDurationFormat (p. 420)": enum,
    "streamInfResolution (p. 420)": enum,
    "timestampDeltaMilliseconds (p. 421)": integer,
    "segmentationMode (p. 421)": enum,
    "baseUrlContent (p. 421)": "string",
    "clientCache (p. 421)": enum,
    "captionLanguageMappings (p. 421)": [
        {
            "languageDescription (p. 389)": "string",
            "captionChannel (p. 390)": integer,
            "languageCode (p. 390)": "string"
        }
    ],
    "codecSpecification (p. 421)": enum,
    "keepSegments (p. 421)": integer,
    "timedMetadataId3Period (p. 422)": integer,
    "programDateTime (p. 422)": enum,
    "directoryStructure (p. 422)": enum,
    "keyFormat (p. 422)": "string"
"inputLossAction (p. 422)" : enum,
"adMarkers (p. 422)" : [ enum
],
"programDateTimePeriod (p. 422)" : integer,
"segmentLength (p. 423)" : integer,
"hlsCdnSettings (p. 423)" : {
  "hlsAkamaiSettings (p. 417)" : {
    "httpTransferMode (p. 415)" : enum,
    "salt (p. 415)" : "string",
    "numRetries (p. 415)" : integer,
    "restartDelay (p. 415)" : integer,
    "connectionRetryInterval (p. 416)" : integer,
    "filecacheDuration (p. 416)" : integer,
    "token (p. 416)" : "string"
  },
  "hlsWebdavSettings (p. 417)" : {
    "httpTransferMode (p. 426)" : enum,
    "numRetries (p. 426)" : integer,
    "restartDelay (p. 427)" : integer,
    "connectionRetryInterval (p. 427)" : integer,
    "filecacheDuration (p. 427)" : integer
  },
  "hlsBasicPutSettings (p. 417)" : {
    "numRetries (p. 416)" : integer,
    "restartDelay (p. 416)" : integer,
    "connectionRetryInterval (p. 416)" : integer,
    "filecacheDuration (p. 417)" : integer
  },
  "hlsMediaStoreSettings (p. 417)" : {
    "mediaStoreStorageClass (p. 424)" : enum,
    "numRetries (p. 424)" : integer,
    "restartDelay (p. 424)" : integer,
    "connectionRetryInterval (p. 424)" : integer,
    "filecacheDuration (p. 425)" : integer
  }
},
"name (p. 451)" : "string"
},
"audioDescriptions (p. 404)" : [
  {
    "languageCodeControl (p. 377)" : enum,
    "audioTypeControl (p. 378)" : enum,
    "remixSettings (p. 378)" : {
      "channelMappings (p. 452)" : [
        {
          "outputChannel (p. 376)" : integer,
          "inputChannelLevels (p. 377)" : [
            {
              "inputChannel (p. 427)" : integer,
              "gain (p. 428)" : integer
            }
          ]
        }
      ],
      "channelsOut (p. 452)" : integer,
      "channelsIn (p. 452)" : integer
    },
    "audioType (p. 378)" : enum,
    "name (p. 378)" : "string",
    "languageCode (p. 378)" : "string",
    "codecSettings (p. 378)" : [
      "aacSettings (p. 377)" : {
        "samplingRate (p. 376)" : integer,
        "mode (p. 376)" : enum,
        "channels (p. 376)" : integer
      }
    ]
  }
}
"vbrQuality (p. 372)": enum,
"codingMode (p. 372)": enum,
"profile (p. 372)": enum,
"bitrate (p. 372)": number,
"inputType (p. 373)": enum,
"rawFormat (p. 373)": enum,
"sampleRate (p. 373)": number,
"rateControlMode (p. 373)": enum,
"spec (p. 373)": enum
},
"ac3Settings (p. 377)": {
"drcProfile (p. 374)": enum,
"dialnorm (p. 374)": integer,
"codingMode (p. 374)": enum,
"metadataControl (p. 375)": enum,
"bitrate (p. 375)": number,
"lifeFilter (p. 375)": enum,
"bitstreamMode (p. 375)": enum
},
"eac3Settings (p. 377)": {
"dialnorm (p. 400)": integer,
"passthroughControl (p. 400)": enum,
"metadataControl (p. 400)": enum,
"drcLine (p. 400)": enum,
"bitrate (p. 400)": number,
"surroundEXMode (p. 400)": enum,
"lRtSurroundMixLevel (p. 401)": number,
"lifeControl (p. 401)": enum,
"codingMode (p. 401)": enum,
"surroundMode (p. 401)": enum,
"attenuationControl (p. 401)": enum,
"lifeFilter (p. 401)": enum,
"lRtCenterMixLevel (p. 401)": number,
"dcFilter (p. 402)": enum,
"phaseControl (p. 402)": enum,
"stereoDownmix (p. 402)": enum,
"bitstreamMode (p. 402)": enum,
"loRoSurroundMixLevel (p. 402)": number,
"drcRf (p. 402)": enum,
"loRoCenterMixLevel (p. 402)": number
},
"passThroughSettings (p. 377)": {
},
"mp2Settings (p. 377)": {
"codingMode (p. 445)": enum,
"bitrate (p. 445)": number,
"sampleRate (p. 445)": number
},
"streamName (p. 378)": "string",
"audioNormalizationSettings (p. 379)": {
"targetLkfs (p. 380)": number,
"algorithmControl (p. 380)": enum,
"algorithm (p. 380)": enum
},
"audioSelectorName (p. 379)": "string"
},
"captionDescriptions (p. 404)": [
{
"captionSelectorName (p. 387)": "string",
"languageDescription (p. 388)": "string",
"name (p. 388)": "string",
"languageCode (p. 388)": "string",
"destinationSettings (p. 388)": {
"scte27DestinationSettings (p. 388)": {
"burnInDestinationSettings (p. 388)": {
  "xPosition (p. 384)": integer,
  "backgroundColor (p. 384)": enum,
  "yPosition (p. 384)": integer,
  "teletextGridControl (p. 385)": enum,
  "backgroundOpacity (p. 385)": integer,
  "fontOpacity (p. 385)": integer,
  "fontResolution (p. 385)": integer,
  "shadowOpacity (p. 385)": integer,
  "shadowYOffset (p. 385)": integer,
  "outlineSize (p. 386)": integer,
  "outlineColor (p. 386)": enum,
  "fontSize (p. 386)": "string",
  "shadowXOffset (p. 386)": integer,
  "alignment (p. 386)": enum,
  "shadowColor (p. 386)": enum,
  "fontColor (p. 387)": enum,
  "font (p. 387)": {
    "passwordParam (p. 428)": "string",
    "uri (p. 428)": "string",
    "username (p. 429)": "string"
  }
},
"teletextDestinationSettings (p. 388)": {
},
"smpteTtDestinationSettings (p. 388)": {
},
"webvttDestinationSettings (p. 389)": {
},
"ttmlDestinationSettings (p. 389)": {
  "styleControl (p. 459)": enum
},
"embeddedPlusScte20DestinationSettings (p. 389)": {
},
"dvbSubDestinationSettings (p. 389)": {
  "xPosition (p. 395)": integer,
  "backgroundColor (p. 395)": enum,
  "yPosition (p. 395)": integer,
  "teletextGridControl (p. 395)": enum,
  "backgroundOpacity (p. 395)": integer,
  "fontOpacity (p. 396)": integer,
  "fontResolution (p. 396)": integer,
  "shadowOpacity (p. 396)": integer,
  "shadowYOffset (p. 396)": integer,
  "outlineSize (p. 396)": integer,
  "outlineColor (p. 397)": enum,
  "fontSize (p. 397)": "string",
  "shadowXOffset (p. 397)": integer,
  "alignment (p. 397)": enum,
  "shadowColor (p. 397)": enum,
  "fontColor (p. 397)": enum,
  "font (p. 398)": {
    "passwordParam (p. 428)": "string",
    "uri (p. 428)": "string",
    "username (p. 429)": "string"
  }
},
"embeddedDestinationSettings (p. 389)": {
},
"rtmpCaptionInfoDestinationSettings (p. 389)": {
},
"aribDestinationSettings (p. 389)": {
},
"scte20PlusEmbeddedDestinationSettings (p. 389)": {
}
"availConfiguration (p. 404)": {
  "availSettings (p. 382)": {
    "scte35TimeSignalApos (p. 382)": {
      "adAvailOffset (p. 457)": integer,
      "webDeliveryAllowedFlag (p. 457)": enum,
      "noRegionalBlackoutFlag (p. 457)": enum
    },
    "scte35SpliceInsert (p. 383)": {
      "adAvailOffset (p. 456)": integer,
      "webDeliveryAllowedFlag (p. 456)": enum,
      "noRegionalBlackoutFlag (p. 456)": enum
    }
  },
  "globalConfiguration (p. 404)": {
    "inputLossBehavior (p. 406)": {
      "inputLossImageType (p. 429)": enum,
      "inputLossImageColor (p. 429)": "string",
      "inputLossImageSlate (p. 429)": {
        "passwordParam (p. 428)": "string",
        "uri (p. 428)": "string",
        "username (p. 429)": "string"
      },
      "blackFrameMsec (p. 430)": integer,
      "repeatFrameMsec (p. 430)": integer
    },
    "supportLowFramerateInputs (p. 406)": enum,
    "initialAudioGain (p. 406)": integer,
    "inputEndAction (p. 406)": enum,
    "outputTimingSource (p. 407)": enum
  },
  "videoDescriptions (p. 405)": [
    {
      "respondToAfd (p. 461)": enum,
      "scalingBehavior (p. 461)": enum,
      "name (p. 462)": "string",
      "width (p. 462)": integer,
      "sharpness (p. 462)": integer,
      "codecSettings (p. 462)": {
        "h264Settings (p. 462)": {
          "minIInterval (p. 409)": integer,
          "slices (p. 409)": integer,
          "parNumerator (p. 409)": integer,
          "gopSizeUnits (p. 410)": enum,
          "maxBitrate (p. 410)": integer,
          "bitrate (p. 410)": integer,
          "bufFillPct (p. 410)": integer,
          "temporalAq (p. 410)": enum,
          "afdSignaling (p. 410)": enum,
          "timecodeInsertion (p. 411)": enum,
          "bufSize (p. 411)": integer,
          "softness (p. 411)": integer,
          "framerateControl (p. 411)": enum,
          "fixedAfd (p. 411)": enum,
          "level (p. 411)": enum,
          "lookAheadRateControl (p. 411)": enum,
          "profile (p. 412)": enum,
          "framerateNumerator (p. 412)": integer,
          "gopClosedCadence (p. 412)": integer,
          "framerateDenominator (p. 412)": integer,
          "entropyEncoding (p. 412)": enum,
          "spatialAq (p. 412)": enum,
          "adaptiveQuantization (p. 412)": enum,
        }
      }
    }
  ]
"colorMetadata (p. 413)": enum,
"gopSize (p. 413)": number,
"numRefFrames (p. 413)": integer,
"gobReference (p. 413)": enum,
"sceneChangeDetect (p. 413)": enum,
"parControl (p. 413)": enum,
"parDenominator (p. 413)": integer,
"syntax (p. 414)": enum,
"scanType (p. 414)": integer,
"gopNumBFrames (p. 414)": integer,
"flickerAq (p. 414)": enum,
"rateControlMode (p. 414)": enum
}
"height (p. 462)": integer
"
"blackoutSlate (p. 405)": {
  "networkEndBlackoutImage (p. 383)": {
    "passwordParam (p. 428)": "string",
    "uri (p. 428)": "string",
    "username (p. 429)": "string"
  },
  "networkEndBlackout (p. 383)": enum,
  "networkId (p. 383)": "string",
  "state (p. 383)": enum,
  "blackoutSlateImage (p. 383)": {
    "passwordParam (p. 428)": "string",
    "uri (p. 428)": "string",
    "username (p. 429)": "string"
  }
},
"availBlanking (p. 405)": {
  "state (p. 382)": enum,
  "availBlankingImage (p. 382)": {
    "passwordParam (p. 428)": "string",
    "uri (p. 428)": "string",
    "username (p. 429)": "string"
  }
},
"pipelinesRunningCount (p. 392)": integer,
"state (p. 392)": enum,
"id (p. 392)": "string",
"egressEndpoints (p. 392)": [
  "sourceIp (p. 392)": "string"
],
"arn (p. 392)": "string",
"inputSpecification (p. 392)": {
  "codec (p. 432)": enum,
  "resolution (p. 432)": enum,
  "maximumBitrate (p. 432)": enum
}

Example InvalidRequest

{
  "message (p. 432)": "string"
}
Example AccessDenied

```json
{
  "message (p. 375)": "string"
}
```

Example ResourceNotFound

```json
{
  "message (p. 453)": "string"
}
```

Example ResourceConflict

```json
{
  "message (p. 453)": "string"
}
```

Example LimitExceeded

```json
{
  "message (p. 433)": "string"
}
```

Example InternalServiceError

```json
{
  "message (p. 432)": "string"
}
```

Example BadGatewayException

```json
{
  "message (p. 383)": "string"
}
```

Example GatewayTimeoutException

```json
{
  "message (p. 406)": "string"
}
```

Properties

AacCodingMode (enum)

- AD_RECEIVER_MIX
- CODING_MODE_1_0
- CODING_MODE_1_1
- CODING_MODE_2_0
- CODING_MODE_5_1
**AacInputType (enum)**

- BROADCASTER_MIXED_AD
- NORMAL

**AacProfile (enum)**

- HEV1
- HEV2
- LC

**AacRateControlMode (enum)**

- CBR
- VBR

**AacRawFormat (enum)**

- LATM_LOAS
- NONE

**AacSettings**

**vbrQuality**

VBR Quality Level - Only used if rateControlMode is VBR.

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

**codingMode**

Mono, Stereo, or 5.1 channel layout. Valid values depend on rate control mode and profile. The adReceiverMix 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:** string
- **Required:** False

**profile**

AAC Profile.

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

**bitrate**

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

- **Type:** number
- **Required:** False
**inputType**
Set to "broadcasterMixedAd" when input contains pre-mixed main audio + AD (narration) as a stereo pair. The Audio Type field (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. The values in audioTypeControl and audioType (in AudioDescription) are ignored when set to broadcasterMixedAd. Leave set to "normal" when input does not contain pre-mixed audio + AD.

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

**rawFormat**
Sets LATM / LOAS AAC output for raw containers.

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

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

  **Type:** number  
  **Required:** False

**rateControlMode**
Rate Control Mode.

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

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

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

**AacSpec (enum)**

  - MPEG2  
  - MPEG4

**AacVbrQuality (enum)**

  - HIGH  
  - LOW  
  - MEDIUM_HIGH  
  - MEDIUM_LOW

**Ac3BitstreamMode (enum)**

  COMMENTARY
COMPLETE_MAIN
DIALOGUE
EMERGENCY
HEARING_IMPAIRED
MUSIC_AND_EFFECTS
VISUALLY_IMPAIRED
VOICE_OVER

**Ac3CodingMode (enum)**

- CODING_MODE_1_0
- CODING_MODE_1_1
- CODING_MODE_2_0
- CODING_MODE_3_2_LFE

**Ac3DrcProfile (enum)**

- FILM_STANDARD
- NONE

**Ac3LfeFilter (enum)**

- DISABLED
- ENABLED

**Ac3MetadataControl (enum)**

- FOLLOW_INPUT
- USE_CONFIGURED

**Ac3Settings**

**drcProfile**

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

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

**dialnorm**

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

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

**codingMode**

Dolby Digital coding mode. Determines number of channels.
Properties

metadataControl

When set to "followInput", 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: string
Required: False

bitrate

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

Type: number
Required: False

lfeFilter

When set to enabled, applies a 120Hz lowpass filter to the LFE channel prior to encoding. Only valid in codingMode32Lfe mode.

