aws

API Reference

AWS Elemental MediaLive API Reference

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

AWS Elemental MediaLive API Reference: API Reference

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

Table of Contents

What Is the AWS Elemental MediaLive API? 1
Resources 2
Batch action: delete 3
URI 3
HTTP methods 4
Schemas 4
Properties 7
See also 11
Batch action: start 11
URI 11
HTTP methods 12
Schemas 12
Properties 15
See also 18
Batch action: stop 19
URI 19
HTTP methods 19
Schemas 20
Properties 22
See also 26
Channels 26
URI 26
HTTP methods 26
Schemas 28
Properties 80
See also 294
Channels: channel ID 295
URI 295
HTTP methods 295
Schemas 298
Properties 367
See also 577
Channels: class 578
URI 578

HTTP methods ... 578
Schemas ... 579
Properties ... 604
See also ... 812
Channels: schedule ... 812
URI ... 812
HTTP methods ... 813
Schemas ... 815
Properties ... 829
See also ... 855
Channels: start ... 857
URI ... 857
HTTP methods ... 857
Schemas ... 858
Properties ... 882
See also ... 1089
Channels: stop ... 1089
URI ... 1089
HTTP methods ... 1090
Schemas ... 1090
Properties ... 1115
See also ... 1321
Input devices ... 1321
URI ... 1321
HTTP methods ... 1322
Schemas ... 1322
Properties ... 1325
See also ... 1335
Input devices: accept device transfer ... 1335
URI ... 1335
HTTP methods ... 1336
Schemas ... 1336
Properties ... 1338
See also ... 1340
Input devices: cancel device transfer ... 1341
URI ... 1341

HTTP methods ... 1341
Schemas ... 1342
Properties ... 1343
See also ... 1345
Input devices: claim device ... 1346
URI ... 1346
HTTP methods ... 1346
Schemas ... 1347
Properties ... 1348
See also ... 1350
Input devices: device ID ... 1351
URI ... 1351
HTTP methods ... 1351
Schemas ... 1353
Properties ... 1356
See also ... 1368
Input devices: input device transfers ... 1369
URI ... 1369
HTTP methods ... 1369
Schemas ... 1370
Properties ... 1371
See also ... 1374
Input devices: reboot ... 1375
URI ... 1375
HTTP methods ... 1375
Schemas ... 1376
Properties ... 1378
See also ... 1380
Input devices: reject device transfer ... 1380
URI ... 1380
HTTP methods ... 1381
Schemas ... 1381
Properties ... 1383
See also ... 1385
Input devices: thumbnail data ... 1386
URI ... 1386

HTTP methods ... 1386
Schemas ... 1387
Properties ... 1388
See also ... 1390
Input devices: transfer device ... 1390
URI ... 1390
HTTP methods ... 1390
Schemas ... 1391
Properties ... 1393
See also ... 1396
Input devices: update ... 1396
URI ... 1396
HTTP methods ... 1396
Schemas ... 1397
Properties ... 1399
See also ... 1401
Input security groups ... 1401
URI ... 1401
HTTP methods ... 1401
Schemas ... 1403
Properties ... 1405
See also ... 1410
Input security groups: group ID ... 1410
URI ... 1410
HTTP methods ... 1411
Schemas ... 1413
Properties ... 1416
See also ... 1420
Inputs ... 1422
URI ... 1422
HTTP methods ... 1422
Schemas ... 1423
Properties ... 1428
See also ... 1440
Inputs: input ID ... 1441
URI ... 1441

HTTP methods ... 1441
Schemas ... 1444
Properties ... 1449
See also ... 1460
Inputs: partners ... 1461
URI ... 1461
HTTP methods ... 1461
Schemas ... 1462
Properties ... 1465
See also ... 1473
Multiplex: describe program ... 1474
URI ... 1474
HTTP methods ... 1474
Schemas ... 1477
Properties ... 1482
See also ... 1491
Multiplex: list programs ... 1492
URI ... 1492
HTTP methods ... 1492
Schemas ... 1494
Properties ... 1498
See also ... 1509
Multiplex: start ... 1509
URI ... 1509
HTTP methods ... 1510
Schemas ... 1510
Properties ... 1513
See also ... 1518
Multiplex: stop ... 1519
URI ... 1519
HTTP methods ... 1519
Schemas ... 1520
Properties ... 1522
See also ... 1528
Multiplexes ... 1528
URI ... 1528

HTTP methods ... 1528
Schemas ... 1530
Properties ... 1533
See also ... 1543
Multiplexes: multiplex ID ... 1544
URI ... 1544
HTTP methods ... 1544
Schemas ... 1547
Properties ... 1550
See also ... 1557
Offerings: describe offering ... 1559
URI ... 1559
HTTP methods ... 1559
Schemas ... 1560
Properties ... 1561
See also ... 1569
Offerings: list offerings ... 1569
URI ... 1569
HTTP methods ... 1569
Schemas ... 1571
Properties ... 1573
See also ... 1580
Offerings: purchase offering ... 1581
URI ... 1581
HTTP methods ... 1581
Schemas ... 1582
Properties ... 1584
See also ... 1594
Reservations: describe reservation ... 1595
URI ... 1595
HTTP methods ... 1595
Schemas ... 1598
Properties ... 1601
See also ... 1610
Reservations: list reservations ... 1612
URI ... 1612

HTTP methods ... 1612
Schemas ... 1613
Properties ... 1615
See also ... 1624
Tags: list tags ... 1625
URI ... 1625
HTTP methods ... 1625
Schemas ... 1627
Properties ... 1628
See also ... 1629
Workflow monitor: CloudWatch alarm template groups ... 1630
URI ... 1630
HTTP methods ... 1631
Schemas ... 1633
Properties ... 1635
See also ... 1641
Workflow monitor: CloudWatch alarm template groups ID ... 1642
URI ... 1642
HTTP methods ... 1642
Schemas ... 1645
Properties ... 1647
See also ... 1652
Workflow monitor: CloudWatch alarm templates ... 1654
URI ... 1654
HTTP methods ... 1654
Schemas ... 1656
Properties ... 1659
See also ... 1673
Workflow monitor: CloudWatch alarm templates ID ... 1674
URI ... 1674
HTTP methods ... 1674
Schemas ... 1678
Properties ... 1680
See also ... 1694
Workflow monitor: EventBridge rule template groups ... 1695
URI ... 1695

HTTP methods ... 1695
Schemas ... 1698
Properties ... 1700
See also ... 1705
Workflow monitor: EventBridge rule template groups ID ... 1707
URI ... 1707
HTTP methods ... 1707
Schemas ... 1710
Properties ... 1712
See also ... 1717
Workflow monitor: EventBridge rule templates ... 1719
URI ... 1719
HTTP methods ... 1719
Schemas ... 1721
Properties ... 1723
See also ... 1732
Workflow monitor: EventBridge rule templates ID ... 1733
URI ... 1733
HTTP methods ... 1733
Schemas ... 1737
Properties ... 1739
See also ... 1747
Workflow monitor: Signal map monitor deployment ... 1749
URI ... 1749
HTTP methods ... 1749
Schemas ... 1751
Properties ... 1754
See also ... 1767
Workflow monitor: Signal maps ... 1769
URI ... 1769
HTTP methods ... 1769
Schemas ... 1771
Properties ... 1774
See also ... 1787
Workflow monitor: Signal maps ID ... 1788
URI ... 1788

HTTP methods 1788
Schemas 1791
Properties 1794
See also 1809
Document History 1811
AWS Glossary 1819

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 you have the IAM permissions that you need 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 *AWS Elemental MediaLive User Guide*.

Resources

The AWS Elemental MediaLive REST API includes the following resources.

Topics

- Batch action: delete
- Batch action: start
- Batch action: stop
- Channels
- Channels: channel ID
- Channels: class
- Channels: schedule
- Channels: start
- Channels: stop
- Input devices
- Input devices: accept device transfer
- Input devices: cancel device transfer
- Input devices: claim device
- Input devices: device ID
- Input devices: input device transfers
- Input devices: reboot
- Input devices: reject device transfer
- Input devices: thumbnail data
- Input devices: transfer device
- Input devices: update
- Input security groups
- Input security groups: group ID
- Inputs
- Inputs: input ID

- Inputs: partners
- Multiplex: describe program
- Multiplex: list programs
- Multiplex: start
- Multiplex: stop
- Multiplexes
- Multiplexes: multiplex ID
- Offerings: describe offering
- Offerings: list offerings
- Offerings: purchase offering
- Reservations: describe reservation
- Reservations: list reservations
- Tags: list tags
- Workflow monitor: CloudWatch alarm template groups
- Workflow monitor: CloudWatch alarm template groups ID
- Workflow monitor: CloudWatch alarm templates
- Workflow monitor: CloudWatch alarm templates ID
- Workflow monitor: EventBridge rule template groups
- Workflow monitor: EventBridge rule template groups ID
- Workflow monitor: EventBridge rule templates
- Workflow monitor: EventBridge rule templates ID
- Workflow monitor: Signal map monitor deployment
- Workflow monitor: Signal maps
- Workflow monitor: Signal maps ID

Batch action: delete

URI

```
/prod/batch/delete
```

HTTP methods

POST

Operation ID: `BatchDelete`

Responses

Status code	Response model	Description
200	BatchDeleteResultModel	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
409	ResourceConflict	409 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

Schemas

Request bodies

POST schema

```
{
  "channelIds": [
    "string"
  ],
  "inputIds": [
```

```
    "string"
  ],
  "inputSecurityGroupIds": [
    "string"
  ],
  "multiplexIds": [
    "string"
  ]
}
```

Response bodies

BatchDeleteResultModel schema

```
{
  "failed": [
    {
      "arn": "string",
      "code": "string",
      "id": "string",
      "message": "string"
    }
  ],
  "successful": [
    {
      "arn": "string",
      "id": "string",
      "state": "string"
    }
  ]
}
```

InvalidRequest schema

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

AccessDenied schema

```
{
```

```
  "message": "string"
}
```

ResourceNotFound schema

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

ResourceConflict schema

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

LimitExceeded schema

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

InternalServiceError schema

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

BadGatewayException schema

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

GatewayTimeoutException schema

```
{
  "message": "string"
```

```
}
```

Properties

AccessDenied

message

Type: string
Required: False

BadGatewayException

message

Type: string
Required: False

BatchDelete

Batch delete resource request.

channelIds

List of channel IDs

Type: Array of type string
Required: False

inputIds

List of input IDs

Type: Array of type string
Required: False

inputSecurityGroupIds

List of input security group IDs

Type: Array of type string
Required: False

multiplexIds

List of multiplex IDs

Type: Array of type string
Required: False

BatchDeleteResultModel

Batch delete resource results

failed

List of failed operations

Type: Array of type BatchFailedResultModel
Required: False

successful

List of successful operations

Type: Array of type BatchSuccessfulResultModel
Required: False

BatchFailedResultModel

Details from a failed operation

arn

ARN of the resource

Type: string
Required: False

code

Error code for the failed operation

Type: string
Required: False

id

ID of the resource

Type: string
Required: False

message

Error message for the failed operation

Type: string
Required: False

BatchSuccessfulResultModel

Details from a successful operation

arn

ARN of the resource

Type: string
Required: False

id

ID of the resource

Type: string
Required: False

state

Current state of the resource

Type: string
Required: False

GatewayTimeoutException

message

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

See also

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

BatchDelete

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

Batch action: start

URI

`/prod/batch/start`

HTTP methods

POST

Operation ID: `BatchStart`

Responses

Status code	Response model	Description
200	BatchStartResultModel	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
409	ResourceConflict	409 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

Schemas

Request bodies

POST schema

```
{
  "channelIds": [
    "string"
  ],
```

```
  "multiplexIds": [
    "string"
  ]
}
```

Response bodies

BatchStartResultModel schema

```
{
  "failed": [
    {
      "arn": "string",
      "code": "string",
      "id": "string",
      "message": "string"
    }
  ],
  "successful": [
    {
      "arn": "string",
      "id": "string",
      "state": "string"
    }
  ]
}
```

InvalidRequest schema

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

AccessDenied schema

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

ResourceNotFound schema

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

ResourceConflict schema

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

LimitExceeded schema

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

InternalServiceError schema

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

BadGatewayException schema

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

GatewayTimeoutException schema

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

Properties

AccessDenied

message

Type: string
Required: False

BadGatewayException

message

Type: string
Required: False

BatchFailedResultModel

Details from a failed operation

arn

ARN of the resource

Type: string
Required: False

code

Error code for the failed operation

Type: string
Required: False

id

ID of the resource

Type: string
Required: False

message

Error message for the failed operation

Type: string
Required: False

BatchStart

Batch start resource request

channelIds

List of channel IDs

Type: Array of type string
Required: False

multiplexIds

List of multiplex IDs

Type: Array of type string
Required: False

BatchStartResultModel

Batch start resource results

failed

List of failed operations

Type: Array of type BatchFailedResultModel
Required: False

successful

List of successful operations

Type: Array of type BatchSuccessfulResultModel

Required: False

BatchSuccessfulResultModel

Details from a successful operation

arn

ARN of the resource

Type: string
Required: False

id

ID of the resource

Type: string
Required: False

state

Current state of the resource

Type: string
Required: False

GatewayTimeoutException

message

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

See also

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

BatchStart

- AWS Command Line Interface

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

Batch action: stop

URI

`/prod/batch/stop`

HTTP methods

POST

Operation ID: `BatchStop`

Responses

Status code	Response model	Description
200	BatchStopResultModel	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
409	ResourceConflict	409 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response

Status code	Response model	Description
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

Schemas

Request bodies

POST schema

```
{
  "channelIds": [
    "string"
  ],
  "multiplexIds": [
    "string"
  ]
}
```

Response bodies

BatchStopResultModel schema

```
{
  "failed": [
    {
      "arn": "string",
      "code": "string",
      "id": "string",
      "message": "string"
    }
  ],
  "successful": [
    {
      "arn": "string",
      "id": "string",
      "state": "string"
    }
```

```
  ]
}
```

InvalidRequest schema

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

AccessDenied schema

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

ResourceNotFound schema

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

ResourceConflict schema

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

LimitExceeded schema

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

InternalServiceError schema

```
{
  "message": "string"
```

```
}
```

BadGatewayException schema

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

GatewayTimeoutException schema

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

Properties

AccessDenied

message

Type: string
Required: False

BadGatewayException

message

Type: string
Required: False

BatchFailedResultModel

Details from a failed operation

arn

ARN of the resource

Type: string

Required: False

code

Error code for the failed operation

Type: string
Required: False

id

ID of the resource

Type: string
Required: False

message

Error message for the failed operation

Type: string
Required: False

BatchStop

Batch stop resource request

channelIds

List of channel IDs

Type: Array of type string
Required: False

multiplexIds

List of multiplex IDs

Type: Array of type string
Required: False

BatchStopResultModel

Batch stop resource results

failed

List of failed operations

Type: Array of type BatchFailedResultModel
Required: False

successful

List of successful operations

Type: Array of type BatchSuccessfulResultModel
Required: False

BatchSuccessfulResultModel

Details from a successful operation

arn

ARN of the resource

Type: string
Required: False

id

ID of the resource

Type: string
Required: False

state

Current state of the resource

Type: string
Required: False

GatewayTimeoutException

message

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

See also

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

BatchStop

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

Channels

URI

`/prod/channels`

HTTP methods

GET

Operation ID: `ListChannels`

Query parameters

Name	Type	Required	Description
sortOrder	String	False	
nextToken	String	False	
maxResults	String	False	
sortBy	String	False	
filterQuery	String	False	

Responses

Status code	Response model	Description
200	ListChannelsResultModel	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

POST

Operation ID: CreateChannel

Responses

Status code	Response model	Description
201	CreateChannelResultModel	201 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
409	ResourceConflict	409 response
422	ChannelConfigurationValidationError	422 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

Schemas

Request bodies

POST schema

```
{
  "cdiInputSpecification": {
    "resolution": enum
  },
  "channelClass": enum,
  "destinations": [
    {
      "id": "string",
      "mediaPackageSettings": [
        {
```

```
          "channelId": "string"
        }
      ],
      "multiplexSettings": {
        "multiplexId": "string",
        "programName": "string"
      },
      "settings": [
        {
          "passwordParam": "string",
          "streamName": "string",
          "url": "string",
          "username": "string"
        }
      ]
    }
  ],
  "encoderSettings": {
    "audioDescriptions": [
      {
        "audioNormalizationSettings": {
          "algorithm": enum,
          "algorithmControl": enum,
          "targetLkfs": number
        },
        "audioSelectorName": "string",
        "audioType": enum,
        "audioTypeControl": enum,
        "audioWatermarkingSettings": {
          "nielsenWatermarksSettings": {
            "nielsenCbetSettings": {
              "cbetCheckDigitString": "string",
              "cbetStepaside": enum,
              "csid": "string"
            },
            "nielsenDistributionType": enum,
            "nielsenNaesIiNwSettings": {
              "checkDigitString": "string",
              "sid": number
            }
          }
        },
        "codecSettings": {
          "aacSettings": {
```

```
            "bitrate": number,
            "codingMode": enum,
            "inputType": enum,
            "profile": enum,
            "rateControlMode": enum,
            "rawFormat": enum,
            "sampleRate": number,
            "spec": enum,
            "vbrQuality": enum
          },
          "ac3Settings": {
            "bitrate": number,
            "bitstreamMode": enum,
            "codingMode": enum,
            "dialnorm": integer,
            "drcProfile": enum,
            "lfeFilter": enum,
            "metadataControl": enum
          },
          "eac3Settings": {
            "attenuationControl": enum,
            "bitrate": number,
            "bitstreamMode": enum,
            "codingMode": enum,
            "dcFilter": enum,
            "dialnorm": integer,
            "drcLine": enum,
            "drcRf": enum,
            "lfeControl": enum,
            "lfeFilter": enum,
            "loRoCenterMixLevel": number,
            "loRoSurroundMixLevel": number,
            "ltRtCenterMixLevel": number,
            "ltRtSurroundMixLevel": number,
            "metadataControl": enum,
            "passthroughControl": enum,
            "phaseControl": enum,
            "stereoDownmix": enum,
            "surroundExMode": enum,
            "surroundMode": enum
          },
          "mp2Settings": {
            "bitrate": number,
            "codingMode": enum,
```

```
            "sampleRate": number
          },
          "passThroughSettings": {
          },
          "wavSettings": {
            "bitDepth": number,
            "codingMode": enum,
            "sampleRate": number
          }
        },
        "languageCode": "string",
        "languageCodeControl": enum,
        "name": "string",
        "remixSettings": {
          "channelMappings": [
            {
              "inputChannelLevels": [
                {
                  "gain": integer,
                  "inputChannel": integer
                }
              ],
              "outputChannel": integer
            }
          ],
          "channelsIn": integer,
          "channelsOut": integer
        },
        "streamName": "string"
      }
    ],
    "availBlanking": {
      "availBlankingImage": {
        "passwordParam": "string",
        "uri": "string",
        "username": "string"
      },
      "state": enum
    },
    "availConfiguration": {
      "availSettings": {
        "scte35SpliceInsert": {
          "adAvailOffset": integer,
          "noRegionalBlackoutFlag": enum,
```

```
          "webDeliveryAllowedFlag": enum
        },
        "scte35TimeSignalApos": {
          "adAvailOffset": integer,
          "noRegionalBlackoutFlag": enum,
          "webDeliveryAllowedFlag": enum
        }
      }
    },
    "blackoutSlate": {
      "blackoutSlateImage": {
        "passwordParam": "string",
        "uri": "string",
        "username": "string"
      },
      "networkEndBlackout": enum,
      "networkEndBlackoutImage": {
        "passwordParam": "string",
        "uri": "string",
        "username": "string"
      },
      "networkId": "string",
      "state": enum
    },
    "captionDescriptions": [
      {
        "captionSelectorName": "string",
        "destinationSettings": {
          "aribDestinationSettings": {
          },
          "burnInDestinationSettings": {
            "alignment": enum,
            "backgroundColor": enum,
            "backgroundOpacity": integer,
            "font": {
              "passwordParam": "string",
              "uri": "string",
              "username": "string"
            },
            "fontColor": enum,
            "fontOpacity": integer,
            "fontResolution": integer,
            "fontSize": "string",
            "outlineColor": enum,
```

```
            "outlineSize": integer,
            "shadowColor": enum,
            "shadowOpacity": integer,
            "shadowXOffset": integer,
            "shadowYOffset": integer,
            "teletextGridControl": enum,
            "xPosition": integer,
            "yPosition": integer
          },
          "dvbSubDestinationSettings": {
            "alignment": enum,
            "backgroundColor": enum,
            "backgroundOpacity": integer,
            "font": {
              "passwordParam": "string",
              "uri": "string",
              "username": "string"
            },
            "fontColor": enum,
            "fontOpacity": integer,
            "fontResolution": integer,
            "fontSize": "string",
            "outlineColor": enum,
            "outlineSize": integer,
            "shadowColor": enum,
            "shadowOpacity": integer,
            "shadowXOffset": integer,
            "shadowYOffset": integer,
            "teletextGridControl": enum,
            "xPosition": integer,
            "yPosition": integer
          },
          "ebuTtDDestinationSettings": {
            "copyrightHolder": "string",
            "fillLineGap": enum,
            "fontFamily": "string",
            "styleControl": enum
          },
          "embeddedDestinationSettings": {
          },
          "embeddedPlusScte20DestinationSettings": {
          },
          "rtmpCaptionInfoDestinationSettings": {
          },
```

```
          "scte20PlusEmbeddedDestinationSettings": {
          },
          "scte27DestinationSettings": {
          },
          "smpteTtDestinationSettings": {
          },
          "teletextDestinationSettings": {
          },
          "ttmlDestinationSettings": {
            "styleControl": enum
          },
          "webvttDestinationSettings": {
            "styleControl": enum
          }
        },
        "languageCode": "string",
        "languageDescription": "string",
        "name": "string"
      }
    ],
    "featureActivations": {
      "inputPrepareScheduleActions": enum
    },
    "globalConfiguration": {
      "initialAudioGain": integer,
      "inputEndAction": enum,
      "inputLossBehavior": {
        "blackFrameMsec": integer,
        "inputLossImageColor": "string",
        "inputLossImageSlate": {
          "passwordParam": "string",
          "uri": "string",
          "username": "string"
        },
        "inputLossImageType": enum,
        "repeatFrameMsec": integer
      },
      "outputLockingMode": enum,
      "outputTimingSource": enum,
      "supportLowFramerateInputs": enum
    },
    "motionGraphicsConfiguration": {
      "motionGraphicsInsertion": enum,
      "motionGraphicsSettings": {
```

```
        "htmlMotionGraphicsSettings": {
        }
      }
    },
    "nielsenConfiguration": {
      "distributorId": "string",
      "nielsenPcmToId3Tagging": enum
    },
    "outputGroups": [
      {
        "name": "string",
        "outputGroupSettings": {
          "archiveGroupSettings": {
            "archiveCdnSettings": {
              "archiveS3Settings": {
                "cannedAcl": enum,
                "logUploads": enum
              }
            },
            "destination": {
              "destinationRefId": "string"
            },
            "rolloverInterval": integer
          },
          "frameCaptureGroupSettings": {
            "destination": {
              "destinationRefId": "string"
            },
            "frameCaptureCdnSettings": {
              "frameCaptureS3Settings": {
                "cannedAcl": enum,
                "logUploads": enum
              }
            }
          },
          "hlsGroupSettings": {
            "adMarkers": [
              enum
            ],
            "baseUrlContent": "string",
            "baseUrlContent1": "string",
            "baseUrlManifest": "string",
            "baseUrlManifest1": "string",
            "captionLanguageMappings": [
```

```
        {
          "captionChannel": integer,
          "languageCode": "string",
          "languageDescription": "string"
        }
      ],
      "captionLanguageSetting": enum,
      "clientCache": enum,
      "codecSpecification": enum,
      "constantIv": "string",
      "destination": {
        "destinationRefId": "string"
      },
      "directoryStructure": enum,
      "discontinuityTags": enum,
      "encryptionType": enum,
      "hlsCdnSettings": {
        "hlsAkamaiSettings": {
          "connectionRetryInterval": integer,
          "filecacheDuration": integer,
          "httpTransferMode": enum,
          "numRetries": integer,
          "restartDelay": integer,
          "salt": "string",
          "token": "string"
        },
        "hlsBasicPutSettings": {
          "connectionRetryInterval": integer,
          "filecacheDuration": integer,
          "numRetries": integer,
          "restartDelay": integer
        },
        "hlsMediaStoreSettings": {
          "connectionRetryInterval": integer,
          "filecacheDuration": integer,
          "mediaStoreStorageClass": enum,
          "numRetries": integer,
          "restartDelay": integer
        },
        "hlsS3Settings": {
          "cannedAcl": enum,
          "logUploads": enum
        },
        "hlsWebdavSettings": {
```

```
            "connectionRetryInterval": integer,
            "filecacheDuration": integer,
            "httpTransferMode": enum,
            "numRetries": integer,
            "restartDelay": integer
          }
        },
        "hlsId3SegmentTagging": enum,
        "iFrameOnlyPlaylists": enum,
        "incompleteSegmentBehavior": enum,
        "indexNSegments": integer,
        "inputLossAction": enum,
        "ivInManifest": enum,
        "ivSource": enum,
        "keepSegments": integer,
        "keyFormat": "string",
        "keyFormatVersions": "string",
        "keyProviderSettings": {
          "staticKeySettings": {
            "keyProviderServer": {
              "passwordParam": "string",
              "uri": "string",
              "username": "string"
            },
            "staticKeyValue": "string"
          }
        },
        "manifestCompression": enum,
        "manifestDurationFormat": enum,
        "minSegmentLength": integer,
        "mode": enum,
        "outputSelection": enum,
        "programDateTime": enum,
        "programDateTimeClock": enum,
        "programDateTimePeriod": integer,
        "redundantManifest": enum,
        "segmentLength": integer,
        "segmentationMode": enum,
        "segmentsPerSubdirectory": integer,
        "streamInfResolution": enum,
        "timedMetadataId3Frame": enum,
        "timedMetadataId3Period": integer,
        "timestampDeltaMilliseconds": integer,
        "tsFileMode": enum
```

```
        },
        "mediaPackageGroupSettings": {
          "destination": {
            "destinationRefId": "string"
          }
        },
        "msSmoothGroupSettings": {
          "acquisitionPointId": "string",
          "audioOnlyTimecodeControl": enum,
          "certificateMode": enum,
          "connectionRetryInterval": integer,
          "destination": {
            "destinationRefId": "string"
          },
          "eventId": "string",
          "eventIdMode": enum,
          "eventStopBehavior": enum,
          "filecacheDuration": integer,
          "fragmentLength": integer,
          "inputLossAction": enum,
          "numRetries": integer,
          "restartDelay": integer,
          "segmentationMode": enum,
          "sendDelayMs": integer,
          "sparseTrackType": enum,
          "streamManifestBehavior": enum,
          "timestampOffset": "string",
          "timestampOffsetMode": enum
        },
        "multiplexGroupSettings": {
        },
        "rtmpGroupSettings": {
          "adMarkers": [
            enum
          ],
          "authenticationScheme": enum,
          "cacheFullBehavior": enum,
          "cacheLength": integer,
          "captionData": enum,
          "inputLossAction": enum,
          "restartDelay": integer
        },
        "udpGroupSettings": {
          "inputLossAction": enum,
```

```
            "timedMetadataId3Frame": enum,
            "timedMetadataId3Period": integer
          }
        },
        "outputs": [
          {
            "audioDescriptionNames": [
              "string"
            ],
            "captionDescriptionNames": [
              "string"
            ],
            "outputName": "string",
            "outputSettings": {
              "archiveOutputSettings": {
                "containerSettings": {
                  "m2tsSettings": {
                    "absentInputAudioBehavior": enum,
                    "arib": enum,
                    "aribCaptionsPid": "string",
                    "aribCaptionsPidControl": enum,
                    "audioBufferModel": enum,
                    "audioFramesPerPes": integer,
                    "audioPids": "string",
                    "audioStreamType": enum,
                    "bitrate": integer,
                    "bufferModel": enum,
                    "ccDescriptor": enum,
                    "dvbNitSettings": {
                      "networkId": integer,
                      "networkName": "string",
                      "repInterval": integer
                    },
                    "dvbSdtSettings": {
                      "outputSdt": enum,
                      "repInterval": integer,
                      "serviceName": "string",
                      "serviceProviderName": "string"
                    },
                    "dvbSubPids": "string",
                    "dvbTdtSettings": {
                      "repInterval": integer
                    },
                    "dvbTeletextPid": "string",
```

```
                    "ebif": enum,
                    "ebpAudioInterval": enum,
                    "ebpLookaheadMs": integer,
                    "ebpPlacement": enum,
                    "ecmPid": "string",
                    "esRateInPes": enum,
                    "etvPlatformPid": "string",
                    "etvSignalPid": "string",
                    "fragmentTime": number,
                    "klv": enum,
                    "klvDataPids": "string",
                    "nielsenId3Behavior": enum,
                    "nullPacketBitrate": number,
                    "patInterval": integer,
                    "pcrControl": enum,
                    "pcrPeriod": integer,
                    "pcrPid": "string",
                    "pmtInterval": integer,
                    "pmtPid": "string",
                    "programNum": integer,
                    "rateMode": enum,
                    "scte27Pids": "string",
                    "scte35Control": enum,
                    "scte35Pid": "string",
                    "segmentationMarkers": enum,
                    "segmentationStyle": enum,
                    "segmentationTime": number,
                    "timedMetadataBehavior": enum,
                    "timedMetadataPid": "string",
                    "transportStreamId": integer,
                    "videoPid": "string"
                  },
                  "rawSettings": {
                  }
                },
                "extension": "string",
                "nameModifier": "string"
              },
              "frameCaptureOutputSettings": {
                "nameModifier": "string"
              },
              "hlsOutputSettings": {
                "h265PackagingType": enum,
                "hlsSettings": {
```

```
            "audioOnlyHlsSettings": {
              "audioGroupId": "string",
              "audioOnlyImage": {
                "passwordParam": "string",
                "uri": "string",
                "username": "string"
              },
              "audioTrackType": enum,
              "segmentType": enum
            },
            "fmp4HlsSettings": {
              "audioRenditionSets": "string",
              "nielsenId3Behavior": enum,
              "timedMetadataBehavior": enum
            },
            "frameCaptureHlsSettings": {
            },
            "standardHlsSettings": {
              "audioRenditionSets": "string",
              "m3u8Settings": {
                "audioFramesPerPes": integer,
                "audioPids": "string",
                "ecmPid": "string",
                "nielsenId3Behavior": enum,
                "patInterval": integer,
                "pcrControl": enum,
                "pcrPeriod": integer,
                "pcrPid": "string",
                "pmtInterval": integer,
                "pmtPid": "string",
                "programNum": integer,
                "scte35Behavior": enum,
                "scte35Pid": "string",
                "timedMetadataBehavior": enum,
                "timedMetadataPid": "string",
                "transportStreamId": integer,
                "videoPid": "string"
              }
            }
          },
          "nameModifier": "string",
          "segmentModifier": "string"
        },
        "mediaPackageOutputSettings": {
```

```
        },
        "msSmoothOutputSettings": {
          "h265PackagingType": enum,
          "nameModifier": "string"
        },
        "multiplexOutputSettings": {
          "destination": {
            "destinationRefId": "string"
          }
        },
        "rtmpOutputSettings": {
          "certificateMode": enum,
          "connectionRetryInterval": integer,
          "destination": {
            "destinationRefId": "string"
          },
          "numRetries": integer
        },
        "udpOutputSettings": {
          "bufferMsec": integer,
          "containerSettings": {
            "m2tsSettings": {
              "absentInputAudioBehavior": enum,
              "arib": enum,
              "aribCaptionsPid": "string",
              "aribCaptionsPidControl": enum,
              "audioBufferModel": enum,
              "audioFramesPerPes": integer,
              "audioPids": "string",
              "audioStreamType": enum,
              "bitrate": integer,
              "bufferModel": enum,
              "ccDescriptor": enum,
              "dvbNitSettings": {
                "networkId": integer,
                "networkName": "string",
                "repInterval": integer
              },
              "dvbSdtSettings": {
                "outputSdt": enum,
                "repInterval": integer,
                "serviceName": "string",
                "serviceProviderName": "string"
              },
```

```
            "dvbSubPids": "string",
            "dvbTdtSettings": {
              "repInterval": integer
            },
            "dvbTeletextPid": "string",
            "ebif": enum,
            "ebpAudioInterval": enum,
            "ebpLookaheadMs": integer,
            "ebpPlacement": enum,
            "ecmPid": "string",
            "esRateInPes": enum,
            "etvPlatformPid": "string",
            "etvSignalPid": "string",
            "fragmentTime": number,
            "klv": enum,
            "klvDataPids": "string",
            "nielsenId3Behavior": enum,
            "nullPacketBitrate": number,
            "patInterval": integer,
            "pcrControl": enum,
            "pcrPeriod": integer,
            "pcrPid": "string",
            "pmtInterval": integer,
            "pmtPid": "string",
            "programNum": integer,
            "rateMode": enum,
            "scte27Pids": "string",
            "scte35Control": enum,
            "scte35Pid": "string",
            "segmentationMarkers": enum,
            "segmentationStyle": enum,
            "segmentationTime": number,
            "timedMetadataBehavior": enum,
            "timedMetadataPid": "string",
            "transportStreamId": integer,
            "videoPid": "string"
          }
        },
        "destination": {
          "destinationRefId": "string"
        },
        "fecOutputSettings": {
          "columnDepth": integer,
          "includeFec": enum,
```

```
                "rowLength": integer
              }
            }
          },
          "videoDescriptionName": "string"
        }
      ]
    }
  ],
  "timecodeConfig": {
    "source": enum,
    "syncThreshold": integer
  },
  "videoDescriptions": [
    {
      "codecSettings": {
        "frameCaptureSettings": {
          "captureInterval": integer,
          "captureIntervalUnits": enum
        },
        "h264Settings": {
          "adaptiveQuantization": enum,
          "afdSignaling": enum,
          "bitrate": integer,
          "bufFillPct": integer,
          "bufSize": integer,
          "colorMetadata": enum,
          "colorSpaceSettings": {
            "colorSpacePassthroughSettings": {
            },
            "rec601Settings": {
            },
            "rec709Settings": {
            }
          },
          "entropyEncoding": enum,
          "filterSettings": {
            "temporalFilterSettings": {
              "postFilterSharpening": enum,
              "strength": enum
            }
          },
          "fixedAfd": enum,
          "flickerAq": enum,
```

```
        "forceFieldPictures": enum,
        "framerateControl": enum,
        "framerateDenominator": integer,
        "framerateNumerator": integer,
        "gopBReference": enum,
        "gopClosedCadence": integer,
        "gopNumBFrames": integer,
        "gopSize": number,
        "gopSizeUnits": enum,
        "level": enum,
        "lookAheadRateControl": enum,
        "maxBitrate": integer,
        "minIInterval": integer,
        "numRefFrames": integer,
        "parControl": enum,
        "parDenominator": integer,
        "parNumerator": integer,
        "profile": enum,
        "qualityLevel": enum,
        "qvbrQualityLevel": integer,
        "rateControlMode": enum,
        "scanType": enum,
        "sceneChangeDetect": enum,
        "slices": integer,
        "softness": integer,
        "spatialAq": enum,
        "subgopLength": enum,
        "syntax": enum,
        "temporalAq": enum,
        "timecodeInsertion": enum
      },
      "h265Settings": {
        "adaptiveQuantization": enum,
        "afdSignaling": enum,
        "alternativeTransferFunction": enum,
        "bitrate": integer,
        "bufSize": integer,
        "colorMetadata": enum,
        "colorSpaceSettings": {
          "colorSpacePassthroughSettings": {
          },
          "hdr10Settings": {
            "maxCll": integer,
            "maxFall": integer
```

```
            },
            "rec601Settings": {
            },
            "rec709Settings": {
            }
          },
          "filterSettings": {
            "temporalFilterSettings": {
              "postFilterSharpening": enum,
              "strength": enum
            }
          },
          "fixedAfd": enum,
          "flickerAq": enum,
          "framerateDenominator": integer,
          "framerateNumerator": integer,
          "gopClosedCadence": integer,
          "gopSize": number,
          "gopSizeUnits": enum,
          "level": enum,
          "lookAheadRateControl": enum,
          "maxBitrate": integer,
          "minIInterval": integer,
          "parDenominator": integer,
          "parNumerator": integer,
          "profile": enum,
          "qvbrQualityLevel": integer,
          "rateControlMode": enum,
          "scanType": enum,
          "sceneChangeDetect": enum,
          "slices": integer,
          "tier": enum,
          "timecodeInsertion": enum
        },
        "mpeg2Settings": {
          "adaptiveQuantization": enum,
          "afdSignaling": enum,
          "colorMetadata": enum,
          "colorSpace": enum,
          "displayAspectRatio": enum,
          "filterSettings": {
            "temporalFilterSettings": {
              "postFilterSharpening": enum,
              "strength": enum
```

```
            }
          },
          "fixedAfd": enum,
          "framerateDenominator": integer,
          "framerateNumerator": integer,
          "gopClosedCadence": integer,
          "gopNumBFrames": integer,
          "gopSize": number,
          "gopSizeUnits": enum,
          "scanType": enum,
          "subgopLength": enum,
          "timecodeInsertion": enum
        }
      },
      "height": integer,
      "name": "string",
      "respondToAfd": enum,
      "scalingBehavior": enum,
      "sharpness": integer,
      "width": integer
    }
  ]
},
"inputAttachments": [
  {
    "automaticInputFailoverSettings": {
      "errorClearTimeMsec": integer,
      "failoverConditions": [
        {
          "failoverConditionSettings": {
            "audioSilenceSettings": {
              "audioSelectorName": "string",
              "audioSilenceThresholdMsec": integer
            },
            "inputLossSettings": {
              "inputLossThresholdMsec": integer
            },
            "videoBlackSettings": {
              "blackDetectThreshold": number,
              "videoBlackThresholdMsec": integer
            }
          }
        },
      ],
```

```
        "inputPreference": enum,
        "secondaryInputId": "string"
      },
      "inputAttachmentName": "string",
      "inputId": "string",
      "inputSettings": {
        "audioSelectors": [
          {
            "name": "string",
            "selectorSettings": {
              "audioHlsRenditionSelection": {
                "groupId": "string",
                "name": "string"
              },
              "audioLanguageSelection": {
                "languageCode": "string",
                "languageSelectionPolicy": enum
              },
              "audioPidSelection": {
                "pid": integer
              },
              "audioTrackSelection": {
                "tracks": [
                  {
                    "track": integer
                  }
                ]
              }
            }
          }
        ],
        "captionSelectors": [
          {
            "languageCode": "string",
            "name": "string",
            "selectorSettings": {
              "ancillarySourceSettings": {
                "sourceAncillaryChannelNumber": integer
              },
              "aribSourceSettings": {
              },
              "dvbSubSourceSettings": {
                "ocrLanguage": enum,
                "pid": integer
```

```
            },
            "embeddedSourceSettings": {
              "convert608To708": enum,
              "scte20Detection": enum,
              "source608ChannelNumber": integer,
              "source608TrackNumber": integer
            },
            "scte20SourceSettings": {
              "convert608To708": enum,
              "source608ChannelNumber": integer
            },
            "scte27SourceSettings": {
              "ocrLanguage": enum,
              "pid": integer
            },
            "teletextSourceSettings": {
              "outputRectangle": {
                "height": number,
                "leftOffset": number,
                "topOffset": number,
                "width": number
              },
              "pageNumber": "string"
            }
          }
        }
      ],
      "deblockFilter": enum,
      "denoiseFilter": enum,
      "filterStrength": integer,
      "inputFilter": enum,
      "networkInputSettings": {
        "hlsInputSettings": {
          "bandwidth": integer,
          "bufferSegments": integer,
          "retries": integer,
          "retryInterval": integer,
          "scte35Source": enum
        },
        "serverValidation": enum
      },
      "smpte2038DataPreference": enum,
      "sourceEndBehavior": enum,
      "videoSelector": {
```

```
          "colorSpace": enum,
          "colorSpaceSettings": {
            "hdr10Settings": {
              "maxCll": integer,
              "maxFall": integer
            }
          },
          "colorSpaceUsage": enum,
          "selectorSettings": {
            "videoSelectorPid": {
              "pid": integer
            },
            "videoSelectorProgramId": {
              "programId": integer
            }
          }
        }
      }
    }
  ],
  "inputSpecification": {
    "codec": enum,
    "maximumBitrate": enum,
    "resolution": enum
  },
  "logLevel": enum,
  "maintenance": {
    "maintenanceDay": enum,
    "maintenanceStartTime": "string"
  },
  "name": "string",
  "requestId": "string",
  "reserved": "string",
  "roleArn": "string",
  "tags": {
  },
  "vpc": {
    "publicAddressAllocationIds": [
      "string"
    ],
    "securityGroupIds": [
      "string"
    ],
    "subnetIds": [
```

```
      "string"
    ]
  }
}
```