Type: string
Required: False

bitstreamMode

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

Type: string
Required: False

AccessDenied

message

Type: string
Required: False

AfdSignaling (enum)

AUTO
FIXED
NONE

ArchiveContainerSettings

m2tsSettings

Type: M2tsSettings (p. 434)
Required: False

**ArchiveGroupSettings**

destination

A directory and base filename where archive files should be written. If the base filename portion of the URI is left blank, the base filename of the first input will be automatically inserted.

- **Type**: OutputLocationRef (p. 451)
- **Required**: True

rolloverInterval

Number of seconds to write to archive file before closing and starting a new one.

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

**ArchiveOutputSettings**

extension

Output file extension. If excluded, this will be auto-selected from the container type.

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

containerSettings

Settings specific to the container type of the file.

- **Type**: ArchiveContainerSettings (p. 375)
- **Required**: True

nameModifier

String concatenated to the end of the destination filename. Required for multiple outputs of the same type.

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

**AribDestinationSettings**

**AribSourceSettings**

**AudioChannelMapping**

outputChannel

The index of the output channel being produced.
**inputChannelLevels**

Indices and gain values for each input channel that should be remixed into this output channel.

- **Type**: Array of type `InputChannelLevel` (p. 427)
- **Required**: True

**AudioCodecSettings**

- **aacSettings**
  - **Type**: `AacSettings` (p. 372)
  - **Required**: False

- **ac3Settings**
  - **Type**: `Ac3Settings` (p. 374)
  - **Required**: False

- **eac3Settings**
  - **Type**: `Eac3Settings` (p. 400)
  - **Required**: False

- **passThroughSettings**
  - **Type**: `PassThroughSettings` (p. 452)
  - **Required**: False

- **mp2Settings**
  - **Type**: `Mp2Settings` (p. 445)
  - **Required**: False

**AudioDescription**

**languageCodeControl**

Choosing followInput will cause the ISO 639 language code of the output to follow the ISO 639 language code of the input. The languageCode will be used when useConfigured is set, or when followInput is selected but there is no ISO 639 language code specified by the input.

- **Type**: string
- **Required**: False
### audioTypeControl
Determines how audio type is determined. followInput: If the input contains an ISO 639 audioType, then that value is passed through to the output. If the input contains no ISO 639 audioType, the value in Audio Type is included in the output. useConfigured: The value in Audio Type is included in the output. Note that this field and audioType are both ignored if inputType is broadcasterMixedAd.

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

### remixSettings
Settings that control how input audio channels are remixed into the output audio channels.

**Type:** RemixSettings (p. 452)  
**Required:** False

### audioType
Applies only if audioTypeControl is useConfigured. The values for audioType are defined in ISO-IEC 13818-1.

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

### name
The name of this AudioDescription. Outputs will use this name to uniquely identify this AudioDescription. Description names should be unique within this Live Event.

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

### languageCode
Indicates the language of the audio output track. Only used if languageControlMode is useConfigured, or there is no ISO 639 language code specified in the input.

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

### codecSettings
Audio codec settings.

**Type:** AudioCodecSettings (p. 377)  
**Required:** False

### streamName
Used for MS Smooth and Apple HLS outputs. Indicates the name displayed by the player (eg. English, or Director Commentary).

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

Advanced audio normalization settings.

*Type:* AudioNormalizationSettings (p. 380)
*Required:* False

**audioSelectorName**

The name of the AudioSelector used as the source for this AudioDescription.

*Type:* string
*Required:* True

**AudioDescriptionAudioTypeControl (enum)**

FOLLOW_INPUT
USE_CONFIGURED

**AudioDescriptionLanguageCodeControl (enum)**

FOLLOW_INPUT
USE_CONFIGURED

**AudioLanguageSelection**

**languageSelectionPolicy**

When set to "strict", the transport stream demux strictly identifies audio streams by their language descriptor. If a PMT update occurs such that an audio stream matching the initially selected language is no longer present then mute will be encoded until the language returns. If "loose", then on a PMT update the demux will choose another audio stream in the program with the same stream type if it can't find one with the same language.

*Type:* string
*Required:* False

**languageCode**

Selects a specific three-letter language code from within an audio source.

*Type:* string
*Required:* True

**AudioLanguageSelectionPolicy (enum)**

LOOSE
STRICT

**AudioNormalizationAlgorithm (enum)**

ITU_1770_1
ITU_1770_2
AudioNormalizationAlgorithmControl (enum)

- CORRECT_AUDIO

AudioNormalizationSettings

targetLkfs

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

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

algorithmControl

When set to correctAudio the output audio is corrected using the chosen algorithm. If set to measureOnly, the audio will be measured but not adjusted.

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

algorithm

Audio normalization algorithm to use. itu17701 conforms to the CALM Act specification, itu17702 conforms to the EBU R-128 specification.

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

AudioOnlyHlsSettings

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

audioGroupId

Specifies the group to which the audio Rendition belongs.

- **Type**: string
Properties

Required: False

**audioOnlyImage**

For use with an audio only Stream. Must be a .jpg or .png file. If given, this image will be used as the cover-art for the audio only output. Ideally, it should be formatted for an iPhone screen for two reasons. The iPhone does not resize the image, it crops a centered image on the top/bottom and left/right. Additionally, this image file gets saved bit-for-bit into every 10-second segment file, so will increase bandwidth by {image file size} * {segment count} * {user count}.

  * **Type:** InputLocation (p. 428)
  * **Required:** False

**AudioOnlyHlsTrackType (enum)**

- ALTERNATE_AUDIO_AUTO_SELECT
- ALTERNATE_AUDIO_AUTO_SELECT_DEFAULT
- ALTERNATE_AUDIO_NOT_AUTO_SELECT
- AUDIO_ONLY_VARIANT_STREAM

**AudioPidSelection**

**pid**

Selects a specific PID from within a source.

  * **Type:** integer
  * **Required:** True
  * **Minimum:** 0
  * **Maximum:** 8191

**AudioSelector**

**name**

The name of this AudioSelector. AudioDescriptions will use this name to uniquely identify this Selector. Selector names should be unique per input.

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

**selectorSettings**

The audio selector settings.

  * **Type:** AudioSelectorSettings (p. 381)
  * **Required:** False

**AudioSelectorSettings**

**audioLanguageSelection**

  * **Type:** AudioLanguageSelection (p. 379)
  * **Required:** False
audioPidSelection
Type: AudioPidSelection (p. 381)
Required: False

AudioType (enum)
CLEAN_EFFECTS
HEARING_IMPAIRED
UNDEFINED
VISUAL_IMPAIREDCOMMENTARY

AuthenticationScheme (enum)
AKAMAI
COMMON

AvailBlanking
state
When set to enabled, causes video, audio and captions to be blanked when insertion metadata is added.
Type: string
Required: False

availBlankingImage
Blanking image to be used. Leave empty for solid black. Only bmp and png images are supported.
Type: InputLocation (p. 428)
Required: False

AvailBlankingState (enum)
DISABLED
ENABLED

AvailConfiguration
availSettings
Ad avail settings.
Type: AvailSettings (p. 382)
Required: False

AvailSettings
scte35TimeSignalApos
Type: Scte35TimeSignalApos (p. 457)
Required: False
scte35SpliceInsert

Type: Scte35SpliceInsert (p. 456)
Required: False

BadGatewayException

message

Type: string
Required: False

BlackoutSlate

networkEndBlackoutImage

Path to local file to use as Network End Blackout image. Image will be scaled to fill the entire output raster.

Type: InputLocation (p. 428)
Required: False

networkEndBlackout

Setting to enabled causes the encoder to blackout the video, audio, and captions, and raise the "Network Blackout Image" slate when an SCTE104/35 Network End Segmentation Descriptor is encountered. The blackout will be lifted when the Network Start Segmentation Descriptor is encountered. The Network End and Network Start descriptors must contain a network ID that matches the value entered in "Network ID".

Type: string
Required: False

networkId

Provides Network ID that matches EIDR ID format (e.g., "10.XXXX/XXXX-XXXX-XXXX-XXXX-XXXX-XXXX-C").

Type: string
Required: False

state

When set to enabled, causes video, audio and captions to be blanked when indicated by program metadata.

Type: string
Required: False

blackoutSlateImage

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

Type: InputLocation (p. 428)
Required: False
**BlackoutSlateNetworkEndBlackout (enum)**

- DISABLED
- ENABLED

**BlackoutSlateState (enum)**

- DISABLED
- ENABLED

**BurnInAlignment (enum)**

- CENTERED
- LEFT
- SMART

**BurnInBackgroundColor (enum)**

- BLACK
- NONE
- WHITE

**BurnInDestinationSettings**

**xPosition**

Specifies the horizontal position of the caption relative to the left side of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the left of the output. If no explicit xPosition is provided, the horizontal caption position will be determined by the alignment parameter. All burn-in and DVB-Sub font settings must match.

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

**groundColor**

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

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

**yPosition**

Specifies the vertical position of the caption relative to the top of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the top of the output. If no explicit yPosition is provided, the caption will be positioned towards the bottom of the output. All burn-in and DVB-Sub font settings must match.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
teletextGridControl

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

Type: string
Required: False

backgroundOpacity

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

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

fontOpacity

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

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

fontResolution

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

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

shadowOpacity

Specifies the opacity of the shadow. 255 is opaque; 0 is transparent. Leaving this parameter out 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

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
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

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: string
Required: False

fontSize

When set to 'auto' fontSize will scale depending on the size of the output. Giving a positive integer will specify the exact font size in points. All burn-in and DVB-Sub font settings must match.

Type: string
Required: False

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

alignment

If no explicit xPosition or yPosition 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. Selecting "smart" justification will left-justify live subtitles and center-justify pre-recorded subtitles. All burn-in and DVB-Sub font settings must match.

Type: string
Required: False

shadowColor

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

Type: string
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:* string  
*Required:* False

**font**

External font file used for caption burn-in. File extension must be 'ttf' or 'tte'. Although the user can select output fonts for many different types of input captions, embedded, STL and teletext sources use a strict grid system. Using external fonts with these caption sources could cause unexpected display of proportional fonts. All burn-in and DVB-Sub font settings must match.

*Type:* InputLocation (p. 428)  
*Required:* False

**BurnInFontColor (enum)**

BLACK  
BLUE  
GREEN  
RED  
WHITE  
YELLOW

**BurnInOutlineColor (enum)**

BLACK  
BLUE  
GREEN  
RED  
WHITE  
YELLOW

**BurnInShadowColor (enum)**

BLACK  
NONE  
WHITE

**BurnInTeletextGridControl (enum)**

FIXED  
Scaled

**CaptionDescription**

**captionSelectorName**

Specifies which input caption selector to use as a caption source when generating output captions. This field should match a captionSelector name.
Properties

**languageDescription**

Human readable information to indicate captions available for players (e.g. English, or Spanish).

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

**name**

Name of the caption description. Used to associate a caption description with an output. Names must be unique within an event.

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

**languageCode**


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

**destinationSettings**

Additional settings for captions destination that depend on the destination type.

- **Type**: CaptionDestinationSettings (p. 388)
- **Required**: False

**CaptionDestinationSettings**

**scte27DestinationSettings**

- **Type**: Scte27DestinationSettings (p. 455)
- **Required**: False

**burnInDestinationSettings**

- **Type**: BurnInDestinationSettings (p. 384)
- **Required**: False

**teletextDestinationSettings**

- **Type**: TeletextDestinationSettings (p. 459)
- **Required**: False

**smpteTtDestinationSettings**

- **Type**: SmpteTtDestinationSettings (p. 458)
**Required**: False

**webvttDestinationSettings**

*Type*: WebvttDestinationSettings (p. 464)

*Required*: False

**ttmlDestinationSettings**

*Type*: TtmlDestinationSettings (p. 459)

*Required*: False

**embeddedPlusScte20DestinationSettings**

*Type*: EmbeddedPlusScte20DestinationSettings (p. 403)

*Required*: False

**dvbSubDestinationSettings**

*Type*: DvbSubDestinationSettings (p. 395)

*Required*: False

**embeddedDestinationSettings**

*Type*: EmbeddedDestinationSettings (p. 403)

*Required*: False

**rtmpCaptionInfoDestinationSettings**

*Type*: RtmpCaptionInfoDestinationSettings (p. 453)

*Required*: False

**aribDestinationSettings**

*Type*: AribDestinationSettings (p. 376)

*Required*: False

**scte20PlusEmbeddedDestinationSettings**

*Type*: Scte20PlusEmbeddedDestinationSettings (p. 455)

*Required*: False

**CaptionLanguageMapping**

**languageDescription**

Textual description of language

*Type*: string

*Required*: True
captionChannel
Channel to insert closed captions. Each channel mapping must have a unique channel number (maximum of 4)

Type: integer
Required: True
Minimum: 1
Maximum: 4

languageCode
Three character ISO 639-2 language code (see http://www.loc.gov/standards/iso639-2)

Type: string
Required: True

CaptionSelector

name
Name identifier for a caption selector. This name is used to associate this caption selector with one or more caption descriptions. Names must be unique within an event.

Type: string
Required: True

languageCode
When specified this field indicates the three letter language code of the caption track to extract from the source.

Type: string
Required: False

selectorSettings
Caption selector settings.

Type: CaptionSelectorSettings (p. 390)
Required: False

CaptionSelectorSettings

embeddedSourceSettings

Type: EmbeddedSourceSettings (p. 403)
Required: False

scte20SourceSettings

Type: Scte20SourceSettings (p. 455)
Required: False

dvbSubSourceSettings
  Type: DvbSubSourceSettings (p. 398)
  Required: False

aribSourceSettings
  Type: AribSourceSettings (p. 376)
  Required: False

teletextSourceSettings
  Type: TeletextSourceSettings (p. 459)
  Required: False

scte27SourceSettings
  Type: Scte27SourceSettings (p. 455)
  Required: False

Channel

inputAttachments
List of input attachments for channel.
  Type: Array of type InputAttachment (p. 427)
  Required: False

roleArn
The Amazon Resource Name (ARN) of the role assumed when running the Channel.
  Type: string
  Required: False

destinations
A list of destinations of the channel. For UDP outputs, there is one destination per output. For other types (HLS, for example), there is one destination per packager.
  Type: Array of type OutputDestination (p. 450)
  Required: False

name
The name of the channel. (user-mutable)
  Type: string
Required: False

**encoderSettings**

Type: EncoderSettings (p. 404)

Required: False

**pipelinesRunningCount**

The number of currently healthy pipelines.