Response bodies

ListChannelsResultModel schema

```
{
  "channels": [
    {
      "arn": "string",
      "cdiInputSpecification": {
        "resolution": enum
      },
      "channelClass": enum,
      "destinations": [
        {
          "id": "string",
          "mediaPackageSettings": [
            {
              "channelId": "string"
            }
          ],
          "multiplexSettings": {
            "multiplexId": "string",
            "programName": "string"
          },
          "settings": [
            {
              "passwordParam": "string",
              "streamName": "string",
              "url": "string",
              "username": "string"
            }
          ]
        }
      ],
      "egressEndpoints": [
        {
          "sourceIp": "string"
        }
```

```
      ],
      "id": "string",
      "inputAttachments": [
        {
          "automaticInputFailoverSettings": {
            "errorClearTimeMsec": integer,
            "failoverConditions": [
              {
                "failoverConditionSettings": {
                  "audioSilenceSettings": {
                    "audioSelectorName": "string",
                    "audioSilenceThresholdMsec": integer
                  },
                  "inputLossSettings": {
                    "inputLossThresholdMsec": integer
                  },
                  "videoBlackSettings": {
                    "blackDetectThreshold": number,
                    "videoBlackThresholdMsec": integer
                  }
                }
              }
            ],
            "inputPreference": enum,
            "secondaryInputId": "string"
          },
          "inputAttachmentName": "string",
          "inputId": "string",
          "inputSettings": {
            "audioSelectors": [
              {
                "name": "string",
                "selectorSettings": {
                  "audioHlsRenditionSelection": {
                    "groupId": "string",
                    "name": "string"
                  },
                  "audioLanguageSelection": {
                    "languageCode": "string",
                    "languageSelectionPolicy": enum
                  },
                  "audioPidSelection": {
                    "pid": integer
                  },
```

```
            "audioTrackSelection": {
              "tracks": [
                {
                  "track": integer
                }
              ]
            }
          }
        }
      ],
      "captionSelectors": [
        {
          "languageCode": "string",
          "name": "string",
          "selectorSettings": {
            "ancillarySourceSettings": {
              "sourceAncillaryChannelNumber": integer
            },
            "aribSourceSettings": {
            },
            "dvbSubSourceSettings": {
              "ocrLanguage": enum,
              "pid": integer
            },
            "embeddedSourceSettings": {
              "convert608To708": enum,
              "scte20Detection": enum,
              "source608ChannelNumber": integer,
              "source608TrackNumber": integer
            },
            "scte20SourceSettings": {
              "convert608To708": enum,
              "source608ChannelNumber": integer
            },
            "scte27SourceSettings": {
              "ocrLanguage": enum,
              "pid": integer
            },
            "teletextSourceSettings": {
              "outputRectangle": {
                "height": number,
                "leftOffset": number,
                "topOffset": number,
                "width": number
```

```
                    },
                    "pageNumber": "string"
                  }
                }
              }
            ],
            "deblockFilter": enum,
            "denoiseFilter": enum,
            "filterStrength": integer,
            "inputFilter": enum,
            "networkInputSettings": {
              "hlsInputSettings": {
                "bandwidth": integer,
                "bufferSegments": integer,
                "retries": integer,
                "retryInterval": integer,
                "scte35Source": enum
              },
              "serverValidation": enum
            },
            "smpte2038DataPreference": enum,
            "sourceEndBehavior": enum,
            "videoSelector": {
              "colorSpace": enum,
              "colorSpaceSettings": {
                "hdr10Settings": {
                  "maxCll": integer,
                  "maxFall": integer
                }
              },
              "colorSpaceUsage": enum,
              "selectorSettings": {
                "videoSelectorPid": {
                  "pid": integer
                },
                "videoSelectorProgramId": {
                  "programId": integer
                }
              }
            }
          }
        ],
        "inputSpecification": {
```

```
        "codec": enum,
        "maximumBitrate": enum,
        "resolution": enum
      },
      "logLevel": enum,
      "maintenance": {
        "maintenanceDay": enum,
        "maintenanceDeadline": "string",
        "maintenanceScheduledDate": "string",
        "maintenanceStartTime": "string"
      },
      "name": "string",
      "pipelinesRunningCount": integer,
      "roleArn": "string",
      "state": enum,
      "tags": {
      },
      "vpc": {
        "availabilityZones": [
          "string"
        ],
        "networkInterfaceIds": [
          "string"
        ],
        "securityGroupIds": [
          "string"
        ],
        "subnetIds": [
          "string"
        ]
      }
    }
  ],
  "nextToken": "string"
}
```

CreateChannelResultModel schema

```
{
  "channel": {
    "arn": "string",
    "cdiInputSpecification": {
      "resolution": enum
```

```
    },
    "channelClass": enum,
    "destinations": [
      {
        "id": "string",
        "mediaPackageSettings": [
          {
            "channelId": "string"
          }
        ],
        "multiplexSettings": {
          "multiplexId": "string",
          "programName": "string"
        },
        "settings": [
          {
            "passwordParam": "string",
            "streamName": "string",
            "url": "string",
            "username": "string"
          }
        ]
      }
    ],
    "egressEndpoints": [
      {
        "sourceIp": "string"
      }
    ],
    "encoderSettings": {
      "audioDescriptions": [
        {
          "audioNormalizationSettings": {
            "algorithm": enum,
            "algorithmControl": enum,
            "targetLkfs": number
          },
          "audioSelectorName": "string",
          "audioType": enum,
          "audioTypeControl": enum,
          "audioWatermarkingSettings": {
            "nielsenWatermarksSettings": {
              "nielsenCbetSettings": {
                "cbetCheckDigitString": "string",
```

```
            "cbetStepaside": enum,
            "csid": "string"
          },
          "nielsenDistributionType": enum,
          "nielsenNaesIiNwSettings": {
            "checkDigitString": "string",
            "sid": number
          }
        }
      },
      "codecSettings": {
        "aacSettings": {
          "bitrate": number,
          "codingMode": enum,
          "inputType": enum,
          "profile": enum,
          "rateControlMode": enum,
          "rawFormat": enum,
          "sampleRate": number,
          "spec": enum,
          "vbrQuality": enum
        },
        "ac3Settings": {
          "bitrate": number,
          "bitstreamMode": enum,
          "codingMode": enum,
          "dialnorm": integer,
          "drcProfile": enum,
          "lfeFilter": enum,
          "metadataControl": enum
        },
        "eac3Settings": {
          "attenuationControl": enum,
          "bitrate": number,
          "bitstreamMode": enum,
          "codingMode": enum,
          "dcFilter": enum,
          "dialnorm": integer,
          "drcLine": enum,
          "drcRf": enum,
          "lfeControl": enum,
          "lfeFilter": enum,
          "loRoCenterMixLevel": number,
          "loRoSurroundMixLevel": number,
```

```
            "ltRtCenterMixLevel": number,
            "ltRtSurroundMixLevel": number,
            "metadataControl": enum,
            "passthroughControl": enum,
            "phaseControl": enum,
            "stereoDownmix": enum,
            "surroundExMode": enum,
            "surroundMode": enum
          },
          "mp2Settings": {
            "bitrate": number,
            "codingMode": enum,
            "sampleRate": number
          },
          "passThroughSettings": {
          },
          "wavSettings": {
            "bitDepth": number,
            "codingMode": enum,
            "sampleRate": number
          }
        },
        "languageCode": "string",
        "languageCodeControl": enum,
        "name": "string",
        "remixSettings": {
          "channelMappings": [
            {
              "inputChannelLevels": [
                {
                  "gain": integer,
                  "inputChannel": integer
                }
              ],
              "outputChannel": integer
            }
          ],
          "channelsIn": integer,
          "channelsOut": integer
        },
        "streamName": "string"
      }
    ],
    "availBlanking": {
```

```
        "availBlankingImage": {
          "passwordParam": "string",
          "uri": "string",
          "username": "string"
        },
        "state": enum
      },
      "availConfiguration": {
        "availSettings": {
          "scte35SpliceInsert": {
            "adAvailOffset": integer,
            "noRegionalBlackoutFlag": enum,
            "webDeliveryAllowedFlag": enum
          },
          "scte35TimeSignalApos": {
            "adAvailOffset": integer,
            "noRegionalBlackoutFlag": enum,
            "webDeliveryAllowedFlag": enum
          }
        }
      },
      "blackoutSlate": {
        "blackoutSlateImage": {
          "passwordParam": "string",
          "uri": "string",
          "username": "string"
        },
        "networkEndBlackout": enum,
        "networkEndBlackoutImage": {
          "passwordParam": "string",
          "uri": "string",
          "username": "string"
        },
        "networkId": "string",
        "state": enum
      },
      "captionDescriptions": [
        {
          "captionSelectorName": "string",
          "destinationSettings": {
            "aribDestinationSettings": {
            },
            "burnInDestinationSettings": {
              "alignment": enum,
```

```
        "backgroundColor": enum,
        "backgroundOpacity": integer,
        "font": {
          "passwordParam": "string",
          "uri": "string",
          "username": "string"
        },
        "fontColor": enum,
        "fontOpacity": integer,
        "fontResolution": integer,
        "fontSize": "string",
        "outlineColor": enum,
        "outlineSize": integer,
        "shadowColor": enum,
        "shadowOpacity": integer,
        "shadowXOffset": integer,
        "shadowYOffset": integer,
        "teletextGridControl": enum,
        "xPosition": integer,
        "yPosition": integer
      },
      "dvbSubDestinationSettings": {
        "alignment": enum,
        "backgroundColor": enum,
        "backgroundOpacity": integer,
        "font": {
          "passwordParam": "string",
          "uri": "string",
          "username": "string"
        },
        "fontColor": enum,
        "fontOpacity": integer,
        "fontResolution": integer,
        "fontSize": "string",
        "outlineColor": enum,
        "outlineSize": integer,
        "shadowColor": enum,
        "shadowOpacity": integer,
        "shadowXOffset": integer,
        "shadowYOffset": integer,
        "teletextGridControl": enum,
        "xPosition": integer,
        "yPosition": integer
      },
```

```
            "ebuTtDDestinationSettings": {
              "copyrightHolder": "string",
              "fillLineGap": enum,
              "fontFamily": "string",
              "styleControl": enum
            },
            "embeddedDestinationSettings": {
            },
            "embeddedPlusScte20DestinationSettings": {
            },
            "rtmpCaptionInfoDestinationSettings": {
            },
            "scte20PlusEmbeddedDestinationSettings": {
            },
            "scte27DestinationSettings": {
            },
            "smpteTtDestinationSettings": {
            },
            "teletextDestinationSettings": {
            },
            "ttmlDestinationSettings": {
              "styleControl": enum
            },
            "webvttDestinationSettings": {
              "styleControl": enum
            }
          },
          "languageCode": "string",
          "languageDescription": "string",
          "name": "string"
        }
      ],
      "featureActivations": {
        "inputPrepareScheduleActions": enum
      },
      "globalConfiguration": {
        "initialAudioGain": integer,
        "inputEndAction": enum,
        "inputLossBehavior": {
          "blackFrameMsec": integer,
          "inputLossImageColor": "string",
          "inputLossImageSlate": {
            "passwordParam": "string",
            "uri": "string",
```

```
            "username": "string"
          },
          "inputLossImageType": enum,
          "repeatFrameMsec": integer
        },
        "outputLockingMode": enum,
        "outputTimingSource": enum,
        "supportLowFramerateInputs": enum
      },
      "motionGraphicsConfiguration": {
        "motionGraphicsInsertion": enum,
        "motionGraphicsSettings": {
          "htmlMotionGraphicsSettings": {
          }
        }
      },
      "nielsenConfiguration": {
        "distributorId": "string",
        "nielsenPcmToId3Tagging": enum
      },
      "outputGroups": [
        {
          "name": "string",
          "outputGroupSettings": {
            "archiveGroupSettings": {
              "archiveCdnSettings": {
                "archiveS3Settings": {
                  "cannedAcl": enum,
                  "logUploads": enum
                }
              },
              "destination": {
                "destinationRefId": "string"
              },
              "rolloverInterval": integer
            },
            "frameCaptureGroupSettings": {
              "destination": {
                "destinationRefId": "string"
              },
              "frameCaptureCdnSettings": {
                "frameCaptureS3Settings": {
                  "cannedAcl": enum,
                  "logUploads": enum
```

```
            }
          }
        },
        "hlsGroupSettings": {
          "adMarkers": [
            enum
          ],
          "baseUrlContent": "string",
          "baseUrlContent1": "string",
          "baseUrlManifest": "string",
          "baseUrlManifest1": "string",
          "captionLanguageMappings": [
            {
              "captionChannel": integer,
              "languageCode": "string",
              "languageDescription": "string"
            }
          ],
          "captionLanguageSetting": enum,
          "clientCache": enum,
          "codecSpecification": enum,
          "constantIv": "string",
          "destination": {
            "destinationRefId": "string"
          },
          "directoryStructure": enum,
          "discontinuityTags": enum,
          "encryptionType": enum,
          "hlsCdnSettings": {
            "hlsAkamaiSettings": {
              "connectionRetryInterval": integer,
              "filecacheDuration": integer,
              "httpTransferMode": enum,
              "numRetries": integer,
              "restartDelay": integer,
              "salt": "string",
              "token": "string"
            },
            "hlsBasicPutSettings": {
              "connectionRetryInterval": integer,
              "filecacheDuration": integer,
              "numRetries": integer,
              "restartDelay": integer
            },
```

```
        "hlsMediaStoreSettings": {
          "connectionRetryInterval": integer,
          "filecacheDuration": integer,
          "mediaStoreStorageClass": enum,
          "numRetries": integer,
          "restartDelay": integer
        },
        "hlsS3Settings": {
          "cannedAcl": enum,
          "logUploads": enum
        },
        "hlsWebdavSettings": {
          "connectionRetryInterval": integer,
          "filecacheDuration": integer,
          "httpTransferMode": enum,
          "numRetries": integer,
          "restartDelay": integer
        }
      },
      "hlsId3SegmentTagging": enum,
      "iFrameOnlyPlaylists": enum,
      "incompleteSegmentBehavior": enum,
      "indexNSegments": integer,
      "inputLossAction": enum,
      "ivInManifest": enum,
      "ivSource": enum,
      "keepSegments": integer,
      "keyFormat": "string",
      "keyFormatVersions": "string",
      "keyProviderSettings": {
        "staticKeySettings": {
          "keyProviderServer": {
            "passwordParam": "string",
            "uri": "string",
            "username": "string"
          },
          "staticKeyValue": "string"
        }
      },
      "manifestCompression": enum,
      "manifestDurationFormat": enum,
      "minSegmentLength": integer,
      "mode": enum,
      "outputSelection": enum,
```

```
        "programDateTime": enum,
        "programDateTimeClock": enum,
        "programDateTimePeriod": integer,
        "redundantManifest": enum,
        "segmentLength": integer,
        "segmentationMode": enum,
        "segmentsPerSubdirectory": integer,
        "streamInfResolution": enum,
        "timedMetadataId3Frame": enum,
        "timedMetadataId3Period": integer,
        "timestampDeltaMilliseconds": integer,
        "tsFileMode": enum
      },
      "mediaPackageGroupSettings": {
        "destination": {
          "destinationRefId": "string"
        }
      },
      "msSmoothGroupSettings": {
        "acquisitionPointId": "string",
        "audioOnlyTimecodeControl": enum,
        "certificateMode": enum,
        "connectionRetryInterval": integer,
        "destination": {
          "destinationRefId": "string"
        },
        "eventId": "string",
        "eventIdMode": enum,
        "eventStopBehavior": enum,
        "filecacheDuration": integer,
        "fragmentLength": integer,
        "inputLossAction": enum,
        "numRetries": integer,
        "restartDelay": integer,
        "segmentationMode": enum,
        "sendDelayMs": integer,
        "sparseTrackType": enum,
        "streamManifestBehavior": enum,
        "timestampOffset": "string",
        "timestampOffsetMode": enum
      },
      "multiplexGroupSettings": {
      },
      "rtmpGroupSettings": {
```

```
            "adMarkers": [
              enum
            ],
            "authenticationScheme": enum,
            "cacheFullBehavior": enum,
            "cacheLength": integer,
            "captionData": enum,
            "inputLossAction": enum,
            "restartDelay": integer
          },
          "udpGroupSettings": {
            "inputLossAction": enum,
            "timedMetadataId3Frame": enum,
            "timedMetadataId3Period": integer
          }
        },
        "outputs": [
          {
            "audioDescriptionNames": [
              "string"
            ],
            "captionDescriptionNames": [
              "string"
            ],
            "outputName": "string",
            "outputSettings": {
              "archiveOutputSettings": {
                "containerSettings": {
                  "m2tsSettings": {
                    "absentInputAudioBehavior": enum,
                    "arib": enum,
                    "aribCaptionsPid": "string",
                    "aribCaptionsPidControl": enum,
                    "audioBufferModel": enum,
                    "audioFramesPerPes": integer,
                    "audioPids": "string",
                    "audioStreamType": enum,
                    "bitrate": integer,
                    "bufferModel": enum,
                    "ccDescriptor": enum,
                    "dvbNitSettings": {
                      "networkId": integer,
                      "networkName": "string",
                      "repInterval": integer
```

```
        },
        "dvbSdtSettings": {
          "outputSdt": enum,
          "repInterval": integer,
          "serviceName": "string",
          "serviceProviderName": "string"
        },
        "dvbSubPids": "string",
        "dvbTdtSettings": {
          "repInterval": integer
        },
        "dvbTeletextPid": "string",
        "ebif": enum,
        "ebpAudioInterval": enum,
        "ebpLookaheadMs": integer,
        "ebpPlacement": enum,
        "ecmPid": "string",
        "esRateInPes": enum,
        "etvPlatformPid": "string",
        "etvSignalPid": "string",
        "fragmentTime": number,
        "klv": enum,
        "klvDataPids": "string",
        "nielsenId3Behavior": enum,
        "nullPacketBitrate": number,
        "patInterval": integer,
        "pcrControl": enum,
        "pcrPeriod": integer,
        "pcrPid": "string",
        "pmtInterval": integer,
        "pmtPid": "string",
        "programNum": integer,
        "rateMode": enum,
        "scte27Pids": "string",
        "scte35Control": enum,
        "scte35Pid": "string",
        "segmentationMarkers": enum,
        "segmentationStyle": enum,
        "segmentationTime": number,
        "timedMetadataBehavior": enum,
        "timedMetadataPid": "string",
        "transportStreamId": integer,
        "videoPid": "string"
      },
```

```
            "rawSettings": {
            }
          },
          "extension": "string",
          "nameModifier": "string"
        },
        "frameCaptureOutputSettings": {
          "nameModifier": "string"
        },
        "hlsOutputSettings": {
          "h265PackagingType": enum,
          "hlsSettings": {
            "audioOnlyHlsSettings": {
              "audioGroupId": "string",
              "audioOnlyImage": {
                "passwordParam": "string",
                "uri": "string",
                "username": "string"
              },
              "audioTrackType": enum,
              "segmentType": enum
            },
            "fmp4HlsSettings": {
              "audioRenditionSets": "string",
              "nielsenId3Behavior": enum,
              "timedMetadataBehavior": enum
            },
            "frameCaptureHlsSettings": {
            },
            "standardHlsSettings": {
              "audioRenditionSets": "string",
              "m3u8Settings": {
                "audioFramesPerPes": integer,
                "audioPids": "string",
                "ecmPid": "string",
                "nielsenId3Behavior": enum,
                "patInterval": integer,
                "pcrControl": enum,
                "pcrPeriod": integer,
                "pcrPid": "string",
                "pmtInterval": integer,
                "pmtPid": "string",
                "programNum": integer,
                "scte35Behavior": enum,
```

```
            "scte35Pid": "string",
            "timedMetadataBehavior": enum,
            "timedMetadataPid": "string",
            "transportStreamId": integer,
            "videoPid": "string"
          }
        }
      },
      "nameModifier": "string",
      "segmentModifier": "string"
    },
    "mediaPackageOutputSettings": {
    },
    "msSmoothOutputSettings": {
      "h265PackagingType": enum,
      "nameModifier": "string"
    },
    "multiplexOutputSettings": {
      "destination": {
        "destinationRefId": "string"
      }
    },
    "rtmpOutputSettings": {
      "certificateMode": enum,
      "connectionRetryInterval": integer,
      "destination": {
        "destinationRefId": "string"
      },
      "numRetries": integer
    },
    "udpOutputSettings": {
      "bufferMsec": integer,
      "containerSettings": {
        "m2tsSettings": {
          "absentInputAudioBehavior": enum,
          "arib": enum,
          "aribCaptionsPid": "string",
          "aribCaptionsPidControl": enum,
          "audioBufferModel": enum,
          "audioFramesPerPes": integer,
          "audioPids": "string",
          "audioStreamType": enum,
          "bitrate": integer,
          "bufferModel": enum,
```

```
                "ccDescriptor": enum,
                "dvbNitSettings": {
                  "networkId": integer,
                  "networkName": "string",
                  "repInterval": integer
                },
                "dvbSdtSettings": {
                  "outputSdt": enum,
                  "repInterval": integer,
                  "serviceName": "string",
                  "serviceProviderName": "string"
                },
                "dvbSubPids": "string",
                "dvbTdtSettings": {
                  "repInterval": integer
                },
                "dvbTeletextPid": "string",
                "ebif": enum,
                "ebpAudioInterval": enum,
                "ebpLookaheadMs": integer,
                "ebpPlacement": enum,
                "ecmPid": "string",
                "esRateInPes": enum,
                "etvPlatformPid": "string",
                "etvSignalPid": "string",
                "fragmentTime": number,
                "klv": enum,
                "klvDataPids": "string",
                "nielsenId3Behavior": enum,
                "nullPacketBitrate": number,
                "patInterval": integer,
                "pcrControl": enum,
                "pcrPeriod": integer,
                "pcrPid": "string",
                "pmtInterval": integer,
                "pmtPid": "string",
                "programNum": integer,
                "rateMode": enum,
                "scte27Pids": "string",
                "scte35Control": enum,
                "scte35Pid": "string",
                "segmentationMarkers": enum,
                "segmentationStyle": enum,
                "segmentationTime": number,
```

```
                    "timedMetadataBehavior": enum,
                    "timedMetadataPid": "string",
                    "transportStreamId": integer,
                    "videoPid": "string"
                  }
                },
                "destination": {
                  "destinationRefId": "string"
                },
                "fecOutputSettings": {
                  "columnDepth": integer,
                  "includeFec": enum,
                  "rowLength": integer
                }
              }
            },
            "videoDescriptionName": "string"
          }
        ]
      }
    ],
    "timecodeConfig": {
      "source": enum,
      "syncThreshold": integer
    },
    "videoDescriptions": [
      {
        "codecSettings": {
          "frameCaptureSettings": {
            "captureInterval": integer,
            "captureIntervalUnits": enum
          },
          "h264Settings": {
            "adaptiveQuantization": enum,
            "afdSignaling": enum,
            "bitrate": integer,
            "bufFillPct": integer,
            "bufSize": integer,
            "colorMetadata": enum,
            "colorSpaceSettings": {
              "colorSpacePassthroughSettings": {
              },
              "rec601Settings": {
              },
```

```
            "rec709Settings": {
            }
          },
          "entropyEncoding": enum,
          "filterSettings": {
            "temporalFilterSettings": {
              "postFilterSharpening": enum,
              "strength": enum
            }
          },
          "fixedAfd": enum,
          "flickerAq": enum,
          "forceFieldPictures": enum,
          "framerateControl": enum,
          "framerateDenominator": integer,
          "framerateNumerator": integer,
          "gopBReference": enum,
          "gopClosedCadence": integer,
          "gopNumBFrames": integer,
          "gopSize": number,
          "gopSizeUnits": enum,
          "level": enum,
          "lookAheadRateControl": enum,
          "maxBitrate": integer,
          "minIInterval": integer,
          "numRefFrames": integer,
          "parControl": enum,
          "parDenominator": integer,
          "parNumerator": integer,
          "profile": enum,
          "qualityLevel": enum,
          "qvbrQualityLevel": integer,
          "rateControlMode": enum,
          "scanType": enum,
          "sceneChangeDetect": enum,
          "slices": integer,
          "softness": integer,
          "spatialAq": enum,
          "subgopLength": enum,
          "syntax": enum,
          "temporalAq": enum,
          "timecodeInsertion": enum
        },
        "h265Settings": {
```

```
            "adaptiveQuantization": enum,
            "afdSignaling": enum,
            "alternativeTransferFunction": enum,
            "bitrate": integer,
            "bufSize": integer,
            "colorMetadata": enum,
            "colorSpaceSettings": {
              "colorSpacePassthroughSettings": {
              },
              "hdr10Settings": {
                "maxCll": integer,
                "maxFall": integer
              },
              "rec601Settings": {
              },
              "rec709Settings": {
              }
            },
            "filterSettings": {
              "temporalFilterSettings": {
                "postFilterSharpening": enum,
                "strength": enum
              }
            },
            "fixedAfd": enum,
            "flickerAq": enum,
            "framerateDenominator": integer,
            "framerateNumerator": integer,
            "gopClosedCadence": integer,
            "gopSize": number,
            "gopSizeUnits": enum,
            "level": enum,
            "lookAheadRateControl": enum,
            "maxBitrate": integer,
            "minIInterval": integer,
            "parDenominator": integer,
            "parNumerator": integer,
            "profile": enum,
            "qvbrQualityLevel": integer,
            "rateControlMode": enum,
            "scanType": enum,
            "sceneChangeDetect": enum,
            "slices": integer,
            "tier": enum,
```

```
            "timecodeInsertion": enum
          },
          "mpeg2Settings": {
            "adaptiveQuantization": enum,
            "afdSignaling": enum,
            "colorMetadata": enum,
            "colorSpace": enum,
            "displayAspectRatio": enum,
            "filterSettings": {
              "temporalFilterSettings": {
                "postFilterSharpening": enum,
                "strength": enum
              }
            },
            "fixedAfd": enum,
            "framerateDenominator": integer,
            "framerateNumerator": integer,
            "gopClosedCadence": integer,
            "gopNumBFrames": integer,
            "gopSize": number,
            "gopSizeUnits": enum,
            "scanType": enum,
            "subgopLength": enum,
            "timecodeInsertion": enum
          }
        },
        "height": integer,
        "name": "string",
        "respondToAfd": enum,
        "scalingBehavior": enum,
        "sharpness": integer,
        "width": integer
      }
    ]
  },
  "id": "string",
  "inputAttachments": [
    {
      "automaticInputFailoverSettings": {
        "errorClearTimeMsec": integer,
        "failoverConditions": [
          {
            "failoverConditionSettings": {
              "audioSilenceSettings": {
```

```
                "audioSelectorName": "string",
                "audioSilenceThresholdMsec": integer
              },
              "inputLossSettings": {
                "inputLossThresholdMsec": integer
              },
              "videoBlackSettings": {
                "blackDetectThreshold": number,
                "videoBlackThresholdMsec": integer
              }
            }
          }
        ],
        "inputPreference": enum,
        "secondaryInputId": "string"
      },
      "inputAttachmentName": "string",
      "inputId": "string",
      "inputSettings": {
        "audioSelectors": [
          {
            "name": "string",
            "selectorSettings": {
              "audioHlsRenditionSelection": {
                "groupId": "string",
                "name": "string"
              },
              "audioLanguageSelection": {
                "languageCode": "string",
                "languageSelectionPolicy": enum
              },
              "audioPidSelection": {
                "pid": integer
              },
              "audioTrackSelection": {
                "tracks": [
                  {
                    "track": integer
                  }
                ]
              }
            }
          }
        ],
```

```
        "captionSelectors": [
          {
            "languageCode": "string",
            "name": "string",
            "selectorSettings": {
              "ancillarySourceSettings": {
                "sourceAncillaryChannelNumber": integer
              },
              "aribSourceSettings": {
              },
              "dvbSubSourceSettings": {
                "ocrLanguage": enum,
                "pid": integer
              },
              "embeddedSourceSettings": {
                "convert608To708": enum,
                "scte20Detection": enum,
                "source608ChannelNumber": integer,
                "source608TrackNumber": integer
              },
              "scte20SourceSettings": {
                "convert608To708": enum,
                "source608ChannelNumber": integer
              },
              "scte27SourceSettings": {
                "ocrLanguage": enum,
                "pid": integer
              },
              "teletextSourceSettings": {
                "outputRectangle": {
                  "height": number,
                  "leftOffset": number,
                  "topOffset": number,
                  "width": number
                },
                "pageNumber": "string"
              }
            }
          }
        ],
        "deblockFilter": enum,
        "denoiseFilter": enum,
        "filterStrength": integer,
        "inputFilter": enum,
```

```
          "networkInputSettings": {
            "hlsInputSettings": {
              "bandwidth": integer,
              "bufferSegments": integer,
              "retries": integer,
              "retryInterval": integer,
              "scte35Source": enum
            },
            "serverValidation": enum
          },
          "smpte2038DataPreference": enum,
          "sourceEndBehavior": enum,
          "videoSelector": {
            "colorSpace": enum,
            "colorSpaceSettings": {
              "hdr10Settings": {
                "maxCll": integer,
                "maxFall": integer
              }
            },
            "colorSpaceUsage": enum,
            "selectorSettings": {
              "videoSelectorPid": {
                "pid": integer
              },
              "videoSelectorProgramId": {
                "programId": integer
              }
            }
          }
        }
      }
    ],
    "inputSpecification": {
      "codec": enum,
      "maximumBitrate": enum,
      "resolution": enum
    },
    "logLevel": enum,
    "maintenance": {
      "maintenanceDay": enum,
      "maintenanceDeadline": "string",
      "maintenanceScheduledDate": "string",
      "maintenanceStartTime": "string"
```

```
    },
    "name": "string",
    "pipelineDetails": [
      {
        "activeInputAttachmentName": "string",
        "activeInputSwitchActionName": "string",
        "activeMotionGraphicsActionName": "string",
        "activeMotionGraphicsUri": "string",
        "pipelineId": "string"
      }
    ],
    "pipelinesRunningCount": integer,
    "roleArn": "string",
    "state": enum,
    "tags": {
    },
    "vpc": {
      "availabilityZones": [
        "string"
      ],
      "networkInterfaceIds": [
        "string"
      ],
      "securityGroupIds": [
        "string"
      ],
      "subnetIds": [
        "string"
      ]
    }
  }
}
```

InvalidRequest schema

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

AccessDenied schema

```
{
```

```
  "message": "string"
}
```

ResourceConflict schema

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

ChannelConfigurationValidationError schema

```
{
  "message": "string",
  "validationErrors": [
    {
      "elementPath": "string",
      "errorMessage": "string"
    }
  ]
}
```

LimitExceeded schema

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

InternalServiceError schema

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

BadGatewayException schema

```
{
  "message": "string"
```

```
}
```

GatewayTimeoutException schema

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

Properties

AacCodingMode

Aac Coding Mode

```
AD_RECEIVER_MIX
CODING_MODE_1_0
CODING_MODE_1_1
CODING_MODE_2_0
CODING_MODE_5_1
```

AacInputType

Aac Input Type

```
BROADCASTER_MIXED_AD
NORMAL
```

AacProfile

Aac Profile

```
HEV1
HEV2
LC
```

AacRateControlMode

Aac Rate Control Mode

CBR

VBR

AacRawFormat

Aac Raw Format

LATM_LOAS

NONE

AacSettings

Aac Settings

bitrate

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

Type: number

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: AacCodingMode

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: AacInputType
Required: False

profile

AAC Profile.

Type: AacProfile
Required: False

rateControlMode

Rate Control Mode.

Type: AacRateControlMode
Required: False

rawFormat

Sets LATM / LOAS AAC output for raw containers.

Type: AacRawFormat
Required: False

sampleRate

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

Type: number
Required: False

spec

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

Type: AacSpec
Required: False

vbrQuality

VBR Quality Level - Only used if rateControlMode is VBR.

Type: AacVbrQuality
Required: False

AacSpec

Aac Spec

MPEG2
MPEG4

AacVbrQuality

Aac Vbr Quality

HIGH
LOW
MEDIUM_HIGH
MEDIUM_LOW

Ac3BitstreamMode

Ac3 Bitstream Mode

COMMENTARY
COMPLETE_MAIN
DIALOGUE
EMERGENCY
HEARING_IMPAIRED
MUSIC_AND_EFFECTS

VISUALLY_IMPAIRED

VOICE_OVER

Ac3CodingMode

Ac3 Coding Mode

CODING_MODE_1_0

CODING_MODE_1_1

CODING_MODE_2_0

CODING_MODE_3_2_LFE

Ac3DrcProfile

Ac3 Drc Profile

FILM_STANDARD

NONE

Ac3LfeFilter

Ac3 Lfe Filter

DISABLED

ENABLED

Ac3MetadataControl

Ac3 Metadata Control

FOLLOW_INPUT

USE_CONFIGURED

Ac3Settings

Ac3 Settings

bitrate

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

Type: number
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: Ac3BitstreamMode
Required: False

codingMode

Dolby Digital coding mode. Determines number of channels.

Type: Ac3CodingMode
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

drcProfile

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

Type: Ac3DrcProfile

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: Ac3LfeFilter
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: Ac3MetadataControl
Required: False

AccessDenied

message

Type: string
Required: False

AfdSignaling

Afd Signaling

AUTO
FIXED
NONE

AncillarySourceSettings

Ancillary Source Settings

sourceAncillaryChannelNumber

Specifies the number (1 to 4) of the captions channel you want to extract from the ancillary captions. If you plan to convert the ancillary captions to another format, complete this field. If you plan to choose Embedded as the captions destination in the output (to pass through all the channels in the ancillary captions), leave this field blank because MediaLive ignores the field.

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

ArchiveCdnSettings

Archive Cdn Settings

archiveS3Settings

Type: ArchiveS3Settings
Required: False

ArchiveContainerSettings

Archive Container Settings

m2tsSettings

Type: M2tsSettings
Required: False

rawSettings

Type: RawSettings
Required: False

ArchiveGroupSettings

Archive Group Settings

archiveCdnSettings

Parameters that control interactions with the CDN.

Type: ArchiveCdnSettings
Required: False

destination

A directory and base filename where archive files should be written.

Type: OutputLocationRef
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

Archive Output Settings

containerSettings

Settings specific to the container type of the file.

Type: ArchiveContainerSettings
Required: True

extension

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

Type: string
Required: False

nameModifier

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

Type: string
Required: False

ArchiveS3LogUploads

Archive S3 Log Uploads

```
DISABLED
ENABLED
```

ArchiveS3Settings

Archive S3 Settings

cannedAcl

Specify the canned ACL to apply to each S3 request. Defaults to none.

Type: S3CannedAcl
Required: False

logUploads

When set to enabled, each upload to CDN or server will be logged.

Type: ArchiveS3LogUploads
Required: False

AribDestinationSettings

Arib Destination Settings

AribSourceSettings

Arib Source Settings

AudioChannelMapping

Audio Channel Mapping

inputChannelLevels

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

Type: Array of type InputChannelLevel
Required: True

outputChannel

The index of the output channel being produced.

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

AudioCodecSettings

Audio Codec Settings

aacSettings

Type: AacSettings
Required: False

ac3Settings

Type: Ac3Settings
Required: False

eac3Settings

Type: Eac3Settings
Required: False

mp2Settings

Type: Mp2Settings
Required: False

passThroughSettings

Type: PassThroughSettings
Required: False

wavSettings

Type: WavSettings
Required: False

AudioDescription

Audio Description

audioNormalizationSettings

Advanced audio normalization settings.

Type: AudioNormalizationSettings
Required: False

audioSelectorName

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

Type: string
Required: True

audioType

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

Type: AudioType

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: AudioDescriptionAudioTypeControl
Required: False

audioWatermarkingSettings

Settings to configure one or more solutions that insert audio watermarks in the audio encode

Type: AudioWatermarkSettings
Required: False

codecSettings

Audio codec settings.

Type: AudioCodecSettings
Required: False

languageCode

RFC 5646 language code representing 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
MinLength: 1
MaxLength: 35

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: AudioDescriptionLanguageCodeControl
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

remixSettings

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

Type: RemixSettings
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

AudioDescriptionAudioTypeControl

Audio Description Audio Type Control

```
FOLLOW_INPUT
USE_CONFIGURED
```

AudioDescriptionLanguageCodeControl

Audio Description Language Code Control

FOLLOW_INPUT
USE_CONFIGURED

AudioHlsRenditionSelection

Audio Hls Rendition Selection

groupId

Specifies the GROUP-ID in the #EXT-X-MEDIA tag of the target HLS audio rendition.

Type: string
Required: True
MinLength: 1

name

Specifies the NAME in the #EXT-X-MEDIA tag of the target HLS audio rendition.

Type: string
Required: True
MinLength: 1

AudioLanguageSelection

Audio Language Selection

languageCode

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

Type: string
Required: True

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: AudioLanguageSelectionPolicy
Required: False

AudioLanguageSelectionPolicy

Audio Language Selection Policy

```
LOOSE
STRICT
```

AudioNormalizationAlgorithm

Audio Normalization Algorithm

```
ITU_1770_1
ITU_1770_2
```

AudioNormalizationAlgorithmControl

Audio Normalization Algorithm Control

```
CORRECT_AUDIO
```

AudioNormalizationSettings

Audio Normalization Settings

algorithm

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

Type: AudioNormalizationAlgorithm
Required: False

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: AudioNormalizationAlgorithmControl
Required: False

targetLkfs

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

Type: number
Required: False
Minimum: -59
Maximum: 0

AudioOnlyHlsSegmentType

Audio Only Hls Segment Type

AAC
FMP4

AudioOnlyHlsSettings

Audio Only Hls Settings

audioGroupId

Specifies the group to which the audio Rendition belongs.

Type: string
Required: False

audioOnlyImage

Optional. Specifies the .jpg or .png image to use as the cover art for an audio-only output. We recommend a low bit-size file because the image increases the output audio bandwidth. The image is attached to the audio as an ID3 tag, frame type APIC, picture type 0x10, as per the "ID3 tag version 2.4.0 - Native Frames" standard.

Type: InputLocation
Required: False

audioTrackType

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

Type: AudioOnlyHlsTrackType
Required: False

segmentType

Specifies the segment type.

Type: AudioOnlyHlsSegmentType
Required: False

AudioOnlyHlsTrackType

Audio Only Hls Track Type

```
ALTERNATE_AUDIO_AUTO_SELECT
ALTERNATE_AUDIO_AUTO_SELECT_DEFAULT
```

```
ALTERNATE_AUDIO_NOT_AUTO_SELECT
AUDIO_ONLY_VARIANT_STREAM
```

AudioPidSelection

Audio Pid Selection

pid

Selects a specific PID from within a source.

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

AudioSelector

Audio Selector

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
MinLength: 1

selectorSettings

The audio selector settings.

Type: AudioSelectorSettings
Required: False

AudioSelectorSettings

Audio Selector Settings

audioHlsRenditionSelection

Type: AudioHlsRenditionSelection
Required: False

audioLanguageSelection

Type: AudioLanguageSelection
Required: False

audioPidSelection

Type: AudioPidSelection
Required: False

audioTrackSelection

Type: AudioTrackSelection
Required: False

AudioSilenceFailoverSettings

audioSelectorName

The name of the audio selector in the input that MediaLive should monitor to detect silence. Select your most important rendition. If you didn't create an audio selector in this input, leave blank.

Type: string
Required: True

audioSilenceThresholdMsec

The amount of time (in milliseconds) that the active input must be silent before automatic input failover occurs. Silence is defined as audio loss or audio quieter than -50 dBFS.

Type: integer
Required: False
Minimum: 1000

AudioTrack

Audio Track

track

1-based integer value that maps to a specific audio track

Type: integer
Required: True
Minimum: 1

AudioTrackSelection

Audio Track Selection

tracks

Selects one or more unique audio tracks from within a source.

Type: Array of type AudioTrack
Required: True

AudioType

Audio Type

```
CLEAN_EFFECTS
HEARING_IMPAIRED
UNDEFINED
VISUAL_IMPAIRED_COMMENTARY
```

AudioWatermarkSettings

Audio Watermark Settings

nielsenWatermarksSettings

Settings to configure Nielsen Watermarks in the audio encode

Type: NielsenWatermarksSettings
Required: False

AuthenticationScheme

Authentication Scheme

```
AKAMAI
COMMON
```

AutomaticInputFailoverSettings

The settings for Automatic Input Failover.

errorClearTimeMsec

This clear time defines the requirement a recovered input must meet to be considered healthy. The input must have no failover conditions for this length of time. Enter a time in milliseconds. This value is particularly important if the input_preference for the failover pair is set to PRIMARY_INPUT_PREFERRED, because after this time, MediaLive will switch back to the primary input.

Type: integer
Required: False
Minimum: 1

failoverConditions

A list of failover conditions. If any of these conditions occur, MediaLive will perform a failover to the other input.

Type: Array of type FailoverCondition
Required: False

inputPreference

Input preference when deciding which input to make active when a previously failed input has recovered.

Type: InputPreference
Required: False

secondaryInputId

The input ID of the secondary input in the automatic input failover pair.

Type: string
Required: True

AvailBlanking

Avail Blanking

availBlankingImage

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

Type: InputLocation
Required: False

state

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

Type: AvailBlankingState
Required: False

AvailBlankingState

Avail Blanking State

```
DISABLED
ENABLED
```

AvailConfiguration

Avail Configuration

availSettings

Ad avail settings.

Type: AvailSettings
Required: False

AvailSettings

Avail Settings

scte35SpliceInsert

Type: Scte35SpliceInsert
Required: False

scte35TimeSignalApos

Type: Scte35TimeSignalApos
Required: False

BadGatewayException

message

Type: string
Required: False

BlackoutSlate

Blackout Slate

blackoutSlateImage

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

Type: InputLocation

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: BlackoutSlateNetworkEndBlackout
Required: False

networkEndBlackoutImage

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

Type: InputLocation
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
MinLength: 34
MaxLength: 34

state

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

Type: BlackoutSlateState
Required: False

BlackoutSlateNetworkEndBlackout

Blackout Slate Network End Blackout

DISABLED
ENABLED

BlackoutSlateState

Blackout Slate State

DISABLED
ENABLED

BurnInAlignment

Burn In Alignment

CENTERED
LEFT
SMART

BurnInBackgroundColor

Burn In Background Color

BLACK
NONE
WHITE

BurnInDestinationSettings

Burn In Destination Settings

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: BurnInAlignment
Required: False

backgroundColor

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

Type: BurnInBackgroundColor
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

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
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: BurnInFontColor
Required: False

fontOpacity

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

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

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

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

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: BurnInOutlineColor
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

shadowColor

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

Type: BurnInShadowColor
Required: False

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

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

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

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: BurnInTeletextGridControl
Required: False

xPosition

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

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

BurnInFontColor

Burn In Font Color

BLACK
BLUE
GREEN
RED
WHITE
YELLOW

BurnInOutlineColor

Burn In Outline Color

BLACK
BLUE
GREEN
RED
WHITE
YELLOW

BurnInShadowColor

Burn In Shadow Color

BLACK

NONE
WHITE

BurnInTeletextGridControl

Burn In Teletext Grid Control

FIXED
SCALED

CaptionDescription

Caption Description

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

destinationSettings

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

Type: CaptionDestinationSettings
Required: False

languageCode

ISO 639-2 three-digit code: http://www.loc.gov/standards/iso639-2/

Type: string
Required: False

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

CaptionDestinationSettings

Caption Destination Settings

aribDestinationSettings

Type: AribDestinationSettings
Required: False

burnInDestinationSettings

Type: BurnInDestinationSettings
Required: False

dvbSubDestinationSettings

Type: DvbSubDestinationSettings
Required: False

ebuTtDDestinationSettings

Type: EbuTtDDestinationSettings
Required: False

embeddedDestinationSettings

Type: EmbeddedDestinationSettings

Required: False

embeddedPlusScte20DestinationSettings

Type: EmbeddedPlusScte20DestinationSettings
Required: False

rtmpCaptionInfoDestinationSettings

Type: RtmpCaptionInfoDestinationSettings
Required: False

scte20PlusEmbeddedDestinationSettings

Type: Scte20PlusEmbeddedDestinationSettings
Required: False

scte27DestinationSettings

Type: Scte27DestinationSettings
Required: False

smpteTtDestinationSettings

Type: SmpteTtDestinationSettings
Required: False

teletextDestinationSettings

Type: TeletextDestinationSettings
Required: False

ttmlDestinationSettings

Type: TtmlDestinationSettings
Required: False

webvttDestinationSettings

Type: WebvttDestinationSettings
Required: False

CaptionLanguageMapping

Maps a caption channel to an ISO 693-2 language code (http://www.loc.gov/standards/iso639-2), with an optional description.

captionChannel

The closed caption channel being described by this CaptionLanguageMapping. 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
MinLength: 3
MaxLength: 3

languageDescription

Textual description of language

Type: string
Required: True
MinLength: 1

CaptionRectangle

Caption Rectangle

height

See the description in leftOffset. For height, specify the entire height of the rectangle as a percentage of the underlying frame height. For example, \"80\" means the rectangle height is 80% of the underlying frame height. The topOffset and rectangleHeight must add up to 100% or less. This field corresponds to tts:extent - Y in the TTML standard.

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

leftOffset

Applies only if you plan to convert these source captions to EBU-TT-D or TTML in an output. (Make sure to leave the default if you don't have either of these formats in the output.) You can define a display rectangle for the captions that is smaller than the underlying video frame. You define the rectangle by specifying the position of the left edge, top edge, bottom edge, and right edge of the rectangle, all within the underlying video frame. The units for the measurements are percentages. If you specify a value for one of these fields, you must specify a value for all of them. For leftOffset, specify the position of the left edge of the rectangle, as a percentage of the underlying frame width, and relative to the left edge of the frame. For example, \"10\" means the measurement is 10% of the underlying frame width. The rectangle left edge starts at that position from the left edge of the frame. This field corresponds to tts:origin - X in the TTML standard.

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

topOffset

See the description in leftOffset. For topOffset, specify the position of the top edge of the rectangle, as a percentage of the underlying frame height, and relative to the top edge of the frame. For example, \"10\" means the measurement is 10% of the underlying frame height. The

rectangle top edge starts at that position from the top edge of the frame. This field corresponds to tts:origin - Y in the TTML standard.

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

width

See the description in leftOffset. For width, specify the entire width of the rectangle as a percentage of the underlying frame width. For example, \"80\" means the rectangle width is 80% of the underlying frame width. The leftOffset and rectangleWidth must add up to 100% or less. This field corresponds to tts:extent - X in the TTML standard.

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

CaptionSelector

Output groups for this Live Event. Output groups contain information about where streams should be distributed.

languageCode

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

Type: string
Required: False

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
MinLength: 1

selectorSettings

Caption selector settings.

Type: CaptionSelectorSettings
Required: False

CaptionSelectorSettings

Caption Selector Settings

ancillarySourceSettings

Type: AncillarySourceSettings
Required: False

aribSourceSettings

Type: AribSourceSettings
Required: False

dvbSubSourceSettings

Type: DvbSubSourceSettings
Required: False

embeddedSourceSettings

Type: EmbeddedSourceSettings
Required: False

scte20SourceSettings

Type: Scte20SourceSettings
Required: False

scte27SourceSettings

Type: Scte27SourceSettings
Required: False

teletextSourceSettings

Type: TeletextSourceSettings
Required: False

CdiInputResolution

Maximum CDI input resolution; SD is 480i and 576i up to 30 frames-per-second (fps), HD is 720p up to 60 fps / 1080i up to 30 fps, FHD is 1080p up to 60 fps, UHD is 2160p up to 60 fps

SD

HD

FHD

UHD

CdiInputSpecification

resolution

Maximum CDI input resolution

Type: CdiInputResolution
Required: False

Channel

arn

The unique arn of the channel.

Type: string
Required: False

cdiInputSpecification

Specification of CDI inputs for this channel

Type: CdiInputSpecification
Required: False

channelClass

The class for this channel. STANDARD for a channel with two pipelines or SINGLE_PIPELINE for a channel with one pipeline.

Type: ChannelClass
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
Required: False

egressEndpoints

The endpoints where outgoing connections initiate from

Type: Array of type ChannelEgressEndpoint
Required: False

encoderSettings

Type: EncoderSettings
Required: False

id

The unique ID of the channel.

Type: string
Required: False

inputAttachments

List of input attachments for channel.

Type: Array of type InputAttachment
Required: False

inputSpecification

Specification of network and file inputs for this channel

Type: InputSpecification
Required: False

logLevel

The log level being written to CloudWatch Logs.

Type: LogLevel
Required: False

maintenance

Maintenance settings for this channel.

Type: MaintenanceStatus
Required: False

name

The name of the channel. (user-mutable)

Type: string
Required: False

pipelineDetails

Runtime details for the pipelines of a running channel.

Type: Array of type PipelineDetail
Required: False

pipelinesRunningCount

The number of currently healthy pipelines.

Type: integer
Required: False

roleArn

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

Type: string
Required: False

state

Type: ChannelState
Required: False

tags

A collection of key-value pairs.

Type: Tags
Required: False

vpc

Settings for VPC output

Type: VpcOutputSettingsDescription
Required: False

ChannelClass

A standard channel has two encoding pipelines and a single pipeline channel only has one.

STANDARD
SINGLE_PIPELINE

ChannelConfigurationValidationError

message

Type: string
Required: False

validationErrors

A collection of validation error responses.

Type: Array of type ValidationError
Required: False

ChannelEgressEndpoint

sourceIp

Public IP of where a channel's output comes from

Type: string
Required: False

ChannelState

CREATING
CREATE_FAILED
IDLE
STARTING
RUNNING

RECOVERING
STOPPING
DELETING
DELETED
UPDATING
UPDATE_FAILED

ChannelSummary

arn

The unique arn of the channel.

Type: string
Required: False

cdiInputSpecification

Specification of CDI inputs for this channel

Type: CdiInputSpecification
Required: False

channelClass

The class for this channel. STANDARD for a channel with two pipelines or SINGLE_PIPELINE for a channel with one pipeline.

Type: ChannelClass
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
Required: False

egressEndpoints

The endpoints where outgoing connections initiate from

Type: Array of type ChannelEgressEndpoint
Required: False

id

The unique id of the channel.

Type: string
Required: False

inputAttachments

List of input attachments for channel.

Type: Array of type InputAttachment
Required: False

inputSpecification

Specification of network and file inputs for this channel

Type: InputSpecification
Required: False

logLevel

The log level being written to CloudWatch Logs.

Type: LogLevel
Required: False

maintenance

Maintenance settings for this channel.

Type: MaintenanceStatus

Required: False

name

The name of the channel. (user-mutable)

Type: string
Required: False

pipelinesRunningCount

The number of currently healthy pipelines.

Type: integer
Required: False

roleArn

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

Type: string
Required: False

state

Type: ChannelState
Required: False

tags

A collection of key-value pairs.

Type: Tags
Required: False

vpc

Settings for any VPC outputs.

Type: VpcOutputSettingsDescription
Required: False

ColorSpacePassthroughSettings

Passthrough applies no color space conversion to the output

CreateChannel

cdiInputSpecification

Specification of CDI inputs for this channel

Type: CdiInputSpecification
Required: False

channelClass

The class for this channel. STANDARD for a channel with two pipelines or SINGLE_PIPELINE for a channel with one pipeline.

Type: ChannelClass
Required: False

destinations

Type: Array of type OutputDestination
Required: False

encoderSettings

Type: EncoderSettings
Required: False

inputAttachments

List of input attachments for channel.

Type: Array of type InputAttachment
Required: False

inputSpecification

Specification of network and file inputs for this channel

Type: InputSpecification
Required: False

logLevel

The log level to write to CloudWatch Logs.

Type: LogLevel
Required: False

maintenance

Maintenance settings for this channel.

Type: MaintenanceCreateSettings
Required: False

name

Name of channel.

Type: string
Required: False

requestId

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

Type: string
Required: False

reserved

Deprecated field that's only usable by whitelisted customers.

Type: string
Required: False

roleArn

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

Type: string
Required: False

tags

A collection of key-value pairs.

Type: Tags
Required: False

vpc

Settings for the VPC outputs

Type: VpcOutputSettings
Required: False

CreateChannelResultModel

channel

Type: Channel
Required: False

DvbNitSettings

DVB Network Information Table (NIT)

networkId

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

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

networkName

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

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

repInterval

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

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

DvbSdtOutputSdt

Dvb Sdt Output Sdt

SDT_FOLLOW
SDT_FOLLOW_IF_PRESENT
SDT_MANUAL
SDT_NONE

DvbSdtSettings

DVB Service Description Table (SDT)

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: DvbSdtOutputSdt
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

serviceName

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

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

serviceProviderName

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

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

DvbSubDestinationAlignment

Dvb Sub Destination Alignment

CENTERED
LEFT
SMART

DvbSubDestinationBackgroundColor

Dvb Sub Destination Background Color

BLACK
NONE
WHITE

DvbSubDestinationFontColor

Dvb Sub Destination Font Color

BLACK
BLUE
GREEN
RED
WHITE
YELLOW

DvbSubDestinationOutlineColor

Dvb Sub Destination Outline Color

BLACK

BLUE
GREEN
RED
WHITE
YELLOW

DvbSubDestinationSettings

Dvb Sub Destination Settings

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: DvbSubDestinationAlignment
Required: False

backgroundColor

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

Type: DvbSubDestinationBackgroundColor
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

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
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: DvbSubDestinationFontColor
Required: False

fontOpacity

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

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

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

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

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: DvbSubDestinationOutlineColor
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

shadowColor

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

Type: DvbSubDestinationShadowColor
Required: False

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

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

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

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: DvbSubDestinationTeletextGridControl
Required: False

xPosition

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

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

DvbSubDestinationShadowColor

Dvb Sub Destination Shadow Color

```
BLACK
NONE
WHITE
```

DvbSubDestinationTeletextGridControl

Dvb Sub Destination Teletext Grid Control

```
FIXED
```

SCALED

DvbSubOcrLanguage

Dvb Sub Ocr Language

DEU
ENG
FRA
NLD
POR
SPA

DvbSubSourceSettings

Dvb Sub Source Settings

ocrLanguage

If you will configure a WebVTT caption description that references this caption selector, use this field to provide the language to consider when translating the image-based source to text.

Type: DvbSubOcrLanguage
Required: False

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

DVB Time and Date Table (SDT)

repInterval

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

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

Eac3AttenuationControl

Eac3 Attenuation Control

ATTENUATE_3_DB
NONE

Eac3BitstreamMode

Eac3 Bitstream Mode

COMMENTARY
COMPLETE_MAIN
EMERGENCY
HEARING_IMPAIRED
VISUALLY_IMPAIRED

Eac3CodingMode

Eac3 Coding Mode

CODING_MODE_1_0
CODING_MODE_2_0
CODING_MODE_3_2

Eac3DcFilter

Eac3 Dc Filter

DISABLED
ENABLED

Eac3DrcLine

Eac3 Drc Line

FILM_LIGHT
FILM_STANDARD
MUSIC_LIGHT
MUSIC_STANDARD
NONE
SPEECH

Eac3DrcRf

Eac3 Drc Rf

FILM_LIGHT
FILM_STANDARD
MUSIC_LIGHT
MUSIC_STANDARD
NONE
SPEECH

Eac3LfeControl

Eac3 Lfe Control

LFE
NO_LFE

Eac3LfeFilter

Eac3 Lfe Filter

DISABLED

ENABLED

Eac3MetadataControl

Eac3 Metadata Control

FOLLOW_INPUT
USE_CONFIGURED

Eac3PassthroughControl

Eac3 Passthrough Control

NO_PASSTHROUGH
WHEN_POSSIBLE

Eac3PhaseControl

Eac3 Phase Control

NO_SHIFT
SHIFT_90_DEGREES

Eac3Settings

Eac3 Settings

attenuationControl

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

Type: Eac3AttenuationControl
Required: False

bitrate

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

Type: number
Required: False

bitstreamMode

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

Type: Eac3BitstreamMode
Required: False

codingMode

Dolby Digital Plus coding mode. Determines number of channels.

Type: Eac3CodingMode
Required: False

dcFilter

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

Type: Eac3DcFilter
Required: False

dialnorm

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

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

drcLine

Sets the Dolby dynamic range compression profile.

Type: Eac3DrcLine
Required: False

drcRf

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

Type: Eac3DrcRf
Required: False

lfeControl

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

Type: Eac3LfeControl
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: Eac3LfeFilter
Required: False

loRoCenterMixLevel

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

Type: number
Required: False

loRoSurroundMixLevel

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

Type: number
Required: False

ltRtCenterMixLevel

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

Type: number
Required: False

ltRtSurroundMixLevel

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

Type: number
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: Eac3MetadataControl
Required: False

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: Eac3PassthroughControl
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: Eac3PhaseControl

Required: False

stereoDownmix

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

Type: Eac3StereoDownmix
Required: False

surroundExMode

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

Type: Eac3SurroundExMode
Required: False

surroundMode

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

Type: Eac3SurroundMode
Required: False

Eac3StereoDownmix

Eac3 Stereo Downmix

```
DPL2
LO_RO
LT_RT
NOT_INDICATED
```

Eac3SurroundExMode

Eac3 Surround Ex Mode

```
DISABLED
```

ENABLED
NOT_INDICATED

Eac3SurroundMode

Eac3 Surround Mode

DISABLED
ENABLED
NOT_INDICATED

EbuTtDDestinationSettings

Ebu Tt DDestination Settings

copyrightHolder

Applies only if you plan to convert these source captions to EBU-TT-D or TTML in an output. Complete this field if you want to include the name of the copyright holder in the copyright metadata tag in the TTML

Type: string
Required: False
MaxLength: 1000

fillLineGap

Specifies how to handle the gap between the lines (in multi-line captions). - enabled: Fill with the captions background color (as specified in the input captions). - disabled: Leave the gap unfilled.

Type: EbuTtDFillLineGapControl
Required: False

fontFamily

Specifies the font family to include in the font data attached to the EBU-TT captions. Valid only if styleControl is set to include. If you leave this field empty, the font family is set to "monospaced". (If styleControl is set to exclude, the font family is always set to "monospaced".) You specify only

the font family. All other style information (color, bold, position and so on) is copied from the input captions. The size is always set to 100% to allow the downstream player to choose the size. - Enter a list of font families, as a comma-separated list of font names, in order of preference. The name can be a font family (such as “Arial”), or a generic font family (such as “serif”), or “default” (to let the downstream player choose the font). - Leave blank to set the family to “monospace”.

Type: string
Required: False

styleControl

Specifies the style information (font color, font position, and so on) to include in the font data that is attached to the EBU-TT captions. - include: Take the style information (font color, font position, and so on) from the source captions and include that information in the font data attached to the EBU-TT captions. This option is valid only if the source captions are Embedded or Teletext. - exclude: In the font data attached to the EBU-TT captions, set the font family to "monospaced". Do not include any other style information.

Type: EbuTtDDestinationStyleControl
Required: False

EbuTtDDestinationStyleControl

Ebu Tt DDestination Style Control

```
EXCLUDE
INCLUDE
```

EbuTtDFillLineGapControl

Ebu Tt DFill Line Gap Control

```
DISABLED
ENABLED
```

EmbeddedConvert608To708

Embedded Convert608 To708

DISABLED

UPCONVERT

EmbeddedDestinationSettings

Embedded Destination Settings

EmbeddedPlusScte20DestinationSettings

Embedded Plus Scte20 Destination Settings

EmbeddedScte20Detection

Embedded Scte20 Detection

AUTO

OFF

EmbeddedSourceSettings

Embedded Source Settings

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: EmbeddedConvert608To708

Required: False

scte20Detection

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

Type: EmbeddedScte20Detection

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

source608TrackNumber

This field is unused and deprecated.

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

EncoderSettings

Encoder Settings

audioDescriptions

Type: Array of type AudioDescription
Required: True

availBlanking

Settings for ad avail blanking.

Type: AvailBlanking
Required: False

availConfiguration

Event-wide configuration settings for ad avail insertion.

Type: AvailConfiguration
Required: False

blackoutSlate

Settings for blackout slate.

Type: BlackoutSlate
Required: False

captionDescriptions

Settings for caption decriptions

Type: Array of type CaptionDescription
Required: False

featureActivations

Feature Activations

Type: FeatureActivations
Required: False

globalConfiguration

Configuration settings that apply to the event as a whole.

Type: GlobalConfiguration
Required: False

motionGraphicsConfiguration

Settings for motion graphics.

Type: MotionGraphicsConfiguration
Required: False

nielsenConfiguration

Nielsen configuration settings.

Type: NielsenConfiguration
Required: False

outputGroups

Type: Array of type OutputGroup
Required: True

timecodeConfig

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

Type: TimecodeConfig
Required: True

videoDescriptions

Type: Array of type VideoDescription
Required: True

FailoverCondition

Failover Condition settings. There can be multiple failover conditions inside AutomaticInputFailoverSettings.

failoverConditionSettings

Failover condition type-specific settings.

Type: FailoverConditionSettings
Required: False

FailoverConditionSettings

Settings for one failover condition.

audioSilenceSettings

MediaLive will perform a failover if the specified audio selector is silent for the specified period.

Type: AudioSilenceFailoverSettings
Required: False

inputLossSettings

MediaLive will perform a failover if content is not detected in this input for the specified period.

Type: InputLossFailoverSettings
Required: False

videoBlackSettings

MediaLive will perform a failover if content is considered black for the specified period.

Type: VideoBlackFailoverSettings
Required: False

FeatureActivations

Feature Activations

inputPrepareScheduleActions

Enables the Input Prepare feature. You can create Input Prepare actions in the schedule only if this feature is enabled. If you disable the feature on an existing schedule, make sure that you first delete all input prepare actions from the schedule.

Type: FeatureActivationsInputPrepareScheduleActions
Required: False

FeatureActivationsInputPrepareScheduleActions

Feature Activations Input Prepare Schedule Actions

```
DISABLED
ENABLED
```

FecOutputIncludeFec

Fec Output Include Fec

COLUMN
COLUMN_AND_ROW

FecOutputSettings

Fec Output Settings

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: FecOutputIncludeFec
Required: False

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

FixedAfd

Fixed Afd

AFD_0000
AFD_0010
AFD_0011
AFD_0100
AFD_1000
AFD_1001
AFD_1010
AFD_1011
AFD_1101
AFD_1110
AFD_1111

Fmp4HlsSettings

Fmp4 Hls Settings

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

nielsenId3Behavior

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

Type: Fmp4NielsenId3Behavior
Required: False

timedMetadataBehavior

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

Type: Fmp4TimedMetadataBehavior
Required: False

Fmp4NielsenId3Behavior

Fmp4 Nielsen Id3 Behavior

NO_PASSTHROUGH
PASSTHROUGH

Fmp4TimedMetadataBehavior

Fmp4 Timed Metadata Behavior

NO_PASSTHROUGH
PASSTHROUGH

FrameCaptureCdnSettings

Frame Capture Cdn Settings

frameCaptureS3Settings

Type: FrameCaptureS3Settings
Required: False

FrameCaptureGroupSettings

Frame Capture Group Settings

destination

The destination for the frame capture files. Either the URI for an Amazon S3 bucket and object, plus a file name prefix (for example, s3ssl://sportsDelivery/highlights/20180820/curling-) or the URI for a MediaStore container, plus a file name prefix (for example, mediastoressl://

sportsDelivery/20180820/curling-). The final file names consist of the prefix from the destination field (for example, "curling-") + name modifier + the counter (5 digits, starting from 00001) + extension (which is always .jpg). For example, curling-low.00001.jpg

Type: OutputLocationRef
Required: True

frameCaptureCdnSettings

Parameters that control interactions with the CDN.

Type: FrameCaptureCdnSettings
Required: False

FrameCaptureHlsSettings

Frame Capture Hls Settings

FrameCaptureIntervalUnit

Frame Capture Interval Unit

```
MILLISECONDS
SECONDS
```

FrameCaptureOutputSettings

Frame Capture Output Settings

nameModifier

Required if the output group contains more than one output. This modifier forms part of the output file name.

Type: string
Required: False

FrameCaptureS3LogUploads

Frame Capture S3 Log Uploads

DISABLED
ENABLED

FrameCaptureS3Settings

Frame Capture S3 Settings

cannedAcl

Specify the canned ACL to apply to each S3 request. Defaults to none.

Type: S3CannedAcl
Required: False

logUploads

When set to enabled, each upload to CDN or server will be logged.

Type: FrameCaptureS3LogUploads
Required: False

FrameCaptureSettings

Frame Capture Settings

captureInterval

The frequency at which to capture frames for inclusion in the output. May be specified in either seconds or milliseconds, as specified by captureIntervalUnits.

Type: integer
Required: False
Minimum: 1
Maximum: 3600000

captureIntervalUnits

Unit for the frame capture interval.

Type: FrameCaptureIntervalUnit

Required: False

GatewayTimeoutException

message

Type: string
Required: False

GlobalConfiguration

Global Configuration

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 the current input completes (e.g. end-of-file). When switchAndLoopInputs is configured the encoder will restart at the beginning of the first input. When "none" is configured the encoder will transcode either black, a solid color, or a user specified slate images per the "Input Loss Behavior" configuration until the next input switch occurs (which is controlled through the Channel Schedule API).

Type: GlobalConfigurationInputEndAction
Required: False

inputLossBehavior

Settings for system actions when input is lost.

Type: InputLossBehavior
Required: False

outputLockingMode

Indicates how MediaLive pipelines are synchronized. PIPELINE_LOCKING - MediaLive will attempt to synchronize the output of each pipeline to the other. EPOCH_LOCKING - MediaLive will attempt to synchronize the output of each pipeline to the Unix epoch.

Type: GlobalConfigurationOutputLockingMode
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: GlobalConfigurationOutputTimingSource
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: GlobalConfigurationLowFramerateInputs
Required: False

GlobalConfigurationInputEndAction

Global Configuration Input End Action

```
NONE
SWITCH_AND_LOOP_INPUTS
```

GlobalConfigurationLowFramerateInputs

Global Configuration Low Framerate Inputs

```
DISABLED
ENABLED
```

GlobalConfigurationOutputLockingMode

Global Configuration Output Locking Mode

EPOCH_LOCKING
PIPELINE_LOCKING

GlobalConfigurationOutputTimingSource

Global Configuration Output Timing Source

INPUT_CLOCK
SYSTEM_CLOCK

H264AdaptiveQuantization

H264 Adaptive Quantization

AUTO
HIGH
HIGHER
LOW
MAX
MEDIUM
OFF

H264ColorMetadata

H264 Color Metadata

IGNORE
INSERT

H264ColorSpaceSettings

H264 Color Space Settings

colorSpacePassthroughSettings

Type: ColorSpacePassthroughSettings
Required: False

rec601Settings

Type: Rec601Settings
Required: False

rec709Settings

Type: Rec709Settings
Required: False

H264EntropyEncoding

H264 Entropy Encoding

CABAC
CAVLC

H264FilterSettings

H264 Filter Settings

temporalFilterSettings

Type: TemporalFilterSettings
Required: False

H264FlickerAq

H264 Flicker Aq

DISABLED
ENABLED

H264ForceFieldPictures

H264 Force Field Pictures

DISABLED
ENABLED

H264FramerateControl

H264 Framerate Control

INITIALIZE_FROM_SOURCE
SPECIFIED

H264GopBReference

H264 Gop BReference

DISABLED
ENABLED

H264GopSizeUnits

H264 Gop Size Units

FRAMES
SECONDS

H264Level

H264 Level

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

H264 Look Ahead Rate Control

HIGH
LOW
MEDIUM

H264ParControl

H264 Par Control

INITIALIZE_FROM_SOURCE
SPECIFIED

H264Profile

H264 Profile

BASELINE
HIGH
HIGH_10BIT
HIGH_422

HIGH_422_10BIT

MAIN

H264QualityLevel

H264 Quality Level

ENHANCED_QUALITY

STANDARD_QUALITY

H264RateControlMode

H264 Rate Control Mode

CBR

MULTIPLEX

QVBR

VBR

H264ScanType

H264 Scan Type

INTERLACED

PROGRESSIVE

H264SceneChangeDetect

H264 Scene Change Detect

DISABLED

ENABLED

H264Settings

H264 Settings

adaptiveQuantization

Enables or disables adaptive quantization, which is a technique MediaLive can apply to video on a frame-by-frame basis to produce more compression without losing quality. There are three types of adaptive quantization: flicker, spatial, and temporal. Set the field in one of these ways: Set to Auto. Recommended. For each type of AQ, MediaLive will determine if AQ is needed, and if so, the appropriate strength. Set a strength (a value other than Auto or Disable). This strength will apply to any of the AQ fields that you choose to enable. Set to Disabled to disable all types of adaptive quantization.

Type: H264AdaptiveQuantization
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: AfdSignaling
Required: False

bitrate

Average bitrate in bits/second. Required when the rate control mode is VBR or CBR. Not used for QVBR. In an MS Smooth output group, each output must have a unique value when its bitrate is 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

bufSize

Size of buffer (HRD buffer model) in bits.

Type: integer
Required: False
Minimum: 0

colorMetadata

Includes colorspace metadata in the output.

Type: H264ColorMetadata
Required: False

colorSpaceSettings

Color Space settings

Type: H264ColorSpaceSettings
Required: False

entropyEncoding

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

Type: H264EntropyEncoding
Required: False

filterSettings

Optional filters that you can apply to an encode.

Type: H264FilterSettings
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: FixedAfd
Required: False

flickerAq

Flicker AQ makes adjustments within each frame to reduce flicker or 'pop' on I-frames. The value to enter in this field depends on the value in the Adaptive quantization field: If you have set the Adaptive quantization field to Auto, MediaLive ignores any value in this field. MediaLive will determine if flicker AQ is appropriate and will apply the appropriate strength. If you have set the Adaptive quantization field to a strength, you can set this field to Enabled or Disabled. Enabled: MediaLive will apply flicker AQ using the specified strength. Disabled: MediaLive won't apply flicker AQ. If you have set the Adaptive quantization to Disabled, MediaLive ignores any value in this field and doesn't apply flicker AQ.

Type: H264FlickerAq
Required: False

forceFieldPictures

This setting applies only when scan type is "interlaced." It controls whether coding is performed on a field basis or on a frame basis. (When the video is progressive, the coding is always performed on a frame basis.) enabled: Force MediaLive to code on a field basis, so that odd and even sets of fields are coded separately. disabled: Code the two sets of fields separately (on a field basis) or together (on a frame basis using PAFF), depending on what is most appropriate for the content.

Type: H264ForceFieldPictures
Required: False

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: H264FramerateControl
Required: False

framerateDenominator

Framerate denominator.

Type: integer
Required: False
Minimum: 1

framerateNumerator

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

Type: integer
Required: False
Minimum: 1

gopBReference

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

Type: H264GopBReference
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

gopNumBFrames

Number of B-frames between reference frames.

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

gopSize

GOP size (keyframe interval) in units of either frames or seconds per gopSizeUnits. If gopSizeUnits is frames, gopSize must be an integer and must be greater than or equal to 1. If gopSizeUnits is seconds, gopSize must be greater than 0, but need not be an integer.

Type: number
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: H264GopSizeUnits
Required: False

level

H.264 Level.

Type: H264Level
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: H264LookAheadRateControl

Required: False

maxBitrate

For QVBR: See the tooltip for Quality level For VBR: Set the maximum bitrate in order to accommodate expected spikes in the complexity of the video.

Type: integer
Required: False
Minimum: 1000

minIInterval

Only meaningful if sceneChangeDetect is set to enabled. Defaults to 5 if multiplex rate control is used. 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

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

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 pixsel aspect ratio will be set equal to the input video pixel aspect ratio of the first input.

Type: H264ParControl
Required: False

parDenominator

Pixel Aspect Ratio denominator.

Type: integer
Required: False
Minimum: 1

parNumerator

Pixel Aspect Ratio numerator.

Type: integer
Required: False
Minimum: 1

profile

H.264 Profile.

Type: H264Profile
Required: False

qualityLevel

Leave as STANDARD_QUALITY or choose a different value (which might result in additional costs to run the channel). - ENHANCED_QUALITY: Produces a slightly better video quality without an increase in the bitrate. Has an effect only when the Rate control mode is QVBR

or CBR. If this channel is in a MediaLive multiplex, the value must be ENHANCED_QUALITY. - STANDARD_QUALITY: Valid for any Rate control mode.

Type: H264QualityLevel
Required: False

qvbrQualityLevel

Controls the target quality for the video encode. Applies only when the rate control mode is QVBR. You can set a target quality or you can let MediaLive determine the best quality. To set a target quality, enter values in the QVBR quality level field and the Max bitrate field. Enter values that suit your most important viewing devices. Recommended values are: - Primary screen: Quality level: 8 to 10. Max bitrate: 4M - PC or tablet: Quality level: 7. Max bitrate: 1.5M to 3M - Smartphone: Quality level: 6. Max bitrate: 1M to 1.5M To let MediaLive decide, leave the QVBR quality level field empty, and in Max bitrate enter the maximum rate you want in the video. For more information, see the section called "Video - rate control mode" in the MediaLive user guide

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

rateControlMode

Rate control mode. QVBR: Quality will match the specified quality level except when it is constrained by the maximum bitrate. Recommended if you or your viewers pay for bandwidth. VBR: Quality and bitrate vary, depending on the video complexity. Recommended instead of QVBR if you want to maintain a specific average bitrate over the duration of the channel. CBR: Quality varies, depending on the video complexity. Recommended only if you distribute your assets to devices that cannot handle variable bitrates. Multiplex: This rate control mode is only supported (and is required) when the video is being delivered to a MediaLive Multiplex in which case the rate control configuration is controlled by the properties within the Multiplex Program.

Type: H264RateControlMode
Required: False

scanType

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

Type: H264ScanType
Required: False

sceneChangeDetect

Scene change detection. - On: inserts I-frames when scene change is detected. - Off: does not force an I-frame when scene change is detected.

Type: H264SceneChangeDetect
Required: False

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

softness

Softness. Selects quantizer matrix, larger values reduce high-frequency content in the encoded image. If not set to zero, must be greater than 15.

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

spatialAq

Spatial AQ makes adjustments within each frame based on spatial variation of content complexity. The value to enter in this field depends on the value in the Adaptive quantization field: If you have set the Adaptive quantization field to Auto, MediaLive ignores any value in this field. MediaLive will determine if spatial AQ is appropriate and will apply the appropriate strength. If you have set the Adaptive quantization field to a strength, you can set this field to Enabled or Disabled. Enabled: MediaLive will apply spatial AQ using the specified strength. Disabled: MediaLive won't apply spatial AQ. If you have set the Adaptive quantization to Disabled, MediaLive ignores any value in this field and doesn't apply spatial AQ.

Type: H264SpatialAq
Required: False

subgopLength

If set to fixed, use gopNumBFrames B-frames per sub-GOP. If set to dynamic, optimize the number of B-frames used for each sub-GOP to improve visual quality.

Type: H264SubGopLength
Required: False

syntax

Produces a bitstream compliant with SMPTE RP-2027.

Type: H264Syntax
Required: False

temporalAq

Temporal makes adjustments within each frame based on temporal variation of content complexity. The value to enter in this field depends on the value in the Adaptive quantization field: If you have set the Adaptive quantization field to Auto, MediaLive ignores any value in this field. MediaLive will determine if temporal AQ is appropriate and will apply the appropriate strength. If you have set the Adaptive quantization field to a strength, you can set this field to Enabled or Disabled. Enabled: MediaLive will apply temporal AQ using the specified strength. Disabled:

MediaLive won't apply temporal AQ. If you have set the Adaptive quantization to Disabled, MediaLive ignores any value in this field and doesn't apply temporal AQ.

Type: H264TemporalAq
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: H264TimecodeInsertionBehavior
Required: False

H264SpatialAq

H264 Spatial Aq

```
DISABLED
ENABLED
```

H264SubGopLength

H264 Sub Gop Length

```
DYNAMIC
FIXED
```

H264Syntax

H264 Syntax

```
DEFAULT
RP2027
```

H264TemporalAq

H264 Temporal Aq

```
DISABLED
ENABLED
```

H264TimecodeInsertionBehavior

H264 Timecode Insertion Behavior

```
DISABLED
PIC_TIMING_SEI
```

H265AdaptiveQuantization

H265 Adaptive Quantization

```
AUTO
HIGH
HIGHER
LOW
MAX
MEDIUM
OFF
```

H265AlternativeTransferFunction

H265 Alternative Transfer Function

```
INSERT
OMIT
```

H265ColorMetadata

H265 Color Metadata

```
IGNORE
```

INSERT

H265ColorSpaceSettings

H265 Color Space Settings

colorSpacePassthroughSettings

Type: ColorSpacePassthroughSettings
Required: False

hdr10Settings

Type: Hdr10Settings
Required: False

rec601Settings

Type: Rec601Settings
Required: False

rec709Settings

Type: Rec709Settings
Required: False

H265FilterSettings

H265 Filter Settings

temporalFilterSettings

Type: TemporalFilterSettings
Required: False

H265FlickerAq

H265 Flicker Aq

DISABLED
ENABLED

H265GopSizeUnits

H265 Gop Size Units

FRAMES
SECONDS

H265Level

H265 Level

H265_LEVEL_1
H265_LEVEL_2
H265_LEVEL_2_1
H265_LEVEL_3
H265_LEVEL_3_1
H265_LEVEL_4
H265_LEVEL_4_1
H265_LEVEL_5
H265_LEVEL_5_1
H265_LEVEL_5_2
H265_LEVEL_6
H265_LEVEL_6_1
H265_LEVEL_6_2
H265_LEVEL_AUTO

H265LookAheadRateControl

H265 Look Ahead Rate Control

HIGH
LOW
MEDIUM

H265Profile

H265 Profile

```
MAIN
MAIN_10BIT
```

H265RateControlMode

H265 Rate Control Mode

```
CBR
MULTIPLEX
QVBR
```

H265ScanType

H265 Scan Type

```
INTERLACED
PROGRESSIVE
```

H265SceneChangeDetect

H265 Scene Change Detect

```
DISABLED
ENABLED
```

H265Settings

H265 Settings

adaptiveQuantization

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

Type: H265AdaptiveQuantization

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: AfdSignaling
Required: False

alternativeTransferFunction

Whether or not EML should insert an Alternative Transfer Function SEI message to support backwards compatibility with non-HDR decoders and displays.

Type: H265AlternativeTransferFunction
Required: False

bitrate

Average bitrate in bits/second. Required when the rate control mode is VBR or CBR. Not used for QVBR. In an MS Smooth output group, each output must have a unique value when its bitrate is rounded down to the nearest multiple of 1000.