Type: integer

Required: False

**state**

Type: string

Required: False

**id**

The unique id of the channel.

Type: string

Required: False

**egressEndpoints**

The endpoints where outgoing connections initiate from

Type: Array of type ChannelEgressEndpoint (p. 392)

Required: False

**arn**

The unique arn of the channel.

Type: string

Required: False

**inputSpecification**

Type: InputSpecification (p. 432)

Required: False

**ChannelEgressEndpoint**

**sourceIp**

Public IP of where a channel's output comes from

Type: string

Required: False
Properties

ChannelState (enum)

CREATING
CREATE_FAILED
IDLE
STARTING
RUNNING
RECOVERING
STOPPING
DELETING
DELETED

DvbNitSettings

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

  Type: string
  Required: True

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

  Type: integer
  Required: True
  Minimum: 0
  Maximum: 65536

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

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

DvbSdtOutputSdt (enum)

SDT_FOLLOW
SDT_FOLLOW_IF_PRESENT
SDT_MANUAL
SDT_NONE

DvbSdtSettings

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

  Type: string
**Required**: False

**serviceProviderName**

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

*Type*: string  
*Required*: False

**repInterval**

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

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

**outputSdt**

Selects method of inserting SDT information into output stream. The sdtFollow setting copies SDT information from input stream to output stream. The sdtFollowIfPresent setting 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. The sdtManual setting means user will enter the SDT information. The sdtNone setting means output stream will not contain SDT information.

*Type*: string  
*Required*: False

**DvbSubDestinationAlignment (enum)**

CENTERED  
LEFT  
SMART

**DvbSubDestinationBackgroundColor (enum)**

BLACK  
NONE  
WHITE

**DvbSubDestinationFontColor (enum)**

BLACK  
BLUE  
GREEN  
RED  
WHITE  
YELLOW

**DvbSubDestinationOutlineColor (enum)**

BLACK
DvbSubDestinationSettings

xPosition

Specifies the horizontal position of the caption relative to the left side of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the left of the output. If no explicit xPosition 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

backgroundColor

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

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

yPosition

Specifies the vertical position of the caption relative to the top of the output in pixels. A value of 10 would result in the captions starting 10 pixels from the top of the output. If no explicit yPosition 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

teletextGridControl

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

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

backgroundOpacity

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

**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

**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

**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

**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

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

**fontSize**

When set to auto fontSize will scale depending on the size of the output. Giving a positive integer will specify the exact font size in points. All burn-in and DVB-Sub font settings must match.

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

**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

**alignment**

If no explicit xPosition or yPosition 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. Selecting "smart" justification will left-justify live subtitles and center-justify pre-recorded subtitles. This option is not valid for source captions that are STL or 608/embedded. These source settings are already pre-defined by the caption stream. All burn-in and DVB-Sub font settings must match.

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

**shadowColor**

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

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

External font file used for caption burn-in. File extension must be 'ttf' or 'tte'. Although the user can select output fonts for many different types of input captions, embedded, STL and teletext sources use a strict grid system. Using external fonts with these caption sources could cause unexpected display of proportional fonts. All burn-in and DVB-Sub font settings must match.

**Type**: InputLocation (p. 428)
**Required**: False

DvbSubDestinationShadowColor (enum)

- BLACK
- NONE
- WHITE

DvbSubDestinationTeletextGridControl (enum)

- FIXED
- SCALED

DvbSubSourceSettings

pid

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

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

DvbTdtSettings

repInterval

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

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

Eac3AttenuationControl (enum)

- ATTENUATE_3_DB
- NONE

Eac3BitstreamMode (enum)

- COMMENTARY
- COMPLETE_MAIN
EMERGENCY
HEARING_IMPAIRED
VISUALLY_IMPAIRED

Eac3CodingMode (enum)

CODING_MODE_1_0
CODING_MODE_2_0
CODING_MODE_3_2

Eac3DcFilter (enum)

DISABLED
ENABLED

Eac3DrcLine (enum)

FILM_LIGHT
FILM_STANDARD
MUSIC_LIGHT
MUSIC_STANDARD
NONE
SPEECH

Eac3DrcRf (enum)

FILM_LIGHT
FILM_STANDARD
MUSIC_LIGHT
MUSIC_STANDARD
NONE
SPEECH

Eac3LfeControl (enum)

LFE
NO_LFE

Eac3LfeFilter (enum)

DISABLED
ENABLED

Eac3MetadataControl (enum)

FOLLOW_INPUT
USE_CONFIGURED

Eac3PassthroughControl (enum)

NO_PASSTHROUGH
WHEN_POSSIBLE
Eac3PhaseControl (enum)

- NO_SHIFT
- SHIFT_90_DEGREES

Eac3Settings

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

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

passthroughControl
When set to whenPossible, 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: string
Required: False

metadataControl
When set to followInput, 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: string
Required: False

drcLine
Sets the Dolby dynamic range compression profile.

Type: string
Required: False

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

Type: number
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.
**Properties**

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

**lRtSurroundMixLevel**

Left total/Right total surround mix level. Only used for 3/2 coding mode.

**Type**: number  
**Required**: False

**lfeControl**

When encoding 3/2 audio, setting to lfe enables the LFE channel

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

**codingMode**

Dolby Digital Plus coding mode. Determines number of channels.

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

**surroundMode**

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

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

**attenuationControl**

When set to attenuate3Db, applies a 3 dB attenuation to the surround channels. Only used for 3/2 coding mode.

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

**lfeFilter**

When set to enabled, applies a 120Hz lowpass filter to the LFE channel prior to encoding. Only valid with codingMode32 coding mode.

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

**lRtCenterMixLevel**

Left total/Right total center mix level. Only used for 3/2 coding mode.

**Type**: number  
**Required**: False
**dcFilter**
When set to enabled, activates a DC highpass filter for all input channels.

*Type*: string  
*Required*: False

**phaseControl**
When set to shift90Degrees, applies a 90-degree phase shift to the surround channels. Only used for 3/2 coding mode.

*Type*: string  
*Required*: False

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

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

**loRoSurroundMixLevel**
Left only/Right only surround mix level. Only used for 3/2 coding mode.

*Type*: number  
*Required*: False

**drcRf**
Sets the profile for heavy Dolby dynamic range compression, ensures that the instantaneous signal peaks do not exceed specified levels.

*Type*: string  
*Required*: False

**loRoCenterMixLevel**
Left only/Right only center mix level. Only used for 3/2 coding mode.

*Type*: number  
*Required*: False

**Eac3StereoDownmix (enum)**

DPL2
Eac3SurroundExMode (enum)

- DISABLED
- ENABLED
- NOT_INDICATED

Eac3SurroundMode (enum)

- DISABLED
- ENABLED
- NOT_INDICATED

EmbeddedConvert608To708 (enum)

- DISABLED
- UPCONVERT

EmbeddedDestinationSettings

EmbeddedPlusScte20DestinationSettings

EmbeddedScte20Detection (enum)

- AUTO
- OFF

EmbeddedSourceSettings

scte20Detection

Set to "auto" to handle streams with intermittent and/or non-aligned SCTE-20 and Embedded captions.

- Type: string
  - Required: False

source608ChannelNumber

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

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

convert608To708

If 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.
source608TrackNumber
This field is unused and deprecated.

EncoderSettings

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

outputGroups

audioDescriptions

captionDescriptions
Settings for caption descriptions

availConfiguration
Event-wide configuration settings for ad avail insertion.

globalConfiguration
Configuration settings that apply to the event as a whole.
videoDescriptions

Type: Array of type VideoDescription (p. 461)
Required: True

blackoutSlate

Settings for blackout slate.

Type: BlackoutSlate (p. 383)
Required: False

availBlanking

Settings for ad avail blanking.

Type: AvailBlanking (p. 382)
Required: False

FecOutputIncludeFec (enum)

COLUMN
COLUMN_AND_ROW

FecOutputSettings

rowLength

Parameter L from SMPTE 2022-1. The width of the FEC protection matrix. Must be between 1 and 20, inclusive. If only Column FEC is used, then larger values increase robustness. If Row FEC is used, then this is the number of transport stream packets per row error correction packet, and the value must be between 4 and 20, inclusive, if includeFec is columnAndRow. If includeFec is column, this value must be 1 to 20, inclusive.

Type: integer
Required: False
Minimum: 1
Maximum: 20

columnDepth

Parameter D from SMPTE 2022-1. The height of the FEC protection matrix. The number of transport stream packets per column error correction packet. Must be between 4 and 20, inclusive.

Type: integer
Required: False
Minimum: 4
Maximum: 20

includeFec

Enables column only or column and row based FEC

Type: string
**Required**: False

**FixedAfd (enum)**

- AFD_0000
- AFD_0010
- AFD_0011
- AFD_0100
- AFD_1000
- AFD_1001
- AFD_1010
- AFD_1011
- AFD_1101
- AFD_1110
- AFD_1111

**GatewayTimeoutException**

**message**

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

**GlobalConfiguration**

**inputLossBehavior**

Settings for system actions when input is lost.

- **Type**: `InputLossBehavior (p. 429)`
- **Required**: False

**supportLowFramerateInputs**

Adjusts video input buffer for streams with very low video framerates. This is commonly set to enabled for music channels with less than one video frame per second.

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

**initialAudioGain**

Value to set the initial audio gain for the Live Event.

- **Type**: integer
- **Required**: False
- **Minimum**: -60
- **Maximum**: 60

**inputEndAction**

Indicates the action to take when an input completes (e.g. end-of-file.) Options include immediately switching to the next sequential input (via "switchInput"), switching to the next input and looping...
Properties

back to the first input when last input ends (via "switchAndLoopInputs") or not switching inputs and instead transcoding black / color / slate images per the "Input Loss Behavior" configuration until an activateInput REST command is received (via "none").

Type: string
Required: False

outputTimingSource

Indicates whether the rate of frames emitted by the Live encoder should be paced by its system clock (which optionally may be locked to another source via NTP) or should be locked to the clock of the source that is providing the input stream.

Type: string
Required: False

GlobalConfigurationInputEndAction (enum)

NONE
SWITCH_AND_LOOP_INPUTS

GlobalConfigurationLowFramerateInputs (enum)

DISABLED
ENABLED

GlobalConfigurationOutputTimingSource (enum)

INPUT_CLOCK
SYSTEM_CLOCK

H264AdaptiveQuantization (enum)

HIGH
HIGHER
LOW
MAX
MEDIUM
OFF

H264ColorMetadata (enum)

IGNORE
INSERT

H264EntropyEncoding (enum)

CABAC
CAVLC

H264FlickerAq (enum)

DISABLED
ENABLED

**H264FramerateControl (enum)**

- INITIALIZE_FROM_SOURCE
- SPECIFIED

**H264GopBReference (enum)**

- DISABLED
- ENABLED

**H264GopSizeUnits (enum)**

- FRAMES
- SECONDS

**H264Level (enum)**

- H264_LEVEL_1
- H264_LEVEL_1_1
- H264_LEVEL_1_2
- H264_LEVEL_1_3
- H264_LEVEL_2
- H264_LEVEL_2_1
- H264_LEVEL_2_2
- H264_LEVEL_3
- H264_LEVEL_3_1
- H264_LEVEL_3_2
- H264_LEVEL_4
- H264_LEVEL_4_1
- H264_LEVEL_4_2
- H264_LEVEL_5
- H264_LEVEL_5_1
- H264_LEVEL_5_2
- H264_LEVEL_AUTO

**H264LookAheadRateControl (enum)**

- HIGH
- LOW
- MEDIUM

**H264ParControl (enum)**

- INITIALIZE_FROM_SOURCE
- SPECIFIED

**H264Profile (enum)**

- BASELINE
HIGH
HIGH_10BIT
HIGH_422
HIGH_422_10BIT
MAIN

**H264RateControlMode (enum)**

CBR
VBR

**H264ScanType (enum)**

INTERLACED
PROGRESSIVE

**H264SceneChangeDetect (enum)**

DISABLED
ENABLED

**H264Settings**

**minIInterval**

Only meaningful if sceneChangeDetect is set to enabled. 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. Note: Maximum GOP stretch = GOP size + Min-I-interval - 1

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

**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. This field is optional; when no value is specified the encoder will choose the number of slices based on encode resolution.

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

**parNumerator**

Pixel Aspect Ratio numerator.
Properties

**Type**: integer  
**Required**: False

**gopSizeUnits**

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

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

**maxBitrate**

Maximum bitrate in bits/second (for VBR mode only).

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

**bitrate**

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

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

**bufFillPct**

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

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

**temporalAq**

If set to enabled, adjust quantization within each frame based on temporal variation of content complexity.

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

**afdSignaling**

Indicates that AFD values will be written into the output stream. If afdSignaling is "auto", the system will try to preserve the input AFD value (in cases where multiple AFD values are valid). If set to “fixed”, the AFD value will be the value configured in the fixedAfD parameter.

**Type**: string  
**Required**: False
timecodeInsertion
Determines how timecodes should be inserted into the video elementary stream. - 'disabled': Do not include timecodes - 'picTimingSei': Pass through picture timing SEI messages from the source specified in Timecode Config
  
  **Type:** string  
  **Required:** False

bufSize
Size of buffer (HRD buffer model) in bits/second.
  
  **Type:** integer  
  **Required:** False  
  **Minimum:** 0

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

framerateControl
This field indicates how the output video frame rate is specified. If "specified" is selected then the output video frame rate is determined by framerateNumerator and framerateDenominator, else if "initializeFromSource" is selected then the output video frame rate will be set equal to the input video frame rate of the first input.
  
  **Type:** string  
  **Required:** False

fixedAfd
Four bit AFD value to write on all frames of video in the output stream. Only valid when afdSignaling is set to 'Fixed'.
  
  **Type:** string  
  **Required:** False

level
H.264 Level.
  
  **Type:** string  
  **Required:** False

lookAheadRateControl
Amount of lookahead. A value of low can decrease latency and memory usage, while high can produce better quality for certain content.
Properties

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

**profile**

H.264 Profile.

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

**framerateNumerator**

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

**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

**framerateDenominator**

Framerate denominator.

**Type**: integer  
**Required**: False

**entropyEncoding**

Entropy encoding mode. Use cabac (must be in Main or High profile) or cavlc.

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

**spatialAq**

If set to enabled, adjust quantization within each frame based on spatial variation of content complexity.

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

**adaptiveQuantization**

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

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

Includes colorspace metadata in the output.

*Type:* string  
*Required:* False

**gopSize**

GOP size (keyframe interval) in units of either frames or seconds per gopSizeUnits. Must be greater than zero.

*Type:* number  
*Required:* False  
*Minimum:* 1.0

**numRefFrames**

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

**gopBReference**

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

*Type:* string  
*Required:* False

**sceneChangeDetect**

Scene change detection. Inserts I-frames on scene changes when enabled.

*Type:* string  
*Required:* False

**parControl**

This field indicates how the output pixel aspect ratio is specified. If "specified" is selected then the output video pixel aspect ratio is determined by parNumerator and parDenominator, else if "initializeFromSource" is selected then the output pixel aspect ratio will be set equal to the input video pixel aspect ratio of the first input.