Type: integer
Required: False
Minimum: 100000
Maximum: 40000000

bufSize

Size of buffer (HRD buffer model) in bits.

Type: integer
Required: False
Minimum: 100000
Maximum: 80000000

colorMetadata

Includes colorspace metadata in the output.

Type: H265ColorMetadata
Required: False

colorSpaceSettings

Color Space settings

Type: H265ColorSpaceSettings
Required: False

filterSettings

Optional filters that you can apply to an encode.

Type: H265FilterSettings
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: FixedAfd
Required: False

flickerAq

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

Type: H265FlickerAq
Required: False

framerateDenominator

Framerate denominator.

Type: integer
Required: True
Minimum: 1
Maximum: 3003

framerateNumerator

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

Type: integer
Required: True
Minimum: 1

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

gopSize

GOP size (keyframe interval) in units of either frames or seconds per gopSizeUnits. If gopSizeUnits is frames, gopSize must be an integer and must be greater than or equal to 1. If gopSizeUnits is seconds, gopSize must be greater than 0, but need not be an integer.

Type: number
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: H265GopSizeUnits

Required: False

level

H.265 Level.

Type: H265Level
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: H265LookAheadRateControl
Required: False

maxBitrate

For QVBR: See the tooltip for Quality level

Type: integer
Required: False
Minimum: 100000
Maximum: 40000000

minIInterval

Only meaningful if sceneChangeDetect is set to enabled. Defaults to 5 if multiplex rate control is used. 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

parDenominator

Pixel Aspect Ratio denominator.

Type: integer
Required: False
Minimum: 1

parNumerator

Pixel Aspect Ratio numerator.

Type: integer
Required: False
Minimum: 1

profile

H.265 Profile.

Type: H265Profile
Required: False

qvbrQualityLevel

Controls the target quality for the video encode. Applies only when the rate control mode is QVBR. Set values for the QVBR quality level field and Max bitrate field that suit your most important viewing devices. Recommended values are: - Primary screen: Quality level: 8 to 10. Max bitrate: 4M - PC or tablet: Quality level: 7. Max bitrate: 1.5M to 3M - Smartphone: Quality level: 6. Max bitrate: 1M to 1.5M

Type: integer
Required: False
Minimum: 1

Maximum: 10

rateControlMode

Rate control mode. QVBR: Quality will match the specified quality level except when it is constrained by the maximum bitrate. Recommended if you or your viewers pay for bandwidth. CBR: Quality varies, depending on the video complexity. Recommended only if you distribute your assets to devices that cannot handle variable bitrates. Multiplex: This rate control mode is only supported (and is required) when the video is being delivered to a MediaLive Multiplex in which case the rate control configuration is controlled by the properties within the Multiplex Program.

Type: H265RateControlMode
Required: False

scanType

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

Type: H265ScanType
Required: False

sceneChangeDetect

Scene change detection.

Type: H265SceneChangeDetect
Required: False

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: 16

tier

H.265 Tier.

Type: H265Tier
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: H265TimecodeInsertionBehavior
Required: False

H265Tier

H265 Tier

```
HIGH
MAIN
```

H265TimecodeInsertionBehavior

H265 Timecode Insertion Behavior

```
DISABLED
PIC_TIMING_SEI
```

Hdr10Settings

Hdr10 Settings

maxCll

Maximum Content Light Level An integer metadata value defining the maximum light level, in nits, of any single pixel within an encoded HDR video stream or file.

Type: integer
Required: False
Minimum: 0
Maximum: 32768

maxFall

Maximum Frame Average Light Level An integer metadata value defining the maximum average light level, in nits, for any single frame within an encoded HDR video stream or file.

Type: integer
Required: False
Minimum: 0
Maximum: 32768

HlsAdMarkers

Hls Ad Markers

ADOBE
ELEMENTAL
ELEMENTAL_SCTE35

HlsAkamaiHttpTransferMode

Hls Akamai Http Transfer Mode

CHUNKED
NON_CHUNKED

HlsAkamaiSettings

Hls Akamai Settings

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

httpTransferMode

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

Type: HlsAkamaiHttpTransferMode
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

salt

Salt for authenticated Akamai.

Type: string
Required: False

token

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

Type: string
Required: False

HlsBasicPutSettings

Hls Basic Put Settings

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

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

HlsCaptionLanguageSetting

Hls Caption Language Setting

```
INSERT
NONE
OMIT
```

HlsCdnSettings

Hls Cdn Settings

hlsAkamaiSettings

Type: HlsAkamaiSettings
Required: False

hlsBasicPutSettings

Type: HlsBasicPutSettings

Required: False

hlsMediaStoreSettings

Type: HlsMediaStoreSettings
Required: False

hlsS3Settings

Type: HlsS3Settings
Required: False

hlsWebdavSettings

Type: HlsWebdavSettings
Required: False

HlsClientCache

Hls Client Cache

DISABLED
ENABLED

HlsCodecSpecification

Hls Codec Specification

RFC_4281
RFC_6381

HlsDirectoryStructure

Hls Directory Structure

SINGLE_DIRECTORY
SUBDIRECTORY_PER_STREAM

HlsDiscontinuityTags

Hls Discontinuity Tags

```
INSERT
NEVER_INSERT
```

HlsEncryptionType

Hls Encryption Type

```
AES128
SAMPLE_AES
```

HlsGroupSettings

Hls Group Settings

adMarkers

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

Type: Array of type HlsAdMarkers
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

baseUrlContent1

Optional. One value per output group. This field is required only if you are completing Base URL content A, and the downstream system has notified you that the media files for pipeline 1 of all outputs are in a location different from the media files for pipeline 0.

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

baseUrlManifest1

Optional. One value per output group. Complete this field only if you are completing Base URL manifest A, and the downstream system has notified you that the child manifest files for pipeline 1 of all outputs are in a location different from the child manifest files for pipeline 0.

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

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: HlsCaptionLanguageSetting
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: HlsClientCache
Required: False

codecSpecification

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

Type: HlsCodecSpecification
Required: False

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
MinLength: 32
MaxLength: 32

destination

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

Type: OutputLocationRef
Required: True

directoryStructure

Place segments in subdirectories.

Type: HlsDirectoryStructure
Required: False

discontinuityTags

Specifies whether to insert EXT-X-DISCONTINUITY tags in the HLS child manifests for this output group. Typically, choose Insert because these tags are required in the manifest (according to the HLS specification) and serve an important purpose. Choose Never Insert only if the downstream system is doing real-time failover (without using the MediaLive automatic failover feature) and only if that downstream system has advised you to exclude the tags.

Type: HlsDiscontinuityTags
Required: False

encryptionType

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

Type: HlsEncryptionType
Required: False

hlsCdnSettings

Parameters that control interactions with the CDN.

Type: HlsCdnSettings
Required: False

hlsId3SegmentTagging

State of HLS ID3 Segment Tagging

Type: HlsId3SegmentTaggingState
Required: False

iFrameOnlyPlaylists

DISABLED: Do not create an I-frame-only manifest, but do create the master and media manifests (according to the Output Selection field). STANDARD: Create an I-frame-only manifest for each output that contains video, as well as the other manifests (according to the Output Selection field). The I-frame manifest contains a #EXT-X-I-FRAMES-ONLY tag to indicate it is I-frame only, and one or more #EXT-X-BYTERANGE entries identifying the I-frame position. For example, #EXT-X-BYTERANGE:160364@1461888"

Type: IFrameOnlyPlaylistType
Required: False

incompleteSegmentBehavior

Specifies whether to include the final (incomplete) segment in the media output when the pipeline stops producing output because of a channel stop, a channel pause or a loss of input to the pipeline. Auto means that MediaLive decides whether to include the final segment, depending on the channel class and the types of output groups. Suppress means to never include the incomplete segment. We recommend you choose Auto and let MediaLive control the behavior.

Type: HlsIncompleteSegmentBehavior
Required: False

indexNSegments

Applies only if Mode field is LIVE. Specifies the maximum number of segments in the media manifest file. After this maximum, older segments are removed from the media manifest. This number must be smaller than the number in the Keep Segments field.

Type: integer
Required: False
Minimum: 3

inputLossAction

Parameter that control output group behavior on input loss.

Type: InputLossActionForHlsOut

Required: False

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: HlsIvInManifest
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: HlsIvSource
Required: False

keepSegments

Applies only if Mode field is LIVE. Specifies the number of media segments to retain in the destination directory. This number should be bigger than indexNSegments (Num segments). We recommend (value = (2 x indexNsegments) + 1). If this "keep segments" number is too low, the following might happen: the player is still reading a media manifest file that lists this segment, but that segment has been removed from the destination directory (as directed by indexNSegments). This situation would result in a 404 HTTP error on the player.

Type: integer
Required: False
Minimum: 1

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

keyFormatVersions

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

Type: string
Required: False

keyProviderSettings

The key provider settings.

Type: KeyProviderSettings
Required: False

manifestCompression

When set to gzip, compresses HLS playlist.

Type: HlsManifestCompression
Required: False

manifestDurationFormat

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

Type: HlsManifestDurationFormat
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 "vod", all segments are indexed and kept permanently in the destination and manifest. If "live", only the number segments specified in keepSegments and indexNSegments are kept; newer segments replace older segments, which may prevent players from rewinding all the way to the beginning of the event. VOD mode uses HLS EXT-X-PLAYLIST-TYPE of EVENT while the channel is running, converting it to a "VOD" type manifest on completion of the stream.

Type: HlsMode
Required: False

outputSelection

MANIFESTS_AND_SEGMENTS: Generates manifests (master manifest, if applicable, and media manifests) for this output group. VARIANT_MANIFESTS_AND_SEGMENTS: Generates media manifests for this output group, but not a master manifest. SEGMENTS_ONLY: Does not generate any manifests for this output group.

Type: HlsOutputSelection
Required: False

programDateTime

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

Type: HlsProgramDateTime
Required: False

programDateTimeClock

Specifies the algorithm used to drive the HLS EXT-X-PROGRAM-DATE-TIME clock. Options include: INITIALIZE_FROM_OUTPUT_TIMECODE: The PDT clock is initialized as a function of the first output timecode, then incremented by the EXTINF duration of each encoded segment. SYSTEM_CLOCK:

The PDT clock is initialized as a function of the UTC wall clock, then incremented by the EXTINF duration of each encoded segment. If the PDT clock diverges from the wall clock by more than 500ms, it is resynchronized to the wall clock.

Type: HlsProgramDateTimeClock
Required: False

programDateTimePeriod

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

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

redundantManifest

ENABLED: The master manifest (.m3u8 file) for each pipeline includes information about both pipelines: first its own media files, then the media files of the other pipeline. This feature allows playout device that support stale manifest detection to switch from one manifest to the other, when the current manifest seems to be stale. There are still two destinations and two master manifests, but both master manifests reference the media files from both pipelines. DISABLED: The master manifest (.m3u8 file) for each pipeline includes information about its own pipeline only. For an HLS output group with MediaPackage as the destination, the DISABLED behavior is always followed. MediaPackage regenerates the manifests it serves to players so a redundant manifest from MediaLive is irrelevant.

Type: HlsRedundantManifest
Required: False

segmentLength

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

Type: integer
Required: False

Minimum: 1

segmentationMode

useInputSegmentation has been deprecated. The configured segment size is always used.

Type: HlsSegmentationMode
Required: False

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

streamInfResolution

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

Type: HlsStreamInfResolution
Required: False

timedMetadataId3Frame

Indicates ID3 frame that has the timecode.

Type: HlsTimedMetadataId3Frame
Required: False

timedMetadataId3Period

Timed Metadata interval in seconds.

Type: integer
Required: False
Minimum: 0

timestampDeltaMilliseconds

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

Type: integer
Required: False
Minimum: 0

tsFileMode

SEGMENTED_FILES: Emit the program as segments - multiple .ts media files. SINGLE_FILE: Applies only if Mode field is VOD. Emit the program as a single .ts media file. The media manifest includes #EXT-X-BYTERANGE tags to index segments for playback. A typical use for this value is when sending the output to AWS Elemental MediaConvert, which can accept only a single media file. Playback while the channel is running is not guaranteed due to HTTP server caching.

Type: HlsTsFileMode
Required: False

HlsH265PackagingType

Hls H265 Packaging Type

HEV1
HVC1

HlsId3SegmentTaggingState

State of HLS ID3 Segment Tagging

DISABLED
ENABLED

HlsIncompleteSegmentBehavior

Hls Incomplete Segment Behavior

AUTO

SUPPRESS

HlsInputSettings

Hls Input Settings

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

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

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

retryInterval

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

Type: integer
Required: False
Minimum: 0

scte35Source

Identifies the source for the SCTE-35 messages that MediaLive will ingest. Messages can be ingested from the content segments (in the stream) or from tags in the playlist (the HLS manifest). MediaLive ignores SCTE-35 information in the source that is not selected.

Type: HlsScte35SourceType
Required: False

HlsIvInManifest

Hls Iv In Manifest

```
EXCLUDE
INCLUDE
```

HlsIvSource

Hls Iv Source

```
EXPLICIT
FOLLOWS_SEGMENT_NUMBER
```

HlsManifestCompression

Hls Manifest Compression

```
GZIP
NONE
```

HlsManifestDurationFormat

Hls Manifest Duration Format

FLOATING_POINT
INTEGER

HlsMediaStoreSettings

Hls Media Store Settings

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

mediaStoreStorageClass

When set to temporal, output files are stored in non-persistent memory for faster reading and writing.

Type: HlsMediaStoreStorageClass
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

HlsMediaStoreStorageClass

Hls Media Store Storage Class

```
TEMPORAL
```

HlsMode

Hls Mode

```
LIVE
VOD
```

HlsOutputSelection

Hls Output Selection

```
MANIFESTS_AND_SEGMENTS
SEGMENTS_ONLY
VARIANT_MANIFESTS_AND_SEGMENTS
```

HlsOutputSettings

Hls Output Settings

h265PackagingType

Only applicable when this output is referencing an H.265 video description. Specifies whether MP4 segments should be packaged as HEV1 or HVC1.

Type: HlsH265PackagingType
Required: False

hlsSettings

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

Type: HlsSettings
Required: True

nameModifier

String concatenated to the end of the destination filename. Accepts \"Format Identifiers \":#formatIdentifierParameters.

Type: string
Required: False
MinLength: 1

segmentModifier

String concatenated to end of segment filenames.

Type: string
Required: False

HlsProgramDateTime

Hls Program Date Time

EXCLUDE
INCLUDE

HlsProgramDateTimeClock

Hls Program Date Time Clock

INITIALIZE_FROM_OUTPUT_TIMECODE

SYSTEM_CLOCK

HlsRedundantManifest

Hls Redundant Manifest

DISABLED
ENABLED

HlsS3LogUploads

Hls S3 Log Uploads

DISABLED
ENABLED

HlsS3Settings

Hls S3 Settings

cannedAcl

Specify the canned ACL to apply to each S3 request. Defaults to none.

Type: S3CannedAcl
Required: False

logUploads

When set to enabled, each fragment upload to CDN or server will be logged.

Type: HlsS3LogUploads
Required: False

HlsScte35SourceType

Hls Scte35 Source Type

MANIFEST

SEGMENTS

HlsSegmentationMode

Hls Segmentation Mode

USE_INPUT_SEGMENTATION
USE_SEGMENT_DURATION

HlsSettings

Hls Settings

audioOnlyHlsSettings

Type: AudioOnlyHlsSettings
Required: False

fmp4HlsSettings

Type: Fmp4HlsSettings
Required: False

frameCaptureHlsSettings

Type: FrameCaptureHlsSettings
Required: False

standardHlsSettings

Type: StandardHlsSettings
Required: False

HlsStreamInfResolution

Hls Stream Inf Resolution

EXCLUDE

INCLUDE

HlsTimedMetadataId3Frame

Hls Timed Metadata Id3 Frame

NONE
PRIV
TDRL

HlsTsFileMode

Hls Ts File Mode

SEGMENTED_FILES
SINGLE_FILE

HlsWebdavHttpTransferMode

Hls Webdav Http Transfer Mode

CHUNKED
NON_CHUNKED

HlsWebdavSettings

Hls Webdav Settings

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

httpTransferMode

Specify whether or not to use chunked transfer encoding to WebDAV.

Type: HlsWebdavHttpTransferMode
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

HtmlMotionGraphicsSettings

Html Motion Graphics Settings

IFrameOnlyPlaylistType

When set to "standard", an I-Frame only playlist will be written out for each video output in the output group. This I-Frame only playlist will contain byte range offsets pointing to the I-frame(s) in each segment.

DISABLED
STANDARD

InputAttachment

automaticInputFailoverSettings

User-specified settings for defining what the conditions are for declaring the input unhealthy and failing over to a different input.

Type: AutomaticInputFailoverSettings
Required: False

inputAttachmentName

User-specified name for the attachment. This is required if the user wants to use this input in an input switch action.

Type: string
Required: False

inputId

The ID of the input

Type: string
Required: False

inputSettings

Settings of an input (caption selector, etc.)

Type: InputSettings
Required: False

InputChannelLevel

Input Channel Level

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

inputChannel

The index of the input channel used as a source.

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

InputCodec

codec in increasing order of complexity

MPEG2
AVC
HEVC

InputDeblockFilter

Input Deblock Filter

DISABLED
ENABLED

InputDenoiseFilter

Input Denoise Filter

DISABLED
ENABLED

InputFilter

Input Filter

AUTO
DISABLED
FORCED

InputLocation

Input Location

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 RTMP destination should have a uri simliar 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

Input Loss Action For Hls Out

```
EMIT_OUTPUT
PAUSE_OUTPUT
```

InputLossActionForMsSmoothOut

Input Loss Action For Ms Smooth Out

```
EMIT_OUTPUT
PAUSE_OUTPUT
```

InputLossActionForRtmpOut

Input Loss Action For Rtmp Out

```
EMIT_OUTPUT
PAUSE_OUTPUT
```

InputLossActionForUdpOut

Input Loss Action For Udp Out

```
DROP_PROGRAM
DROP_TS
EMIT_PROGRAM
```

InputLossBehavior

Input Loss Behavior

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

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
MinLength: 6
MaxLength: 6

inputLossImageSlate

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

Type: InputLocation
Required: False

inputLossImageType

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

Type: InputLossImageType
Required: False

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

InputLossFailoverSettings

MediaLive will perform a failover if content is not detected in this input for the specified period.

inputLossThresholdMsec

The amount of time (in milliseconds) that no input is detected. After that time, an input failover will occur.

Type: integer
Required: False
Minimum: 100

InputLossImageType

Input Loss Image Type

COLOR
SLATE

InputMaximumBitrate

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

MAX_10_MBPS
MAX_20_MBPS
MAX_50_MBPS

InputPreference

Input preference when deciding which input to make active when a previously failed input has recovered. If \"EQUAL_INPUT_PREFERENCE\", then the active input will stay active as long as it is healthy. If \"PRIMARY_INPUT_PREFERRED\", then always switch back to the primary input when it is healthy.

EQUAL_INPUT_PREFERENCE

PRIMARY_INPUT_PREFERRED

InputResolution

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

Live Event input parameters. There can be multiple inputs in a single Live Event.

audioSelectors

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

Type: Array of type AudioSelector
Required: False

captionSelectors

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

Type: Array of type CaptionSelector
Required: False

deblockFilter

Enable or disable the deblock filter when filtering.

Type: InputDeblockFilter
Required: False

denoiseFilter

Enable or disable the denoise filter when filtering.

Type: InputDenoiseFilter
Required: False

filterStrength

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

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

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: InputFilter
Required: False

networkInputSettings

Input settings.

Type: NetworkInputSettings
Required: False

smpte2038DataPreference

Specifies whether to extract applicable ancillary data from a SMPTE-2038 source in this input. Applicable data types are captions, timecode, AFD, and SCTE-104 messages. - PREFER: Extract from SMPTE-2038 if present in this input, otherwise extract from another source (if any). - IGNORE: Never extract any ancillary data from SMPTE-2038.

Type: Smpte2038DataPreference
Required: False

sourceEndBehavior

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

Type: InputSourceEndBehavior
Required: False

videoSelector

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

Type: VideoSelector
Required: False

InputSourceEndBehavior

Input Source End Behavior

CONTINUE
LOOP

InputSpecification

codec

Input codec

Type: InputCodec
Required: False

maximumBitrate

Maximum input bitrate, categorized coarsely

Type: InputMaximumBitrate
Required: False

resolution

Input resolution, categorized coarsely

Type: InputResolution
Required: False

InternalServiceError

message

Type: string
Required: False

InvalidRequest

message

Type: string
Required: False

KeyProviderSettings

Key Provider Settings

staticKeySettings

Type: StaticKeySettings
Required: False

LimitExceeded

message

Type: string
Required: False

ListChannelsResultModel

channels

Type: Array of type ChannelSummary
Required: False

nextToken

Type: string
Required: False

LogLevel

The log level the user wants for their channel.

ERROR
WARNING
INFO
DEBUG
DISABLED

M2tsAbsentInputAudioBehavior

M2ts Absent Input Audio Behavior

DROP
ENCODE_SILENCE

M2tsArib

M2ts Arib

DISABLED
ENABLED

M2tsAribCaptionsPidControl

M2ts Arib Captions Pid Control

AUTO
USE_CONFIGURED

M2tsAudioBufferModel

M2ts Audio Buffer Model

ATSC
DVB

M2tsAudioInterval

M2ts Audio Interval

VIDEO_AND_FIXED_INTERVALS
VIDEO_INTERVAL

M2tsAudioStreamType

M2ts Audio Stream Type

ATSC
DVB

M2tsBufferModel

M2ts Buffer Model

MULTIPLEX
NONE

M2tsCcDescriptor

M2ts Cc Descriptor

DISABLED
ENABLED

M2tsEbifControl

M2ts Ebif Control

NONE
PASSTHROUGH

M2tsEbpPlacement

M2ts Ebp Placement

VIDEO_AND_AUDIO_PIDS
VIDEO_PID

M2tsEsRateInPes

M2ts Es Rate In Pes

EXCLUDE
INCLUDE

M2tsKlv

M2ts Klv

NONE
PASSTHROUGH

M2tsNielsenId3Behavior

M2ts Nielsen Id3 Behavior

NO_PASSTHROUGH
PASSTHROUGH

M2tsPcrControl

M2ts Pcr Control

CONFIGURED_PCR_PERIOD
PCR_EVERY_PES_PACKET

M2tsRateMode

M2ts Rate Mode

CBR
VBR

M2tsScte35Control

M2ts Scte35 Control

NONE
PASSTHROUGH

M2tsSegmentationMarkers

M2ts Segmentation Markers

EBP
EBP_LEGACY
NONE
PSI_SEGSTART
RAI_ADAPT
RAI_SEGSTART

M2tsSegmentationStyle

M2ts Segmentation Style

MAINTAIN_CADENCE

RESET_CADENCE

M2tsSettings

M2ts Settings

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: M2tsAbsentInputAudioBehavior
Required: False

arib

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

Type: M2tsArib
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

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: M2tsAribCaptionsPidControl
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: M2tsAudioBufferModel
Required: False

audioFramesPerPes

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

Type: integer
Required: False
Minimum: 0

audioPids

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

Type: string
Required: False

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: M2tsAudioStreamType
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

bufferModel

Controls the timing accuracy for output network traffic. Leave as MULTIPLEX to ensure accurate network packet timing. Or set to NONE, which might result in lower latency but will result in more variability in output network packet timing. This variability might cause interruptions, jitter, or bursty behavior in your playback or receiving devices.

Type: M2tsBufferModel

Required: False

ccDescriptor

When set to enabled, generates captionServiceDescriptor in PMT.

Type: M2tsCcDescriptor

Required: False

dvbNitSettings

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

Type: DvbNitSettings

Required: False

dvbSdtSettings

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

Type: DvbSdtSettings

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

dvbTdtSettings

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

Type: DvbTdtSettings
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

ebif

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

Type: M2tsEbifControl
Required: False

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: M2tsAudioInterval

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

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: M2tsEbpPlacement
Required: False

ecmPid

This field is unused and deprecated.

Type: string
Required: False

esRateInPes

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

Type: M2tsEsRateInPes
Required: False

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

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

fragmentTime

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

Type: number
Required: False
Minimum: 0

klv

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

Type: M2tsKlv
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

nielsenId3Behavior

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

Type: M2tsNielsenId3Behavior
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

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

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: M2tsPcrControl
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

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

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

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

programNum

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

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

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: M2tsRateMode
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

scte35Control

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

Type: M2tsScte35Control
Required: False

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

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: M2tsSegmentationMarkers
Required: False

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: M2tsSegmentationStyle
Required: False

segmentationTime

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

Type: number
Required: False
Minimum: 1

timedMetadataBehavior

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

Type: M2tsTimedMetadataBehavior
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

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. Valid values are 32 (or 0x20)..8182 (or 0x1ff6).

Type: string
Required: False

M2tsTimedMetadataBehavior

M2ts Timed Metadata Behavior

NO_PASSTHROUGH
PASSTHROUGH

M3u8NielsenId3Behavior

M3u8 Nielsen Id3 Behavior

NO_PASSTHROUGH
PASSTHROUGH

M3u8PcrControl

M3u8 Pcr Control

CONFIGURED_PCR_PERIOD
PCR_EVERY_PES_PACKET

M3u8Scte35Behavior

M3u8 Scte35 Behavior

NO_PASSTHROUGH
PASSTHROUGH

M3u8Settings

Settings information for the .m3u8 container

audioFramesPerPes

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

Type: integer
Required: False
Minimum: 0

audioPids

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

Type: string
Required: False

ecmPid

This parameter is unused and deprecated.

Type: string
Required: False

nielsenId3Behavior

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

Type: M3u8NielsenId3Behavior
Required: False

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

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: M3u8PcrControl
Required: False

pcrPeriod

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

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

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

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

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

programNum

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

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

scte35Behavior

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

Type: M3u8Scte35Behavior
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

timedMetadataBehavior

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

Type: M3u8TimedMetadataBehavior
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

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

M3u8TimedMetadataBehavior

M3u8 Timed Metadata Behavior

NO_PASSTHROUGH
PASSTHROUGH

MaintenanceCreateSettings

maintenanceDay

Choose one day of the week for maintenance. The chosen day is used for all future maintenance windows.

Type: MaintenanceDay
Required: False

maintenanceStartTime

Choose the hour that maintenance will start. The chosen time is used for all future maintenance windows.

Type: string
Required: False
Pattern: `^([0,1]?[0-9]|2[0-3]):00$`

MaintenanceDay

The currently selected maintenance day.

`MONDAY`
`TUESDAY`
`WEDNESDAY`
`THURSDAY`
`FRIDAY`
`SATURDAY`
`SUNDAY`

MaintenanceStatus

maintenanceDay

The currently selected maintenance day.

Type: MaintenanceDay
Required: False

maintenanceDeadline

Maintenance is required by the displayed date and time. Date and time is in ISO.

Type: string
Required: False
Format: string

maintenanceScheduledDate

The currently scheduled maintenance date and time. Date and time is in ISO.

Type: string

Required: False
Format: string

maintenanceStartTime

The currently selected maintenance start time. Time is in UTC.

Type: string
Required: False

MediaPackageGroupSettings

Media Package Group Settings

destination

MediaPackage channel destination.

Type: OutputLocationRef
Required: True

MediaPackageOutputDestinationSettings

MediaPackage Output Destination Settings

channelId

ID of the channel in MediaPackage that is the destination for this output group. You do not need to specify the individual inputs in MediaPackage; MediaLive will handle the connection of the two MediaLive pipelines to the two MediaPackage inputs. The MediaPackage channel and MediaLive channel must be in the same region.

Type: string
Required: False
MinLength: 1

MediaPackageOutputSettings

Media Package Output Settings

MotionGraphicsConfiguration

Motion Graphics Configuration

motionGraphicsInsertion

Type: MotionGraphicsInsertion
Required: False

motionGraphicsSettings

Motion Graphics Settings

Type: MotionGraphicsSettings
Required: True

MotionGraphicsInsertion

Motion Graphics Insertion

```
DISABLED
ENABLED
```

MotionGraphicsSettings

Motion Graphics Settings

htmlMotionGraphicsSettings

Type: HtmlMotionGraphicsSettings
Required: False

Mp2CodingMode

Mp2 Coding Mode

```
CODING_MODE_1_0
CODING_MODE_2_0
```

Mp2Settings

Mp2 Settings

bitrate

Average bitrate in bits/second.

Type: number
Required: False

codingMode

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

Type: Mp2CodingMode
Required: False

sampleRate

Sample rate in Hz.

Type: number
Required: False

Mpeg2AdaptiveQuantization

Mpeg2 Adaptive Quantization

```
AUTO
HIGH
LOW
MEDIUM
OFF
```

Mpeg2ColorMetadata

Mpeg2 Color Metadata

IGNORE
INSERT

Mpeg2ColorSpace

Mpeg2 Color Space

AUTO
PASSTHROUGH

Mpeg2DisplayRatio

Mpeg2 Display Ratio

DISPLAYRATIO16X9
DISPLAYRATIO4X3

Mpeg2FilterSettings

Mpeg2 Filter Settings

temporalFilterSettings

Type: TemporalFilterSettings
Required: False

Mpeg2GopSizeUnits

Mpeg2 Gop Size Units

FRAMES
SECONDS

Mpeg2ScanType

Mpeg2 Scan Type

INTERLACED

PROGRESSIVE

Mpeg2Settings

Mpeg2 Settings

adaptiveQuantization

Choose Off to disable adaptive quantization. Or choose another value to enable the quantizer and set its strength. The strengths are: Auto, Off, Low, Medium, High. When you enable this field, MediaLive allows intra-frame quantizers to vary, which might improve visual quality.

Type: Mpeg2AdaptiveQuantization
Required: False

afdSignaling

Indicates the AFD values that MediaLive will write into the video encode. If you do not know what AFD signaling is, or if your downstream system has not given you guidance, choose AUTO. AUTO: MediaLive will try to preserve the input AFD value (in cases where multiple AFD values are valid). FIXED: MediaLive will use the value you specify in fixedAFD.

Type: AfdSignaling
Required: False

colorMetadata

Specifies whether to include the color space metadata. The metadata describes the color space that applies to the video (the colorSpace field). We recommend that you insert the metadata.

Type: Mpeg2ColorMetadata
Required: False

colorSpace

Choose the type of color space conversion to apply to the output. For detailed information on setting up both the input and the output to obtain the desired color space in the output, see the section on \"MediaLive Features - Video - color space\" in the MediaLive User Guide.

PASSTHROUGH: Keep the color space of the input content - do not convert it. AUTO:Convert all content that is SD to rec 601, and convert all content that is HD to rec 709.

Type: Mpeg2ColorSpace
Required: False

displayAspectRatio

Sets the pixel aspect ratio for the encode.

Type: Mpeg2DisplayRatio
Required: False

filterSettings

Optionally specify a noise reduction filter, which can improve quality of compressed content. If you do not choose a filter, no filter will be applied. TEMPORAL: This filter is useful for both source content that is noisy (when it has excessive digital artifacts) and source content that is clean. When the content is noisy, the filter cleans up the source content before the encoding phase, with these two effects: First, it improves the output video quality because the content has been cleaned up. Secondly, it decreases the bandwidth because MediaLive does not waste bits on encoding noise. When the content is reasonably clean, the filter tends to decrease the bitrate.

Type: Mpeg2FilterSettings
Required: False

fixedAfd

Complete this field only when afdSignaling is set to FIXED. Enter the AFD value (4 bits) to write on all frames of the video encode.

Type: FixedAfd
Required: False

framerateDenominator

description": "The framerate denominator. For example, 1001. The framerate is the numerator divided by the denominator. For example, 24000 / 1001 = 23.976 FPS.

Type: integer
Required: True
Minimum: 1

framerateNumerator

The framerate numerator. For example, 24000. The framerate is the numerator divided by the denominator. For example, 24000 / 1001 = 23.976 FPS.

Type: integer
Required: True
Minimum: 1

gopClosedCadence

MPEG2: default is open GOP.

Type: integer
Required: False
Minimum: 0

gopNumBFrames

Relates to the GOP structure. The number of B-frames between reference frames. If you do not know what a B-frame is, use the default.

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

gopSize

Relates to the GOP structure. The GOP size (keyframe interval) in the units specified in gopSizeUnits. If you do not know what GOP is, use the default. If gopSizeUnits is frames, then the gopSize must be an integer and must be greater than or equal to 1. If gopSizeUnits is seconds, the gopSize must be greater than 0, but does not need to be an integer.

Type: number

Required: False

gopSizeUnits

Relates to the GOP structure. Specifies whether the gopSize is specified in frames or seconds. If you do not plan to change the default gopSize, leave the default. If you specify SECONDS, MediaLive will internally convert the gop size to a frame count.

Type: Mpeg2GopSizeUnits

Required: False

scanType

Set the scan type of the output to PROGRESSIVE or INTERLACED (top field first).

Type: Mpeg2ScanType

Required: False

subgopLength

Relates to the GOP structure. If you do not know what GOP is, use the default. FIXED: Set the number of B-frames in each sub-GOP to the value in gopNumBFrames. DYNAMIC: Let MediaLive optimize the number of B-frames in each sub-GOP, to improve visual quality.

Type: Mpeg2SubGopLength

Required: False

timecodeInsertion

Determines how MediaLive inserts timecodes in the output video. For detailed information about setting up the input and the output for a timecode, see the section on \"MediaLive Features - Timecode configuration\" in the MediaLive User Guide. DISABLED: do not include timecodes. GOP_TIMECODE: Include timecode metadata in the GOP header.

Type: Mpeg2TimecodeInsertionBehavior

Required: False

Mpeg2SubGopLength

Mpeg2 Sub Gop Length

DYNAMIC
FIXED

Mpeg2TimecodeInsertionBehavior

Mpeg2 Timecode Insertion Behavior

DISABLED
GOP_TIMECODE

MsSmoothGroupSettings

Ms Smooth Group Settings

acquisitionPointId

The ID to include in each message in the sparse track. Ignored if sparseTrackType is NONE.

Type: string
Required: False

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: SmoothGroupAudioOnlyTimecodeControl
Required: False

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.

Type: SmoothGroupCertificateMode
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

destination

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

Type: OutputLocationRef
Required: True

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

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: SmoothGroupEventIdMode
Required: False

eventStopBehavior

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

Type: SmoothGroupEventStopBehavior
Required: False

filecacheDuration

Size in seconds of file cache for streaming outputs.

Type: integer
Required: False
Minimum: 0

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

inputLossAction

Parameter that control output group behavior on input loss.

Type: InputLossActionForMsSmoothOut
Required: False

numRetries

Number of retry attempts.

Type: integer
Required: False
Minimum: 0

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

segmentationMode

useInputSegmentation has been deprecated. The configured segment size is always used.

Type: SmoothGroupSegmentationMode
Required: False

sendDelayMs

Number of milliseconds to delay the output from the second pipeline.

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

sparseTrackType

Identifies the type of data to place in the sparse track: - SCTE35: Insert SCTE-35 messages from the source content. With each message, insert an IDR frame to start a new segment. - SCTE35_WITHOUT_SEGMENTATION: Insert SCTE-35 messages from the source content. With each message, insert an IDR frame but don't start a new segment. - NONE: Don't generate a sparse track for any outputs in this output group.

Type: SmoothGroupSparseTrackType
Required: False

streamManifestBehavior

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

Type: SmoothGroupStreamManifestBehavior
Required: False

timestampOffset

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

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: SmoothGroupTimestampOffsetMode
Required: False

MsSmoothH265PackagingType

Ms Smooth H265 Packaging Type

HEV1
HVC1

MsSmoothOutputSettings

Ms Smooth Output Settings

h265PackagingType

Only applicable when this output is referencing an H.265 video description. Specifies whether MP4 segments should be packaged as HEV1 or HVC1.

Type: MsSmoothH265PackagingType
Required: False

nameModifier

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

Type: string
Required: False

MultiplexGroupSettings

Multiplex Group Settings

MultiplexOutputSettings

Multiplex Output Settings

destination

Destination is a Multiplex.

Type: OutputLocationRef
Required: True

MultiplexProgramChannelDestinationSettings

Multiplex Program Input Destination Settings for outputting a Channel to a Multiplex

multiplexId

The ID of the Multiplex that the encoder is providing output to. You do not need to specify the individual inputs to the Multiplex; MediaLive will handle the connection of the two MediaLive pipelines to the two Multiplex instances. The Multiplex must be in the same region as the Channel.

Type: string
Required: False
MinLength: 1

programName

The program name of the Multiplex program that the encoder is providing output to.

Type: string
Required: False
MinLength: 1

NetworkInputServerValidation

Network Input Server Validation

`CHECK_CRYPTOGRAPHY_AND_VALIDATE_NAME`
`CHECK_CRYPTOGRAPHY_ONLY`

NetworkInputSettings

Network source to transcode. Must be accessible to the Elemental Live node that is running the live event through a network connection.

hlsInputSettings

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

Type: HlsInputSettings
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: NetworkInputServerValidation
Required: False

NielsenCBET

Nielsen CBET

cbetCheckDigitString

Enter the CBET check digits to use in the watermark.

Type: string
Required: True
MinLength: 2
MaxLength: 2

cbetStepaside

Determines the method of CBET insertion mode when prior encoding is detected on the same layer.

Type: NielsenWatermarksCbetStepaside
Required: True

csid

Enter the CBET Source ID (CSID) to use in the watermark

Type: string
Required: True
MinLength: 1
MaxLength: 7

NielsenConfiguration

Nielsen Configuration

distributorId

Enter the Distributor ID assigned to your organization by Nielsen.

Type: string
Required: False

nielsenPcmToId3Tagging

Enables Nielsen PCM to ID3 tagging

Type: NielsenPcmToId3TaggingState
Required: False

NielsenNaesIiNw

Nielsen Naes Ii Nw

checkDigitString

Enter the check digit string for the watermark

Type: string
Required: True
MinLength: 2
MaxLength: 2

sid

Enter the Nielsen Source ID (SID) to include in the watermark

Type: number
Required: True
Minimum: 1
Maximum: 65535

NielsenPcmToId3TaggingState

State of Nielsen PCM to ID3 tagging

```
DISABLED
ENABLED
```

NielsenWatermarksCbetStepaside

Nielsen Watermarks Cbet Stepaside

```
DISABLED
ENABLED
```

NielsenWatermarksDistributionTypes

Nielsen Watermarks Distribution Types

FINAL_DISTRIBUTOR

PROGRAM_CONTENT

NielsenWatermarksSettings

Nielsen Watermarks Settings

nielsenCbetSettings

Complete these fields only if you want to insert watermarks of type Nielsen CBET

Type: NielsenCBET

Required: False

nielsenDistributionType

Choose the distribution types that you want to assign to the watermarks: - PROGRAM_CONTENT - FINAL_DISTRIBUTOR

Type: NielsenWatermarksDistributionTypes

Required: False

nielsenNaesIiNwSettings

Complete these fields only if you want to insert watermarks of type Nielsen NAES II (N2) and Nielsen NAES VI (NW).

Type: NielsenNaesIiNw

Required: False

Output

Output settings. There can be multiple outputs within a group.

audioDescriptionNames

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

Type: Array of 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
MinLength: 1
MaxLength: 255

outputSettings

Output type-specific settings.

Type: OutputSettings
Required: True

videoDescriptionName

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

Type: string
Required: False

OutputDestination

id

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

Type: string
Required: False

mediaPackageSettings

Destination settings for a MediaPackage output; one destination for both encoders.

Type: Array of type MediaPackageOutputDestinationSettings
Required: False

multiplexSettings

Destination settings for a Multiplex output; one destination for both encoders.

Type: MultiplexProgramChannelDestinationSettings
Required: False

settings

Destination settings for a standard output; one destination for each redundant encoder.

Type: Array of type OutputDestinationSettings
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

Output groups for this Live Event. Output groups contain information about where streams should be distributed.

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
MaxLength: 32

outputGroupSettings

Settings associated with the output group.

Type: OutputGroupSettings

Required: True

outputs

Type: Array of type Output

Required: True

OutputGroupSettings

Output Group Settings

archiveGroupSettings

Type: ArchiveGroupSettings

Required: False

frameCaptureGroupSettings

Type: FrameCaptureGroupSettings

Required: False

hlsGroupSettings

Type: HlsGroupSettings

Required: False

mediaPackageGroupSettings

Type: MediaPackageGroupSettings

Required: False

msSmoothGroupSettings

Type: MsSmoothGroupSettings

Required: False

multiplexGroupSettings

Type: MultiplexGroupSettings
Required: False

rtmpGroupSettings

Type: RtmpGroupSettings
Required: False

udpGroupSettings

Type: UdpGroupSettings
Required: False

OutputLocationRef

Reference to an OutputDestination ID defined in the channel

destinationRefId

Type: string
Required: False

OutputSettings

Output Settings

archiveOutputSettings

Type: ArchiveOutputSettings
Required: False

frameCaptureOutputSettings

Type: FrameCaptureOutputSettings
Required: False

hlsOutputSettings

Type: HlsOutputSettings
Required: False

mediaPackageOutputSettings

Type: MediaPackageOutputSettings
Required: False

msSmoothOutputSettings

Type: MsSmoothOutputSettings
Required: False

multiplexOutputSettings

Type: MultiplexOutputSettings
Required: False

rtmpOutputSettings

Type: RtmpOutputSettings
Required: False

udpOutputSettings

Type: UdpOutputSettings
Required: False

PassThroughSettings

Pass Through Settings

PipelineDetail

Runtime details of a pipeline when a channel is running.

activeInputAttachmentName

The name of the active input attachment currently being ingested by this pipeline.

Type: string
Required: False

activeInputSwitchActionName

The name of the input switch schedule action that occurred most recently and that resulted in the switch to the current input attachment for this pipeline.

Type: string
Required: False

activeMotionGraphicsActionName

The name of the motion graphics activate action that occurred most recently and that resulted in the current graphics URI for this pipeline.

Type: string
Required: False

activeMotionGraphicsUri

The current URI being used for HTML5 motion graphics for this pipeline.

Type: string
Required: False

pipelineId

Pipeline ID

Type: string
Required: False

RawSettings

Raw Settings

Rec601Settings

Rec601 Settings

Rec709Settings

Rec709 Settings

RemixSettings

Remix Settings

channelMappings

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

Type: Array of type AudioChannelMapping
Required: True

channelsIn

Number of input channels to be used.

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

channelsOut

Number of output channels to be produced. Valid values: 1, 2, 4, 6, 8

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

ResourceConflict

message

Type: string
Required: False

RtmpAdMarkers

Rtmp Ad Markers

```
ON_CUE_POINT_SCTE35
```

RtmpCacheFullBehavior

Rtmp Cache Full Behavior

```
DISCONNECT_IMMEDIATELY
WAIT_FOR_SERVER
```

RtmpCaptionData

Rtmp Caption Data

```
ALL
FIELD1_608
FIELD1_AND_FIELD2_608
```

RtmpCaptionInfoDestinationSettings

Rtmp Caption Info Destination Settings

RtmpGroupSettings

Rtmp Group Settings

adMarkers

Choose the ad marker type for this output group. MediaLive will create a message based on the content of each SCTE-35 message, format it for that marker type, and insert it in the datastream.

Type: Array of type RtmpAdMarkers
Required: False

authenticationScheme

Authentication scheme to use when connecting with CDN

Type: AuthenticationScheme
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: RtmpCacheFullBehavior
Required: False

cacheLength

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

Type: integer
Required: False
Minimum: 30

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: RtmpCaptionData

Required: False

inputLossAction

Controls the behavior of this RTMP group if input becomes unavailable. - emitOutput: Emit a slate until input returns. - pauseOutput: Stop transmitting data until input returns. This does not close the underlying RTMP connection.

Type: InputLossActionForRtmpOut
Required: False

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

RtmpOutputCertificateMode

Rtmp Output Certificate Mode

```
SELF_SIGNED
VERIFY_AUTHENTICITY
```

RtmpOutputSettings

Rtmp Output Settings

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: RtmpOutputCertificateMode
Required: False

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

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
Required: True

numRetries

Number of retry attempts.

Type: integer
Required: False
Minimum: 0

S3CannedAcl

S3 Canned Acl

```
AUTHENTICATED_READ
BUCKET_OWNER_FULL_CONTROL
BUCKET_OWNER_READ
PUBLIC_READ
```

Scte20Convert608To708

Scte20 Convert608 To708

```
DISABLED
```

UPCONVERT

Scte20PlusEmbeddedDestinationSettings

Scte20 Plus Embedded Destination Settings

Scte20SourceSettings

Scte20 Source Settings

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: Scte20Convert608To708
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

Scte27DestinationSettings

Scte27 Destination Settings

Scte27OcrLanguage

Scte27 Ocr Language

DEU
ENG
FRA

NLD
POR
SPA

Scte27SourceSettings

Scte27 Source Settings

ocrLanguage

If you will configure a WebVTT caption description that references this caption selector, use this field to provide the language to consider when translating the image-based source to text.

Type: Scte27OcrLanguage
Required: False

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

Scte35 Apos No Regional Blackout Behavior

FOLLOW
IGNORE

Scte35AposWebDeliveryAllowedBehavior

Scte35 Apos Web Delivery Allowed Behavior

FOLLOW
IGNORE

Scte35SpliceInsert

Scte35 Splice Insert

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

noRegionalBlackoutFlag

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

Type: Scte35SpliceInsertNoRegionalBlackoutBehavior
Required: False

webDeliveryAllowedFlag

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

Type: Scte35SpliceInsertWebDeliveryAllowedBehavior
Required: False

Scte35SpliceInsertNoRegionalBlackoutBehavior

Scte35 Splice Insert No Regional Blackout Behavior

FOLLOW

IGNORE

Scte35SpliceInsertWebDeliveryAllowedBehavior

Scte35 Splice Insert Web Delivery Allowed Behavior

FOLLOW
IGNORE

Scte35TimeSignalApos

Scte35 Time Signal Apos

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

noRegionalBlackoutFlag

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

Type: Scte35AposNoRegionalBlackoutBehavior
Required: False

webDeliveryAllowedFlag

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

Type: Scte35AposWebDeliveryAllowedBehavior
Required: False

SmoothGroupAudioOnlyTimecodeControl

Smooth Group Audio Only Timecode Control

PASSTHROUGH

USE_CONFIGURED_CLOCK

SmoothGroupCertificateMode

Smooth Group Certificate Mode

SELF_SIGNED

VERIFY_AUTHENTICITY

SmoothGroupEventIdMode

Smooth Group Event Id Mode

NO_EVENT_ID

USE_CONFIGURED

USE_TIMESTAMP

SmoothGroupEventStopBehavior

Smooth Group Event Stop Behavior

NONE

SEND_EOS

SmoothGroupSegmentationMode

Smooth Group Segmentation Mode

USE_INPUT_SEGMENTATION

USE_SEGMENT_DURATION

SmoothGroupSparseTrackType

Smooth Group Sparse Track Type

NONE
SCTE_35
SCTE_35_WITHOUT_SEGMENTATION

SmoothGroupStreamManifestBehavior

Smooth Group Stream Manifest Behavior

DO_NOT_SEND
SEND

SmoothGroupTimestampOffsetMode

Smooth Group Timestamp Offset Mode

USE_CONFIGURED_OFFSET
USE_EVENT_START_DATE

Smpte2038DataPreference

Smpte2038 Data Preference

IGNORE
PREFER

SmpteTtDestinationSettings

Smpte Tt Destination Settings

StandardHlsSettings

Standard Hls Settings

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

m3u8Settings

Type: M3u8Settings
Required: True

StaticKeySettings

Static Key Settings

keyProviderServer

The URL of the license server used for protecting content.

Type: InputLocation
Required: False

staticKeyValue

Static key value as a 32 character hexadecimal string.

Type: string
Required: True
MinLength: 32
MaxLength: 32

Tags

key-value pairs

Type: string

TeletextDestinationSettings

Teletext Destination Settings

TeletextSourceSettings

Teletext Source Settings

outputRectangle

Optionally defines a region where TTML style captions will be displayed

Type: CaptionRectangle
Required: False

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

TemporalFilterPostFilterSharpening

Temporal Filter Post Filter Sharpening

AUTO
DISABLED
ENABLED

TemporalFilterSettings

Temporal Filter Settings

postFilterSharpening

If you enable this filter, the results are the following: - If the source content is noisy (it contains excessive digital artifacts), the filter cleans up the source. - If the source content is already clean, the filter tends to decrease the bitrate, especially when the rate control mode is QVBR.

Type: TemporalFilterPostFilterSharpening
Required: False

strength

Choose a filter strength. We recommend a strength of 1 or 2. A higher strength might take out good information, resulting in an image that is overly soft.

Type: TemporalFilterStrength
Required: False

TemporalFilterStrength

Temporal Filter Strength

AUTO
STRENGTH_1
STRENGTH_2
STRENGTH_3
STRENGTH_4
STRENGTH_5
STRENGTH_6
STRENGTH_7
STRENGTH_8
STRENGTH_9
STRENGTH_10
STRENGTH_11
STRENGTH_12
STRENGTH_13
STRENGTH_14
STRENGTH_15
STRENGTH_16

TimecodeConfig

Timecode Config

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: TimecodeConfigSource
Required: True

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

TimecodeConfigSource

Timecode Config Source

```
EMBEDDED
SYSTEMCLOCK
ZEROBASED
```

TtmlDestinationSettings

Ttml Destination Settings

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: TtmlDestinationStyleControl
Required: False

TtmlDestinationStyleControl

Ttml Destination Style Control

```
PASSTHROUGH
USE_CONFIGURED
```

UdpContainerSettings

Udp Container Settings

m2tsSettings

Type: M2tsSettings
Required: False

UdpGroupSettings

Udp Group Settings

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: InputLossActionForUdpOut
Required: False

timedMetadataId3Frame

Indicates ID3 frame that has the timecode.

Type: UdpTimedMetadataId3Frame
Required: False

timedMetadataId3Period

Timed Metadata interval in seconds.

Type: integer
Required: False
Minimum: 0

UdpOutputSettings

Udp Output Settings

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

containerSettings

Type: UdpContainerSettings
Required: True

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

Required: True

fecOutputSettings

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

Type: FecOutputSettings

Required: False

UdpTimedMetadataId3Frame

Udp Timed Metadata Id3 Frame

`NONE`

`PRIV`

`TDRL`

ValidationError

elementPath

Path to the source of the error.

Type: string

Required: False

errorMessage

The error message.

Type: string

Required: False

VideoBlackFailoverSettings

blackDetectThreshold

A value used in calculating the threshold below which MediaLive considers a pixel to be 'black'. For the input to be considered black, every pixel in a frame must be below this threshold. The threshold is calculated as a percentage (expressed as a decimal) of white. Therefore .1 means 10% white (or 90% black). Note how the formula works for any color depth. For example, if you set this field to 0.1 in 10-bit color depth: (1023*0.1=102.3), which means a pixel value of 102 or less is 'black'. If you set this field to .1 in an 8-bit color depth: (255*0.1=25.5), which means a pixel value of 25 or less is 'black'. The range is 0.0 to 1.0, with any number of decimal places.

Type: number
Required: False
Minimum: 0
Maximum: 1

videoBlackThresholdMsec

The amount of time (in milliseconds) that the active input must be black before automatic input failover occurs.

Type: integer
Required: False
Minimum: 1000

VideoCodecSettings

Video Codec Settings

frameCaptureSettings

Type: FrameCaptureSettings
Required: False

h264Settings

Type: H264Settings
Required: False

h265Settings

Type: H265Settings
Required: False

mpeg2Settings

Type: Mpeg2Settings
Required: False

VideoDescription

Video settings for this stream.

codecSettings

Video codec settings.

Type: VideoCodecSettings
Required: False

height

Output video height, in pixels. Must be an even number. For most codecs, you can leave this field and width blank in order to use the height and width (resolution) from the source. Note, however, that leaving blank is not recommended. For the Frame Capture codec, height and width are required.

Type: integer
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

respondToAfd

Indicates how MediaLive will respond to the AFD values that might be in the input video. If you do not know what AFD signaling is, or if your downstream system has not given you guidance, choose PASSTHROUGH. RESPOND: MediaLive clips the input video using a formula that uses the AFD values (configured in afdSignaling), the input display aspect ratio, and the output display aspect ratio. MediaLive also includes the AFD values in the output, unless the codec for this encode is FRAME_CAPTURE. PASSTHROUGH: MediaLive ignores the AFD values and does not clip the video. But MediaLive does include the values in the output. NONE: MediaLive does not clip the input video and does not include the AFD values in the output

Type: VideoDescriptionRespondToAfd
Required: False

scalingBehavior

STRETCH_TO_OUTPUT configures the output position to stretch the video to the specified output resolution (height and width). This option will override any position value. DEFAULT may insert black boxes (pillar boxes or letter boxes) around the video to provide the specified output resolution.

Type: VideoDescriptionScalingBehavior
Required: False

sharpness

Changes the strength of the anti-alias filter used for scaling. 0 is the softest setting, 100 is the sharpest. A setting of 50 is recommended for most content.

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

width

Output video width, in pixels. Must be an even number. For most codecs, you can leave this field and height blank in order to use the height and width (resolution) from the source. Note, however,

that leaving blank is not recommended. For the Frame Capture codec, height and width are required.

Type: integer
Required: False

VideoDescriptionRespondToAfd

Video Description Respond To Afd

NONE
PASSTHROUGH
RESPOND

VideoDescriptionScalingBehavior

Video Description Scaling Behavior

DEFAULT
STRETCH_TO_OUTPUT

VideoSelector

Specifies a particular video stream within an input source. An input may have only a single video selector.

colorSpace

Specifies the color space of an input. This setting works in tandem with colorSpaceUsage and a video description's colorSpaceSettingsChoice to determine if any conversion will be performed.

Type: VideoSelectorColorSpace
Required: False

colorSpaceSettings

Color space settings

Type: VideoSelectorColorSpaceSettings

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: VideoSelectorColorSpaceUsage
Required: False

selectorSettings

The video selector settings.

Type: VideoSelectorSettings
Required: False

VideoSelectorColorSpace

Video Selector Color Space

```
FOLLOW
HDR10
HLG_2020
REC_601
REC_709
```

VideoSelectorColorSpaceSettings

Video Selector Color Space Settings

hdr10Settings

Type: Hdr10Settings

Required: False

VideoSelectorColorSpaceUsage

Video Selector Color Space Usage

FALLBACK
FORCE

VideoSelectorPid

Video Selector Pid

pid

Selects a specific PID from within a video source.

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

VideoSelectorProgramId

Video Selector Program Id

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

Video Selector Settings

videoSelectorPid

Type: VideoSelectorPid
Required: False

videoSelectorProgramId

Type: VideoSelectorProgramId
Required: False

VpcOutputSettings

The properties for a private VPC Output When this property is specified, the output egress addresses will be created in a user specified VPC

publicAddressAllocationIds

List of public address allocation ids to associate with ENIs that will be created in Output VPC. Must specify one for SINGLE_PIPELINE, two for STANDARD channels

Type: Array of type string
Required: False

securityGroupIds

A list of up to 5 EC2 VPC security group IDs to attach to the Output VPC network interfaces. If none are specified then the VPC default security group will be used

Type: Array of type string
Required: False

subnetIds

A list of VPC subnet IDs from the same VPC. If STANDARD channel, subnet IDs must be mapped to two unique availability zones (AZ).

Type: Array of type string
Required: True

VpcOutputSettingsDescription

The properties for a private VPC Output

availabilityZones

The Availability Zones where the vpc subnets are located. The first Availability Zone applies to the first subnet in the list of subnets. The second Availability Zone applies to the second subnet.

Type: Array of type string
Required: False

networkInterfaceIds

A list of Elastic Network Interfaces created by MediaLive in the customer's VPC

Type: Array of type string
Required: False

securityGroupIds

A list of up EC2 VPC security group IDs attached to the Output VPC network interfaces.

Type: Array of type string
Required: False

subnetIds

A list of VPC subnet IDs from the same VPC. If STANDARD channel, subnet IDs must be mapped to two unique availability zones (AZ).

Type: Array of type string
Required: False

WavCodingMode

Wav Coding Mode

```
CODING_MODE_1_0
```

CODING_MODE_2_0
CODING_MODE_4_0
CODING_MODE_8_0

WavSettings

Wav Settings

bitDepth

Bits per sample.

Type: number
Required: False

codingMode

The audio coding mode for the WAV audio. The mode determines the number of channels in the audio.

Type: WavCodingMode
Required: False

sampleRate

Sample rate in Hz.

Type: number
Required: False

WebvttDestinationSettings

Webvtt Destination Settings

styleControl

Controls whether the color and position of the source captions is passed through to the WebVTT output captions. PASSTHROUGH - Valid only if the source captions are EMBEDDED or TELETEXT.

NO_STYLE_DATA - Don't pass through the style. The output captions will not contain any font styling information.

Type: WebvttDestinationStyleControl
Required: False

WebvttDestinationStyleControl

Webvtt Destination Style Control

```
NO_STYLE_DATA
PASSTHROUGH
```

See also

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

ListChannels

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

CreateChannel

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++

- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Channels: channel ID

URI

`/prod/channels/channelId`

HTTP methods

DELETE

Operation ID: `DeleteChannel`

Path parameters

Name	Type	Required	Description
channelId	String	True	

Responses

Status code	Response model	Description
200	Channel	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response

Status code	Response model	Description
409	ResourceConflict	409 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

GET

Operation ID: `DescribeChannel`

Path parameters

Name	Type	Required	Description
channelId	String	True	

Responses

Status code	Response model	Description
200	Channel	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response

Status code	Response model	Description
504	GatewayTimeoutException	504 response

PUT

Operation ID: `UpdateChannel`

Path parameters

Name	Type	Required	Description
channelId	String	True	

Responses

Status code	Response model	Description
200	UpdateChannelResultModel	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
409	ResourceConflict	409 response
422	ChannelConfigurationValidationError	422 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

Schemas

Request bodies

PUT schema

```
{
  "cdiInputSpecification": {
    "resolution": enum
  },
  "destinations": [
    {
      "id": "string",
      "mediaPackageSettings": [
        {
          "channelId": "string"
        }
      ],
      "multiplexSettings": {
        "multiplexId": "string",
        "programName": "string"
      },
      "settings": [
        {
          "passwordParam": "string",
          "streamName": "string",
          "url": "string",
          "username": "string"
        }
      ]
    }
  ],
  "encoderSettings": {
    "audioDescriptions": [
      {
        "audioNormalizationSettings": {
          "algorithm": enum,
          "algorithmControl": enum,
          "targetLkfs": number
        },
        "audioSelectorName": "string",
        "audioType": enum,
        "audioTypeControl": enum,
```

```
        "audioWatermarkingSettings": {
          "nielsenWatermarksSettings": {
            "nielsenCbetSettings": {
              "cbetCheckDigitString": "string",
              "cbetStepaside": enum,
              "csid": "string"
            },
            "nielsenDistributionType": enum,
            "nielsenNaesIiNwSettings": {
              "checkDigitString": "string",
              "sid": number
            }
          }
        },
        "codecSettings": {
          "aacSettings": {
            "bitrate": number,
            "codingMode": enum,
            "inputType": enum,
            "profile": enum,
            "rateControlMode": enum,
            "rawFormat": enum,
            "sampleRate": number,
            "spec": enum,
            "vbrQuality": enum
          },
          "ac3Settings": {
            "bitrate": number,
            "bitstreamMode": enum,
            "codingMode": enum,
            "dialnorm": integer,
            "drcProfile": enum,
            "lfeFilter": enum,
            "metadataControl": enum
          },
          "eac3Settings": {
            "attenuationControl": enum,
            "bitrate": number,
            "bitstreamMode": enum,
            "codingMode": enum,
            "dcFilter": enum,
            "dialnorm": integer,
            "drcLine": enum,
            "drcRf": enum,
```

```
            "lfeControl": enum,
            "lfeFilter": enum,
            "loRoCenterMixLevel": number,
            "loRoSurroundMixLevel": number,
            "ltRtCenterMixLevel": number,
            "ltRtSurroundMixLevel": number,
            "metadataControl": enum,
            "passthroughControl": enum,
            "phaseControl": enum,
            "stereoDownmix": enum,
            "surroundExMode": enum,
            "surroundMode": enum
          },
          "mp2Settings": {
            "bitrate": number,
            "codingMode": enum,
            "sampleRate": number
          },
          "passThroughSettings": {
          },
          "wavSettings": {
            "bitDepth": number,
            "codingMode": enum,
            "sampleRate": number
          }
        },
        "languageCode": "string",
        "languageCodeControl": enum,
        "name": "string",
        "remixSettings": {
          "channelMappings": [
            {
              "inputChannelLevels": [
                {
                  "gain": integer,
                  "inputChannel": integer
                }
              ],
              "outputChannel": integer
            }
          ],
          "channelsIn": integer,
          "channelsOut": integer
        },
```

```
        "streamName": "string"
      }
    ],
    "availBlanking": {
      "availBlankingImage": {
        "passwordParam": "string",
        "uri": "string",
        "username": "string"
      },
      "state": enum
    },
    "availConfiguration": {
      "availSettings": {
        "scte35SpliceInsert": {
          "adAvailOffset": integer,
          "noRegionalBlackoutFlag": enum,
          "webDeliveryAllowedFlag": enum
        },
        "scte35TimeSignalApos": {
          "adAvailOffset": integer,
          "noRegionalBlackoutFlag": enum,
          "webDeliveryAllowedFlag": enum
        }
      }
    },
    "blackoutSlate": {
      "blackoutSlateImage": {
        "passwordParam": "string",
        "uri": "string",
        "username": "string"
      },
      "networkEndBlackout": enum,
      "networkEndBlackoutImage": {
        "passwordParam": "string",
        "uri": "string",
        "username": "string"
      },
      "networkId": "string",
      "state": enum
    },
    "captionDescriptions": [
      {
        "captionSelectorName": "string",
        "destinationSettings": {
```

```
        "aribDestinationSettings": {
        },
        "burnInDestinationSettings": {
          "alignment": enum,
          "backgroundColor": enum,
          "backgroundOpacity": integer,
          "font": {
            "passwordParam": "string",
            "uri": "string",
            "username": "string"
          },
          "fontColor": enum,
          "fontOpacity": integer,
          "fontResolution": integer,
          "fontSize": "string",
          "outlineColor": enum,
          "outlineSize": integer,
          "shadowColor": enum,
          "shadowOpacity": integer,
          "shadowXOffset": integer,
          "shadowYOffset": integer,
          "teletextGridControl": enum,
          "xPosition": integer,
          "yPosition": integer
        },
        "dvbSubDestinationSettings": {
          "alignment": enum,
          "backgroundColor": enum,
          "backgroundOpacity": integer,
          "font": {
            "passwordParam": "string",
            "uri": "string",
            "username": "string"
          },
          "fontColor": enum,
          "fontOpacity": integer,
          "fontResolution": integer,
          "fontSize": "string",
          "outlineColor": enum,
          "outlineSize": integer,
          "shadowColor": enum,
          "shadowOpacity": integer,
          "shadowXOffset": integer,
          "shadowYOffset": integer,
```

```
            "teletextGridControl": enum,
            "xPosition": integer,
            "yPosition": integer
          },
          "ebuTtDDestinationSettings": {
            "copyrightHolder": "string",
            "fillLineGap": enum,
            "fontFamily": "string",
            "styleControl": enum
          },
          "embeddedDestinationSettings": {
          },
          "embeddedPlusScte20DestinationSettings": {
          },
          "rtmpCaptionInfoDestinationSettings": {
          },
          "scte20PlusEmbeddedDestinationSettings": {
          },
          "scte27DestinationSettings": {
          },
          "smpteTtDestinationSettings": {
          },
          "teletextDestinationSettings": {
          },
          "ttmlDestinationSettings": {
            "styleControl": enum
          },
          "webvttDestinationSettings": {
            "styleControl": enum
          }
        },
        "languageCode": "string",
        "languageDescription": "string",
        "name": "string"
      }
    ],
    "featureActivations": {
      "inputPrepareScheduleActions": enum
    },
    "globalConfiguration": {
      "initialAudioGain": integer,
      "inputEndAction": enum,
      "inputLossBehavior": {
        "blackFrameMsec": integer,
```

```
        "inputLossImageColor": "string",
        "inputLossImageSlate": {
          "passwordParam": "string",
          "uri": "string",
          "username": "string"
        },
        "inputLossImageType": enum,
        "repeatFrameMsec": integer
      },
      "outputLockingMode": enum,
      "outputTimingSource": enum,
      "supportLowFramerateInputs": enum
    },
    "motionGraphicsConfiguration": {
      "motionGraphicsInsertion": enum,
      "motionGraphicsSettings": {
        "htmlMotionGraphicsSettings": {
        }
      }
    },
    "nielsenConfiguration": {
      "distributorId": "string",
      "nielsenPcmToId3Tagging": enum
    },
    "outputGroups": [
      {
        "name": "string",
        "outputGroupSettings": {
          "archiveGroupSettings": {
            "archiveCdnSettings": {
              "archiveS3Settings": {
                "cannedAcl": enum,
                "logUploads": enum
              }
            },
            "destination": {
              "destinationRefId": "string"
            },
            "rolloverInterval": integer
          },
          "frameCaptureGroupSettings": {
            "destination": {
              "destinationRefId": "string"
            },
```

```
        "frameCaptureCdnSettings": {
          "frameCaptureS3Settings": {
            "cannedAcl": enum,
            "logUploads": enum
          }
        }
      },
      "hlsGroupSettings": {
        "adMarkers": [
          enum
        ],
        "baseUrlContent": "string",
        "baseUrlContent1": "string",
        "baseUrlManifest": "string",
        "baseUrlManifest1": "string",
        "captionLanguageMappings": [
          {
            "captionChannel": integer,
            "languageCode": "string",
            "languageDescription": "string"
          }
        ],
        "captionLanguageSetting": enum,
        "clientCache": enum,
        "codecSpecification": enum,
        "constantIv": "string",
        "destination": {
          "destinationRefId": "string"
        },
        "directoryStructure": enum,
        "discontinuityTags": enum,
        "encryptionType": enum,
        "hlsCdnSettings": {
          "hlsAkamaiSettings": {
            "connectionRetryInterval": integer,
            "filecacheDuration": integer,
            "httpTransferMode": enum,
            "numRetries": integer,
            "restartDelay": integer,
            "salt": "string",
            "token": "string"
          },
          "hlsBasicPutSettings": {
            "connectionRetryInterval": integer,
```

```
            "filecacheDuration": integer,
            "numRetries": integer,
            "restartDelay": integer
          },
          "hlsMediaStoreSettings": {
            "connectionRetryInterval": integer,
            "filecacheDuration": integer,
            "mediaStoreStorageClass": enum,
            "numRetries": integer,
            "restartDelay": integer
          },
          "hlsS3Settings": {
            "cannedAcl": enum,
            "logUploads": enum
          },
          "hlsWebdavSettings": {
            "connectionRetryInterval": integer,
            "filecacheDuration": integer,
            "httpTransferMode": enum,
            "numRetries": integer,
            "restartDelay": integer
          }
        },
        "hlsId3SegmentTagging": enum,
        "iFrameOnlyPlaylists": enum,
        "incompleteSegmentBehavior": enum,
        "indexNSegments": integer,
        "inputLossAction": enum,
        "ivInManifest": enum,
        "ivSource": enum,
        "keepSegments": integer,
        "keyFormat": "string",
        "keyFormatVersions": "string",
        "keyProviderSettings": {
          "staticKeySettings": {
            "keyProviderServer": {
              "passwordParam": "string",
              "uri": "string",
              "username": "string"
            },
            "staticKeyValue": "string"
          }
        },
        "manifestCompression": enum,
```

```
            "manifestDurationFormat": enum,
            "minSegmentLength": integer,
            "mode": enum,
            "outputSelection": enum,
            "programDateTime": enum,
            "programDateTimeClock": enum,
            "programDateTimePeriod": integer,
            "redundantManifest": enum,
            "segmentLength": integer,
            "segmentationMode": enum,
            "segmentsPerSubdirectory": integer,
            "streamInfResolution": enum,
            "timedMetadataId3Frame": enum,
            "timedMetadataId3Period": integer,
            "timestampDeltaMilliseconds": integer,
            "tsFileMode": enum
          },
          "mediaPackageGroupSettings": {
            "destination": {
              "destinationRefId": "string"
            }
          },
          "msSmoothGroupSettings": {
            "acquisitionPointId": "string",
            "audioOnlyTimecodeControl": enum,
            "certificateMode": enum,
            "connectionRetryInterval": integer,
            "destination": {
              "destinationRefId": "string"
            },
            "eventId": "string",
            "eventIdMode": enum,
            "eventStopBehavior": enum,
            "filecacheDuration": integer,
            "fragmentLength": integer,
            "inputLossAction": enum,
            "numRetries": integer,
            "restartDelay": integer,
            "segmentationMode": enum,
            "sendDelayMs": integer,
            "sparseTrackType": enum,
            "streamManifestBehavior": enum,
            "timestampOffset": "string",
            "timestampOffsetMode": enum
```

```
        },
        "multiplexGroupSettings": {
        },
        "rtmpGroupSettings": {
          "adMarkers": [
            enum
          ],
          "authenticationScheme": enum,
          "cacheFullBehavior": enum,
          "cacheLength": integer,
          "captionData": enum,
          "inputLossAction": enum,
          "restartDelay": integer
        },
        "udpGroupSettings": {
          "inputLossAction": enum,
          "timedMetadataId3Frame": enum,
          "timedMetadataId3Period": integer
        }
      },
      "outputs": [
        {
          "audioDescriptionNames": [
            "string"
          ],
          "captionDescriptionNames": [
            "string"
          ],
          "outputName": "string",
          "outputSettings": {
            "archiveOutputSettings": {
              "containerSettings": {
                "m2tsSettings": {
                  "absentInputAudioBehavior": enum,
                  "arib": enum,
                  "aribCaptionsPid": "string",
                  "aribCaptionsPidControl": enum,
                  "audioBufferModel": enum,
                  "audioFramesPerPes": integer,
                  "audioPids": "string",
                  "audioStreamType": enum,
                  "bitrate": integer,
                  "bufferModel": enum,
                  "ccDescriptor": enum,
```

```
                "dvbNitSettings": {
                  "networkId": integer,
                  "networkName": "string",
                  "repInterval": integer
                },
                "dvbSdtSettings": {
                  "outputSdt": enum,
                  "repInterval": integer,
                  "serviceName": "string",
                  "serviceProviderName": "string"
                },
                "dvbSubPids": "string",
                "dvbTdtSettings": {
                  "repInterval": integer
                },
                "dvbTeletextPid": "string",
                "ebif": enum,
                "ebpAudioInterval": enum,
                "ebpLookaheadMs": integer,
                "ebpPlacement": enum,
                "ecmPid": "string",
                "esRateInPes": enum,
                "etvPlatformPid": "string",
                "etvSignalPid": "string",
                "fragmentTime": number,
                "klv": enum,
                "klvDataPids": "string",
                "nielsenId3Behavior": enum,
                "nullPacketBitrate": number,
                "patInterval": integer,
                "pcrControl": enum,
                "pcrPeriod": integer,
                "pcrPid": "string",
                "pmtInterval": integer,
                "pmtPid": "string",
                "programNum": integer,
                "rateMode": enum,
                "scte27Pids": "string",
                "scte35Control": enum,
                "scte35Pid": "string",
                "segmentationMarkers": enum,
                "segmentationStyle": enum,
                "segmentationTime": number,
                "timedMetadataBehavior": enum,
```

```
            "timedMetadataPid": "string",
            "transportStreamId": integer,
            "videoPid": "string"
          },
          "rawSettings": {
          }
        },
        "extension": "string",
        "nameModifier": "string"
      },
      "frameCaptureOutputSettings": {
        "nameModifier": "string"
      },
      "hlsOutputSettings": {
        "h265PackagingType": enum,
        "hlsSettings": {
          "audioOnlyHlsSettings": {
            "audioGroupId": "string",
            "audioOnlyImage": {
              "passwordParam": "string",
              "uri": "string",
              "username": "string"
            },
            "audioTrackType": enum,
            "segmentType": enum
          },
          "fmp4HlsSettings": {
            "audioRenditionSets": "string",
            "nielsenId3Behavior": enum,
            "timedMetadataBehavior": enum
          },
          "frameCaptureHlsSettings": {
          },
          "standardHlsSettings": {
            "audioRenditionSets": "string",
            "m3u8Settings": {
              "audioFramesPerPes": integer,
              "audioPids": "string",
              "ecmPid": "string",
              "nielsenId3Behavior": enum,
              "patInterval": integer,
              "pcrControl": enum,
              "pcrPeriod": integer,
              "pcrPid": "string",
```

```
                "pmtInterval": integer,
                "pmtPid": "string",
                "programNum": integer,
                "scte35Behavior": enum,
                "scte35Pid": "string",
                "timedMetadataBehavior": enum,
                "timedMetadataPid": "string",
                "transportStreamId": integer,
                "videoPid": "string"
              }
            }
          },
          "nameModifier": "string",
          "segmentModifier": "string"
        },
        "mediaPackageOutputSettings": {
        },
        "msSmoothOutputSettings": {
          "h265PackagingType": enum,
          "nameModifier": "string"
        },
        "multiplexOutputSettings": {
          "destination": {
            "destinationRefId": "string"
          }
        },
        "rtmpOutputSettings": {
          "certificateMode": enum,
          "connectionRetryInterval": integer,
          "destination": {
            "destinationRefId": "string"
          },
          "numRetries": integer
        },
        "udpOutputSettings": {
          "bufferMsec": integer,
          "containerSettings": {
            "m2tsSettings": {
              "absentInputAudioBehavior": enum,
              "arib": enum,
              "aribCaptionsPid": "string",
              "aribCaptionsPidControl": enum,
              "audioBufferModel": enum,
              "audioFramesPerPes": integer,
```

```
                "audioPids": "string",
                "audioStreamType": enum,
                "bitrate": integer,
                "bufferModel": enum,
                "ccDescriptor": enum,
                "dvbNitSettings": {
                  "networkId": integer,
                  "networkName": "string",
                  "repInterval": integer
                },
                "dvbSdtSettings": {
                  "outputSdt": enum,
                  "repInterval": integer,
                  "serviceName": "string",
                  "serviceProviderName": "string"
                },
                "dvbSubPids": "string",
                "dvbTdtSettings": {
                  "repInterval": integer
                },
                "dvbTeletextPid": "string",
                "ebif": enum,
                "ebpAudioInterval": enum,
                "ebpLookaheadMs": integer,
                "ebpPlacement": enum,
                "ecmPid": "string",
                "esRateInPes": enum,
                "etvPlatformPid": "string",
                "etvSignalPid": "string",
                "fragmentTime": number,
                "klv": enum,
                "klvDataPids": "string",
                "nielsenId3Behavior": enum,
                "nullPacketBitrate": number,
                "patInterval": integer,
                "pcrControl": enum,
                "pcrPeriod": integer,
                "pcrPid": "string",
                "pmtInterval": integer,
                "pmtPid": "string",
                "programNum": integer,
                "rateMode": enum,
                "scte27Pids": "string",
                "scte35Control": enum,
```

```
                    "scte35Pid": "string",
                    "segmentationMarkers": enum,
                    "segmentationStyle": enum,
                    "segmentationTime": number,
                    "timedMetadataBehavior": enum,
                    "timedMetadataPid": "string",
                    "transportStreamId": integer,
                    "videoPid": "string"
                  }
                },
                "destination": {
                  "destinationRefId": "string"
                },
                "fecOutputSettings": {
                  "columnDepth": integer,
                  "includeFec": enum,
                  "rowLength": integer
                }
              }
            },
            "videoDescriptionName": "string"
          }
        ]
      }
    ],
    "timecodeConfig": {
      "source": enum,
      "syncThreshold": integer
    },
    "videoDescriptions": [
      {
        "codecSettings": {
          "frameCaptureSettings": {
            "captureInterval": integer,
            "captureIntervalUnits": enum
          },
          "h264Settings": {
            "adaptiveQuantization": enum,
            "afdSignaling": enum,
            "bitrate": integer,
            "bufFillPct": integer,
            "bufSize": integer,
            "colorMetadata": enum,
            "colorSpaceSettings": {
```

```
            "colorSpacePassthroughSettings": {
            },
            "rec601Settings": {
            },
            "rec709Settings": {
            }
          },
          "entropyEncoding": enum,
          "filterSettings": {
            "temporalFilterSettings": {
              "postFilterSharpening": enum,
              "strength": enum
            }
          },
          "fixedAfd": enum,
          "flickerAq": enum,
          "forceFieldPictures": enum,
          "framerateControl": enum,
          "framerateDenominator": integer,
          "framerateNumerator": integer,
          "gopBReference": enum,
          "gopClosedCadence": integer,
          "gopNumBFrames": integer,
          "gopSize": number,
          "gopSizeUnits": enum,
          "level": enum,
          "lookAheadRateControl": enum,
          "maxBitrate": integer,
          "minIInterval": integer,
          "numRefFrames": integer,
          "parControl": enum,
          "parDenominator": integer,
          "parNumerator": integer,
          "profile": enum,
          "qualityLevel": enum,
          "qvbrQualityLevel": integer,
          "rateControlMode": enum,
          "scanType": enum,
          "sceneChangeDetect": enum,
          "slices": integer,
          "softness": integer,
          "spatialAq": enum,
          "subgopLength": enum,
          "syntax": enum,
```

```
            "temporalAq": enum,
            "timecodeInsertion": enum
          },
          "h265Settings": {
            "adaptiveQuantization": enum,
            "afdSignaling": enum,
            "alternativeTransferFunction": enum,
            "bitrate": integer,
            "bufSize": integer,
            "colorMetadata": enum,
            "colorSpaceSettings": {
              "colorSpacePassthroughSettings": {
              },
              "hdr10Settings": {
                "maxCll": integer,
                "maxFall": integer
              },
              "rec601Settings": {
              },
              "rec709Settings": {
              }
            },
            "filterSettings": {
              "temporalFilterSettings": {
                "postFilterSharpening": enum,
                "strength": enum
              }
            },
            "fixedAfd": enum,
            "flickerAq": enum,
            "framerateDenominator": integer,
            "framerateNumerator": integer,
            "gopClosedCadence": integer,
            "gopSize": number,
            "gopSizeUnits": enum,
            "level": enum,
            "lookAheadRateControl": enum,
            "maxBitrate": integer,
            "minIInterval": integer,
            "parDenominator": integer,
            "parNumerator": integer,
            "profile": enum,
            "qvbrQualityLevel": integer,
            "rateControlMode": enum,
```

```
            "scanType": enum,
            "sceneChangeDetect": enum,
            "slices": integer,
            "tier": enum,
            "timecodeInsertion": enum
          },
          "mpeg2Settings": {
            "adaptiveQuantization": enum,
            "afdSignaling": enum,
            "colorMetadata": enum,
            "colorSpace": enum,
            "displayAspectRatio": enum,
            "filterSettings": {
              "temporalFilterSettings": {
                "postFilterSharpening": enum,
                "strength": enum
              }
            },
            "fixedAfd": enum,
            "framerateDenominator": integer,
            "framerateNumerator": integer,
            "gopClosedCadence": integer,
            "gopNumBFrames": integer,
            "gopSize": number,
            "gopSizeUnits": enum,
            "scanType": enum,
            "subgopLength": enum,
            "timecodeInsertion": enum
          }
        },
        "height": integer,
        "name": "string",
        "respondToAfd": enum,
        "scalingBehavior": enum,
        "sharpness": integer,
        "width": integer
      }
    ]
  },
  "inputAttachments": [
    {
      "automaticInputFailoverSettings": {
        "errorClearTimeMsec": integer,
        "failoverConditions": [
```

```
          {
            "failoverConditionSettings": {
              "audioSilenceSettings": {
                "audioSelectorName": "string",
                "audioSilenceThresholdMsec": integer
              },
              "inputLossSettings": {
                "inputLossThresholdMsec": integer
              },
              "videoBlackSettings": {
                "blackDetectThreshold": number,
                "videoBlackThresholdMsec": integer
              }
            }
          }
        ],
        "inputPreference": enum,
        "secondaryInputId": "string"
      },
      "inputAttachmentName": "string",
      "inputId": "string",
      "inputSettings": {
        "audioSelectors": [
          {
            "name": "string",
            "selectorSettings": {
              "audioHlsRenditionSelection": {
                "groupId": "string",
                "name": "string"
              },
              "audioLanguageSelection": {
                "languageCode": "string",
                "languageSelectionPolicy": enum
              },
              "audioPidSelection": {
                "pid": integer
              },
              "audioTrackSelection": {
                "tracks": [
                  {
                    "track": integer
                  }
                ]
              }
```

```
            }
          }
        ],
        "captionSelectors": [
          {
            "languageCode": "string",
            "name": "string",
            "selectorSettings": {
              "ancillarySourceSettings": {
                "sourceAncillaryChannelNumber": integer
              },
              "aribSourceSettings": {
              },
              "dvbSubSourceSettings": {
                "ocrLanguage": enum,
                "pid": integer
              },
              "embeddedSourceSettings": {
                "convert608To708": enum,
                "scte20Detection": enum,
                "source608ChannelNumber": integer,
                "source608TrackNumber": integer
              },
              "scte20SourceSettings": {
                "convert608To708": enum,
                "source608ChannelNumber": integer
              },
              "scte27SourceSettings": {
                "ocrLanguage": enum,
                "pid": integer
              },
              "teletextSourceSettings": {
                "outputRectangle": {
                  "height": number,
                  "leftOffset": number,
                  "topOffset": number,
                  "width": number
                },
                "pageNumber": "string"
              }
            }
          }
        ],
        "deblockFilter": enum,
```

```
        "denoiseFilter": enum,
        "filterStrength": integer,
        "inputFilter": enum,
        "networkInputSettings": {
          "hlsInputSettings": {
            "bandwidth": integer,
            "bufferSegments": integer,
            "retries": integer,
            "retryInterval": integer,
            "scte35Source": enum
          },
          "serverValidation": enum
        },
        "smpte2038DataPreference": enum,
        "sourceEndBehavior": enum,
        "videoSelector": {
          "colorSpace": enum,
          "colorSpaceSettings": {
            "hdr10Settings": {
              "maxCll": integer,
              "maxFall": integer
            }
          },
          "colorSpaceUsage": enum,
          "selectorSettings": {
            "videoSelectorPid": {
              "pid": integer
            },
            "videoSelectorProgramId": {
              "programId": integer
            }
          }
        }
      }
    ],
    "inputSpecification": {
      "codec": enum,
      "maximumBitrate": enum,
      "resolution": enum
    },
    "logLevel": enum,
    "maintenance": {
      "maintenanceDay": enum,
```

```
    "maintenanceScheduledDate": "string",
    "maintenanceStartTime": "string"
  },
  "name": "string",
  "roleArn": "string"
}
```