*Type:* string  
*Required:* False

**parDenominator**

Pixel Aspect Ratio denominator.

*Type:* integer
Required: False
Minimum: 1

**syntax**

Produces a bitstream compliant with SMPTE RP-2027.

Type: string
Required: False

**scanType**

Sets the scan type of the output to progressive or top-field-first interlaced.

Type: string
Required: False

**gopNumBFrames**

Number of B-frames between reference frames.

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

**flickerAq**

If set to enabled, adjust quantization within each frame to reduce flicker or 'pop' on I-frames.

Type: string
Required: False

**rateControlMode**

Rate control mode.

Type: string
Required: False

**H264SpatialAq (enum)**

DISABLED
ENABLED

**H264Syntax (enum)**

DEFAULT
RP2027

**H264TemporalAq (enum)**

DISABLED
ENABLED

**H264TimecodeInsertionBehavior (enum)**

DISABLED
PIC_TIMING_SEI

**HlsAdMarkers (enum)**

ADOBE
ELEMENTAL
ELEMENTAL_SCTE35

**HlsAkamaiHttpTransferMode (enum)**

CHUNKED
NON_CHUNKED

**HlsAkamaiSettings**

**httpTransferMode**

Specify whether or not to use chunked transfer encoding to Akamai. User should contact Akamai to enable this feature.

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

**salt**

Salt for authenticated Akamai.

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

**numRetries**

Number of retry attempts that will be made before the Live Event is put into an error state.

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

**restartDelay**

If a streaming output fails, number of seconds to wait until a restart is initiated. A value of 0 means never restart.

- **Type:** integer
  - **Required:** False
  - **Minimum:** 0
Properties

Maximum: 15

connectionRetryInterval
Number of seconds to wait before retrying connection to the CDN if the connection is lost.

Type: integer
Required: False
Minimum: 0

filecacheDuration
Size in seconds of file cache for streaming outputs.

Type: integer
Required: False
Minimum: 0
Maximum: 600

token
Token parameter for authenticated akamai. If not specified, _gda_ is used.

Type: string
Required: False

HlsBasicPutSettings

numRetries
Number of retry attempts that will be made before the Live Event is put into an error state.

Type: integer
Required: False
Minimum: 0

restartDelay
If a streaming output fails, number of seconds to wait until a restart is initiated. A value of 0 means never restart.

Type: integer
Required: False
Minimum: 0
Maximum: 15

connectionRetryInterval
Number of seconds to wait before retrying connection to the CDN if the connection is lost.

Type: integer
Required: False
Minimum: 0
filecacheDuration
Size in seconds of file cache for streaming outputs.
  Type: integer
  Required: False
  Minimum: 0
  Maximum: 600

HlsCaptionLanguageSetting (enum)
  INSERT
  NONE
  OMIT

HlsCdnSettings

hlsAkamaiSettings
  Type: HlsAkamaiSettings (p. 415)
  Required: False

hlsWebdavSettings
  Type: HlsWebdavSettings (p. 426)
  Required: False

hlsBasicPutSettings
  Type: HlsBasicPutSettings (p. 416)
  Required: False

hlsMediaStoreSettings
  Type: HlsMediaStoreSettings (p. 424)
  Required: False

HlsClientCache (enum)
  DISABLED
  ENABLED

HlsCodecSpecification (enum)
  RFC_4281
  RFC_6381

HlsDirectoryStructure (enum)
  SINGLE_DIRECTORY
  SUBDIRECTORY_PER_STREAM
HlsEncryptionType (enum)

AES128
SAMPLE_AES

HlsGroupSettings

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

Type: integer
Required: False
Minimum: 1

ivInManifest
For use with encryptionType. The IV (Initialization Vector) is a 128-bit number used in conjunction with the key for encrypting blocks. If set to "include", IV is listed in the manifest, otherwise the IV is not in the manifest.

Type: string
Required: False

outputSelection
Generates the .m3u8 playlist file for this HLS output group. The segmentsOnly option will output segments without the .m3u8 file.

Type: string
Required: False

encryptionType
Encrypts the segments with the given encryption scheme. Exclude this parameter if no encryption is desired.

Type: string
Required: False

indexNSegments
Number of segments to keep in the playlist (.m3u8) file. mode must be "vod" for this setting to have an effect, and this number should be less than or equal to keepSegments.

Type: integer
Required: False
Minimum: 3

destination
A directory or HTTP destination for the HLS segments, manifest files, and encryption keys (if enabled).
Properties

**Type:** OutputLocationRef (p. 451)
**Required:** True

**constantIv**
For use with encryptionType. This is a 128-bit, 16-byte hex value represented by a 32-character text string. If ivSource is set to "explicit" then this parameter is required and is used as the IV for encryption.

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

**timedMetadataId3Frame**
Indicates ID3 frame that has the timecode.

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

**baseUrlManifest**
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

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

**minSegmentLength**
When set, minimumSegmentLength 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

**mode**
If set to "vod", keeps and indexes all segments starting with the first segment. If set to "live" segments will age out and only the last keepSegments number of segments will be retained.

**Type:** string
ivSource
For use with encryptionType. The IV (Initialization Vector) is a 128-bit number used in conjunction with the key for encrypting blocks. If this setting is "followsSegmentNumber", it will cause the IV to change every segment (to match the segment number). If this is set to "explicit", you must enter a constantIV value.

Type: string
Required: False

manifestCompression
When set to gzip, compresses HLS playlist.

Type: string
Required: False

keyProviderSettings
The key provider settings.

Type: KeyProviderSettings (p. 432)
Required: False

tsFileMode
When set to "singleFile", emits the program as a single media resource (.ts) file, and uses #EXT-X-BYTERANGE tags to index segment for playback. Playback of VOD mode content during event is not guaranteed due to HTTP server caching.

Type: string
Required: False

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

Type: string
Required: False

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

Type: string
Required: False

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

Type: string
Required: False
timestampDeltaMilliseconds

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

Type: integer
  Required: False
  Minimum: 0

segmentationMode

When set to useInputSegmentation, the output segment or fragment points are set by the RAI markers from the input streams.

Type: string
  Required: False

baseUrlContent

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

clientCache

When set to "disabled", sets the #EXT-X-ALLOW-CACHE:no tag in the manifest, which prevents clients from saving media segments for later replay.

Type: string
  Required: False

captionLanguageMappings

Mapping of up to 4 caption channels to caption languages. Is only meaningful if captionLanguageSetting is set to "insert".

Type: Array of type CaptionLanguageMapping (p. 389)
  Required: False

codecSpecification

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

Type: string
  Required: False

keepSegments

Number of segments to retain in the destination directory. mode must be "live" for this setting to have an effect.

Type: integer
  Required: False
  Minimum: 1
timedMetadataId3Period
Timed Metadata interval in seconds.

Type: integer
Required: False
Minimum: 0

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 timestampOffset.

Type: string
Required: False

directoryStructure
Place segments in subdirectories.

Type: string
Required: False

keyFormat
The value specifies how the key is represented in the resource identified by the URI. If parameter is absent, an implicit value of "identity" is used. A reverse DNS string can also be given.

Type: string
Required: False

inputLossAction
Parameter that control output group behavior on input loss.

Type: string
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 string
Required: False

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

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

**hlsCdnSettings**

Parameters that control interactions with the CDN.

- **Type**: HlsCdnSettings (p. 417)
- **Required**: False

**HlsInputSettings**

**retries**

The number of consecutive times that attempts to read a manifest or segment must fail before the input is considered unavailable.

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

**bandwidth**

When specified the HLS stream with the m3u8 BANDWIDTH that most closely matches this value will be chosen, otherwise the highest bandwidth stream in the m3u8 will be chosen. The bitrate is specified in bits per second, as in an HLS manifest.

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

**retryInterval**

The number of seconds between retries when an attempt to read a manifest or segment fails.

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

**bufferSegments**

When specified, reading of the HLS input will begin this many buffer segments from the end (most recently written segment). When not specified, the HLS input will begin with the first segment specified in the m3u8.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
HlsIvInManifest (enum)

    EXCLUDE
    INCLUDE

HlsIvSource (enum)

    EXPLICIT
    FOLLOWS_SEGMENT_NUMBER

HlsManifestCompression (enum)

    GZIP
    NONE

HlsManifestDurationFormat (enum)

    FLOATING_POINT
    INTEGER

HlsMediaStoreSettings

mediaStoreStorageClass

When set to temporal, output files are stored in non-persistent memory for faster reading and writing.

    Type: string
    Required: False

numRetries

Number of retry attempts that will be made before the Live Event is put into an error state.

    Type: integer
    Required: False
    Minimum: 0

restartDelay

If a streaming output fails, number of seconds to wait until a restart is initiated. A value of 0 means never restart.

    Type: integer
    Required: False
    Minimum: 0
    Maximum: 15

collectionRetryInterval

Number of seconds to wait before retrying connection to the CDN if the connection is lost.

    Type: integer
    Required: False
Minimum: 0

filecacheDuration
Size in seconds of file cache for streaming outputs.
  - Type: integer
  - Required: False
  - Minimum: 0
  - Maximum: 600

HlsMediaStoreStorageClass (enum)
  - TEMPORAL

HlsMode (enum)
  - LIVE
  - VOD

HlsOutputSelection (enum)
  - MANIFESTS_AND_SEGMENTS
  - SEGMENTS_ONLY

HlsOutputSettings

segmentModifier
String concatenated to end of segment filenames.
  - Type: string
  - Required: False

hlsSettings
Settings regarding the underlying stream. These settings are different for audio-only outputs.
  - Type: HlsSettings (p. 426)
  - Required: True

nameModifier
String concatenated to the end of the destination filename. Accepts "Format Identifiers \"Format Identifiers \r\n \":#formatIdentifierParameters.
  - Type: string
  - Required: False

HlsProgramDateTime (enum)
  - EXCLUDE
  - INCLUDE
HlsSegmentationMode (enum)

USE_INPUT_SEGMENTATION
USE_SEGMENT_DURATION

HlsSettings

audioOnlyHlsSettings

Type: AudioOnlyHlsSettings (p. 380)
Required: False

standardHlsSettings

Type: StandardHlsSettings (p. 458)
Required: False

HlsStreamInfResolution (enum)

EXCLUDE
INCLUDE

HlsTimedMetadataId3Frame (enum)

NONE
PRIV
TDRL

HlsTsFileMode (enum)

SEGMENTED_FILES
SINGLE_FILE

HlsWebdavHttpTransferMode (enum)

CHUNKED
NON_CHUNKED

HlsWebdavSettings

httpTransferMode

Specify whether or not to use chunked transfer encoding to WebDAV.

Type: string
Required: False

numRetries

Number of retry attempts that will be made before the Live Event is put into an error state.

Type: integer
**Properties**

---

**Required**: False  
**Minimum**: 0

**restartDelay**
If a streaming output fails, number of seconds to wait until a restart is initiated. A value of 0 means never restart.

- **Type**: integer  
  - **Required**: False  
  - **Minimum**: 0  
  - **Maximum**: 15

**connectionRetryInterval**
Number of seconds to wait before retrying connection to the CDN if the connection is lost.

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

**filecacheDuration**
Size in seconds of file cache for streaming outputs.

- **Type**: integer  
  - **Required**: False  
  - **Minimum**: 0  
  - **Maximum**: 600

---

**InputAttachment**

---

**inputId**
The ID of the input

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

**inputSettings**
Settings of an input (caption selector, etc.)

- **Type**: InputSettings (p. 430)  
  - **Required**: False

---

**InputChannelLevel**

---

**inputChannel**
The index of the input channel used as a source.

- **Type**: integer
**Properties**

**Required**: True  
**Minimum**: 0  
**Maximum**: 15

**gain**

Remixing value. Units are in dB and acceptable values are within the range from -60 (mute) and 6 dB.

**Type**: integer  
**Required**: True  
**Minimum**: -60  
**Maximum**: 6

**InputCodec (enum)**

codec in increasing order of complexity  
- MPEG2  
- AVC  
- HEVC

**InputDeblockFilter (enum)**

- DISABLED  
- ENABLED

**InputDenoiseFilter (enum)**

- DISABLED  
- ENABLED

**InputFilter (enum)**

- AUTO  
- DISABLED  
- FORCED

**InputLocation**

**passwordParam**

key used to extract the password from EC2 Parameter store  
**Type**: string  
**Required**: False

**uri**

Uniform Resource Identifier - This should be a path to a file accessible to the Live system (eg. a http:// URI) depending on the output type. For example, a rtmpEndpoint should have a uri similar to: "rtmp:// fmsserver/live".
**Properties**

**username**

Username if credentials are required to access a file or publishing point. This can be either a plaintext username, or a reference to an AWS parameter store name from which the username can be retrieved. AWS Parameter store format: "ssm://<parameter name>"

**InputLossActionForHlsOut (enum)**

- EMIT_OUTPUT
- PAUSE_OUTPUT

**InputLossActionForMsSmoothOut (enum)**

- EMIT_OUTPUT
- PAUSE_OUTPUT

**InputLossActionForUdpOut (enum)**

- DROP_PROGRAM
- DROP_TS
- EMIT_PROGRAM

**InputLossBehavior**

**inputLossImageType**

Indicates whether to substitute a solid color or a slate into the output after input loss exceeds blackFrameMsec.

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

**inputLossImageColor**

When input loss image type is "color" this field specifies the color to use. Value: 6 hex characters representing the values of RGB.

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

**inputLossImageSlate**

When input loss image type is "slate" these fields specify the parameters for accessing the slate.

**Type**: InputLocation (p. 428)
**Required**: False
blackFrameMsec
On input loss, the number of milliseconds to substitute black into the output before switching to the frame specified by inputLossImageType. A value x, where 0 <= x <= 1,000,000 and a value of 1,000,000 will be interpreted as infinite.

Type: integer
Required: False
Minimum: 0
Maximum: 1000000

repeatFrameMsec
On input loss, the number of milliseconds to repeat the previous picture before substituting black into the output. A value x, where 0 <= x <= 1,000,000 and a value of 1,000,000 will be interpreted as infinite.

Type: integer
Required: False
Minimum: 0
Maximum: 1000000

InputLossImageType (enum)
COLOR
SLATE

InputMaximumBitrate (enum)
Maximum input bitrate in megabits per second. Bitrates up to 50 Mbps are supported currently.

MAX_10_MBPS
MAX_20_MBPS
MAX_50_MBPS

InputResolution (enum)
Input resolution based on lines of vertical resolution in the input; SD is less than 720 lines, HD is 720 to 1080 lines, UHD is greater than 1080 lines

SD
HD
UHD

InputSettings

sourceEndBehavior
Loop input if it is a file. This allows a file input to be streamed indefinitely.

Type: string
Required: False

audioSelectors
Used to select the audio stream to decode for inputs that have multiple available.
**Properties**

- **Type**: Array of type `AudioSelector` *(p. 381)*
  - **Required**: False

**deblockFilter**

Enable or disable the deblock filter when filtering.

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

**networkInputSettings**

Input settings.

- **Type**: `NetworkInputSettings` *(p. 448)*
  - **Required**: False

**inputFilter**

Turns on the filter for this input. MPEG-2 inputs have the deblurring filter enabled by default. 1) auto - filtering will be applied depending on input type/quality 2) disabled - no filtering will be applied to the input 3) forced - filtering will be applied regardless of input type

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

**videoSelector**

Informs which video elementary stream to decode for input types that have multiple available.

- **Type**: `VideoSelector` *(p. 463)*
  - **Required**: False

**filterStrength**

Adjusts the magnitude of filtering from 1 (minimal) to 5 (strongest).

- **Type**: integer
  - **Required**: False
  - **Minimum**: 1
  - **Maximum**: 5

**captionSelectors**

Used to select the caption input to use for inputs that have multiple available.

- **Type**: Array of type `CaptionSelector` *(p. 390)*
  - **Required**: False

**denoiseFilter**

Enable or disable the denoise filter when filtering.

- **Type**: string
Properties

**Required:** False

**InputSourceEndBehavior (enum)**
- CONTINUE
- LOOP

**InputSpecification**

**codec**
Input codec
- **Type:** string
- **Required:** False

**resolution**
Input resolution, categorized coarsely
- **Type:** string
- **Required:** False

**maximumBitrate**
Maximum input bitrate, categorized coarsely
- **Type:** string
- **Required:** False

**InternalServiceError**

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

**InvalidRequest**

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

**KeyProviderSettings**

**staticKeySettings**
- **Type:** StaticKeySettings (p. 458)
- **Required:** False
LimitExceeded

message

  Type: string
  Required: False

M2tsAbsentInputAudioBehavior (enum)

  DROP
  ENCODE_SILENCE

M2tsArib (enum)

  DISABLED
  ENABLED

M2tsAribCaptionsPidControl (enum)

  AUTO
  USE_CONFIGURED

M2tsAudioBufferModel (enum)

  ATSC
  DVB

M2tsAudioInterval (enum)

  VIDEO_AND_FIXED_INTERVALS
  VIDEO_INTERVAL

M2tsAudioStreamType (enum)

  ATSC
  DVB

M2tsBufferModel (enum)

  MULTIPLEX
  NONE

M2tsCcDescriptor (enum)

  DISABLED
  ENABLED

M2tsEbifControl (enum)

  NONE
  PASSTHROUGH
M2tsEbpPlacement (enum)

- VIDEO_AND_AUDIO_PIDS
- VIDEO_PID

M2tsEsRateInPes (enum)

- EXCLUDE
- INCLUDE

M2tsKlv (enum)

- NONE
- PASSTHROUGH

M2tsPcrControl (enum)

- CONFIGURED_PCR_PERIOD
- PCR_EVERY_PES_PACKET

M2tsRateMode (enum)

- CBR
- VBR

M2tsScte35Control (enum)

- NONE
- PASSTHROUGH

M2tsSegmentationMarkers (enum)

- EBP
- EBP_LEGACY
- NONE
- PSI_SEGSTART
- RAI_ADAPT
- RAI_SEGSTART

M2tsSegmentationStyle (enum)

- MAINTAIN_CADENCE
- RESET_CADENCE

M2tsSettings

audioStreamType

When set to atsc, uses stream type = 0x81 for AC3 and stream type = 0x87 for EAC3. When set to dvb, uses stream type = 0x06.

Type: string
**Properties**

**Required**: False

**ecmPid**
Packet Identifier (PID) for ECM in the transport stream. Only enabled when Simulcrypt is enabled. Can be entered as a decimal or hexadecimal value. Valid values are 32 (0x20)..8182 (0x1ff6).

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

**dvbTeletextPid**
Packet Identifier (PID) for input source DVB Teletext data to this output. Can be entered as a decimal or hexadecimal value. Valid values are 32 (0x20)..8182 (0x1ff6).

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

**aribCaptionsPidControl**
If set to auto, pid number used for ARIB Captions will be auto-selected from unused pids. If set to useConfigured, ARIB Captions will be on the configured pid number.

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

**bitrate**
The output bitrate of the transport stream in bits per second. Setting to 0 lets the muxer automatically determine the appropriate bitrate.

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

**segmentationTime**
The length in seconds of each segment. Required unless markers is set to None_.

**Type**: number  
**Required**: False  
**Minimum**: 1.0

**rateMode**
When vbr, does not insert null packets into transport stream to fill specified bitrate. The bitrate setting acts as the maximum bitrate when vbr is set.

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

Packet Identifier (PID) of the elementary audio stream(s) in the transport stream. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values. Each PID specified must be in the range of 32 (or 0x20)..8182 (or 0x1ff6).

**Type**: string

**Required**: False

**fragmentTime**

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

**Type**: number

**Required**: False

**Minimum**: 0.0

**ebpAudioInterval**

When `videoAndFixedIntervals` is selected, 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. Only available when EBP Cablelabs segmentation markers are selected. Partitions 1 and 2 will always follow the video interval.

**Type**: string

**Required**: False

**ebpLookaheadMs**

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

**audioFramesPerPes**

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

**Type**: integer

**Required**: False

**Minimum**: 0

**scte35Pid**

Packet Identifier (PID) of the SCTE-35 stream in the transport stream. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20)..8182 (or 0x1ff6).

**Type**: string

**Required**: False
pcrPeriod
Maximum time in milliseconds between Program Clock Reference (PCRs) inserted into the transport stream.

- **Type**: integer
- **Required**: False
- **Minimum**: 0
- **Maximum**: 500

pmtInterval
The number of milliseconds between instances of this table in the output transport stream. Valid values are 0, 10..1000.

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

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

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

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 "resetCadence" 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 $segmentationTime seconds. When a segmentation style of "maintainCadence" 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 $segmentationTime seconds. Note that EBP lookahead is a slight exception to this rule.

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

ebif
If set to passthrough, passes any EBIF data from the input source to this output.

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

audioBufferModel
When set to dvb, uses DVB buffer model for Dolby Digital audio. When set to atsc, the ATSC model is used.
Properties

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

**dvbNitSettings**

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

**Type**: DvbNitSettings (p. 393)  
**Required**: False

**absentInputAudioBehavior**

When set to drop, output audio streams will be removed from the program if the selected input audio stream is removed from the input. This allows the output audio configuration to dynamically change based on input configuration. If this is set to encodeSilence, all output audio streams will output encoded silence when not connected to an active input stream.

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

**timedMetadataPid**

Packet Identifier (PID) of the timed metadata stream in the transport stream. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20)..8182 (or 0x1ff6).

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

**timedMetadataBehavior**

When set to passthrough, timed metadata will be passed through from input to output.

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

**etvSignalPid**

Packet Identifier (PID) for input source ETV Signal data to this output. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20)..8182 (or 0x1ff6).

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

**pmtPid**

Packet Identifier (PID) for the Program Map Table (PMT) in the transport stream. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20)..8182 (or 0x1ff6).

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

If set to multiplex, use multiplex buffer model for accurate interleaving. Setting to bufferModel to none can lead to lower latency, but low-memory devices may not be able to play back the stream without interruptions.

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

**scte35Control**

Optionally pass SCTE-35 signals from the input source to this output.

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

**ebpPlacement**

Controls placement of EBP on Audio PIDs. If set to videoAndAudioPids, EBP markers will be placed on the video PID and all audio PIDs. If set to videoPid, EBP markers will be placed on only the video PID.

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

**arib**

When set to enabled, uses ARIB-compliant field muxing and removes video descriptor.

- **Type**: string
- **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
- **Minimum**: 0.0

**dvbSdtSettings**

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

- **Type**: [DvbSdtSettings](p. 393)
- **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. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20)..8182 (or 0x1ff6).

- **Type**: string
Required: False

**transportStreamId**

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

_Type:_ integer  
_Required:_ False  
_Minimum:_ 0  
_Maximum:_ 65535

**pcrControl**

When set to `pcrEveryPesPacket`, 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:_ string  
_Required:_ False

**videoPid**

Packet Identifier (PID) of the elementary video stream in the transport stream. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20)..8182 (or 0x1ff6).

_Type:_ string  
_Required:_ False

**esRateInPes**

Include or exclude the ES Rate field in the PES header.

_Type:_ string  
_Required:_ False

**segmentationMarkers**

Inserts segmentation markers at each `segmentationTime` period. `raiSegstart` sets the Random Access Indicator bit in the adaptation field. `raiAdapt` sets the RAI bit and adds the current timecode in the private data bytes. `psiSegstart` 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. `ebpLegacy` adds Encoder Boundary Point information to the adaptation field using a legacy proprietary format.

_Type:_ string  
_Required:_ False

**dvbTdtSettings**

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

_Type:_ [DvbTdtSettings](p. 398)  
_Required:_ False
**klv**

If set to passthrough, passes any KLV data from the input source to this output.

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

**ccDescriptor**

When set to enabled, generates captionServiceDescriptor in PMT.

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

**patInterval**

The number of milliseconds between instances of this table in the output transport stream. Valid values are 0, 10..1000.

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

**etvPlatformPid**

Packet Identifier (PID) for input source ETV Platform data to this output. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20)..8182 (or 0x1ff6).

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

**dvbSubPids**

Packet Identifier (PID) for input source DVB Subtitle data to this output. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values. Each PID specified must be in the range of 32 (or 0x20)..8182 (or 0x1ff6).

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

**aribCaptionsPid**

Packet Identifier (PID) for ARIB Captions in the transport stream. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20)..8182 (or 0x1ff6).

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

**scte27Pids**

Packet Identifier (PID) for input source SCTE-27 data to this output. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values. Each PID specified must be in the range of 32 (or 0x20)..8182 (or 0x1ff6).
Type: string
Required: False

**klvDataPids**

Packet Identifier (PID) for input source KLV data to this output. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values. Each PID specified must be in the range of 32 (or 0x20)..8182 (or 0x1ff6).

Type: string
Required: False

**M2tsTimedMetadataBehavior (enum)**

- NO_PASSTHROUGH
- PASSTHROUGH

**M3u8PcrControl (enum)**

- CONFIGURED_PCR_PERIOD
- PCR_EVERY_PES_PACKET

**M3u8Scte35Behavior (enum)**

- NO_PASSTHROUGH
- PASSTHROUGH

**M3u8Settings**

**pmtPid**

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

Type: string
Required: False

**ecmPid**

ThePlatform-protected transport streams using 'microsoft' as Target Client include an ECM stream. This ECM stream contains the size, IV, and PTS of every sample in the transport stream. This stream PID is specified here. This PID has no effect on non ThePlatform-protected streams.

Type: string
Required: False

**scte35Behavior**

If set to passthrough, passes any SCTE-35 signals from the input source to this output.

Type: string
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. Can be entered as a decimal or hexadecimal value.

*Type:* string  
*Required:* False

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

*Type:* string  
*Required:* False

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

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

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

*Type:* string  
*Required:* False

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

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

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

*Type:* string  
*Required:* False

**pcrControl**
When set to `pcrEveryPesPacket`, 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.
## Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>pmtInterval</td>
<td>string</td>
<td>False</td>
<td>The number of milliseconds between instances of this table in the output transport stream. A value of &quot;0&quot; writes out the PMT once per segment file.</td>
</tr>
<tr>
<td>pcrPeriod</td>
<td>integer</td>
<td>False</td>
<td>Maximum time in milliseconds between Program Clock References (PCRs) inserted into the transport stream.</td>
</tr>
<tr>
<td>programNum</td>
<td>integer</td>
<td>False</td>
<td>The value of the program number field in the Program Map Table.</td>
</tr>
<tr>
<td>patInterval</td>
<td>integer</td>
<td>False</td>
<td>The number of milliseconds between instances of this table in the output transport stream. A value of &quot;0&quot; writes out the PMT once per segment file.</td>
</tr>
<tr>
<td>timedMetadataPid</td>
<td>string</td>
<td>False</td>
<td>Packet Identifier (PID) of the timed metadata stream in the transport stream. Can be entered as a decimal or hexadecimal value. Valid values are 32 (or 0x20).8182 (or 0x1ff6).</td>
</tr>
<tr>
<td>timedMetadataBehavior</td>
<td>string</td>
<td>False</td>
<td>When set to passthrough, timed metadata is passed through from input to output.</td>
</tr>
</tbody>
</table>
**Properties**

**Required**: False

**M3u8TimedMetadataBehavior (enum)**

- NO_PASSTHROUGH
- PASSTHROUGH

**Mp2CodingMode (enum)**

- CODING_MODE_1_0
- CODING_MODE_2_0

**Mp2Settings**

**codingMode**

The MPEG2 Audio coding mode. Valid values are codingMode10 (for mono) or codingMode20 (for stereo).

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

**bitrate**

Average bitrate in bits/second.

- **Type**: number
- **Required**: False

**sampleRate**

Sample rate in Hz.

- **Type**: number
- **Required**: False

**MsSmoothGroupSettings**

**eventId**

MS Smooth event ID to be sent to the IIS server. Should only be specified if eventIdMode is set to useConfigured.

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

**fragmentLength**

Length of mp4 fragments to generate (in seconds). Fragment length must be compatible with GOP size and framerate.

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

**timestampOffset**
Timestamp offset for the event. Only used if `timestampOffsetMode` is set to `useConfiguredOffset`.

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

**segmentationMode**
When set to `useInputSegmentation`, the output segment or fragment points are set by the RAI markers from the input streams.

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

**numRetries**
Number of retry attempts.

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

**eventStopBehavior**
When set to `sendEos`, send EOS signal to IIS server when stopping the event.

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

**acquisitionPointId**
The value of the "Acquisition Point Identity" element used in each message placed in the sparse track. Only enabled if `sparseTrackType` is not "none".

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

**sparseTrackType**
If set to `scte35`, use incoming SCTE-35 messages to generate a sparse track in this group of MS-Smooth outputs.

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

**timestampOffsetMode**
Type of timestamp date offset to use. - `useEventStartDate`: Use the date the event was started as the offset - `useConfiguredOffset`: Use an explicitly configured date as the offset

  - **Type:** string
  - **Required:** False
**destination**
Smooth Streaming publish point on an IIS server. Elemental Live acts as a "Push" encoder to IIS.

*Type:* OutputLocationRef (p. 451)
*Required:* True

**audioOnlyTimecodeControl**
If set to passthrough for an audio-only MS Smooth output, the fragment absolute time will be set to the current timecode. This option does not write timecodes to the audio elementary stream.

*Type:* string
*Required:* False

**connectionRetryInterval**
Number of seconds to wait before retrying connection to the IIS server if the connection is lost. Content will be cached during this time and the cache will be be delivered to the IIS server once the connection is re-established.

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

**filecacheDuration**
Size in seconds of file cache for streaming outputs.

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

**certificateMode**
If set to verifyAuthenticity, verify the https certificate chain to a trusted Certificate Authority (CA). This will cause https outputs to self-signed certificates to fail unless those certificates are manually added to the OS trusted keystore.