Response bodies

Channel schema

```
{
  "arn": "string",
  "cdiInputSpecification": {
    "resolution": enum
  },
  "channelClass": enum,
  "destinations": [
    {
      "id": "string",
      "mediaPackageSettings": [
        {
          "channelId": "string"
        }
      ],
      "multiplexSettings": {
        "multiplexId": "string",
        "programName": "string"
      },
      "settings": [
        {
          "passwordParam": "string",
          "streamName": "string",
          "url": "string",
          "username": "string"
        }
      ]
    }
  ],
  "egressEndpoints": [
    {
      "sourceIp": "string"
    }
```

```
  ],
  "encoderSettings": {
    "audioDescriptions": [
      {
        "audioNormalizationSettings": {
          "algorithm": enum,
          "algorithmControl": enum,
          "targetLkfs": number
        },
        "audioSelectorName": "string",
        "audioType": enum,
        "audioTypeControl": enum,
        "audioWatermarkingSettings": {
          "nielsenWatermarksSettings": {
            "nielsenCbetSettings": {
              "cbetCheckDigitString": "string",
              "cbetStepaside": enum,
              "csid": "string"
            },
            "nielsenDistributionType": enum,
            "nielsenNaesIiNwSettings": {
              "checkDigitString": "string",
              "sid": number
            }
          }
        },
        "codecSettings": {
          "aacSettings": {
            "bitrate": number,
            "codingMode": enum,
            "inputType": enum,
            "profile": enum,
            "rateControlMode": enum,
            "rawFormat": enum,
            "sampleRate": number,
            "spec": enum,
            "vbrQuality": enum
          },
          "ac3Settings": {
            "bitrate": number,
            "bitstreamMode": enum,
            "codingMode": enum,
            "dialnorm": integer,
            "drcProfile": enum,
```

```
            "lfeFilter": enum,
            "metadataControl": enum
          },
          "eac3Settings": {
            "attenuationControl": enum,
            "bitrate": number,
            "bitstreamMode": enum,
            "codingMode": enum,
            "dcFilter": enum,
            "dialnorm": integer,
            "drcLine": enum,
            "drcRf": enum,
            "lfeControl": enum,
            "lfeFilter": enum,
            "loRoCenterMixLevel": number,
            "loRoSurroundMixLevel": number,
            "ltRtCenterMixLevel": number,
            "ltRtSurroundMixLevel": number,
            "metadataControl": enum,
            "passthroughControl": enum,
            "phaseControl": enum,
            "stereoDownmix": enum,
            "surroundExMode": enum,
            "surroundMode": enum
          },
          "mp2Settings": {
            "bitrate": number,
            "codingMode": enum,
            "sampleRate": number
          },
          "passThroughSettings": {
          },
          "wavSettings": {
            "bitDepth": number,
            "codingMode": enum,
            "sampleRate": number
          }
        },
        "languageCode": "string",
        "languageCodeControl": enum,
        "name": "string",
        "remixSettings": {
          "channelMappings": [
            {
```

```
            "inputChannelLevels": [
              {
                "gain": integer,
                "inputChannel": integer
              }
            ],
            "outputChannel": integer
          }
        ],
        "channelsIn": integer,
        "channelsOut": integer
      },
      "streamName": "string"
    }
  ],
  "availBlanking": {
    "availBlankingImage": {
      "passwordParam": "string",
      "uri": "string",
      "username": "string"
    },
    "state": enum
  },
  "availConfiguration": {
    "availSettings": {
      "scte35SpliceInsert": {
        "adAvailOffset": integer,
        "noRegionalBlackoutFlag": enum,
        "webDeliveryAllowedFlag": enum
      },
      "scte35TimeSignalApos": {
        "adAvailOffset": integer,
        "noRegionalBlackoutFlag": enum,
        "webDeliveryAllowedFlag": enum
      }
    }
  },
  "blackoutSlate": {
    "blackoutSlateImage": {
      "passwordParam": "string",
      "uri": "string",
      "username": "string"
    },
    "networkEndBlackout": enum,
```

```
      "networkEndBlackoutImage": {
        "passwordParam": "string",
        "uri": "string",
        "username": "string"
      },
      "networkId": "string",
      "state": enum
    },
    "captionDescriptions": [
      {
        "captionSelectorName": "string",
        "destinationSettings": {
          "aribDestinationSettings": {
          },
          "burnInDestinationSettings": {
            "alignment": enum,
            "backgroundColor": enum,
            "backgroundOpacity": integer,
            "font": {
              "passwordParam": "string",
              "uri": "string",
              "username": "string"
            },
            "fontColor": enum,
            "fontOpacity": integer,
            "fontResolution": integer,
            "fontSize": "string",
            "outlineColor": enum,
            "outlineSize": integer,
            "shadowColor": enum,
            "shadowOpacity": integer,
            "shadowXOffset": integer,
            "shadowYOffset": integer,
            "teletextGridControl": enum,
            "xPosition": integer,
            "yPosition": integer
          },
          "dvbSubDestinationSettings": {
            "alignment": enum,
            "backgroundColor": enum,
            "backgroundOpacity": integer,
            "font": {
              "passwordParam": "string",
              "uri": "string",
```

```
            "username": "string"
          },
          "fontColor": enum,
          "fontOpacity": integer,
          "fontResolution": integer,
          "fontSize": "string",
          "outlineColor": enum,
          "outlineSize": integer,
          "shadowColor": enum,
          "shadowOpacity": integer,
          "shadowXOffset": integer,
          "shadowYOffset": integer,
          "teletextGridControl": enum,
          "xPosition": integer,
          "yPosition": integer
        },
        "ebuTtDDestinationSettings": {
          "copyrightHolder": "string",
          "fillLineGap": enum,
          "fontFamily": "string",
          "styleControl": enum
        },
        "embeddedDestinationSettings": {
        },
        "embeddedPlusScte20DestinationSettings": {
        },
        "rtmpCaptionInfoDestinationSettings": {
        },
        "scte20PlusEmbeddedDestinationSettings": {
        },
        "scte27DestinationSettings": {
        },
        "smpteTtDestinationSettings": {
        },
        "teletextDestinationSettings": {
        },
        "ttmlDestinationSettings": {
          "styleControl": enum
        },
        "webvttDestinationSettings": {
          "styleControl": enum
        }
      },
      "languageCode": "string",
```

```
        "languageDescription": "string",
        "name": "string"
      }
    ],
    "featureActivations": {
      "inputPrepareScheduleActions": enum
    },
    "globalConfiguration": {
      "initialAudioGain": integer,
      "inputEndAction": enum,
      "inputLossBehavior": {
        "blackFrameMsec": integer,
        "inputLossImageColor": "string",
        "inputLossImageSlate": {
          "passwordParam": "string",
          "uri": "string",
          "username": "string"
        },
        "inputLossImageType": enum,
        "repeatFrameMsec": integer
      },
      "outputLockingMode": enum,
      "outputTimingSource": enum,
      "supportLowFramerateInputs": enum
    },
    "motionGraphicsConfiguration": {
      "motionGraphicsInsertion": enum,
      "motionGraphicsSettings": {
        "htmlMotionGraphicsSettings": {
        }
      }
    },
    "nielsenConfiguration": {
      "distributorId": "string",
      "nielsenPcmToId3Tagging": enum
    },
    "outputGroups": [
      {
        "name": "string",
        "outputGroupSettings": {
          "archiveGroupSettings": {
            "archiveCdnSettings": {
              "archiveS3Settings": {
                "cannedAcl": enum,
```

```
            "logUploads": enum
          }
        },
        "destination": {
          "destinationRefId": "string"
        },
        "rolloverInterval": integer
      },
      "frameCaptureGroupSettings": {
        "destination": {
          "destinationRefId": "string"
        },
        "frameCaptureCdnSettings": {
          "frameCaptureS3Settings": {
            "cannedAcl": enum,
            "logUploads": enum
          }
        }
      },
      "hlsGroupSettings": {
        "adMarkers": [
          enum
        ],
        "baseUrlContent": "string",
        "baseUrlContent1": "string",
        "baseUrlManifest": "string",
        "baseUrlManifest1": "string",
        "captionLanguageMappings": [
          {
            "captionChannel": integer,
            "languageCode": "string",
            "languageDescription": "string"
          }
        ],
        "captionLanguageSetting": enum,
        "clientCache": enum,
        "codecSpecification": enum,
        "constantIv": "string",
        "destination": {
          "destinationRefId": "string"
        },
        "directoryStructure": enum,
        "discontinuityTags": enum,
        "encryptionType": enum,
```

```
            "hlsCdnSettings": {
              "hlsAkamaiSettings": {
                "connectionRetryInterval": integer,
                "filecacheDuration": integer,
                "httpTransferMode": enum,
                "numRetries": integer,
                "restartDelay": integer,
                "salt": "string",
                "token": "string"
              },
              "hlsBasicPutSettings": {
                "connectionRetryInterval": integer,
                "filecacheDuration": integer,
                "numRetries": integer,
                "restartDelay": integer
              },
              "hlsMediaStoreSettings": {
                "connectionRetryInterval": integer,
                "filecacheDuration": integer,
                "mediaStoreStorageClass": enum,
                "numRetries": integer,
                "restartDelay": integer
              },
              "hlsS3Settings": {
                "cannedAcl": enum,
                "logUploads": enum
              },
              "hlsWebdavSettings": {
                "connectionRetryInterval": integer,
                "filecacheDuration": integer,
                "httpTransferMode": enum,
                "numRetries": integer,
                "restartDelay": integer
              }
            },
            "hlsId3SegmentTagging": enum,
            "iFrameOnlyPlaylists": enum,
            "incompleteSegmentBehavior": enum,
            "indexNSegments": integer,
            "inputLossAction": enum,
            "ivInManifest": enum,
            "ivSource": enum,
            "keepSegments": integer,
            "keyFormat": "string",
```

```
            "keyFormatVersions": "string",
            "keyProviderSettings": {
              "staticKeySettings": {
                "keyProviderServer": {
                  "passwordParam": "string",
                  "uri": "string",
                  "username": "string"
                },
                "staticKeyValue": "string"
              }
            },
            "manifestCompression": enum,
            "manifestDurationFormat": enum,
            "minSegmentLength": integer,
            "mode": enum,
            "outputSelection": enum,
            "programDateTime": enum,
            "programDateTimeClock": enum,
            "programDateTimePeriod": integer,
            "redundantManifest": enum,
            "segmentLength": integer,
            "segmentationMode": enum,
            "segmentsPerSubdirectory": integer,
            "streamInfResolution": enum,
            "timedMetadataId3Frame": enum,
            "timedMetadataId3Period": integer,
            "timestampDeltaMilliseconds": integer,
            "tsFileMode": enum
          },
          "mediaPackageGroupSettings": {
            "destination": {
              "destinationRefId": "string"
            }
          },
          "msSmoothGroupSettings": {
            "acquisitionPointId": "string",
            "audioOnlyTimecodeControl": enum,
            "certificateMode": enum,
            "connectionRetryInterval": integer,
            "destination": {
              "destinationRefId": "string"
            },
            "eventId": "string",
            "eventIdMode": enum,
```

```
            "eventStopBehavior": enum,
            "filecacheDuration": integer,
            "fragmentLength": integer,
            "inputLossAction": enum,
            "numRetries": integer,
            "restartDelay": integer,
            "segmentationMode": enum,
            "sendDelayMs": integer,
            "sparseTrackType": enum,
            "streamManifestBehavior": enum,
            "timestampOffset": "string",
            "timestampOffsetMode": enum
          },
          "multiplexGroupSettings": {
          },
          "rtmpGroupSettings": {
            "adMarkers": [
              enum
            ],
            "authenticationScheme": enum,
            "cacheFullBehavior": enum,
            "cacheLength": integer,
            "captionData": enum,
            "inputLossAction": enum,
            "restartDelay": integer
          },
          "udpGroupSettings": {
            "inputLossAction": enum,
            "timedMetadataId3Frame": enum,
            "timedMetadataId3Period": integer
          }
        },
        "outputs": [
          {
            "audioDescriptionNames": [
              "string"
            ],
            "captionDescriptionNames": [
              "string"
            ],
            "outputName": "string",
            "outputSettings": {
              "archiveOutputSettings": {
                "containerSettings": {
```

```
        "m2tsSettings": {
          "absentInputAudioBehavior": enum,
          "arib": enum,
          "aribCaptionsPid": "string",
          "aribCaptionsPidControl": enum,
          "audioBufferModel": enum,
          "audioFramesPerPes": integer,
          "audioPids": "string",
          "audioStreamType": enum,
          "bitrate": integer,
          "bufferModel": enum,
          "ccDescriptor": enum,
          "dvbNitSettings": {
            "networkId": integer,
            "networkName": "string",
            "repInterval": integer
          },
          "dvbSdtSettings": {
            "outputSdt": enum,
            "repInterval": integer,
            "serviceName": "string",
            "serviceProviderName": "string"
          },
          "dvbSubPids": "string",
          "dvbTdtSettings": {
            "repInterval": integer
          },
          "dvbTeletextPid": "string",
          "ebif": enum,
          "ebpAudioInterval": enum,
          "ebpLookaheadMs": integer,
          "ebpPlacement": enum,
          "ecmPid": "string",
          "esRateInPes": enum,
          "etvPlatformPid": "string",
          "etvSignalPid": "string",
          "fragmentTime": number,
          "klv": enum,
          "klvDataPids": "string",
          "nielsenId3Behavior": enum,
          "nullPacketBitrate": number,
          "patInterval": integer,
          "pcrControl": enum,
          "pcrPeriod": integer,
```

```
            "pcrPid": "string",
            "pmtInterval": integer,
            "pmtPid": "string",
            "programNum": integer,
            "rateMode": enum,
            "scte27Pids": "string",
            "scte35Control": enum,
            "scte35Pid": "string",
            "segmentationMarkers": enum,
            "segmentationStyle": enum,
            "segmentationTime": number,
            "timedMetadataBehavior": enum,
            "timedMetadataPid": "string",
            "transportStreamId": integer,
            "videoPid": "string"
          },
          "rawSettings": {
          }
        },
        "extension": "string",
        "nameModifier": "string"
      },
      "frameCaptureOutputSettings": {
        "nameModifier": "string"
      },
      "hlsOutputSettings": {
        "h265PackagingType": enum,
        "hlsSettings": {
          "audioOnlyHlsSettings": {
            "audioGroupId": "string",
            "audioOnlyImage": {
              "passwordParam": "string",
              "uri": "string",
              "username": "string"
            },
            "audioTrackType": enum,
            "segmentType": enum
          },
          "fmp4HlsSettings": {
            "audioRenditionSets": "string",
            "nielsenId3Behavior": enum,
            "timedMetadataBehavior": enum
          },
          "frameCaptureHlsSettings": {
```

```
            },
            "standardHlsSettings": {
              "audioRenditionSets": "string",
              "m3u8Settings": {
                "audioFramesPerPes": integer,
                "audioPids": "string",
                "ecmPid": "string",
                "nielsenId3Behavior": enum,
                "patInterval": integer,
                "pcrControl": enum,
                "pcrPeriod": integer,
                "pcrPid": "string",
                "pmtInterval": integer,
                "pmtPid": "string",
                "programNum": integer,
                "scte35Behavior": enum,
                "scte35Pid": "string",
                "timedMetadataBehavior": enum,
                "timedMetadataPid": "string",
                "transportStreamId": integer,
                "videoPid": "string"
              }
            }
          },
          "nameModifier": "string",
          "segmentModifier": "string"
        },
        "mediaPackageOutputSettings": {
        },
        "msSmoothOutputSettings": {
          "h265PackagingType": enum,
          "nameModifier": "string"
        },
        "multiplexOutputSettings": {
          "destination": {
            "destinationRefId": "string"
          }
        },
        "rtmpOutputSettings": {
          "certificateMode": enum,
          "connectionRetryInterval": integer,
          "destination": {
            "destinationRefId": "string"
          },
```

```
            "numRetries": integer
          },
          "udpOutputSettings": {
            "bufferMsec": integer,
            "containerSettings": {
              "m2tsSettings": {
                "absentInputAudioBehavior": enum,
                "arib": enum,
                "aribCaptionsPid": "string",
                "aribCaptionsPidControl": enum,
                "audioBufferModel": enum,
                "audioFramesPerPes": integer,
                "audioPids": "string",
                "audioStreamType": enum,
                "bitrate": integer,
                "bufferModel": enum,
                "ccDescriptor": enum,
                "dvbNitSettings": {
                  "networkId": integer,
                  "networkName": "string",
                  "repInterval": integer
                },
                "dvbSdtSettings": {
                  "outputSdt": enum,
                  "repInterval": integer,
                  "serviceName": "string",
                  "serviceProviderName": "string"
                },
                "dvbSubPids": "string",
                "dvbTdtSettings": {
                  "repInterval": integer
                },
                "dvbTeletextPid": "string",
                "ebif": enum,
                "ebpAudioInterval": enum,
                "ebpLookaheadMs": integer,
                "ebpPlacement": enum,
                "ecmPid": "string",
                "esRateInPes": enum,
                "etvPlatformPid": "string",
                "etvSignalPid": "string",
                "fragmentTime": number,
                "klv": enum,
                "klvDataPids": "string",
```

```
                    "nielsenId3Behavior": enum,
                    "nullPacketBitrate": number,
                    "patInterval": integer,
                    "pcrControl": enum,
                    "pcrPeriod": integer,
                    "pcrPid": "string",
                    "pmtInterval": integer,
                    "pmtPid": "string",
                    "programNum": integer,
                    "rateMode": enum,
                    "scte27Pids": "string",
                    "scte35Control": enum,
                    "scte35Pid": "string",
                    "segmentationMarkers": enum,
                    "segmentationStyle": enum,
                    "segmentationTime": number,
                    "timedMetadataBehavior": enum,
                    "timedMetadataPid": "string",
                    "transportStreamId": integer,
                    "videoPid": "string"
                  }
                },
                "destination": {
                  "destinationRefId": "string"
                },
                "fecOutputSettings": {
                  "columnDepth": integer,
                  "includeFec": enum,
                  "rowLength": integer
                }
              }
            },
            "videoDescriptionName": "string"
          }
        ]
      }
    ],
    "timecodeConfig": {
      "source": enum,
      "syncThreshold": integer
    },
    "videoDescriptions": [
      {
        "codecSettings": {
```

```
          "frameCaptureSettings": {
            "captureInterval": integer,
            "captureIntervalUnits": enum
          },
          "h264Settings": {
            "adaptiveQuantization": enum,
            "afdSignaling": enum,
            "bitrate": integer,
            "bufFillPct": integer,
            "bufSize": integer,
            "colorMetadata": enum,
            "colorSpaceSettings": {
              "colorSpacePassthroughSettings": {
              },
              "rec601Settings": {
              },
              "rec709Settings": {
              }
            },
            "entropyEncoding": enum,
            "filterSettings": {
              "temporalFilterSettings": {
                "postFilterSharpening": enum,
                "strength": enum
              }
            },
            "fixedAfd": enum,
            "flickerAq": enum,
            "forceFieldPictures": enum,
            "framerateControl": enum,
            "framerateDenominator": integer,
            "framerateNumerator": integer,
            "gopBReference": enum,
            "gopClosedCadence": integer,
            "gopNumBFrames": integer,
            "gopSize": number,
            "gopSizeUnits": enum,
            "level": enum,
            "lookAheadRateControl": enum,
            "maxBitrate": integer,
            "minIInterval": integer,
            "numRefFrames": integer,
            "parControl": enum,
            "parDenominator": integer,
```

```
            "parNumerator": integer,
            "profile": enum,
            "qualityLevel": enum,
            "qvbrQualityLevel": integer,
            "rateControlMode": enum,
            "scanType": enum,
            "sceneChangeDetect": enum,
            "slices": integer,
            "softness": integer,
            "spatialAq": enum,
            "subgopLength": enum,
            "syntax": enum,
            "temporalAq": enum,
            "timecodeInsertion": enum
          },
          "h265Settings": {
            "adaptiveQuantization": enum,
            "afdSignaling": enum,
            "alternativeTransferFunction": enum,
            "bitrate": integer,
            "bufSize": integer,
            "colorMetadata": enum,
            "colorSpaceSettings": {
              "colorSpacePassthroughSettings": {
              },
              "hdr10Settings": {
                "maxCll": integer,
                "maxFall": integer
              },
              "rec601Settings": {
              },
              "rec709Settings": {
              }
            },
            "filterSettings": {
              "temporalFilterSettings": {
                "postFilterSharpening": enum,
                "strength": enum
              }
            },
            "fixedAfd": enum,
            "flickerAq": enum,
            "framerateDenominator": integer,
            "framerateNumerator": integer,
```

```
            "gopClosedCadence": integer,
            "gopSize": number,
            "gopSizeUnits": enum,
            "level": enum,
            "lookAheadRateControl": enum,
            "maxBitrate": integer,
            "minIInterval": integer,
            "parDenominator": integer,
            "parNumerator": integer,
            "profile": enum,
            "qvbrQualityLevel": integer,
            "rateControlMode": enum,
            "scanType": enum,
            "sceneChangeDetect": enum,
            "slices": integer,
            "tier": enum,
            "timecodeInsertion": enum
          },
          "mpeg2Settings": {
            "adaptiveQuantization": enum,
            "afdSignaling": enum,
            "colorMetadata": enum,
            "colorSpace": enum,
            "displayAspectRatio": enum,
            "filterSettings": {
              "temporalFilterSettings": {
                "postFilterSharpening": enum,
                "strength": enum
              }
            },
            "fixedAfd": enum,
            "framerateDenominator": integer,
            "framerateNumerator": integer,
            "gopClosedCadence": integer,
            "gopNumBFrames": integer,
            "gopSize": number,
            "gopSizeUnits": enum,
            "scanType": enum,
            "subgopLength": enum,
            "timecodeInsertion": enum
          }
        },
        "height": integer,
        "name": "string",
```

```
        "respondToAfd": enum,
        "scalingBehavior": enum,
        "sharpness": integer,
        "width": integer
      }
    ]
  },
  "id": "string",
  "inputAttachments": [
    {
      "automaticInputFailoverSettings": {
        "errorClearTimeMsec": integer,
        "failoverConditions": [
          {
            "failoverConditionSettings": {
              "audioSilenceSettings": {
                "audioSelectorName": "string",
                "audioSilenceThresholdMsec": integer
              },
              "inputLossSettings": {
                "inputLossThresholdMsec": integer
              },
              "videoBlackSettings": {
                "blackDetectThreshold": number,
                "videoBlackThresholdMsec": integer
              }
            }
          }
        ],
        "inputPreference": enum,
        "secondaryInputId": "string"
      },
      "inputAttachmentName": "string",
      "inputId": "string",
      "inputSettings": {
        "audioSelectors": [
          {
            "name": "string",
            "selectorSettings": {
              "audioHlsRenditionSelection": {
                "groupId": "string",
                "name": "string"
              },
              "audioLanguageSelection": {
```

```
            "languageCode": "string",
            "languageSelectionPolicy": enum
          },
          "audioPidSelection": {
            "pid": integer
          },
          "audioTrackSelection": {
            "tracks": [
              {
                "track": integer
              }
            ]
          }
        }
      }
    ],
    "captionSelectors": [
      {
        "languageCode": "string",
        "name": "string",
        "selectorSettings": {
          "ancillarySourceSettings": {
            "sourceAncillaryChannelNumber": integer
          },
          "aribSourceSettings": {
          },
          "dvbSubSourceSettings": {
            "ocrLanguage": enum,
            "pid": integer
          },
          "embeddedSourceSettings": {
            "convert608To708": enum,
            "scte20Detection": enum,
            "source608ChannelNumber": integer,
            "source608TrackNumber": integer
          },
          "scte20SourceSettings": {
            "convert608To708": enum,
            "source608ChannelNumber": integer
          },
          "scte27SourceSettings": {
            "ocrLanguage": enum,
            "pid": integer
          },
```

```
            "teletextSourceSettings": {
              "outputRectangle": {
                "height": number,
                "leftOffset": number,
                "topOffset": number,
                "width": number
              },
              "pageNumber": "string"
            }
          }
        }
      ],
      "deblockFilter": enum,
      "denoiseFilter": enum,
      "filterStrength": integer,
      "inputFilter": enum,
      "networkInputSettings": {
        "hlsInputSettings": {
          "bandwidth": integer,
          "bufferSegments": integer,
          "retries": integer,
          "retryInterval": integer,
          "scte35Source": enum
        },
        "serverValidation": enum
      },
      "smpte2038DataPreference": enum,
      "sourceEndBehavior": enum,
      "videoSelector": {
        "colorSpace": enum,
        "colorSpaceSettings": {
          "hdr10Settings": {
            "maxCll": integer,
            "maxFall": integer
          }
        },
        "colorSpaceUsage": enum,
        "selectorSettings": {
          "videoSelectorPid": {
            "pid": integer
          },
          "videoSelectorProgramId": {
            "programId": integer
          }
```

```
          }
        }
      }
    }
  ],
  "inputSpecification": {
    "codec": enum,
    "maximumBitrate": enum,
    "resolution": enum
  },
  "logLevel": enum,
  "maintenance": {
    "maintenanceDay": enum,
    "maintenanceDeadline": "string",
    "maintenanceScheduledDate": "string",
    "maintenanceStartTime": "string"
  },
  "name": "string",
  "pipelineDetails": [
    {
      "activeInputAttachmentName": "string",
      "activeInputSwitchActionName": "string",
      "activeMotionGraphicsActionName": "string",
      "activeMotionGraphicsUri": "string",
      "pipelineId": "string"
    }
  ],
  "pipelinesRunningCount": integer,
  "roleArn": "string",
  "state": enum,
  "tags": {
  },
  "vpc": {
    "availabilityZones": [
      "string"
    ],
    "networkInterfaceIds": [
      "string"
    ],
    "securityGroupIds": [
      "string"
    ],
    "subnetIds": [
      "string"
```

```
    ]
  }
}
```