*Type:* string
*Required:* False

**inputLossAction**
Parameter that control output group behavior on input loss.

*Type:* string
*Required:* False

**sendDelayMs**
Outputs that are "output locked" can use this delay. Assign a delay to the output that is "secondary". Do not assign a delay to the "primary" output. The delay means that the primary output will always reach the downstream system before the secondary, which helps ensure that the downstream system always uses the primary output. (If there were no delay, the downstream system might flip-flop between whichever output happens to arrive first.) If the primary fails, the downstream system will switch to the
secondary output. When the primary is restarted, the downstream system will switch back to the primary
(because once again it is always arriving first)

**eventIdMode**

Specifies whether or not to send an event ID to the IIS server. If no event ID is sent and the same Live
Event is used without changing the publishing point, clients might see cached video from the previous
run. Options: - "useConfigured" - use the value provided in eventId - "useTimestamp" - generate and send
an event ID based on the current timestamp - "noEventId" - do not send an event ID to the IIS server.

**Type:** string

**Required:** False

**restartDelay**

Number of seconds before initiating a restart due to output failure, due to exhausting the numRetries on
one segment, or exceeding filecacheDuration.

**Type:** integer

**Required:** False

**streamManifestBehavior**

When set to send, send stream manifest so publishing point doesn't start until all streams start.

**Type:** string

**Required:** False

**MsSmoothOutputSettings**

**nameModifier**

String concatenated to the end of the destination filename. Required for multiple outputs of the same
type.

**Type:** string

**Required:** False

**NetworkInputServerValidation (enum)**

- CHECK_CRYPTOGRAPHY_AND_VALIDATE_NAME
- CHECK_CRYPTOGRAPHY_ONLY

**NetworkInputSettings**

**hlsInputSettings**

Specifies HLS input settings when the uri is for a HLS manifest.
Properties

Type: HlsInputSettings (p. 423)
Required: False

serverValidation

Check HTTPS server certificates. When set to checkCryptographyOnly, cryptography in the certificate will be checked, but not the server's name. Certain subdomains (notably S3 buckets that use dots in the bucket name) do not strictly match the corresponding certificate's wildcard pattern and would otherwise cause the event to error. This setting is ignored for protocols that do not use https.

  Type: string
  Required: False

Output

videoDescriptionName

The name of the VideoDescription used as the source for this output.

  Type: string
  Required: False

captionDescriptionNames

The names of the CaptionDescriptions used as caption sources for this output.

  Type: Array of type string
  Required: False

outputName

The name used to identify an output.

  Type: string
  Required: False

outputSettings

Output type-specific settings.

  Type: OutputSettings (p. 452)
  Required: True

audioDescriptionNames

The names of the AudioDescriptions used as audio sources for this output.

  Type: Array of type string
  Required: False
OutputDestination

settings

Destination settings for output; one for each redundant encoder.

  Type: Array of type OutputDestinationSettings (p. 450)
  Required: False

id

User-specified id. This is used in an output group or an output.

  Type: string
  Required: False

OutputDestinationSettings

passwordParam

key used to extract the password from EC2 Parameter store

  Type: string
  Required: False

streamName

Stream name for RTMP destinations (URLs of type rtmp://)

  Type: string
  Required: False

url

A URL specifying a destination

  Type: string
  Required: False

username

username for destination

  Type: string
  Required: False

OutputGroup

outputs

  Type: Array of type Output (p. 449)
Required: True

outputGroupSettings
Settings associated with the output group.
  Type: OutputGroupSettings (p. 451)
  Required: True

name
Custom output group name optionally defined by the user. Only letters, numbers, and the underscore character allowed; only 32 characters allowed.
  Type: string
  Required: False

OutputGroupSettings

archiveGroupSettings
  Type: ArchiveGroupSettings (p. 376)
  Required: False

rtmpGroupSettings
  Type: RtmpGroupSettings (p. 453)
  Required: False

udpGroupSettings
  Type: UdpGroupSettings (p. 460)
  Required: False

msSmoothGroupSettings
  Type: MsSmoothGroupSettings (p. 445)
  Required: False

hlsGroupSettings
  Type: HlsGroupSettings (p. 418)
  Required: False

OutputLocationRef

destinationRefId
  Type: string
  Required: False
OutputSettings

rtmpOutputSettings

Type: RtmpOutputSettings (p. 454)
Required: False

archiveOutputSettings

Type: ArchiveOutputSettings (p. 376)
Required: False

msSmoothOutputSettings

Type: MsSmoothOutputSettings (p. 448)
Required: False

udpOutputSettings

Type: UdpOutputSettings (p. 460)
Required: False

hlsOutputSettings

Type: HlsOutputSettings (p. 425)
Required: False

PassThroughSettings

RemixSettings

channelMappings
Mapping of input channels to output channels, with appropriate gain adjustments.

Type: Array of type AudioChannelMapping (p. 376)
Required: True

channelsOut
Number of output channels to be produced. Valid values: 1, 2, 4, 6, 8

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

channelsIn
Number of input channels to be used.

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

**ResourceConflict**

*message*

Type: string
Required: False

**ResourceNotFound**

*message*

Type: string
Required: False

**RtmpCacheFullBehavior (enum)**

- DISCONNECT_IMMEDIATELY
- WAIT_FOR_SERVER

**RtmpCaptionData (enum)**

- ALL
- FIELD1_608
- FIELD1_AND_FIELD2_608

**RtmpCaptionInfoDestinationSettings**

**RtmpGroupSettings**

*captionData*

Controls the types of data that passes to onCaptionInfo outputs. If set to 'all' then 608 and 708 carried DTVCC data will be passed. If set to 'field1AndField2608' then DTVCC data will be stripped out, but 608 data from both fields will be passed. If set to 'field1608' then only the data carried in 608 from field 1 video will be passed.

Type: string
Required: False

*authenticationScheme*

Authentication scheme to use when connecting with CDN

Type: string
Required: False

*cacheLength*

Cache length, in seconds, is used to calculate buffer size.
### Properties

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

**restartDelay**  
If a streaming output fails, number of seconds to wait until a restart is initiated. A value of 0 means never restart.  
**Type**: integer  
**Required**: False  
**Minimum**: 0

**cacheFullBehavior**  
Controls behavior when content cache fills up. If remote origin server stalls the RTMP connection and does not accept content fast enough the 'Media Cache' will fill up. When the cache reaches the duration specified by cacheLength the cache will stop accepting new content. If set to disconnectImmediately, the RTMP output will force a disconnect. Clear the media cache, and reconnect after restartDelay seconds. If set to waitForServer, the RTMP output will wait up to 5 minutes to allow the origin server to begin accepting data again.  
**Type**: string  
**Required**: False

**RtmpOutputCertificateMode (enum)**  
- SELF_SIGNED  
- VERIFY_AUTHENTICITY

### RtmpOutputSettings

**certificateMode**  
If set to verifyAuthenticity, verify the tls certificate chain to a trusted Certificate Authority (CA). This will cause rtmps outputs with self-signed certificates to fail.  
**Type**: string  
**Required**: False

**numRetries**  
Number of retry attempts.  
**Type**: integer  
**Required**: False  
**Minimum**: 0

**destination**  
The RTMP endpoint excluding the stream name (eg. rtmp://host/appname). For connection to Akamai, a username and password must be supplied. URI fields accept format identifiers.  
**Type**: OutputLocationRef (p. 451)
**Properties**

**required**: True

**connectionRetryInterval**

Number of seconds to wait before retrying a connection to the Flash Media server if the connection is lost.

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

**Scte20Convert608To708 (enum)**

DISABLED  
UPCONVERT

**Scte20PlusEmbeddedDestinationSettings**

**Scte20SourceSettings**

**source608ChannelNumber**

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

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

**convert608To708**

If 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:* string  
*Required:* False

**Scte27DestinationSettings**

**Scte27SourceSettings**

**pid**

The pid field is used in conjunction with the caption selector languageCode field as follows:  
- Specify PID and Language: Extracts captions from that PID; the language is "informational".  
- Specify PID and omit Language: Extracts the specified PID.  
- Omit PID and specify Language: Extracts the specified language, whichever PID that happens to be.  
- Omit PID and omit Language: Valid only if source is DVB-Sub that is being passed through; all languages will be passed through.

*Type:* integer  
*Required:* False  
*Minimum:* 1
**Scte35AposNoRegionalBlackoutBehavior (enum)**

- FOLLOW
- IGNORE

**Scte35AposWebDeliveryAllowedBehavior (enum)**

- FOLLOW
- IGNORE

**Scte35SpliceInsert**

**adAvailOffset**

When specified, this offset (in milliseconds) is added to the input Ad Avail PTS time. This only applies to embedded SCTE 104/35 messages and does not apply to OOB messages.

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

**webDeliveryAllowedFlag**

When set to ignore, Segment Descriptors with webDeliveryAllowedFlag set to 0 will no longer trigger blackouts or Ad Avail slates

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

**noRegionalBlackoutFlag**

When set to ignore, Segment Descriptors with noRegionalBlackoutFlag set to 0 will no longer trigger blackouts or Ad Avail slates

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

**Scte35SpliceInsertNoRegionalBlackoutBehavior (enum)**

- FOLLOW
- IGNORE

**Scte35SpliceInsertWebDeliveryAllowedBehavior (enum)**

- FOLLOW
- IGNORE
**Scte35TimeSignalApos**

**adAvailOffset**

When specified, this offset (in milliseconds) is added to the input Ad Avail PTS time. This only applies to embedded SCTE 104/35 messages and does not apply to OOB messages.

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

**webDeliveryAllowedFlag**

When set to ignore, Segment Descriptors with webDeliveryAllowedFlag set to 0 will no longer trigger blackouts or Ad Avail slates.

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

**noRegionalBlackoutFlag**

When set to ignore, Segment Descriptors with noRegionalBlackoutFlag set to 0 will no longer trigger blackouts or Ad Avail slates.

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

**SmoothGroupAudioOnlyTimecodeControl (enum)**

- PASSTHROUGH
- USE_CONFIGURED_CLOCK

**SmoothGroupCertificateMode (enum)**

- SELF_SIGNED
- VERIFY_AUTHENTICITY

**SmoothGroupEventIdMode (enum)**

- NO_EVENT_ID
- USE_CONFIGURED
- USE_TIMESTAMP

**SmoothGroupEventStopBehavior (enum)**

- NONE
- SEND_EOS

**SmoothGroupSegmentationMode (enum)**

- USE_INPUT_SEGMENTATION
- USE_SEGMENT_DURATION
Properties

**SmoothGroupSparseTrackType (enum)**
- NONE
- SCTE_35

**SmoothGroupStreamManifestBehavior (enum)**
- DO_NOT_SEND
- SEND

**SmoothGroupTimestampOffsetMode (enum)**
- USE_CONFIGURED_OFFSET
- USE_EVENT_START_DATE

**SmpteTtDestinationSettings**

**StandardHlsSettings**

**m3u8Settings**
- **Type**: M3u8Settings (p. 442)
- **Required**: True

**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

**StaticKeySettings**

**staticKeyValue**

Static key value as a 32 character hexadecimal string.

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

**keyProviderServer**

The URL of the license server used for protecting content.

- **Type**: InputLocation (p. 428)
- **Required**: False
**TeletextDestinationSettings**

**TeletextSourceSettings**

**pageNumber**

Specifies the teletext page number within the data stream from which to extract captions. Range of 0x100 (256) to 0x8FF (2303). Unused for passthrough. Should be specified as a hexadecimal string with no "0x" prefix.

*Type: string*

*Required: False*

**TimecodeConfig**

**syncThreshold**

Threshold in frames beyond which output timecode is resynchronized to the input timecode. Discrepancies below this threshold are permitted to avoid unnecessary discontinuities in the output timecode. No timecode sync when this is not specified.

*Type: integer*

*Required: False*

*Minimum: 1*

*Maximum: 1000000*

**source**

Identifies the source for the timecode that will be associated with the events outputs. -Embedded (embedded): Initialize the output timecode with timecode from the the source. If no embedded timecode is detected in the source, the system falls back to using "Start at 0" (zerobased). -System Clock (systemclock): Use the UTC time. -Start at 0 (zerobased): The time of the first frame of the event will be 00:00:00:00.

*Type: string*

*Required: True*

**TimecodeConfigSource (enum)**

- EMBEDDED
- SYSTEMCLOCK
- ZEROBASED

**TtmlDestinationSettings**

**styleControl**

When set to passthrough, passes 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: string*

*Required: False*
TtmlDestinationStyleControl (enum)

- PASSTHROUGH
- USE_CONFIGURED

UdpContainerSettings

m2tsSettings

Type: M2tsSettings (p. 434)
Required: False

UdpGroupSettings

inputLossAction

Specifies behavior of last resort when input video is lost, and no more backup inputs are available. When dropTS is selected the entire transport stream will stop being emitted. When dropProgram is selected the program can be dropped from the transport stream (and replaced with null packets to meet the TS bitrate requirement). Or, when emitProgram is chosen the transport stream will continue to be produced normally with repeat frames, black frames, or slate frames substituted for the absent input video.

Type: string
Required: False

timedMetadataId3Frame

Indicates ID3 frame that has the timecode.

Type: string
Required: False

timedMetadataId3Period

Timed Metadata interval in seconds.

Type: integer
Required: False
Minimum: 0

UdpOutputSettings

bufferMsec

UDP output buffering in milliseconds. Larger values increase latency through the transcoder but simultaneously assist the transcoder in maintaining a constant, low-jitter UDP/RTP output while accommodating clock recovery, input switching, input disruptions, picture reordering, etc.

Type: integer
Required: False
Minimum: 0
Maximum: 10000
destination

Destination address and port number for RTP or UDP packets. Can be unicast or multicast RTP or UDP (e.g. rtp://239.10.10.10:5001 or udp://10.100.100.100:5002).

  Type: OutputLocationRef (p. 451)
  Required: True

containerSettings

  Type: UdpContainerSettings (p. 460)
  Required: True

fecOutputSettings

Settings for enabling and adjusting Forward Error Correction on UDP outputs.

  Type: FecOutputSettings (p. 405)
  Required: False

UdpTimedMetadataId3Frame (enum)

  NONE
  PRIV
  TDRL

VideoCodecSettings

h264Settings

  Type: H264Settings (p. 409)
  Required: False

VideoDescription

respondToAfd

Indicates how to respond to the AFD values in the input stream. Setting to "respond" causes input video to be clipped, depending on AFD value, input display aspect ratio and output display aspect ratio.

  Type: string
  Required: False

scalingBehavior

When set to "stretchToOutput", automatically configures the output position to stretch the video to the specified output resolution. This option will override any position value.

  Type: string
  Required: False
name

The name of this VideoDescription. Outputs will use this name to uniquely identify this Description. Description names should be unique within this Live Event.

Type: string
Required: True

width

Output video width (in pixels). Leave out to use source video width. If left out, height must also be left out. Display aspect ratio is always preserved by letterboxing or pillarboxing when necessary.

Type: integer
Required: False

sharpness

Changes the width of the anti-alias filter kernel used for scaling. Only applies if scaling is being performed and antiAlias is set to true. 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.

Type: VideoCodecSettings (p. 461)
Required: False

height

Output video height (in pixels). Leave blank to use source video height. If left blank, width must also be unspecified.

Type: integer
Required: False

VideoDescriptionRespondToAfd (enum)

NONE
PASSTHROUGH
RESPOND

VideoDescriptionScalingBehavior (enum)

DEFAULT
STRETCH_TO_OUTPUT
Properties

VideoSelector

colorSpace

Specifies the colorspace of an input. This setting works in tandem with colorSpaceConversion to determine if any conversion will be performed.

Type: string
Required: False

selectorSettings

The video selector settings.

Type: VideoSelectorSettings (p. 464)
Required: False

colorSpaceUsage

Applies only if colorSpace is a value other than follow. This field controls how the value in the colorSpace field will be used. fallback means that when the input does include color space data, that data will be used, but when the input has no color space data, the value in colorSpace will be used. Choose fallback if your input is sometimes missing color space data, but when it does have color space data, that data is correct. force means to always use the value in colorSpace. Choose force if your input usually has no color space data or might have unreliable color space data.

Type: string
Required: False

VideoSelectorColorSpace (enum)

FOLLOW
REC_601
REC_709

VideoSelectorColorSpaceUsage (enum)

FALLBACK
FORCE

VideoSelectorPid

pid

Selects a specific PID from within a video source.

Type: integer
Required: False
Minimum: 0
Maximum: 8191
VideoSelectorProgramId

programId

Selects a specific program from within a multi-program transport stream. If the program doesn't exist, the first program within the transport stream will be selected by default.

Type: integer
Required: False
Minimum: 0
Maximum: 65536

VideoSelectorSettings

videoSelectorPid

Type: VideoSelectorPid (p. 463)
Required: False

videoSelectorProgramId

Type: VideoSelectorProgramId (p. 464)
Required: False

WebvttDestinationSettings

InputSecurityGroups

URI

/prod/inputSecurityGroups

HTTP Methods

GET

Operation ID: ListInputSecurityGroups

Produces a list of Input Security Groups for an account

Query Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<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>
## Responses

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>ListInputSecurityGroupsResultModel (p. 466)</td>
<td>An array of Input Security Groups</td>
</tr>
<tr>
<td>400</td>
<td>InvalidRequest (p. 466)</td>
<td>This request was invalid.</td>
</tr>
<tr>
<td>500</td>
<td>InternalServiceError (p. 467)</td>
<td>Unexpected internal service error.</td>
</tr>
<tr>
<td>502</td>
<td>BadGatewayException (p. 467)</td>
<td>Bad Gateway Error</td>
</tr>
<tr>
<td>403</td>
<td>AccessDenied (p. 466)</td>
<td>You do not have permission to list channels.</td>
</tr>
<tr>
<td>504</td>
<td>GatewayTimeoutException (p. 467)</td>
<td>Gateway Timeout Error</td>
</tr>
<tr>
<td>429</td>
<td>LimitExceeded (p. 467)</td>
<td>Request limit exceeded on list channel calls to channel service.</td>
</tr>
</tbody>
</table>

### POST

Operation ID: CreateInputSecurityGroup

Creates a Input Security Group

#### Responses

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>400</td>
<td>InvalidRequest (p. 466)</td>
<td>This request was invalid.</td>
</tr>
<tr>
<td>500</td>
<td>InternalServiceError (p. 467)</td>
<td>Unexpected internal service error.</td>
</tr>
<tr>
<td>502</td>
<td>BadGatewayException (p. 467)</td>
<td>Bad Gateway Error</td>
</tr>
<tr>
<td>403</td>
<td>AccessDenied (p. 466)</td>
<td>You do not have permission to list channels.</td>
</tr>
<tr>
<td>504</td>
<td>GatewayTimeoutException (p. 467)</td>
<td>Gateway Timeout Error</td>
</tr>
<tr>
<td>429</td>
<td>LimitExceeded (p. 467)</td>
<td>Request limit exceeded on list channel calls to channel service.</td>
</tr>
</tbody>
</table>

### Schemas

#### Request Bodies

#### Example POST

```
{
  "whitelistRules (p. 468)": [
```
Schemas

Example ListInputSecurityGroupsResultModel

```json
{
  "inputSecurityGroups (p. 469)": [
    {
      "inputs (p. 468)": [
        "string"
      ],
      "state (p. 468)": enum,
      "id (p. 468)": "string",
      "arn (p. 468)": "string",
      "whitelistRules (p. 468)": [
        {
          "cidr (p. 469)": "string"
        }
      ],
      "nextToken (p. 470)": "string"
    }
  ]
}
```

Example CreateInputSecurityGroupResultModel

```json
{
  "securityGroup (p. 467)": {
    "inputs (p. 468)": [
      "string"
    ],
    "state (p. 468)": enum,
    "id (p. 468)": "string",
    "arn (p. 468)": "string",
    "whitelistRules (p. 468)": [
      {
        "cidr (p. 469)": "string"
      }
    ]
  }
}
```

Example InvalidRequest

```json
{
  "message (p. 469)": "string"
}
```

Example AccessDenied

```json
{
  "message (p. 467)": "string"
}
```
Example LimitExceeded

```
{
  "message (p. 469)": "string"
}
```

Example InternalServiceError

```
{
  "message (p. 469)": "string"
}
```

Example BadGatewayException

```
{
  "message (p. 467)": "string"
}
```

Example GatewayTimeoutException

```
{
  "message (p. 467)": "string"
}
```

Properties

AccessDenied

message

Type: string
Required: False

BadGatewayException

message

Type: string
Required: False

CreateInputSecurityGroupResultModel

securityGroup

Type: InputSecurityGroup (p. 468)
Required: False

GatewayTimeoutException

message

Type: string
Required: False

**InputSecurityGroup**

**inputs**
The list of inputs currently using this Input Security Group.

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

**state**
The current state of the Input Security Group.

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

**id**
The Id of the Input Security Group

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

**arn**
Unique ARN of Input Security Group

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

** whitelistRules**
Whitelist rules and their sync status

**Type**: Array of type InputWhitelistRule (p. 469)
**Required**: False

**InputSecurityGroupState (enum)**

IDLE
IN_USE
UPDATING
DELETED

**InputSecurityGroupWhitelistRequest**

**whitelistRules**
List of IPv4 CIDR addresses to whitelist
Type: Array of type InputWhitelistRuleCidr (p. 469)
Required: False

InputWhitelistRule

cidr
The IPv4 CIDR that's whitelisted.
Type: string
Required: False

InputWhitelistRuleCidr

cidr
The IPv4 CIDR to whitelist
Type: string
Required: False

InternalServerError

message
Type: string
Required: False

InvalidRequest

message
Type: string
Required: False

LimitExceeded

message
Type: string
Required: False

ListInputSecurityGroupsResultModel

inputSecurityGroups
List of input security groups
Type: Array of type InputSecurityGroup (p. 468)
Required: False
nextToken

Type: string
Required: False

InputSecurityGroups inputSecurityGroupId

URI

/prod/inputSecurityGroups/ inputSecurityGroupId

HTTP Methods

GET

Operation ID: DescribeInputSecurityGroup

Produces a summary of an Input Security Group

Path Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>inputSecurityGroupId</td>
<td>String</td>
<td>True</td>
<td>The id of the Input Security Group to describe</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>InputSecurityGroup (p. 472)</td>
<td>An Input Security Group</td>
</tr>
<tr>
<td>400</td>
<td>InvalidRequest (p. 473)</td>
<td>This request was invalid.</td>
</tr>
<tr>
<td>500</td>
<td>InternalServiceError (p. 473)</td>
<td>Unexpected internal service error.</td>
</tr>
<tr>
<td>502</td>
<td>BadGatewayException (p. 474)</td>
<td>Bad Gateway Error</td>
</tr>
<tr>
<td>403</td>
<td>AccessDenied (p. 473)</td>
<td>You do not have permission to list channels.</td>
</tr>
<tr>
<td>404</td>
<td>ResourceNotFound (p. 473)</td>
<td>The channel you’re requesting to describe does not exist.</td>
</tr>
<tr>
<td>504</td>
<td>GatewayTimeoutException (p. 474)</td>
<td>Gateway Timeout Error</td>
</tr>
<tr>
<td>429</td>
<td>LimitExceeded (p. 473)</td>
<td>Request limit exceeded on list channel calls to channel service.</td>
</tr>
</tbody>
</table>

PUT

Operation ID: UpdateInputSecurityGroup

**Path Parameters**

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>inputSecurityGroupId</code></td>
<td><code>String</code></td>
<td>True</td>
<td>The id of the Input Security Group to describe</td>
</tr>
</tbody>
</table>

**Responses**

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>400</td>
<td><code>InvalidRequest (p. 473)</code></td>
<td>This request was invalid.</td>
</tr>
<tr>
<td>500</td>
<td><code>InternalServerError (p. 473)</code></td>
<td>Unexpected internal service error.</td>
</tr>
<tr>
<td>502</td>
<td><code>BadGatewayException (p. 474)</code></td>
<td>Bad Gateway Error</td>
</tr>
<tr>
<td>403</td>
<td><code>AccessDenied (p. 473)</code></td>
<td>You do not have permission to list channels.</td>
</tr>
<tr>
<td>404</td>
<td>None</td>
<td>The channel you're requesting to describe does not exist.</td>
</tr>
<tr>
<td>504</td>
<td><code>GatewayTimeoutException (p. 474)</code></td>
<td>Gateway Timeout Error</td>
</tr>
<tr>
<td>409</td>
<td><code>ResourceConflict (p. 473)</code></td>
<td>The channel is unable to create due to an issue with channel resources.</td>
</tr>
</tbody>
</table>

**DELETE**

Operation ID: `DeleteInputSecurityGroup`

Deletes an Input Security Group

**Path Parameters**

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>inputSecurityGroupId</code></td>
<td><code>String</code></td>
<td>True</td>
<td>The id of the Input Security Group to describe</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><code>Empty (p. 473)</code></td>
<td>An Input Security Group</td>
</tr>
<tr>
<td>400</td>
<td><code>InvalidRequest (p. 473)</code></td>
<td>This request was invalid.</td>
</tr>
</tbody>
</table>
### Status Codes

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>500</td>
<td>InternalServiceError</td>
<td>Unexpected internal service error.</td>
</tr>
<tr>
<td>502</td>
<td>BadGatewayException</td>
<td>Bad Gateway Error</td>
</tr>
<tr>
<td>403</td>
<td>AccessDenied</td>
<td>You do not have permission to list channels.</td>
</tr>
<tr>
<td>404</td>
<td>ResourceNotFound</td>
<td>The channel you’re requesting to describe does not exist.</td>
</tr>
<tr>
<td>504</td>
<td>GatewayTimeoutException</td>
<td>Gateway Timeout Error</td>
</tr>
<tr>
<td>429</td>
<td>LimitExceeded</td>
<td>Request limit exceeded on list channel calls to channel service.</td>
</tr>
</tbody>
</table>

### Schemas

#### Request Bodies

**Example PUT**

```json
{
    "whitelistRules (p. 475)": [
        {
            "cidr (p. 475)" : "string"
        }
    ]
}
```

#### Response Bodies

**Example InputSecurityGroup**

```json
{
    "inputs (p. 474)": [
        "string"
    ],
    "state (p. 474)" : enum,
    "id (p. 475)" : "string",
    "arn (p. 475)" : "string",
    "whitelistRules (p. 475)": [
        {
            "cidr (p. 475)" : "string"
        }
    ]
}
```

**Example UpdateInputSecurityGroupResultModel**

```json
{
    "securityGroup (p. 476)": {
        "inputs (p. 474)": [
            "string"
        ]
    }
}
```
Schemas

Example Empty

{
}

Example InvalidRequest

{  
  "message (p. 476)": "string"
}

Example AccessDenied

{  
  "message (p. 474)": "string"
}

Example ResourceNotFound

{  
  "message (p. 476)": "string"
}

Example ResourceConflict

{  
  "message (p. 476)": "string"
}

Example LimitExceeded

{  
  "message (p. 476)": "string"
}

Example InternalServiceError

{  
  "message (p. 476)": "string"
}
Example BadGatewayException

```json
{
  "message (p. 474)": "string"
}
```

Example GatewayTimeoutException

```json
{
  "message (p. 474)": "string"
}
```

Properties

**AccessDenied**

**message**

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

**BadGatewayException**

**message**

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

**Empty**

**GatewayTimeoutException**

**message**

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

**InputSecurityGroup**

**inputs**

The list of inputs currently using this Input Security Group.

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

**state**

The current state of the Input Security Group.

- **Type:** string
- **Required:** False
id
The Id of the Input Security Group

  Type: string
  Required: False

arn
Unique ARN of Input Security Group

  Type: string
  Required: False

whitelistRules
Whitelist rules and their sync status

  Type: Array of type InputWhitelistRule (p. 475)
  Required: False

InputSecurityGroupState (enum)
IDLE
IN_USE
UPDATING
DELETED

InputSecurityGroupWhitelistRequest
whitelistRules
List of IPv4 CIDR addresses to whitelist

  Type: Array of type InputWhitelistRuleCidr (p. 475)
  Required: False

InputWhitelistRule
cidr
The IPv4 CIDR that's whitelisted.

  Type: string
  Required: False

InputWhitelistRuleCidr
cidr
The IPv4 CIDR to whitelist

  Type: string
  Required: False
InternalServiceError

message

Type: string
Required: False

InvalidRequest

message

Type: string
Required: False

LimitExceeded

message

Type: string
Required: False

ResourceConflict

message

Type: string
Required: False

ResourceNotFound

message

Type: string
Required: False

UpdateInputSecurityGroupResultModel

securityGroup

Type: InputSecurityGroup (p. 474)
Required: False

Inputs

URI

/prod/inputs
HTTP Methods

GET

Operation ID: ListInputs

Produces list of inputs that have been created

Query Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<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>

Responses

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>ListInputsResultModel (p. 478)</td>
<td>An array of inputs</td>
</tr>
<tr>
<td>400</td>
<td>InvalidRequest (p. 479)</td>
<td>This request was invalid.</td>
</tr>
<tr>
<td>500</td>
<td>InternalServiceError (p. 480)</td>
<td>Unexpected internal service error.</td>
</tr>
<tr>
<td>502</td>
<td>BadGatewayException (p. 480)</td>
<td>Bad Gateway Error</td>
</tr>
<tr>
<td>403</td>
<td>AccessDenied (p. 479)</td>
<td>You do not have permission to list channels.</td>
</tr>
<tr>
<td>504</td>
<td>GatewayTimeoutException (p. 480)</td>
<td>Gateway Timeout Error</td>
</tr>
<tr>
<td>429</td>
<td>LimitExceeded (p. 479)</td>
<td>Request limit exceeded on list channel calls to channel service.</td>
</tr>
</tbody>
</table>

POST

Operation ID: CreateInput

Create an input

Responses

<table>
<thead>
<tr>
<th>Status Code</th>
<th>Response Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>201</td>
<td>CreateInputResultModel (p. 479)</td>
<td>Creation of channel is started.</td>
</tr>
<tr>
<td>400</td>
<td>InvalidRequest (p. 479)</td>
<td>This request was invalid.</td>
</tr>
<tr>
<td>500</td>
<td>InternalServiceError (p. 480)</td>
<td>Unexpected internal service error.</td>
</tr>
<tr>
<td>502</td>
<td>BadGatewayException (p. 480)</td>
<td>Bad Gateway Error</td>
</tr>
<tr>
<td>403</td>
<td>AccessDenied (p. 479)</td>
<td>You do not have permission to list channels.</td>
</tr>
</tbody>
</table>
Schemas