UpdateChannelResultModel schema

```
{
  "channel": {
    "arn": "string",
    "cdiInputSpecification": {
      "resolution": enum
    },
    "channelClass": enum,
    "destinations": [
      {
        "id": "string",
        "mediaPackageSettings": [
          {
            "channelId": "string"
          }
        ],
        "multiplexSettings": {
          "multiplexId": "string",
          "programName": "string"
        },
        "settings": [
          {
            "passwordParam": "string",
            "streamName": "string",
            "url": "string",
            "username": "string"
          }
        ]
      }
    ],
    "egressEndpoints": [
      {
        "sourceIp": "string"
      }
    ],
    "encoderSettings": {
      "audioDescriptions": [
        {
```

```
        "audioNormalizationSettings": {
          "algorithm": enum,
          "algorithmControl": enum,
          "targetLkfs": number
        },
        "audioSelectorName": "string",
        "audioType": enum,
        "audioTypeControl": enum,
        "audioWatermarkingSettings": {
          "nielsenWatermarksSettings": {
            "nielsenCbetSettings": {
              "cbetCheckDigitString": "string",
              "cbetStepaside": enum,
              "csid": "string"
            },
            "nielsenDistributionType": enum,
            "nielsenNaesIiNwSettings": {
              "checkDigitString": "string",
              "sid": number
            }
          }
        },
        "codecSettings": {
          "aacSettings": {
            "bitrate": number,
            "codingMode": enum,
            "inputType": enum,
            "profile": enum,
            "rateControlMode": enum,
            "rawFormat": enum,
            "sampleRate": number,
            "spec": enum,
            "vbrQuality": enum
          },
          "ac3Settings": {
            "bitrate": number,
            "bitstreamMode": enum,
            "codingMode": enum,
            "dialnorm": integer,
            "drcProfile": enum,
            "lfeFilter": enum,
            "metadataControl": enum
          },
          "eac3Settings": {
```

```
            "attenuationControl": enum,
            "bitrate": number,
            "bitstreamMode": enum,
            "codingMode": enum,
            "dcFilter": enum,
            "dialnorm": integer,
            "drcLine": enum,
            "drcRf": enum,
            "lfeControl": enum,
            "lfeFilter": enum,
            "loRoCenterMixLevel": number,
            "loRoSurroundMixLevel": number,
            "ltRtCenterMixLevel": number,
            "ltRtSurroundMixLevel": number,
            "metadataControl": enum,
            "passthroughControl": enum,
            "phaseControl": enum,
            "stereoDownmix": enum,
            "surroundExMode": enum,
            "surroundMode": enum
          },
          "mp2Settings": {
            "bitrate": number,
            "codingMode": enum,
            "sampleRate": number
          },
          "passThroughSettings": {
          },
          "wavSettings": {
            "bitDepth": number,
            "codingMode": enum,
            "sampleRate": number
          }
        },
        "languageCode": "string",
        "languageCodeControl": enum,
        "name": "string",
        "remixSettings": {
          "channelMappings": [
            {
              "inputChannelLevels": [
                {
                  "gain": integer,
                  "inputChannel": integer
```

```
                }
              ],
              "outputChannel": integer
            }
          ],
          "channelsIn": integer,
          "channelsOut": integer
        },
        "streamName": "string"
      }
    ],
    "availBlanking": {
      "availBlankingImage": {
        "passwordParam": "string",
        "uri": "string",
        "username": "string"
      },
      "state": enum
    },
    "availConfiguration": {
      "availSettings": {
        "scte35SpliceInsert": {
          "adAvailOffset": integer,
          "noRegionalBlackoutFlag": enum,
          "webDeliveryAllowedFlag": enum
        },
        "scte35TimeSignalApos": {
          "adAvailOffset": integer,
          "noRegionalBlackoutFlag": enum,
          "webDeliveryAllowedFlag": enum
        }
      }
    },
    "blackoutSlate": {
      "blackoutSlateImage": {
        "passwordParam": "string",
        "uri": "string",
        "username": "string"
      },
      "networkEndBlackout": enum,
      "networkEndBlackoutImage": {
        "passwordParam": "string",
        "uri": "string",
        "username": "string"
```

```
        },
        "networkId": "string",
        "state": enum
      },
      "captionDescriptions": [
        {
          "captionSelectorName": "string",
          "destinationSettings": {
            "aribDestinationSettings": {
            },
            "burnInDestinationSettings": {
              "alignment": enum,
              "backgroundColor": enum,
              "backgroundOpacity": integer,
              "font": {
                "passwordParam": "string",
                "uri": "string",
                "username": "string"
              },
              "fontColor": enum,
              "fontOpacity": integer,
              "fontResolution": integer,
              "fontSize": "string",
              "outlineColor": enum,
              "outlineSize": integer,
              "shadowColor": enum,
              "shadowOpacity": integer,
              "shadowXOffset": integer,
              "shadowYOffset": integer,
              "teletextGridControl": enum,
              "xPosition": integer,
              "yPosition": integer
            },
            "dvbSubDestinationSettings": {
              "alignment": enum,
              "backgroundColor": enum,
              "backgroundOpacity": integer,
              "font": {
                "passwordParam": "string",
                "uri": "string",
                "username": "string"
              },
              "fontColor": enum,
              "fontOpacity": integer,
```

```
            "fontResolution": integer,
            "fontSize": "string",
            "outlineColor": enum,
            "outlineSize": integer,
            "shadowColor": enum,
            "shadowOpacity": integer,
            "shadowXOffset": integer,
            "shadowYOffset": integer,
            "teletextGridControl": enum,
            "xPosition": integer,
            "yPosition": integer
          },
          "ebuTtDDestinationSettings": {
            "copyrightHolder": "string",
            "fillLineGap": enum,
            "fontFamily": "string",
            "styleControl": enum
          },
          "embeddedDestinationSettings": {
          },
          "embeddedPlusScte20DestinationSettings": {
          },
          "rtmpCaptionInfoDestinationSettings": {
          },
          "scte20PlusEmbeddedDestinationSettings": {
          },
          "scte27DestinationSettings": {
          },
          "smpteTtDestinationSettings": {
          },
          "teletextDestinationSettings": {
          },
          "ttmlDestinationSettings": {
            "styleControl": enum
          },
          "webvttDestinationSettings": {
            "styleControl": enum
          }
        },
        "languageCode": "string",
        "languageDescription": "string",
        "name": "string"
      }
    ],
```

```
      "featureActivations": {
        "inputPrepareScheduleActions": enum
      },
      "globalConfiguration": {
        "initialAudioGain": integer,
        "inputEndAction": enum,
        "inputLossBehavior": {
          "blackFrameMsec": integer,
          "inputLossImageColor": "string",
          "inputLossImageSlate": {
            "passwordParam": "string",
            "uri": "string",
            "username": "string"
          },
          "inputLossImageType": enum,
          "repeatFrameMsec": integer
        },
        "outputLockingMode": enum,
        "outputTimingSource": enum,
        "supportLowFramerateInputs": enum
      },
      "motionGraphicsConfiguration": {
        "motionGraphicsInsertion": enum,
        "motionGraphicsSettings": {
          "htmlMotionGraphicsSettings": {
          }
        }
      },
      "nielsenConfiguration": {
        "distributorId": "string",
        "nielsenPcmToId3Tagging": enum
      },
      "outputGroups": [
        {
          "name": "string",
          "outputGroupSettings": {
            "archiveGroupSettings": {
              "archiveCdnSettings": {
                "archiveS3Settings": {
                  "cannedAcl": enum,
                  "logUploads": enum
                }
              },
              "destination": {
```

```
            "destinationRefId": "string"
          },
          "rolloverInterval": integer
        },
        "frameCaptureGroupSettings": {
          "destination": {
            "destinationRefId": "string"
          },
          "frameCaptureCdnSettings": {
            "frameCaptureS3Settings": {
              "cannedAcl": enum,
              "logUploads": enum
            }
          }
        },
        "hlsGroupSettings": {
          "adMarkers": [
            enum
          ],
          "baseUrlContent": "string",
          "baseUrlContent1": "string",
          "baseUrlManifest": "string",
          "baseUrlManifest1": "string",
          "captionLanguageMappings": [
            {
              "captionChannel": integer,
              "languageCode": "string",
              "languageDescription": "string"
            }
          ],
          "captionLanguageSetting": enum,
          "clientCache": enum,
          "codecSpecification": enum,
          "constantIv": "string",
          "destination": {
            "destinationRefId": "string"
          },
          "directoryStructure": enum,
          "discontinuityTags": enum,
          "encryptionType": enum,
          "hlsCdnSettings": {
            "hlsAkamaiSettings": {
              "connectionRetryInterval": integer,
              "filecacheDuration": integer,
```

```
            "httpTransferMode": enum,
            "numRetries": integer,
            "restartDelay": integer,
            "salt": "string",
            "token": "string"
          },
          "hlsBasicPutSettings": {
            "connectionRetryInterval": integer,
            "filecacheDuration": integer,
            "numRetries": integer,
            "restartDelay": integer
          },
          "hlsMediaStoreSettings": {
            "connectionRetryInterval": integer,
            "filecacheDuration": integer,
            "mediaStoreStorageClass": enum,
            "numRetries": integer,
            "restartDelay": integer
          },
          "hlsS3Settings": {
            "cannedAcl": enum,
            "logUploads": enum
          },
          "hlsWebdavSettings": {
            "connectionRetryInterval": integer,
            "filecacheDuration": integer,
            "httpTransferMode": enum,
            "numRetries": integer,
            "restartDelay": integer
          }
        },
        "hlsId3SegmentTagging": enum,
        "iFrameOnlyPlaylists": enum,
        "incompleteSegmentBehavior": enum,
        "indexNSegments": integer,
        "inputLossAction": enum,
        "ivInManifest": enum,
        "ivSource": enum,
        "keepSegments": integer,
        "keyFormat": "string",
        "keyFormatVersions": "string",
        "keyProviderSettings": {
          "staticKeySettings": {
            "keyProviderServer": {
```

```
                "passwordParam": "string",
                "uri": "string",
                "username": "string"
              },
              "staticKeyValue": "string"
            }
          },
          "manifestCompression": enum,
          "manifestDurationFormat": enum,
          "minSegmentLength": integer,
          "mode": enum,
          "outputSelection": enum,
          "programDateTime": enum,
          "programDateTimeClock": enum,
          "programDateTimePeriod": integer,
          "redundantManifest": enum,
          "segmentLength": integer,
          "segmentationMode": enum,
          "segmentsPerSubdirectory": integer,
          "streamInfResolution": enum,
          "timedMetadataId3Frame": enum,
          "timedMetadataId3Period": integer,
          "timestampDeltaMilliseconds": integer,
          "tsFileMode": enum
        },
        "mediaPackageGroupSettings": {
          "destination": {
            "destinationRefId": "string"
          }
        },
        "msSmoothGroupSettings": {
          "acquisitionPointId": "string",
          "audioOnlyTimecodeControl": enum,
          "certificateMode": enum,
          "connectionRetryInterval": integer,
          "destination": {
            "destinationRefId": "string"
          },
          "eventId": "string",
          "eventIdMode": enum,
          "eventStopBehavior": enum,
          "filecacheDuration": integer,
          "fragmentLength": integer,
          "inputLossAction": enum,
```

```
            "numRetries": integer,
            "restartDelay": integer,
            "segmentationMode": enum,
            "sendDelayMs": integer,
            "sparseTrackType": enum,
            "streamManifestBehavior": enum,
            "timestampOffset": "string",
            "timestampOffsetMode": enum
          },
          "multiplexGroupSettings": {
          },
          "rtmpGroupSettings": {
            "adMarkers": [
              enum
            ],
            "authenticationScheme": enum,
            "cacheFullBehavior": enum,
            "cacheLength": integer,
            "captionData": enum,
            "inputLossAction": enum,
            "restartDelay": integer
          },
          "udpGroupSettings": {
            "inputLossAction": enum,
            "timedMetadataId3Frame": enum,
            "timedMetadataId3Period": integer
          }
        },
        "outputs": [
          {
            "audioDescriptionNames": [
              "string"
            ],
            "captionDescriptionNames": [
              "string"
            ],
            "outputName": "string",
            "outputSettings": {
              "archiveOutputSettings": {
                "containerSettings": {
                  "m2tsSettings": {
                    "absentInputAudioBehavior": enum,
                    "arib": enum,
                    "aribCaptionsPid": "string",
```

```
        "aribCaptionsPidControl": enum,
        "audioBufferModel": enum,
        "audioFramesPerPes": integer,
        "audioPids": "string",
        "audioStreamType": enum,
        "bitrate": integer,
        "bufferModel": enum,
        "ccDescriptor": enum,
        "dvbNitSettings": {
          "networkId": integer,
          "networkName": "string",
          "repInterval": integer
        },
        "dvbSdtSettings": {
          "outputSdt": enum,
          "repInterval": integer,
          "serviceName": "string",
          "serviceProviderName": "string"
        },
        "dvbSubPids": "string",
        "dvbTdtSettings": {
          "repInterval": integer
        },
        "dvbTeletextPid": "string",
        "ebif": enum,
        "ebpAudioInterval": enum,
        "ebpLookaheadMs": integer,
        "ebpPlacement": enum,
        "ecmPid": "string",
        "esRateInPes": enum,
        "etvPlatformPid": "string",
        "etvSignalPid": "string",
        "fragmentTime": number,
        "klv": enum,
        "klvDataPids": "string",
        "nielsenId3Behavior": enum,
        "nullPacketBitrate": number,
        "patInterval": integer,
        "pcrControl": enum,
        "pcrPeriod": integer,
        "pcrPid": "string",
        "pmtInterval": integer,
        "pmtPid": "string",
        "programNum": integer,
```

```
                "rateMode": enum,
                "scte27Pids": "string",
                "scte35Control": enum,
                "scte35Pid": "string",
                "segmentationMarkers": enum,
                "segmentationStyle": enum,
                "segmentationTime": number,
                "timedMetadataBehavior": enum,
                "timedMetadataPid": "string",
                "transportStreamId": integer,
                "videoPid": "string"
              },
              "rawSettings": {
              }
            },
            "extension": "string",
            "nameModifier": "string"
          },
          "frameCaptureOutputSettings": {
            "nameModifier": "string"
          },
          "hlsOutputSettings": {
            "h265PackagingType": enum,
            "hlsSettings": {
              "audioOnlyHlsSettings": {
                "audioGroupId": "string",
                "audioOnlyImage": {
                  "passwordParam": "string",
                  "uri": "string",
                  "username": "string"
                },
                "audioTrackType": enum,
                "segmentType": enum
              },
              "fmp4HlsSettings": {
                "audioRenditionSets": "string",
                "nielsenId3Behavior": enum,
                "timedMetadataBehavior": enum
              },
              "frameCaptureHlsSettings": {
              },
              "standardHlsSettings": {
                "audioRenditionSets": "string",
                "m3u8Settings": {
```

```
                    "audioFramesPerPes": integer,
                    "audioPids": "string",
                    "ecmPid": "string",
                    "nielsenId3Behavior": enum,
                    "patInterval": integer,
                    "pcrControl": enum,
                    "pcrPeriod": integer,
                    "pcrPid": "string",
                    "pmtInterval": integer,
                    "pmtPid": "string",
                    "programNum": integer,
                    "scte35Behavior": enum,
                    "scte35Pid": "string",
                    "timedMetadataBehavior": enum,
                    "timedMetadataPid": "string",
                    "transportStreamId": integer,
                    "videoPid": "string"
                  }
                }
              },
              "nameModifier": "string",
              "segmentModifier": "string"
            },
            "mediaPackageOutputSettings": {
            },
            "msSmoothOutputSettings": {
              "h265PackagingType": enum,
              "nameModifier": "string"
            },
            "multiplexOutputSettings": {
              "destination": {
                "destinationRefId": "string"
              }
            },
            "rtmpOutputSettings": {
              "certificateMode": enum,
              "connectionRetryInterval": integer,
              "destination": {
                "destinationRefId": "string"
              },
              "numRetries": integer
            },
            "udpOutputSettings": {
              "bufferMsec": integer,
```

```
      "containerSettings": {
        "m2tsSettings": {
          "absentInputAudioBehavior": enum,
          "arib": enum,
          "aribCaptionsPid": "string",
          "aribCaptionsPidControl": enum,
          "audioBufferModel": enum,
          "audioFramesPerPes": integer,
          "audioPids": "string",
          "audioStreamType": enum,
          "bitrate": integer,
          "bufferModel": enum,
          "ccDescriptor": enum,
          "dvbNitSettings": {
            "networkId": integer,
            "networkName": "string",
            "repInterval": integer
          },
          "dvbSdtSettings": {
            "outputSdt": enum,
            "repInterval": integer,
            "serviceName": "string",
            "serviceProviderName": "string"
          },
          "dvbSubPids": "string",
          "dvbTdtSettings": {
            "repInterval": integer
          },
          "dvbTeletextPid": "string",
          "ebif": enum,
          "ebpAudioInterval": enum,
          "ebpLookaheadMs": integer,
          "ebpPlacement": enum,
          "ecmPid": "string",
          "esRateInPes": enum,
          "etvPlatformPid": "string",
          "etvSignalPid": "string",
          "fragmentTime": number,
          "klv": enum,
          "klvDataPids": "string",
          "nielsenId3Behavior": enum,
          "nullPacketBitrate": number,
          "patInterval": integer,
          "pcrControl": enum,
```

```
                    "pcrPeriod": integer,
                    "pcrPid": "string",
                    "pmtInterval": integer,
                    "pmtPid": "string",
                    "programNum": integer,
                    "rateMode": enum,
                    "scte27Pids": "string",
                    "scte35Control": enum,
                    "scte35Pid": "string",
                    "segmentationMarkers": enum,
                    "segmentationStyle": enum,
                    "segmentationTime": number,
                    "timedMetadataBehavior": enum,
                    "timedMetadataPid": "string",
                    "transportStreamId": integer,
                    "videoPid": "string"
                  }
                },
                "destination": {
                  "destinationRefId": "string"
                },
                "fecOutputSettings": {
                  "columnDepth": integer,
                  "includeFec": enum,
                  "rowLength": integer
                }
              }
            },
            "videoDescriptionName": "string"
          }
        ]
      }
    ],
    "timecodeConfig": {
      "source": enum,
      "syncThreshold": integer
    },
    "videoDescriptions": [
      {
        "codecSettings": {
          "frameCaptureSettings": {
            "captureInterval": integer,
            "captureIntervalUnits": enum
          },
```

```
        "h264Settings": {
          "adaptiveQuantization": enum,
          "afdSignaling": enum,
          "bitrate": integer,
          "bufFillPct": integer,
          "bufSize": integer,
          "colorMetadata": enum,
          "colorSpaceSettings": {
            "colorSpacePassthroughSettings": {
            },
            "rec601Settings": {
            },
            "rec709Settings": {
            }
          },
          "entropyEncoding": enum,
          "filterSettings": {
            "temporalFilterSettings": {
              "postFilterSharpening": enum,
              "strength": enum
            }
          },
          "fixedAfd": enum,
          "flickerAq": enum,
          "forceFieldPictures": enum,
          "framerateControl": enum,
          "framerateDenominator": integer,
          "framerateNumerator": integer,
          "gopBReference": enum,
          "gopClosedCadence": integer,
          "gopNumBFrames": integer,
          "gopSize": number,
          "gopSizeUnits": enum,
          "level": enum,
          "lookAheadRateControl": enum,
          "maxBitrate": integer,
          "minIInterval": integer,
          "numRefFrames": integer,
          "parControl": enum,
          "parDenominator": integer,
          "parNumerator": integer,
          "profile": enum,
          "qualityLevel": enum,
          "qvbrQualityLevel": integer,
```

```
            "rateControlMode": enum,
            "scanType": enum,
            "sceneChangeDetect": enum,
            "slices": integer,
            "softness": integer,
            "spatialAq": enum,
            "subgopLength": enum,
            "syntax": enum,
            "temporalAq": enum,
            "timecodeInsertion": enum
          },
          "h265Settings": {
            "adaptiveQuantization": enum,
            "afdSignaling": enum,
            "alternativeTransferFunction": enum,
            "bitrate": integer,
            "bufSize": integer,
            "colorMetadata": enum,
            "colorSpaceSettings": {
              "colorSpacePassthroughSettings": {
              },
              "hdr10Settings": {
                "maxCll": integer,
                "maxFall": integer
              },
              "rec601Settings": {
              },
              "rec709Settings": {
              }
            },
            "filterSettings": {
              "temporalFilterSettings": {
                "postFilterSharpening": enum,
                "strength": enum
              }
            },
            "fixedAfd": enum,
            "flickerAq": enum,
            "framerateDenominator": integer,
            "framerateNumerator": integer,
            "gopClosedCadence": integer,
            "gopSize": number,
            "gopSizeUnits": enum,
            "level": enum,
```

```
            "lookAheadRateControl": enum,
            "maxBitrate": integer,
            "minIInterval": integer,
            "parDenominator": integer,
            "parNumerator": integer,
            "profile": enum,
            "qvbrQualityLevel": integer,
            "rateControlMode": enum,
            "scanType": enum,
            "sceneChangeDetect": enum,
            "slices": integer,
            "tier": enum,
            "timecodeInsertion": enum
          },
          "mpeg2Settings": {
            "adaptiveQuantization": enum,
            "afdSignaling": enum,
            "colorMetadata": enum,
            "colorSpace": enum,
            "displayAspectRatio": enum,
            "filterSettings": {
              "temporalFilterSettings": {
                "postFilterSharpening": enum,
                "strength": enum
              }
            },
            "fixedAfd": enum,
            "framerateDenominator": integer,
            "framerateNumerator": integer,
            "gopClosedCadence": integer,
            "gopNumBFrames": integer,
            "gopSize": number,
            "gopSizeUnits": enum,
            "scanType": enum,
            "subgopLength": enum,
            "timecodeInsertion": enum
          }
        },
        "height": integer,
        "name": "string",
        "respondToAfd": enum,
        "scalingBehavior": enum,
        "sharpness": integer,
        "width": integer
```

```
        }
      ]
    },
    "id": "string",
    "inputAttachments": [
      {
        "automaticInputFailoverSettings": {
          "errorClearTimeMsec": integer,
          "failoverConditions": [
            {
              "failoverConditionSettings": {
                "audioSilenceSettings": {
                  "audioSelectorName": "string",
                  "audioSilenceThresholdMsec": integer
                },
                "inputLossSettings": {
                  "inputLossThresholdMsec": integer
                },
                "videoBlackSettings": {
                  "blackDetectThreshold": number,
                  "videoBlackThresholdMsec": integer
                }
              }
            }
          ],
          "inputPreference": enum,
          "secondaryInputId": "string"
        },
        "inputAttachmentName": "string",
        "inputId": "string",
        "inputSettings": {
          "audioSelectors": [
            {
              "name": "string",
              "selectorSettings": {
                "audioHlsRenditionSelection": {
                  "groupId": "string",
                  "name": "string"
                },
                "audioLanguageSelection": {
                  "languageCode": "string",
                  "languageSelectionPolicy": enum
                },
                "audioPidSelection": {
```

```
            "pid": integer
          },
          "audioTrackSelection": {
            "tracks": [
              {
                "track": integer
              }
            ]
          }
        }
      }
    ],
    "captionSelectors": [
      {
        "languageCode": "string",
        "name": "string",
        "selectorSettings": {
          "ancillarySourceSettings": {
            "sourceAncillaryChannelNumber": integer
          },
          "aribSourceSettings": {
          },
          "dvbSubSourceSettings": {
            "ocrLanguage": enum,
            "pid": integer
          },
          "embeddedSourceSettings": {
            "convert608To708": enum,
            "scte20Detection": enum,
            "source608ChannelNumber": integer,
            "source608TrackNumber": integer
          },
          "scte20SourceSettings": {
            "convert608To708": enum,
            "source608ChannelNumber": integer
          },
          "scte27SourceSettings": {
            "ocrLanguage": enum,
            "pid": integer
          },
          "teletextSourceSettings": {
            "outputRectangle": {
              "height": number,
              "leftOffset": number,
```

```
                    "topOffset": number,
                    "width": number
                  },
                  "pageNumber": "string"
                }
              }
            }
          ],
          "deblockFilter": enum,
          "denoiseFilter": enum,
          "filterStrength": integer,
          "inputFilter": enum,
          "networkInputSettings": {
            "hlsInputSettings": {
              "bandwidth": integer,
              "bufferSegments": integer,
              "retries": integer,
              "retryInterval": integer,
              "scte35Source": enum
            },
            "serverValidation": enum
          },
          "smpte2038DataPreference": enum,
          "sourceEndBehavior": enum,
          "videoSelector": {
            "colorSpace": enum,
            "colorSpaceSettings": {
              "hdr10Settings": {
                "maxCll": integer,
                "maxFall": integer
              }
            },
            "colorSpaceUsage": enum,
            "selectorSettings": {
              "videoSelectorPid": {
                "pid": integer
              },
              "videoSelectorProgramId": {
                "programId": integer
              }
            }
          }
        }
      }
```

```
    ],
    "inputSpecification": {
      "codec": enum,
      "maximumBitrate": enum,
      "resolution": enum
    },
    "logLevel": enum,
    "maintenance": {
      "maintenanceDay": enum,
      "maintenanceDeadline": "string",
      "maintenanceScheduledDate": "string",
      "maintenanceStartTime": "string"
    },
    "name": "string",
    "pipelineDetails": [
      {
        "activeInputAttachmentName": "string",
        "activeInputSwitchActionName": "string",
        "activeMotionGraphicsActionName": "string",
        "activeMotionGraphicsUri": "string",
        "pipelineId": "string"
      }
    ],
    "pipelinesRunningCount": integer,
    "roleArn": "string",
    "state": enum,
    "tags": {
    },
    "vpc": {
      "availabilityZones": [
        "string"
      ],
      "networkInterfaceIds": [
        "string"
      ],
      "securityGroupIds": [
        "string"
      ],
      "subnetIds": [
        "string"
      ]
    }
  }
```

```
}
```

InvalidRequest schema

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

AccessDenied schema

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

ResourceNotFound schema

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

ResourceConflict schema

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

ChannelConfigurationValidationError schema

```
{
  "message": "string",
  "validationErrors": [
    {
      "elementPath": "string",
      "errorMessage": "string"
    }
  ]
```

```
}
```

LimitExceeded schema

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

InternalServiceError schema

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

BadGatewayException schema

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

GatewayTimeoutException schema

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

Properties

AacCodingMode

Aac Coding Mode

```
AD_RECEIVER_MIX
CODING_MODE_1_0
CODING_MODE_1_1
CODING_MODE_2_0
CODING_MODE_5_1
```

AacInputType

Aac Input Type

BROADCASTER_MIXED_AD
NORMAL

AacProfile

Aac Profile

HEV1
HEV2
LC

AacRateControlMode

Aac Rate Control Mode

CBR
VBR

AacRawFormat

Aac Raw Format

LATM_LOAS
NONE

AacSettings

Aac Settings

bitrate

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

Type: number
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: AacCodingMode
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: AacInputType
Required: False

profile

AAC Profile.

Type: AacProfile
Required: False

rateControlMode

Rate Control Mode.

Type: AacRateControlMode
Required: False

rawFormat

Sets LATM / LOAS AAC output for raw containers.

Type: AacRawFormat
Required: False

sampleRate

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

Type: number
Required: False

spec

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

Type: AacSpec
Required: False

vbrQuality

VBR Quality Level - Only used if rateControlMode is VBR.

Type: AacVbrQuality
Required: False

AacSpec

Aac Spec

`MPEG2`
`MPEG4`

AacVbrQuality

Aac Vbr Quality

`HIGH`
`LOW`

MEDIUM_HIGH
MEDIUM_LOW

Ac3BitstreamMode

Ac3 Bitstream Mode

COMMENTARY
COMPLETE_MAIN
DIALOGUE
EMERGENCY
HEARING_IMPAIRED
MUSIC_AND_EFFECTS
VISUALLY_IMPAIRED
VOICE_OVER

Ac3CodingMode

Ac3 Coding Mode

CODING_MODE_1_0
CODING_MODE_1_1
CODING_MODE_2_0
CODING_MODE_3_2_LFE

Ac3DrcProfile

Ac3 Drc Profile

FILM_STANDARD
NONE

Ac3LfeFilter

Ac3 Lfe Filter

DISABLED

ENABLED

Ac3MetadataControl

Ac3 Metadata Control

FOLLOW_INPUT
USE_CONFIGURED

Ac3Settings

Ac3 Settings

bitrate

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

Type: number
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: Ac3BitstreamMode
Required: False

codingMode

Dolby Digital coding mode. Determines number of channels.

Type: Ac3CodingMode
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

drcProfile

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

Type: Ac3DrcProfile

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: Ac3LfeFilter

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: Ac3MetadataControl

Required: False

AccessDenied

message

Type: string

Required: False

AfdSignaling

Afd Signaling

AUTO
FIXED
NONE

AncillarySourceSettings

Ancillary Source Settings

sourceAncillaryChannelNumber

Specifies the number (1 to 4) of the captions channel you want to extract from the ancillary captions. If you plan to convert the ancillary captions to another format, complete this field. If you plan to choose Embedded as the captions destination in the output (to pass through all the channels in the ancillary captions), leave this field blank because MediaLive ignores the field.

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

ArchiveCdnSettings

Archive Cdn Settings

archiveS3Settings

Type: ArchiveS3Settings
Required: False

ArchiveContainerSettings

Archive Container Settings

m2tsSettings

Type: M2tsSettings

Required: False

rawSettings

Type: RawSettings
Required: False

ArchiveGroupSettings

Archive Group Settings

archiveCdnSettings

Parameters that control interactions with the CDN.

Type: ArchiveCdnSettings
Required: False

destination

A directory and base filename where archive files should be written.

Type: OutputLocationRef
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

Archive Output Settings

containerSettings

Settings specific to the container type of the file.

Type: ArchiveContainerSettings
Required: True

extension

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

Type: string
Required: False

nameModifier

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

Type: string
Required: False

ArchiveS3LogUploads

Archive S3 Log Uploads

```
DISABLED
ENABLED
```

ArchiveS3Settings

Archive S3 Settings

cannedAcl

Specify the canned ACL to apply to each S3 request. Defaults to none.

Type: S3CannedAcl
Required: False

logUploads

When set to enabled, each upload to CDN or server will be logged.

Type: ArchiveS3LogUploads
Required: False

AribDestinationSettings

Arib Destination Settings

AribSourceSettings

Arib Source Settings

AudioChannelMapping

Audio Channel Mapping

inputChannelLevels

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

Type: Array of type InputChannelLevel
Required: True

outputChannel

The index of the output channel being produced.

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

AudioCodecSettings

Audio Codec Settings

aacSettings

Type: AacSettings

Required: False

ac3Settings

Type: Ac3Settings
Required: False

eac3Settings

Type: Eac3Settings
Required: False

mp2Settings

Type: Mp2Settings
Required: False

passThroughSettings

Type: PassThroughSettings
Required: False

wavSettings

Type: WavSettings
Required: False

AudioDescription

Audio Description

audioNormalizationSettings

Advanced audio normalization settings.

Type: AudioNormalizationSettings
Required: False

audioSelectorName

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

Type: string
Required: True

audioType

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

Type: AudioType
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: AudioDescriptionAudioTypeControl
Required: False

audioWatermarkingSettings

Settings to configure one or more solutions that insert audio watermarks in the audio encode

Type: AudioWatermarkSettings
Required: False

codecSettings

Audio codec settings.

Type: AudioCodecSettings
Required: False

languageCode

RFC 5646 language code representing 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
MinLength: 1
MaxLength: 35

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: AudioDescriptionLanguageCodeControl
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

remixSettings

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

Type: RemixSettings
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

AudioDescriptionAudioTypeControl

Audio Description Audio Type Control

`FOLLOW_INPUT`

`USE_CONFIGURED`

AudioDescriptionLanguageCodeControl

Audio Description Language Code Control

`FOLLOW_INPUT`

`USE_CONFIGURED`

AudioHlsRenditionSelection

Audio Hls Rendition Selection

groupId

Specifies the GROUP-ID in the #EXT-X-MEDIA tag of the target HLS audio rendition.

Type: string

Required: True

MinLength: 1

name

Specifies the NAME in the #EXT-X-MEDIA tag of the target HLS audio rendition.

Type: string

Required: True

MinLength: 1

AudioLanguageSelection

Audio Language Selection

languageCode

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

Type: string
Required: True

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: AudioLanguageSelectionPolicy
Required: False

AudioLanguageSelectionPolicy

Audio Language Selection Policy

```
LOOSE
STRICT
```

AudioNormalizationAlgorithm

Audio Normalization Algorithm

```
ITU_1770_1
ITU_1770_2
```

AudioNormalizationAlgorithmControl

Audio Normalization Algorithm Control

CORRECT_AUDIO

AudioNormalizationSettings

Audio Normalization Settings

algorithm

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

Type: AudioNormalizationAlgorithm
Required: False

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: AudioNormalizationAlgorithmControl
Required: False

targetLkfs

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

Type: number
Required: False
Minimum: -59
Maximum: 0

AudioOnlyHlsSegmentType

Audio Only Hls Segment Type

AAC
FMP4

AudioOnlyHlsSettings

Audio Only Hls Settings

audioGroupId

Specifies the group to which the audio Rendition belongs.

Type: string
Required: False

audioOnlyImage

Optional. Specifies the .jpg or .png image to use as the cover art for an audio-only output. We recommend a low bit-size file because the image increases the output audio bandwidth. The image is attached to the audio as an ID3 tag, frame type APIC, picture type 0x10, as per the "ID3 tag version 2.4.0 - Native Frames" standard.

Type: InputLocation
Required: False

audioTrackType

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

Type: AudioOnlyHlsTrackType
Required: False

segmentType

Specifies the segment type.

Type: AudioOnlyHlsSegmentType
Required: False

AudioOnlyHlsTrackType

Audio Only Hls Track Type

`ALTERNATE_AUDIO_AUTO_SELECT`
`ALTERNATE_AUDIO_AUTO_SELECT_DEFAULT`
`ALTERNATE_AUDIO_NOT_AUTO_SELECT`
`AUDIO_ONLY_VARIANT_STREAM`

AudioPidSelection

Audio Pid Selection

pid

Selects a specific PID from within a source.

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

AudioSelector

Audio Selector

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
MinLength: 1

selectorSettings

The audio selector settings.

Type: AudioSelectorSettings
Required: False

AudioSelectorSettings

Audio Selector Settings

audioHlsRenditionSelection

Type: AudioHlsRenditionSelection
Required: False

audioLanguageSelection

Type: AudioLanguageSelection
Required: False

audioPidSelection

Type: AudioPidSelection
Required: False

audioTrackSelection

Type: AudioTrackSelection
Required: False

AudioSilenceFailoverSettings

audioSelectorName

The name of the audio selector in the input that MediaLive should monitor to detect silence. Select your most important rendition. If you didn't create an audio selector in this input, leave blank.

Type: string
Required: True

audioSilenceThresholdMsec

The amount of time (in milliseconds) that the active input must be silent before automatic input failover occurs. Silence is defined as audio loss or audio quieter than -50 dBFS.

Type: integer
Required: False
Minimum: 1000

AudioTrack

Audio Track

track

1-based integer value that maps to a specific audio track

Type: integer
Required: True
Minimum: 1

AudioTrackSelection

Audio Track Selection

tracks

Selects one or more unique audio tracks from within a source.

Type: Array of type AudioTrack
Required: True

AudioType

Audio Type

```
CLEAN_EFFECTS
HEARING_IMPAIRED
UNDEFINED
VISUAL_IMPAIRED_COMMENTARY
```

AudioWatermarkSettings

Audio Watermark Settings

nielsenWatermarksSettings

Settings to configure Nielsen Watermarks in the audio encode

Type: NielsenWatermarksSettings
Required: False

AuthenticationScheme

Authentication Scheme

```
AKAMAI
COMMON
```

AutomaticInputFailoverSettings

The settings for Automatic Input Failover.

errorClearTimeMsec

This clear time defines the requirement a recovered input must meet to be considered healthy. The input must have no failover conditions for this length of time. Enter a time in milliseconds. This value is particularly important if the input_preference for the failover pair is set to PRIMARY_INPUT_PREFERRED, because after this time, MediaLive will switch back to the primary input.

Type: integer
Required: False
Minimum: 1

failoverConditions

A list of failover conditions. If any of these conditions occur, MediaLive will perform a failover to the other input.

Type: Array of type FailoverCondition
Required: False

inputPreference

Input preference when deciding which input to make active when a previously failed input has recovered.

Type: InputPreference
Required: False

secondaryInputId

The input ID of the secondary input in the automatic input failover pair.

Type: string
Required: True

AvailBlanking

Avail Blanking

availBlankingImage

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

Type: InputLocation
Required: False

state

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

Type: AvailBlankingState

Required: False

AvailBlankingState

Avail Blanking State

DISABLED
ENABLED

AvailConfiguration

Avail Configuration

availSettings

Ad avail settings.

Type: AvailSettings
Required: False

AvailSettings

Avail Settings

scte35SpliceInsert

Type: Scte35SpliceInsert
Required: False

scte35TimeSignalApos

Type: Scte35TimeSignalApos
Required: False

BadGatewayException

message

Type: string

Required: False

BlackoutSlate

Blackout Slate

blackoutSlateImage

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

Type: InputLocation
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: BlackoutSlateNetworkEndBlackout
Required: False

networkEndBlackoutImage

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

Type: InputLocation
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
MinLength: 34
MaxLength: 34

state

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

Type: BlackoutSlateState
Required: False

BlackoutSlateNetworkEndBlackout

Blackout Slate Network End Blackout

DISABLED
ENABLED

BlackoutSlateState

Blackout Slate State

DISABLED
ENABLED

BurnInAlignment

Burn In Alignment

CENTERED
LEFT
SMART

BurnInBackgroundColor

Burn In Background Color

BLACK

NONE

WHITE

BurnInDestinationSettings

Burn In Destination Settings

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: BurnInAlignment
Required: False

backgroundColor

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

Type: BurnInBackgroundColor
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

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
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: BurnInFontColor
Required: False

fontOpacity

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

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

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

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

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: BurnInOutlineColor
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

shadowColor

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

Type: BurnInShadowColor
Required: False

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

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

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

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: BurnInTeletextGridControl
Required: False

xPosition

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

explicit 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

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

BurnInFontColor

Burn In Font Color

BLACK
BLUE
GREEN
RED
WHITE
YELLOW

BurnInOutlineColor

Burn In Outline Color

BLACK
BLUE
GREEN

RED

WHITE

YELLOW

BurnInShadowColor

Burn In Shadow Color

BLACK

NONE

WHITE

BurnInTeletextGridControl

Burn In Teletext Grid Control

FIXED

SCALED

CaptionDescription

Caption Description

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

destinationSettings

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

Type: CaptionDestinationSettings
Required: False

languageCode

ISO 639-2 three-digit code: http://www.loc.gov/standards/iso639-2/

Type: string
Required: False

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

CaptionDestinationSettings

Caption Destination Settings

aribDestinationSettings

Type: AribDestinationSettings
Required: False

burnInDestinationSettings

Type: BurnInDestinationSettings
Required: False

dvbSubDestinationSettings

Type: DvbSubDestinationSettings

Required: False

ebuTtDDestinationSettings

Type: EbuTtDDestinationSettings
Required: False

embeddedDestinationSettings

Type: EmbeddedDestinationSettings
Required: False

embeddedPlusScte20DestinationSettings

Type: EmbeddedPlusScte20DestinationSettings
Required: False

rtmpCaptionInfoDestinationSettings

Type: RtmpCaptionInfoDestinationSettings
Required: False

scte20PlusEmbeddedDestinationSettings

Type: Scte20PlusEmbeddedDestinationSettings
Required: False

scte27DestinationSettings

Type: Scte27DestinationSettings
Required: False

smpteTtDestinationSettings

Type: SmpteTtDestinationSettings
Required: False

teletextDestinationSettings

Type: TeletextDestinationSettings
Required: False

ttmlDestinationSettings

Type: TtmlDestinationSettings
Required: False

webvttDestinationSettings

Type: WebvttDestinationSettings
Required: False

CaptionLanguageMapping

Maps a caption channel to an ISO 693-2 language code (http://www.loc.gov/standards/iso639-2), with an optional description.

captionChannel

The closed caption channel being described by this CaptionLanguageMapping. 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
MinLength: 3
MaxLength: 3

languageDescription

Textual description of language

Type: string
Required: True
MinLength: 1

CaptionRectangle

Caption Rectangle

height

See the description in leftOffset. For height, specify the entire height of the rectangle as a percentage of the underlying frame height. For example, \"80\" means the rectangle height is 80% of the underlying frame height. The topOffset and rectangleHeight must add up to 100% or less. This field corresponds to tts:extent - Y in the TTML standard.

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

leftOffset

Applies only if you plan to convert these source captions to EBU-TT-D or TTML in an output. (Make sure to leave the default if you don't have either of these formats in the output.) You can define a display rectangle for the captions that is smaller than the underlying video frame. You define the rectangle by specifying the position of the left edge, top edge, bottom edge, and right edge of the rectangle, all within the underlying video frame. The units for the measurements are percentages. If you specify a value for one of these fields, you must specify a value for all of them. For leftOffset, specify the position of the left edge of the rectangle, as a percentage of the underlying frame width, and relative to the left edge of the frame. For example, \"10\" means the measurement is 10% of the underlying frame width. The rectangle left edge starts at that position from the left edge of the frame. This field corresponds to tts:origin - X in the TTML standard.

Type: number

Required: True
Minimum: 0
Maximum: 100

topOffset

See the description in leftOffset. For topOffset, specify the position of the top edge of the rectangle, as a percentage of the underlying frame height, and relative to the top edge of the frame. For example, \"10\" means the measurement is 10% of the underlying frame height. The rectangle top edge starts at that position from the top edge of the frame. This field corresponds to tts:origin - Y in the TTML standard.

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

width

See the description in leftOffset. For width, specify the entire width of the rectangle as a percentage of the underlying frame width. For example, \"80\" means the rectangle width is 80% of the underlying frame width. The leftOffset and rectangleWidth must add up to 100% or less. This field corresponds to tts:extent - X in the TTML standard.

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

CaptionSelector

Output groups for this Live Event. Output groups contain information about where streams should be distributed.

languageCode

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

Type: string
Required: False

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
MinLength: 1

selectorSettings

Caption selector settings.

Type: CaptionSelectorSettings
Required: False

CaptionSelectorSettings

Caption Selector Settings

ancillarySourceSettings

Type: AncillarySourceSettings
Required: False

aribSourceSettings

Type: AribSourceSettings
Required: False

dvbSubSourceSettings

Type: DvbSubSourceSettings
Required: False

embeddedSourceSettings

Type: EmbeddedSourceSettings
Required: False

scte20SourceSettings

Type: Scte20SourceSettings
Required: False

scte27SourceSettings

Type: Scte27SourceSettings
Required: False

teletextSourceSettings

Type: TeletextSourceSettings
Required: False

CdiInputResolution

Maximum CDI input resolution; SD is 480i and 576i up to 30 frames-per-second (fps), HD is 720p up to 60 fps / 1080i up to 30 fps, FHD is 1080p up to 60 fps, UHD is 2160p up to 60 fps

SD
HD
FHD
UHD

CdiInputSpecification

resolution

Maximum CDI input resolution

Type: CdiInputResolution
Required: False

Channel

arn

The unique arn of the channel.

Type: string
Required: False

cdiInputSpecification

Specification of CDI inputs for this channel

Type: CdiInputSpecification
Required: False

channelClass

The class for this channel. STANDARD for a channel with two pipelines or SINGLE_PIPELINE for a channel with one pipeline.

Type: ChannelClass
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
Required: False

egressEndpoints

The endpoints where outgoing connections initiate from

Type: Array of type ChannelEgressEndpoint
Required: False

encoderSettings

Type: EncoderSettings
Required: False

id

The unique ID of the channel.

Type: string
Required: False

inputAttachments

List of input attachments for channel.

Type: Array of type InputAttachment
Required: False

inputSpecification

Specification of network and file inputs for this channel

Type: InputSpecification
Required: False

logLevel

The log level being written to CloudWatch Logs.

Type: LogLevel
Required: False

maintenance

Maintenance settings for this channel.

Type: MaintenanceStatus
Required: False

name

The name of the channel. (user-mutable)

Type: string
Required: False

pipelineDetails

Runtime details for the pipelines of a running channel.

Type: Array of type PipelineDetail
Required: False

pipelinesRunningCount

The number of currently healthy pipelines.

Type: integer
Required: False

roleArn

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

Type: string
Required: False

state

Type: ChannelState
Required: False

tags

A collection of key-value pairs.

Type: Tags

Required: False

vpc

Settings for VPC output

Type: VpcOutputSettingsDescription
Required: False

ChannelClass

A standard channel has two encoding pipelines and a single pipeline channel only has one.

```
STANDARD
SINGLE_PIPELINE
```

ChannelConfigurationValidationError

message

Type: string
Required: False

validationErrors

A collection of validation error responses.

Type: Array of type ValidationError
Required: False

ChannelEgressEndpoint

sourceIp

Public IP of where a channel's output comes from

Type: string
Required: False

ChannelState

```
CREATING
CREATE_FAILED
IDLE
STARTING
RUNNING
RECOVERING
STOPPING
DELETING
DELETED
UPDATING
UPDATE_FAILED
```

ColorSpacePassthroughSettings

Passthrough applies no color space conversion to the output

DvbNitSettings

DVB Network Information Table (NIT)

networkId

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

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

networkName

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

Type: string
Required: True
MinLength: 1

MaxLength: 256

repInterval

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

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

DvbSdtOutputSdt

Dvb Sdt Output Sdt

```
SDT_FOLLOW
SDT_FOLLOW_IF_PRESENT
SDT_MANUAL
SDT_NONE
```

DvbSdtSettings

DVB Service Description Table (SDT)

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: DvbSdtOutputSdt
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

serviceName

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

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

serviceProviderName

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

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

DvbSubDestinationAlignment

Dvb Sub Destination Alignment

CENTERED
LEFT
SMART

DvbSubDestinationBackgroundColor

Dvb Sub Destination Background Color

BLACK

NONE
WHITE

DvbSubDestinationFontColor

Dvb Sub Destination Font Color

BLACK
BLUE
GREEN
RED
WHITE
YELLOW

DvbSubDestinationOutlineColor

Dvb Sub Destination Outline Color

BLACK
BLUE
GREEN
RED
WHITE
YELLOW

DvbSubDestinationSettings

Dvb Sub Destination Settings

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: DvbSubDestinationAlignment
Required: False

backgroundColor

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

Type: DvbSubDestinationBackgroundColor
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

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
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: DvbSubDestinationFontColor
Required: False

fontOpacity

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

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

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

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

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: DvbSubDestinationOutlineColor

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

shadowColor

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

Type: DvbSubDestinationShadowColor
Required: False

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

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

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

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: DvbSubDestinationTeletextGridControl
Required: False

xPosition

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

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

DvbSubDestinationShadowColor

Dvb Sub Destination Shadow Color

BLACK
NONE
WHITE

DvbSubDestinationTeletextGridControl

Dvb Sub Destination Teletext Grid Control

FIXED
SCALED

DvbSubOcrLanguage

Dvb Sub Ocr Language

DEU
ENG
FRA
NLD
POR
SPA

DvbSubSourceSettings

Dvb Sub Source Settings

ocrLanguage

If you will configure a WebVTT caption description that references this caption selector, use this field to provide the language to consider when translating the image-based source to text.

Type: DvbSubOcrLanguage
Required: False

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

DVB Time and Date Table (SDT)

repInterval

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

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

Eac3AttenuationControl

Eac3 Attenuation Control

```
ATTENUATE_3_DB
NONE
```

Eac3BitstreamMode

Eac3 Bitstream Mode

```
COMMENTARY
COMPLETE_MAIN
```

EMERGENCY
HEARING_IMPAIRED
VISUALLY_IMPAIRED

Eac3CodingMode

Eac3 Coding Mode

CODING_MODE_1_0
CODING_MODE_2_0
CODING_MODE_3_2

Eac3DcFilter

Eac3 Dc Filter

DISABLED
ENABLED

Eac3DrcLine

Eac3 Drc Line

FILM_LIGHT
FILM_STANDARD
MUSIC_LIGHT
MUSIC_STANDARD
NONE
SPEECH

Eac3DrcRf

Eac3 Drc Rf

FILM_LIGHT
FILM_STANDARD
MUSIC_LIGHT

MUSIC_STANDARD
NONE
SPEECH

Eac3LfeControl

Eac3 Lfe Control

LFE
NO_LFE

Eac3LfeFilter

Eac3 Lfe Filter

DISABLED
ENABLED

Eac3MetadataControl

Eac3 Metadata Control

FOLLOW_INPUT
USE_CONFIGURED

Eac3PassthroughControl

Eac3 Passthrough Control

NO_PASSTHROUGH
WHEN_POSSIBLE

Eac3PhaseControl

Eac3 Phase Control

NO_SHIFT
SHIFT_90_DEGREES

Eac3Settings

Eac3 Settings

attenuationControl

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

Type: Eac3AttenuationControl
Required: False

bitrate

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

Type: number
Required: False

bitstreamMode

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

Type: Eac3BitstreamMode
Required: False

codingMode

Dolby Digital Plus coding mode. Determines number of channels.

Type: Eac3CodingMode
Required: False

dcFilter

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

Type: Eac3DcFilter

Required: False

dialnorm

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

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

drcLine

Sets the Dolby dynamic range compression profile.

Type: Eac3DrcLine
Required: False

drcRf

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

Type: Eac3DrcRf
Required: False

lfeControl

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

Type: Eac3LfeControl
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: Eac3LfeFilter
Required: False

loRoCenterMixLevel

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

Type: number
Required: False

loRoSurroundMixLevel

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

Type: number
Required: False

ltRtCenterMixLevel

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

Type: number
Required: False

ltRtSurroundMixLevel

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

Type: number
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: Eac3MetadataControl

Required: False

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: Eac3PassthroughControl
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: Eac3PhaseControl
Required: False

stereoDownmix

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

Type: Eac3StereoDownmix
Required: False

surroundExMode

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

Type: Eac3SurroundExMode
Required: False

surroundMode

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

Type: Eac3SurroundMode
Required: False

Eac3StereoDownmix

Eac3 Stereo Downmix

DPL2
LO_RO
LT_RT
NOT_INDICATED

Eac3SurroundExMode

Eac3 Surround Ex Mode

DISABLED
ENABLED
NOT_INDICATED

Eac3SurroundMode

Eac3 Surround Mode

DISABLED
ENABLED
NOT_INDICATED

EbuTtDDestinationSettings

Ebu Tt DDestination Settings

copyrightHolder

Applies only if you plan to convert these source captions to EBU-TT-D or TTML in an output. Complete this field if you want to include the name of the copyright holder in the copyright metadata tag in the TTML

Type: string

Required: False

MaxLength: 1000

fillLineGap

Specifies how to handle the gap between the lines (in multi-line captions). - enabled: Fill with the captions background color (as specified in the input captions). - disabled: Leave the gap unfilled.