Request Bodies

Example POST

```json
{
    "inputSecurityGroups (p. 480)": [
        "string"
    ],
    "sources (p. 480)": [
        {
            "passwordParam (p. 484)": "string",
            "url (p. 484)": "string",
            "username (p. 484)": "string"
        }
    ],
    "requestId (p. 481)": "string",
    "destinations (p. 481)": [
        {
            "streamName (p. 483)": "string"
        }
    ],
    "name (p. 481)": "string",
    "type (p. 481)": enum
}
```

Response Bodies

Example ListInputsResultModel

```json
{
    "nextToken (p. 485)": "string",
    "inputs (p. 485)": [
        {
            "attachedChannels (p. 481)": [
                "string"
            ],
            "sources (p. 482)": [
                {
                    "passwordParam (p. 483)": "string",
                    "url (p. 483)": "string",
                    "username (p. 483)": "string"
                }
            ],
            "destinations (p. 482)": [
                {
                    "port (p. 483)": "string",
                    "ip (p. 483)": "string",
                    "url (p. 483)": "string"
                }
            ]
        }
    ]
}
```
Example CreateInputResultModel

```
{
  "input (p. 481)": {
    "attachedChannels (p. 481)": [ 
      "string"
    ],
    "sources (p. 482)": [ 
      { 
        "passwordParam (p. 483)": "string",
        "url (p. 483)": "string",
        "username (p. 483)": "string"
      }
    ],
    "destinations (p. 482)": [ 
      { 
        "port (p. 483)": "string",
        "ip (p. 483)": "string",
        "url (p. 483)": "string"
      }
    ],
    "name (p. 482)": "string",
    "securityGroups (p. 482)": [ 
      "string"
    ],
    "state (p. 482)": enum,
    "id (p. 482)": "string",
    "type (p. 482)": enum,
    "arn (p. 482)": "string"
  }
}
```

Example InvalidRequest

```
{
  "message (p. 484)": "string"
}
```

Example AccessDenied

```
{
  "message (p. 480)": "string"
}
```

Example LimitExceeded

```
{
}
```
Example InternalServiceError

```
{
    "message (p. 484)": "string"
}
```

Example BadGatewayException

```
{
    "message (p. 480)": "string"
}
```

Example GatewayTimeoutException

```
{
    "message (p. 481)": "string"
}
```

Properties

AccessDenied

message

Type: string
Required: False

BadGatewayException

message

Type: string
Required: False

CreateInput

inputSecurityGroups

A list of security groups referenced by IDs to attach to the input.

Type: Array of type string
Required: False

sources

settings required for PULL-type inputs; one per redundancy group Only one of sources and destinations can be specified

Type: string
Required: False
Properties

- **Type**: Array of type InputSourceRequest (p. 484)
  - **Required**: False

- **requestId**
  Unique identifier of the request to ensure the request is handled exactly once in case of retries
  - **Type**: string
  - **Required**: False

- **destinations**
  settings required for PUSH-type inputs; one per redundancy group. Only one of sources and destinations can be specified. Note: there are currently no settings required for PUSH-type inputs
  - **Type**: Array of type InputDestinationRequest (p. 483)
  - **Required**: False

- **name**
  Name of the input.
  - **Type**: string
  - **Required**: False

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

CreateInputResultModel

- **input**
  - **Type**: Input (p. 481)
  - **Required**: False

GatewayTimeoutException

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

Input

- **attachedChannels**
  List of channel IDs that that input is attached to (currently an input can only be attached to one channel)
  - **Type**: Array of type string
**Required**: False

**sources**
List of sources of input (PULL-type)

Type: Array of type `InputSource (p. 483)`
Required: False

**destinations**
List of destinations of input (PULL-type)

Type: Array of type `InputDestination (p. 483)`
Required: False

**name**
user-assigned name (mutable)

Type: string
Required: False

**securityGroups**
List of IDs for all the security groups attached to the input.

Type: Array of type string
Required: False

**state**

Type: string
Required: False

**id**
generated ID of input (unique for user account, immutable)

Type: string
Required: False

**type**

Type: string
Required: False

**arn**
Unique ARN of input (generated, immutable)

Type: string
Required: False
**InputDestination**

**port**

port for input

*Type: string*

*Required: False*

**ip**

system-generated static IP address of endpoint. Remains fixed for the lifetime of the input

*Type: string*

*Required: False*

**url**

This represents the endpoint that the customer stream will be pushed to.

*Type: string*

*Required: False*

**InputDestinationRequest**

**streamName**

A unique name for the location the RTMP stream is being pushed to.

*Type: string*

*Required: False*

**InputSource**

**passwordParam**

key used to extract the password from EC2 Parameter store

*Type: string*

*Required: False*

**url**

This represents the customer’s source URL where stream is pulled from.

*Type: string*

*Required: False*

**username**

username for input source

*Type: string*

*Required: False*
InputSourceRequest

passwordParam
key used to extract the password from EC2 Parameter store
  Type: string
  Required: False

url
This represents the customer's source URL where stream is pulled from.
  Type: string
  Required: False

username
username for input source
  Type: string
  Required: False

InputState (enum)
  CREATING
  DETACHED
  ATTACHED
  DELETING
  DELETED

InputType (enum)
  UDP_PUSH
  RTP_PUSH
  RTMP_PUSH
  RTMP_PULL
  URL_PULL

InternalServiceError

message
  Type: string
  Required: False

InvalidRequest

message
  Type: string
**Inputs inputId**

**URI**

/prod/inputs/ inputId

**HTTP Methods**

**GET**

Operation ID: DescribeInput

Produces details about an input

**Path Parameters**

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>inputId</td>
<td>String</td>
<td>True</td>
<td>Unique ID of the input</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>Input (p. 488)</td>
<td>Input details</td>
</tr>
<tr>
<td>400</td>
<td>InvalidRequest (p. 489)</td>
<td>This request was invalid.</td>
</tr>
<tr>
<td>500</td>
<td>InternalServiceError (p. 489)</td>
<td>Unexpected internal service error.</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>502</td>
<td>BadGatewayException (p. 489)</td>
<td>Bad Gateway Error</td>
</tr>
<tr>
<td>403</td>
<td>AccessDenied (p. 489)</td>
<td>You do not have permission to list channels.</td>
</tr>
<tr>
<td>404</td>
<td>ResourceNotFound (p. 489)</td>
<td>The channel you're requesting to describe does not exist.</td>
</tr>
<tr>
<td>504</td>
<td>GatewayTimeoutException (p. 489)</td>
<td>Gateway Timeout Error</td>
</tr>
<tr>
<td>429</td>
<td>LimitExceeded (p. 489)</td>
<td>Request limit exceeded on list channel calls to channel service.</td>
</tr>
</tbody>
</table>

**PUT**

Operation ID: UpdateInput

Updates an input.

**Path Parameters**

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>inputId</td>
<td>String</td>
<td>True</td>
<td>Unique ID of the input</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>UpdateInputResultModel (p. 489)</td>
<td>The input update is successfully initiated.</td>
</tr>
<tr>
<td>400</td>
<td>InvalidRequest (p. 489)</td>
<td>This request was invalid.</td>
</tr>
<tr>
<td>500</td>
<td>InternalServiceError (p. 489)</td>
<td>Unexpected internal service error.</td>
</tr>
<tr>
<td>502</td>
<td>BadGatewayException (p. 489)</td>
<td>Bad Gateway Error</td>
</tr>
<tr>
<td>403</td>
<td>AccessDenied (p. 489)</td>
<td>You do not have permission to list channels.</td>
</tr>
<tr>
<td>404</td>
<td>ResourceNotFound (p. 489)</td>
<td>The channel you're requesting to describe does not exist.</td>
</tr>
<tr>
<td>504</td>
<td>GatewayTimeoutException (p. 489)</td>
<td>Gateway Timeout Error</td>
</tr>
<tr>
<td>409</td>
<td>ResourceConflict (p. 489)</td>
<td>The channel is unable to create due to an issue with channel resources.</td>
</tr>
</tbody>
</table>

**DELETE**

Operation ID: DeleteInput

Deletes the input end point
Path Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>inputId</td>
<td>String</td>
<td>True</td>
<td>Unique ID of the input</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>Empty (p. 489)</td>
<td>Successful deletion</td>
</tr>
<tr>
<td>400</td>
<td>InvalidRequest (p. 489)</td>
<td>This request was invalid.</td>
</tr>
<tr>
<td>500</td>
<td>InternalServiceError (p. 489)</td>
<td>Unexpected internal service error.</td>
</tr>
<tr>
<td>502</td>
<td>BadGatewayException (p. 489)</td>
<td>Bad Gateway Error</td>
</tr>
<tr>
<td>403</td>
<td>AccessDenied (p. 489)</td>
<td>You do not have permission to list channels.</td>
</tr>
<tr>
<td>404</td>
<td>ResourceNotFound (p. 489)</td>
<td>The channel you’re requesting to describe does not exist.</td>
</tr>
<tr>
<td>504</td>
<td>GatewayTimeoutException (p. 489)</td>
<td>Gateway Timeout Error</td>
</tr>
<tr>
<td>429</td>
<td>LimitExceeded (p. 489)</td>
<td>Request limit exceeded on list channel calls to channel service.</td>
</tr>
<tr>
<td>409</td>
<td>ResourceConflict (p. 489)</td>
<td>The channel is unable to create due to an issue with channel resources.</td>
</tr>
</tbody>
</table>

Schemas

Request Bodies

Example PUT

```json
{
  "inputSecurityGroups (p. 494)": [
    "string"
  ],
  "sources (p. 494)": [
    {
      "passwordParam (p. 492)": "string",
      "url (p. 492)": "string",
      "username (p. 493)": "string"
    }
  ],
  "destinations (p. 494)": [
    {
      "streamName (p. 492)": "string"
    }
  ],
  "name (p. 494)": "string"
}
```
Response Bodies

Example Input

```json
{
  "attachedChannels (p. 490)" : [
    "string"
  ],
  "sources (p. 490)" : [
    {
      "passwordParam (p. 492)" : "string",
      "url (p. 492)" : "string",
      "username (p. 492)" : "string"
    }
  ],
  "destinations (p. 490)" : [
    {
      "port (p. 491)" : "string",
      "ip (p. 491)" : "string",
      "url (p. 492)" : "string"
    }
  ],
  "name (p. 491)" : "string",
  "securityGroups (p. 491)" : [
    "string"
  ],
  "state (p. 491)" : enum,
  "id (p. 491)" : "string",
  "type (p. 491)" : enum,
  "arn (p. 491)" : "string"
}
```

Example UpdateInputResultModel

```json
{
  "input (p. 494)" : {
    "attachedChannels (p. 490)" : [
      "string"
    ],
    "sources (p. 490)" : [
      {
        "passwordParam (p. 492)" : "string",
        "url (p. 492)" : "string",
        "username (p. 492)" : "string"
      }
    ],
    "destinations (p. 490)" : [
      {
        "port (p. 491)" : "string",
        "ip (p. 491)" : "string",
        "url (p. 492)" : "string"
      }
    ],
    "name (p. 491)" : "string",
    "securityGroups (p. 491)" : [
      "string"
    ],
    "state (p. 491)" : enum,
    "id (p. 491)" : "string",
    "type (p. 491)" : enum,
    "arn (p. 491)" : "string"
  }
}
```
Example Empty

```json
{}
```

Example InvalidRequest

```json
{
  "message (p. 493)": "string"
}
```

Example AccessDenied

```json
{
  "message (p. 490)": "string"
}
```

Example ResourceNotFound

```json
{
  "message (p. 494)": "string"
}
```

Example ResourceConflict

```json
{
  "message (p. 493)": "string"
}
```

Example LimitExceeded

```json
{
  "message (p. 493)": "string"
}
```

Example InternalServiceError

```json
{
  "message (p. 493)": "string"
}
```

Example BadGatewayException

```json
{
  "message (p. 490)": "string"
}
```

Example GatewayTimeoutException

```json
{}
```
Properties

AccessDenied

message

Type: string
Required: False

BadGatewayException

message

Type: string
Required: False

Empty

GatewayTimeoutException

message

Type: string
Required: False

Input

attachedChannels

List of channel IDs that that input is attached to (currently an input can only be attached to one channel)

Type: Array of type string
Required: False

sources

List of sources of input (PULL-type)

Type: Array of type InputSource (p. 492)
Required: False

destinations

List of destinations of input (PULL-type)

Type: Array of type InputDestination (p. 491)
Required: False
name
user-assigned name (mutable)
  Type: string
  Required: False

securityGroups
List of IDs for all the security groups attached to the input.
  Type: Array of type string
  Required: False

state
  Type: string
  Required: False

id
generated ID of input (unique for user account, immutable)
  Type: string
  Required: False

type
  Type: string
  Required: False

arn
Unique ARN of input (generated, immutable)
  Type: string
  Required: False

InputDestination

port
port for input
  Type: string
  Required: False

ip
system-generated static IP address of endpoint. Remains fixed for the lifetime of the input
  Type: string
  Required: False
url

This represents the endpoint that the customer stream will be pushed to.

  Type: string
  Required: False

InputDestinationRequest

streamName

A unique name for the location the RTMP stream is being pushed to.

  Type: string
  Required: False

InputSource

passwordParam

key used to extract the password from EC2 Parameter store

  Type: string
  Required: False

url

This represents the customer's source URL where stream is pulled from.

  Type: string
  Required: False

username

username for input source

  Type: string
  Required: False

InputSourceRequest

passwordParam

key used to extract the password from EC2 Parameter store

  Type: string
  Required: False

url

This represents the customer's source URL where stream is pulled from.

  Type: string
Required: False

**username**

username for input source

Type: string
Required: False

**InputState (enum)**

CREATING
DETACHED
ATTACHED
DELETING
DELETED

**InputType (enum)**

UDP_PUSH
RTP_PUSH
RTMP_PUSH
RTMP_PULL
URL_PULL

**InternalServiceError**

message

Type: string
Required: False

**InvalidRequest**

message

Type: string
Required: False

**LimitExceeded**

message

Type: string
Required: False

**ResourceConflict**

message

Type: string

493
**Required**: False

**ResourceNotFound**

**message**

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

**UpdateInput**

**inputSecurityGroups**

A list of security groups referenced by IDs to attach to the input.

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

**sources**

The source URLs for a PULL-type input. Every PULL type input needs exactly two source URLs for redundancy. Only specify sources for PULL type Inputs. Leave Destinations empty.

**Type**: Array of type `InputSourceRequest (p. 492)`  
**Required**: False

**destinations**

Destination settings for PUSH type inputs.

**Type**: Array of type `InputDestinationRequest (p. 492)`  
**Required**: False

**name**

Name of the input.

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

**UpdateInputResultModel**

**input**

**Type**: `Input (p. 490)`  
**Required**: False
Document History

The following table describes important changes to this documentation.

- **API version: latest**

<table>
<thead>
<tr>
<th>Change</th>
<th>Description</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>New AWS Elemental MediaLive service release</td>
<td>Initial documentation for the AWS Elemental MediaLive 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.