Type: EbuTtDFillLineGapControl

Required: False

fontFamily

Specifies the font family to include in the font data attached to the EBU-TT captions. Valid only if styleControl is set to include. If you leave this field empty, the font family is set to "monospaced". (If styleControl is set to exclude, the font family is always set to "monospaced".) You specify only the font family. All other style information (color, bold, position and so on) is copied from the input captions. The size is always set to 100% to allow the downstream player to choose the size. - Enter a list of font families, as a comma-separated list of font names, in order of preference. The name can be a font family (such as “Arial”), or a generic font family (such as “serif”), or “default” (to let the downstream player choose the font). - Leave blank to set the family to “monospace”.

Type: string

Required: False

styleControl

Specifies the style information (font color, font position, and so on) to include in the font data that is attached to the EBU-TT captions. - include: Take the style information (font color, font position, and so on) from the source captions and include that information in the font data attached to the EBU-TT captions. This option is valid only if the source captions are Embedded or Teletext. - exclude: In the font data attached to the EBU-TT captions, set the font family to "monospaced". Do not include any other style information.

Type: EbuTtDDestinationStyleControl

Required: False

EbuTtDDestinationStyleControl

Ebu Tt DDestination Style Control

EXCLUDE
INCLUDE

EbuTtDFillLineGapControl

Ebu Tt DFill Line Gap Control

DISABLED
ENABLED

EmbeddedConvert608To708

Embedded Convert608 To708

DISABLED
UPCONVERT

EmbeddedDestinationSettings

Embedded Destination Settings

EmbeddedPlusScte20DestinationSettings

Embedded Plus Scte20 Destination Settings

EmbeddedScte20Detection

Embedded Scte20 Detection

AUTO
OFF

EmbeddedSourceSettings

Embedded Source Settings

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: EmbeddedConvert608To708
Required: False

scte20Detection

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

Type: EmbeddedScte20Detection
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

source608TrackNumber

This field is unused and deprecated.

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

EncoderSettings

Encoder Settings

audioDescriptions

Type: Array of type AudioDescription
Required: True

availBlanking

Settings for ad avail blanking.

Type: AvailBlanking
Required: False

availConfiguration

Event-wide configuration settings for ad avail insertion.

Type: AvailConfiguration
Required: False

blackoutSlate

Settings for blackout slate.

Type: BlackoutSlate
Required: False

captionDescriptions

Settings for caption decriptions

Type: Array of type CaptionDescription
Required: False

featureActivations

Feature Activations

Type: FeatureActivations

Required: False

globalConfiguration

Configuration settings that apply to the event as a whole.

Type: GlobalConfiguration
Required: False

motionGraphicsConfiguration

Settings for motion graphics.

Type: MotionGraphicsConfiguration
Required: False

nielsenConfiguration

Nielsen configuration settings.

Type: NielsenConfiguration
Required: False

outputGroups

Type: Array of type OutputGroup
Required: True

timecodeConfig

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

Type: TimecodeConfig
Required: True

videoDescriptions

Type: Array of type VideoDescription

Required: True

FailoverCondition

Failover Condition settings. There can be multiple failover conditions inside AutomaticInputFailoverSettings.

failoverConditionSettings

Failover condition type-specific settings.

Type: FailoverConditionSettings
Required: False

FailoverConditionSettings

Settings for one failover condition.

audioSilenceSettings

MediaLive will perform a failover if the specified audio selector is silent for the specified period.

Type: AudioSilenceFailoverSettings
Required: False

inputLossSettings

MediaLive will perform a failover if content is not detected in this input for the specified period.

Type: InputLossFailoverSettings
Required: False

videoBlackSettings

MediaLive will perform a failover if content is considered black for the specified period.

Type: VideoBlackFailoverSettings
Required: False

FeatureActivations

Feature Activations

inputPrepareScheduleActions

Enables the Input Prepare feature. You can create Input Prepare actions in the schedule only if this feature is enabled. If you disable the feature on an existing schedule, make sure that you first delete all input prepare actions from the schedule.

Type: FeatureActivationsInputPrepareScheduleActions
Required: False

FeatureActivationsInputPrepareScheduleActions

Feature Activations Input Prepare Schedule Actions

```
DISABLED
ENABLED
```

FecOutputIncludeFec

Fec Output Include Fec

```
COLUMN
COLUMN_AND_ROW
```

FecOutputSettings

Fec Output Settings

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: FecOutputIncludeFec
Required: False

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

FixedAfd

Fixed Afd

AFD_0000
AFD_0010
AFD_0011
AFD_0100
AFD_1000
AFD_1001
AFD_1010
AFD_1011
AFD_1101
AFD_1110

AFD_1111

Fmp4HlsSettings

Fmp4 Hls Settings

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

nielsenId3Behavior

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

Type: Fmp4NielsenId3Behavior
Required: False

timedMetadataBehavior

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

Type: Fmp4TimedMetadataBehavior
Required: False

Fmp4NielsenId3Behavior

Fmp4 Nielsen Id3 Behavior

NO_PASSTHROUGH
PASSTHROUGH

Fmp4TimedMetadataBehavior

Fmp4 Timed Metadata Behavior

NO_PASSTHROUGH
PASSTHROUGH

FrameCaptureCdnSettings

Frame Capture Cdn Settings

frameCaptureS3Settings

Type: FrameCaptureS3Settings
Required: False

FrameCaptureGroupSettings

Frame Capture Group Settings

destination

The destination for the frame capture files. Either the URI for an Amazon S3 bucket and object, plus a file name prefix (for example, s3ssl://sportsDelivery/highlights/20180820/curling-) or the URI for a MediaStore container, plus a file name prefix (for example, mediastoressl://sportsDelivery/20180820/curling-). The final file names consist of the prefix from the destination field (for example, "curling-") + name modifier + the counter (5 digits, starting from 00001) + extension (which is always .jpg). For example, curling-low.00001.jpg

Type: OutputLocationRef
Required: True

frameCaptureCdnSettings

Parameters that control interactions with the CDN.

Type: FrameCaptureCdnSettings
Required: False

FrameCaptureHlsSettings

Frame Capture Hls Settings

FrameCaptureIntervalUnit

Frame Capture Interval Unit

```
MILLISECONDS
SECONDS
```

FrameCaptureOutputSettings

Frame Capture Output Settings

nameModifier

Required if the output group contains more than one output. This modifier forms part of the output file name.

Type: string
Required: False

FrameCaptureS3LogUploads

Frame Capture S3 Log Uploads

```
DISABLED
ENABLED
```

FrameCaptureS3Settings

Frame Capture S3 Settings

cannedAcl

Specify the canned ACL to apply to each S3 request. Defaults to none.

Type: S3CannedAcl
Required: False

logUploads

When set to enabled, each upload to CDN or server will be logged.

Type: FrameCaptureS3LogUploads
Required: False

FrameCaptureSettings

Frame Capture Settings

captureInterval

The frequency at which to capture frames for inclusion in the output. May be specified in either seconds or milliseconds, as specified by captureIntervalUnits.

Type: integer
Required: False
Minimum: 1
Maximum: 3600000

captureIntervalUnits

Unit for the frame capture interval.

Type: FrameCaptureIntervalUnit
Required: False

GatewayTimeoutException

message

Type: string
Required: False

GlobalConfiguration

Global Configuration

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 the current input completes (e.g. end-of-file). When switchAndLoopInputs is configured the encoder will restart at the beginning of the first input. When "none" is configured the encoder will transcode either black, a solid color, or a user specified slate images per the "Input Loss Behavior" configuration until the next input switch occurs (which is controlled through the Channel Schedule API).

Type: GlobalConfigurationInputEndAction
Required: False

inputLossBehavior

Settings for system actions when input is lost.

Type: InputLossBehavior
Required: False

outputLockingMode

Indicates how MediaLive pipelines are synchronized. PIPELINE_LOCKING - MediaLive will attempt to synchronize the output of each pipeline to the other. EPOCH_LOCKING - MediaLive will attempt to synchronize the output of each pipeline to the Unix epoch.

Type: GlobalConfigurationOutputLockingMode
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: GlobalConfigurationOutputTimingSource
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: GlobalConfigurationLowFramerateInputs
Required: False

GlobalConfigurationInputEndAction

Global Configuration Input End Action

```
NONE
SWITCH_AND_LOOP_INPUTS
```

GlobalConfigurationLowFramerateInputs

Global Configuration Low Framerate Inputs

```
DISABLED
ENABLED
```

GlobalConfigurationOutputLockingMode

Global Configuration Output Locking Mode

```
EPOCH_LOCKING
PIPELINE_LOCKING
```

GlobalConfigurationOutputTimingSource

Global Configuration Output Timing Source

```
INPUT_CLOCK
```

SYSTEM_CLOCK

H264AdaptiveQuantization

H264 Adaptive Quantization

AUTO
HIGH
HIGHER
LOW
MAX
MEDIUM
OFF

H264ColorMetadata

H264 Color Metadata

IGNORE
INSERT

H264ColorSpaceSettings

H264 Color Space Settings

colorSpacePassthroughSettings

Type: ColorSpacePassthroughSettings
Required: False

rec601Settings

Type: Rec601Settings
Required: False

rec709Settings

Type: Rec709Settings

Required: False

H264EntropyEncoding

H264 Entropy Encoding

CABAC

CAVLC

H264FilterSettings

H264 Filter Settings

temporalFilterSettings

Type: TemporalFilterSettings

Required: False

H264FlickerAq

H264 Flicker Aq

DISABLED

ENABLED

H264ForceFieldPictures

H264 Force Field Pictures

DISABLED

ENABLED

H264FramerateControl

H264 Framerate Control

INITIALIZE_FROM_SOURCE

SPECIFIED

H264GopBReference

H264 Gop BReference

DISABLED
ENABLED

H264GopSizeUnits

H264 Gop Size Units

FRAMES
SECONDS

H264Level

H264 Level

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

H264 Look Ahead Rate Control

HIGH
LOW
MEDIUM

H264ParControl

H264 Par Control

INITIALIZE_FROM_SOURCE
SPECIFIED

H264Profile

H264 Profile

BASELINE
HIGH
HIGH_10BIT
HIGH_422
HIGH_422_10BIT
MAIN

H264QualityLevel

H264 Quality Level

ENHANCED_QUALITY
STANDARD_QUALITY

H264RateControlMode

H264 Rate Control Mode

CBR

MULTIPLEX

QVBR

VBR

H264ScanType

H264 Scan Type

INTERLACED

PROGRESSIVE

H264SceneChangeDetect

H264 Scene Change Detect

DISABLED

ENABLED

H264Settings

H264 Settings

adaptiveQuantization

Enables or disables adaptive quantization, which is a technique MediaLive can apply to video on a frame-by-frame basis to produce more compression without losing quality. There are three types of adaptive quantization: flicker, spatial, and temporal. Set the field in one of these ways: Set to Auto. Recommended. For each type of AQ, MediaLive will determine if AQ is needed, and if so, the appropriate strength. Set a strength (a value other than Auto or Disable). This strength will apply to any of the AQ fields that you choose to enable. Set to Disabled to disable all types of adaptive quantization.

Type: H264AdaptiveQuantization

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: AfdSignaling
Required: False

bitrate

Average bitrate in bits/second. Required when the rate control mode is VBR or CBR. Not used for QVBR. In an MS Smooth output group, each output must have a unique value when its bitrate is 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

bufSize

Size of buffer (HRD buffer model) in bits.

Type: integer
Required: False
Minimum: 0

colorMetadata

Includes colorspace metadata in the output.

Type: H264ColorMetadata
Required: False

colorSpaceSettings

Color Space settings

Type: H264ColorSpaceSettings
Required: False

entropyEncoding

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

Type: H264EntropyEncoding
Required: False

filterSettings

Optional filters that you can apply to an encode.

Type: H264FilterSettings
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: FixedAfd
Required: False

flickerAq

Flicker AQ makes adjustments within each frame to reduce flicker or 'pop' on I-frames. The value to enter in this field depends on the value in the Adaptive quantization field: If you have set the Adaptive quantization field to Auto, MediaLive ignores any value in this field. MediaLive will determine if flicker AQ is appropriate and will apply the appropriate strength. If you have set the

Adaptive quantization field to a strength, you can set this field to Enabled or Disabled. Enabled: MediaLive will apply flicker AQ using the specified strength. Disabled: MediaLive won't apply flicker AQ. If you have set the Adaptive quantization to Disabled, MediaLive ignores any value in this field and doesn't apply flicker AQ.

Type: H264FlickerAq
Required: False

forceFieldPictures

This setting applies only when scan type is "interlaced." It controls whether coding is performed on a field basis or on a frame basis. (When the video is progressive, the coding is always performed on a frame basis.) enabled: Force MediaLive to code on a field basis, so that odd and even sets of fields are coded separately. disabled: Code the two sets of fields separately (on a field basis) or together (on a frame basis using PAFF), depending on what is most appropriate for the content.

Type: H264ForceFieldPictures
Required: False

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: H264FramerateControl
Required: False

framerateDenominator

Framerate denominator.

Type: integer
Required: False
Minimum: 1

framerateNumerator

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

Type: integer
Required: False
Minimum: 1

gopBReference

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

Type: H264GopBReference
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

gopNumBFrames

Number of B-frames between reference frames.

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

gopSize

GOP size (keyframe interval) in units of either frames or seconds per gopSizeUnits. If gopSizeUnits is frames, gopSize must be an integer and must be greater than or equal to 1. If gopSizeUnits is seconds, gopSize must be greater than 0, but need not be an integer.

Type: number
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: H264GopSizeUnits
Required: False

level

H.264 Level.

Type: H264Level
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: H264LookAheadRateControl
Required: False

maxBitrate

For QVBR: See the tooltip for Quality level For VBR: Set the maximum bitrate in order to accommodate expected spikes in the complexity of the video.

Type: integer
Required: False
Minimum: 1000

minIInterval

Only meaningful if sceneChangeDetect is set to enabled. Defaults to 5 if multiplex rate control is used. 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

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

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 pixsel aspect ratio will be set equal to the input video pixel aspect ratio of the first input.

Type: H264ParControl
Required: False

parDenominator

Pixel Aspect Ratio denominator.

Type: integer

Required: False
Minimum: 1

parNumerator

Pixel Aspect Ratio numerator.

Type: integer
Required: False
Minimum: 1

profile

H.264 Profile.

Type: H264Profile
Required: False

qualityLevel

Leave as STANDARD_QUALITY or choose a different value (which might result in additional costs to run the channel). - ENHANCED_QUALITY: Produces a slightly better video quality without an increase in the bitrate. Has an effect only when the Rate control mode is QVBR or CBR. If this channel is in a MediaLive multiplex, the value must be ENHANCED_QUALITY. - STANDARD_QUALITY: Valid for any Rate control mode.

Type: H264QualityLevel
Required: False

qvbrQualityLevel

Controls the target quality for the video encode. Applies only when the rate control mode is QVBR. You can set a target quality or you can let MediaLive determine the best quality. To set a target quality, enter values in the QVBR quality level field and the Max bitrate field. Enter values that suit your most important viewing devices. Recommended values are: - Primary screen: Quality level: 8 to 10. Max bitrate: 4M - PC or tablet: Quality level: 7. Max bitrate: 1.5M to 3M - Smartphone: Quality level: 6. Max bitrate: 1M to 1.5M To let MediaLive decide, leave the QVBR quality level field

empty, and in Max bitrate enter the maximum rate you want in the video. For more information, see the section called "Video - rate control mode" in the MediaLive user guide

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

rateControlMode

Rate control mode. QVBR: Quality will match the specified quality level except when it is constrained by the maximum bitrate. Recommended if you or your viewers pay for bandwidth. VBR: Quality and bitrate vary, depending on the video complexity. Recommended instead of QVBR if you want to maintain a specific average bitrate over the duration of the channel. CBR: Quality varies, depending on the video complexity. Recommended only if you distribute your assets to devices that cannot handle variable bitrates. Multiplex: This rate control mode is only supported (and is required) when the video is being delivered to a MediaLive Multiplex in which case the rate control configuration is controlled by the properties within the Multiplex Program.

Type: H264RateControlMode
Required: False

scanType

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

Type: H264ScanType
Required: False

sceneChangeDetect

Scene change detection. - On: inserts I-frames when scene change is detected. - Off: does not force an I-frame when scene change is detected.

Type: H264SceneChangeDetect
Required: False

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

softness

Softness. Selects quantizer matrix, larger values reduce high-frequency content in the encoded image. If not set to zero, must be greater than 15.

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

spatialAq

Spatial AQ makes adjustments within each frame based on spatial variation of content complexity. The value to enter in this field depends on the value in the Adaptive quantization field: If you have set the Adaptive quantization field to Auto, MediaLive ignores any value in this field. MediaLive will determine if spatial AQ is appropriate and will apply the appropriate strength. If you have set the Adaptive quantization field to a strength, you can set this field to Enabled or Disabled. Enabled: MediaLive will apply spatial AQ using the specified strength. Disabled: MediaLive won't apply spatial AQ. If you have set the Adaptive quantization to Disabled, MediaLive ignores any value in this field and doesn't apply spatial AQ.

Type: H264SpatialAq
Required: False

subgopLength

If set to fixed, use gopNumBFrames B-frames per sub-GOP. If set to dynamic, optimize the number of B-frames used for each sub-GOP to improve visual quality.

Type: H264SubGopLength
Required: False

syntax

Produces a bitstream compliant with SMPTE RP-2027.

Type: H264Syntax
Required: False

temporalAq

Temporal makes adjustments within each frame based on temporal variation of content complexity. The value to enter in this field depends on the value in the Adaptive quantization field: If you have set the Adaptive quantization field to Auto, MediaLive ignores any value in this field. MediaLive will determine if temporal AQ is appropriate and will apply the appropriate strength. If you have set the Adaptive quantization field to a strength, you can set this field to Enabled or Disabled. Enabled: MediaLive will apply temporal AQ using the specified strength. Disabled: MediaLive won't apply temporal AQ. If you have set the Adaptive quantization to Disabled, MediaLive ignores any value in this field and doesn't apply temporal AQ.

Type: H264TemporalAq
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: H264TimecodeInsertionBehavior
Required: False

H264SpatialAq

H264 Spatial Aq

DISABLED
ENABLED

H264SubGopLength

H264 Sub Gop Length

DYNAMIC
FIXED

H264Syntax

H264 Syntax

DEFAULT
RP2027

H264TemporalAq

H264 Temporal Aq

DISABLED
ENABLED

H264TimecodeInsertionBehavior

H264 Timecode Insertion Behavior

DISABLED
PIC_TIMING_SEI

H265AdaptiveQuantization

H265 Adaptive Quantization

AUTO

HIGH

HIGHER

LOW

MAX

MEDIUM

OFF

H265AlternativeTransferFunction

H265 Alternative Transfer Function

INSERT

OMIT

H265ColorMetadata

H265 Color Metadata

IGNORE

INSERT

H265ColorSpaceSettings

H265 Color Space Settings

colorSpacePassthroughSettings

Type: ColorSpacePassthroughSettings

Required: False

hdr10Settings

Type: Hdr10Settings

Required: False

rec601Settings

Type: Rec601Settings
Required: False

rec709Settings

Type: Rec709Settings
Required: False

H265FilterSettings

H265 Filter Settings

temporalFilterSettings

Type: TemporalFilterSettings
Required: False

H265FlickerAq

H265 Flicker Aq

```
DISABLED
ENABLED
```

H265GopSizeUnits

H265 Gop Size Units

```
FRAMES
SECONDS
```

H265Level

H265 Level

```
H265_LEVEL_1
```

H265_LEVEL_2
H265_LEVEL_2_1
H265_LEVEL_3
H265_LEVEL_3_1
H265_LEVEL_4
H265_LEVEL_4_1
H265_LEVEL_5
H265_LEVEL_5_1
H265_LEVEL_5_2
H265_LEVEL_6
H265_LEVEL_6_1
H265_LEVEL_6_2
H265_LEVEL_AUTO

H265LookAheadRateControl

H265 Look Ahead Rate Control

HIGH
LOW
MEDIUM

H265Profile

H265 Profile

MAIN
MAIN_10BIT

H265RateControlMode

H265 Rate Control Mode

CBR
MULTIPLEX
QVBR

H265ScanType

H265 Scan Type

```
INTERLACED
PROGRESSIVE
```

H265SceneChangeDetect

H265 Scene Change Detect

```
DISABLED
ENABLED
```

H265Settings

H265 Settings

adaptiveQuantization

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

Type: H265AdaptiveQuantization
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: AfdSignaling
Required: False

alternativeTransferFunction

Whether or not EML should insert an Alternative Transfer Function SEI message to support backwards compatibility with non-HDR decoders and displays.

Type: H265AlternativeTransferFunction
Required: False

bitrate

Average bitrate in bits/second. Required when the rate control mode is VBR or CBR. Not used for QVBR. In an MS Smooth output group, each output must have a unique value when its bitrate is rounded down to the nearest multiple of 1000.

Type: integer
Required: False
Minimum: 100000
Maximum: 40000000

bufSize

Size of buffer (HRD buffer model) in bits.

Type: integer
Required: False
Minimum: 100000
Maximum: 80000000

colorMetadata

Includes colorspace metadata in the output.

Type: H265ColorMetadata
Required: False

colorSpaceSettings

Color Space settings

Type: H265ColorSpaceSettings
Required: False

filterSettings

Optional filters that you can apply to an encode.

Type: H265FilterSettings
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: FixedAfd
Required: False

flickerAq

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

Type: H265FlickerAq
Required: False

framerateDenominator

Framerate denominator.

Type: integer
Required: True
Minimum: 1
Maximum: 3003

framerateNumerator

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

Type: integer
Required: True
Minimum: 1

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

gopSize

GOP size (keyframe interval) in units of either frames or seconds per gopSizeUnits. If gopSizeUnits is frames, gopSize must be an integer and must be greater than or equal to 1. If gopSizeUnits is seconds, gopSize must be greater than 0, but need not be an integer.

Type: number
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: H265GopSizeUnits
Required: False

level

H.265 Level.

Type: H265Level
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: H265LookAheadRateControl
Required: False

maxBitrate

For QVBR: See the tooltip for Quality level

Type: integer
Required: False
Minimum: 100000
Maximum: 40000000

minIInterval

Only meaningful if sceneChangeDetect is set to enabled. Defaults to 5 if multiplex rate control is used. 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

parDenominator

Pixel Aspect Ratio denominator.

Type: integer
Required: False
Minimum: 1

parNumerator

Pixel Aspect Ratio numerator.

Type: integer

Required: False

Minimum: 1

profile

H.265 Profile.

Type: H265Profile

Required: False

qvbrQualityLevel

Controls the target quality for the video encode. Applies only when the rate control mode is QVBR. Set values for the QVBR quality level field and Max bitrate field that suit your most important viewing devices. Recommended values are: - Primary screen: Quality level: 8 to 10. Max bitrate: 4M - PC or tablet: Quality level: 7. Max bitrate: 1.5M to 3M - Smartphone: Quality level: 6. Max bitrate: 1M to 1.5M

Type: integer

Required: False

Minimum: 1

Maximum: 10

rateControlMode

Rate control mode. QVBR: Quality will match the specified quality level except when it is constrained by the maximum bitrate. Recommended if you or your viewers pay for bandwidth. CBR: Quality varies, depending on the video complexity. Recommended only if you distribute your assets to devices that cannot handle variable bitrates. Multiplex: This rate control mode is only supported (and is required) when the video is being delivered to a MediaLive Multiplex in which case the rate control configuration is controlled by the properties within the Multiplex Program.

Type: H265RateControlMode

Required: False

scanType

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

Type: H265ScanType
Required: False

sceneChangeDetect

Scene change detection.

Type: H265SceneChangeDetect
Required: False

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: 16

tier

H.265 Tier.

Type: H265Tier
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: H265TimecodeInsertionBehavior
Required: False

H265Tier

H265 Tier

```
HIGH
MAIN
```

H265TimecodeInsertionBehavior

H265 Timecode Insertion Behavior

```
DISABLED
PIC_TIMING_SEI
```

Hdr10Settings

Hdr10 Settings

maxCll

Maximum Content Light Level An integer metadata value defining the maximum light level, in nits, of any single pixel within an encoded HDR video stream or file.

Type: integer
Required: False
Minimum: 0
Maximum: 32768

maxFall

Maximum Frame Average Light Level An integer metadata value defining the maximum average light level, in nits, for any single frame within an encoded HDR video stream or file.

Type: integer
Required: False

Minimum: 0
Maximum: 32768

HlsAdMarkers

Hls Ad Markers

ADOBE
ELEMENTAL
ELEMENTAL_SCTE35

HlsAkamaiHttpTransferMode

Hls Akamai Http Transfer Mode

CHUNKED
NON_CHUNKED

HlsAkamaiSettings

Hls Akamai Settings

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

httpTransferMode

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

Type: HlsAkamaiHttpTransferMode
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

salt

Salt for authenticated Akamai.

Type: string
Required: False

token

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

Type: string

Required: False

HlsBasicPutSettings

Hls Basic Put Settings

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

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

HlsCaptionLanguageSetting

Hls Caption Language Setting

INSERT
NONE
OMIT

HlsCdnSettings

Hls Cdn Settings

hlsAkamaiSettings

Type: HlsAkamaiSettings
Required: False

hlsBasicPutSettings

Type: HlsBasicPutSettings
Required: False

hlsMediaStoreSettings

Type: HlsMediaStoreSettings
Required: False

hlsS3Settings

Type: HlsS3Settings
Required: False

hlsWebdavSettings

Type: HlsWebdavSettings

Required: False

HlsClientCache

Hls Client Cache

DISABLED

ENABLED

HlsCodecSpecification

Hls Codec Specification

RFC_4281

RFC_6381

HlsDirectoryStructure

Hls Directory Structure

SINGLE_DIRECTORY

SUBDIRECTORY_PER_STREAM

HlsDiscontinuityTags

Hls Discontinuity Tags

INSERT

NEVER_INSERT

HlsEncryptionType

Hls Encryption Type

AES128

SAMPLE_AES

HlsGroupSettings

Hls Group Settings

adMarkers

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

Type: Array of type HlsAdMarkers
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

baseUrlContent1

Optional. One value per output group. This field is required only if you are completing Base URL content A, and the downstream system has notified you that the media files for pipeline 1 of all outputs are in a location different from the media files for pipeline 0.

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

baseUrlManifest1

Optional. One value per output group. Complete this field only if you are completing Base URL manifest A, and the downstream system has notified you that the child manifest files for pipeline 1 of all outputs are in a location different from the child manifest files for pipeline 0.

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

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: HlsCaptionLanguageSetting

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: HlsClientCache

Required: False

codecSpecification

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

Type: HlsCodecSpecification
Required: False

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
MinLength: 32
MaxLength: 32

destination

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

Type: OutputLocationRef
Required: True

directoryStructure

Place segments in subdirectories.

Type: HlsDirectoryStructure
Required: False

discontinuityTags

Specifies whether to insert EXT-X-DISCONTINUITY tags in the HLS child manifests for this output group. Typically, choose Insert because these tags are required in the manifest (according to the HLS specification) and serve an important purpose. Choose Never Insert only if the downstream

system is doing real-time failover (without using the MediaLive automatic failover feature) and only if that downstream system has advised you to exclude the tags.

Type: HlsDiscontinuityTags
Required: False

encryptionType

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

Type: HlsEncryptionType
Required: False

hlsCdnSettings

Parameters that control interactions with the CDN.

Type: HlsCdnSettings
Required: False

hlsId3SegmentTagging

State of HLS ID3 Segment Tagging

Type: HlsId3SegmentTaggingState
Required: False

iFrameOnlyPlaylists

DISABLED: Do not create an I-frame-only manifest, but do create the master and media manifests (according to the Output Selection field). STANDARD: Create an I-frame-only manifest for each output that contains video, as well as the other manifests (according to the Output Selection field). The I-frame manifest contains a #EXT-X-I-FRAMES-ONLY tag to indicate it is I-frame only, and one or more #EXT-X-BYTERANGE entries identifying the I-frame position. For example, #EXT-X-BYTERANGE:160364@1461888"

Type: IFrameOnlyPlaylistType

Required: False

incompleteSegmentBehavior

Specifies whether to include the final (incomplete) segment in the media output when the pipeline stops producing output because of a channel stop, a channel pause or a loss of input to the pipeline. Auto means that MediaLive decides whether to include the final segment, depending on the channel class and the types of output groups. Suppress means to never include the incomplete segment. We recommend you choose Auto and let MediaLive control the behavior.

Type: HlsIncompleteSegmentBehavior
Required: False

indexNSegments

Applies only if Mode field is LIVE. Specifies the maximum number of segments in the media manifest file. After this maximum, older segments are removed from the media manifest. This number must be smaller than the number in the Keep Segments field.

Type: integer
Required: False
Minimum: 3

inputLossAction

Parameter that control output group behavior on input loss.

Type: InputLossActionForHlsOut
Required: False

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: HlsIvInManifest
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: HlsIvSource
Required: False

keepSegments

Applies only if Mode field is LIVE. Specifies the number of media segments to retain in the destination directory. This number should be bigger than indexNSegments (Num segments). We recommend (value = (2 x indexNsegments) + 1). If this "keep segments" number is too low, the following might happen: the player is still reading a media manifest file that lists this segment, but that segment has been removed from the destination directory (as directed by indexNSegments). This situation would result in a 404 HTTP error on the player.

Type: integer
Required: False
Minimum: 1

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

keyFormatVersions

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

Type: string
Required: False

keyProviderSettings

The key provider settings.

Type: KeyProviderSettings
Required: False

manifestCompression

When set to gzip, compresses HLS playlist.

Type: HlsManifestCompression
Required: False

manifestDurationFormat

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

Type: HlsManifestDurationFormat
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 "vod", all segments are indexed and kept permanently in the destination and manifest. If "live", only the number segments specified in keepSegments and indexNSegments are kept; newer segments replace older segments, which may prevent players from rewinding all the way to the beginning of the event. VOD mode uses HLS EXT-X-PLAYLIST-TYPE of EVENT while the channel is running, converting it to a "VOD" type manifest on completion of the stream.

Type: HlsMode
Required: False

outputSelection

MANIFESTS_AND_SEGMENTS: Generates manifests (master manifest, if applicable, and media manifests) for this output group. VARIANT_MANIFESTS_AND_SEGMENTS: Generates media manifests for this output group, but not a master manifest. SEGMENTS_ONLY: Does not generate any manifests for this output group.

Type: HlsOutputSelection
Required: False

programDateTime

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

Type: HlsProgramDateTime
Required: False

programDateTimeClock

Specifies the algorithm used to drive the HLS EXT-X-PROGRAM-DATE-TIME clock. Options include: INITIALIZE_FROM_OUTPUT_TIMECODE: The PDT clock is initialized as a function of the first output timecode, then incremented by the EXTINF duration of each encoded segment. SYSTEM_CLOCK: The PDT clock is initialized as a function of the UTC wall clock, then incremented by the EXTINF duration of each encoded segment. If the PDT clock diverges from the wall clock by more than 500ms, it is resynchronized to the wall clock.

Type: HlsProgramDateTimeClock
Required: False

programDateTimePeriod

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

Type: integer

Required: False

Minimum: 0

Maximum: 3600

redundantManifest

ENABLED: The master manifest (.m3u8 file) for each pipeline includes information about both pipelines: first its own media files, then the media files of the other pipeline. This feature allows playout device that support stale manifest detection to switch from one manifest to the other, when the current manifest seems to be stale. There are still two destinations and two master manifests, but both master manifests reference the media files from both pipelines. DISABLED: The master manifest (.m3u8 file) for each pipeline includes information about its own pipeline only. For an HLS output group with MediaPackage as the destination, the DISABLED behavior is always followed. MediaPackage regenerates the manifests it serves to players so a redundant manifest from MediaLive is irrelevant.

Type: HlsRedundantManifest

Required: False

segmentLength

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

Type: integer

Required: False

Minimum: 1

segmentationMode

useInputSegmentation has been deprecated. The configured segment size is always used.

Type: HlsSegmentationMode

Required: False

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

streamInfResolution

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

Type: HlsStreamInfResolution
Required: False

timedMetadataId3Frame

Indicates ID3 frame that has the timecode.

Type: HlsTimedMetadataId3Frame
Required: False

timedMetadataId3Period

Timed Metadata interval in seconds.

Type: integer
Required: False
Minimum: 0

timestampDeltaMilliseconds

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

Type: integer
Required: False
Minimum: 0

tsFileMode

SEGMENTED_FILES: Emit the program as segments - multiple .ts media files. SINGLE_FILE: Applies only if Mode field is VOD. Emit the program as a single .ts media file. The media manifest includes #EXT-X-BYTERANGE tags to index segments for playback. A typical use for this value is when sending the output to AWS Elemental MediaConvert, which can accept only a single media file. Playback while the channel is running is not guaranteed due to HTTP server caching.

Type: HlsTsFileMode

Required: False

HlsH265PackagingType

Hls H265 Packaging Type

HEV1

HVC1

HlsId3SegmentTaggingState

State of HLS ID3 Segment Tagging

DISABLED

ENABLED

HlsIncompleteSegmentBehavior

Hls Incomplete Segment Behavior

AUTO

SUPPRESS

HlsInputSettings

Hls Input Settings

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

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

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

retryInterval

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

Type: integer
Required: False
Minimum: 0

scte35Source

Identifies the source for the SCTE-35 messages that MediaLive will ingest. Messages can be ingested from the content segments (in the stream) or from tags in the playlist (the HLS manifest). MediaLive ignores SCTE-35 information in the source that is not selected.

Type: HlsScte35SourceType

Required: False

HlsIvInManifest

Hls Iv In Manifest

```
EXCLUDE
INCLUDE
```

HlsIvSource

Hls Iv Source

```
EXPLICIT
FOLLOWS_SEGMENT_NUMBER
```

HlsManifestCompression

Hls Manifest Compression

```
GZIP
NONE
```

HlsManifestDurationFormat

Hls Manifest Duration Format

```
FLOATING_POINT
INTEGER
```

HlsMediaStoreSettings

Hls Media Store Settings

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

mediaStoreStorageClass

When set to temporal, output files are stored in non-persistent memory for faster reading and writing.

Type: HlsMediaStoreStorageClass
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

HlsMediaStoreStorageClass

Hls Media Store Storage Class

```
TEMPORAL
```

HlsMode

Hls Mode

```
LIVE
VOD
```

HlsOutputSelection

Hls Output Selection

```
MANIFESTS_AND_SEGMENTS
SEGMENTS_ONLY
VARIANT_MANIFESTS_AND_SEGMENTS
```

HlsOutputSettings

Hls Output Settings

h265PackagingType

Only applicable when this output is referencing an H.265 video description. Specifies whether MP4 segments should be packaged as HEV1 or HVC1.

Type: HlsH265PackagingType
Required: False

hlsSettings

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

Type: HlsSettings
Required: True

nameModifier

String concatenated to the end of the destination filename. Accepts \"Format Identifiers \":#formatIdentifierParameters.

Type: string
Required: False
MinLength: 1

segmentModifier

String concatenated to end of segment filenames.

Type: string
Required: False

HlsProgramDateTime

Hls Program Date Time

EXCLUDE
INCLUDE

HlsProgramDateTimeClock

Hls Program Date Time Clock

INITIALIZE_FROM_OUTPUT_TIMECODE

SYSTEM_CLOCK

HlsRedundantManifest

Hls Redundant Manifest

DISABLED
ENABLED

HlsS3LogUploads

Hls S3 Log Uploads

DISABLED
ENABLED

HlsS3Settings

Hls S3 Settings

cannedAcl

Specify the canned ACL to apply to each S3 request. Defaults to none.

Type: S3CannedAcl
Required: False

logUploads

When set to enabled, each fragment upload to CDN or server will be logged.

Type: HlsS3LogUploads
Required: False

HlsScte35SourceType

Hls Scte35 Source Type

MANIFEST

SEGMENTS

HlsSegmentationMode

Hls Segmentation Mode

USE_INPUT_SEGMENTATION
USE_SEGMENT_DURATION

HlsSettings

Hls Settings

audioOnlyHlsSettings

Type: AudioOnlyHlsSettings
Required: False

fmp4HlsSettings

Type: Fmp4HlsSettings
Required: False

frameCaptureHlsSettings

Type: FrameCaptureHlsSettings
Required: False

standardHlsSettings

Type: StandardHlsSettings
Required: False

HlsStreamInfResolution

Hls Stream Inf Resolution

EXCLUDE

INCLUDE

HlsTimedMetadataId3Frame

Hls Timed Metadata Id3 Frame

NONE
PRIV
TDRL

HlsTsFileMode

Hls Ts File Mode

SEGMENTED_FILES
SINGLE_FILE

HlsWebdavHttpTransferMode

Hls Webdav Http Transfer Mode

CHUNKED
NON_CHUNKED

HlsWebdavSettings

Hls Webdav Settings

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

httpTransferMode

Specify whether or not to use chunked transfer encoding to WebDAV.

Type: HlsWebdavHttpTransferMode
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

HtmlMotionGraphicsSettings

Html Motion Graphics Settings

IFrameOnlyPlaylistType

When set to "standard", an I-Frame only playlist will be written out for each video output in the output group. This I-Frame only playlist will contain byte range offsets pointing to the I-frame(s) in each segment.

DISABLED
STANDARD

InputAttachment

automaticInputFailoverSettings

User-specified settings for defining what the conditions are for declaring the input unhealthy and failing over to a different input.

Type: AutomaticInputFailoverSettings
Required: False

inputAttachmentName

User-specified name for the attachment. This is required if the user wants to use this input in an input switch action.

Type: string
Required: False

inputId

The ID of the input

Type: string
Required: False

inputSettings

Settings of an input (caption selector, etc.)

Type: InputSettings
Required: False

InputChannelLevel

Input Channel Level

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

inputChannel

The index of the input channel used as a source.

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

InputCodec

codec in increasing order of complexity

MPEG2
AVC
HEVC

InputDeblockFilter

Input Deblock Filter

DISABLED
ENABLED

InputDenoiseFilter

Input Denoise Filter

DISABLED
ENABLED

InputFilter

Input Filter

AUTO
DISABLED
FORCED

InputLocation

Input Location

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 RTMP destination should have a uri simliar 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

Input Loss Action For Hls Out

```
EMIT_OUTPUT
PAUSE_OUTPUT
```

InputLossActionForMsSmoothOut

Input Loss Action For Ms Smooth Out

```
EMIT_OUTPUT
PAUSE_OUTPUT
```

InputLossActionForRtmpOut

Input Loss Action For Rtmp Out

```
EMIT_OUTPUT
PAUSE_OUTPUT
```

InputLossActionForUdpOut

Input Loss Action For Udp Out

```
DROP_PROGRAM
DROP_TS
EMIT_PROGRAM
```

InputLossBehavior

Input Loss Behavior

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

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
MinLength: 6
MaxLength: 6

inputLossImageSlate

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

Type: InputLocation
Required: False

inputLossImageType

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

Type: InputLossImageType
Required: False

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

InputLossFailoverSettings

MediaLive will perform a failover if content is not detected in this input for the specified period.

inputLossThresholdMsec

The amount of time (in milliseconds) that no input is detected. After that time, an input failover will occur.

Type: integer
Required: False
Minimum: 100

InputLossImageType

Input Loss Image Type

COLOR
SLATE

InputMaximumBitrate

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

MAX_10_MBPS
MAX_20_MBPS
MAX_50_MBPS

InputPreference

Input preference when deciding which input to make active when a previously failed input has recovered. If \"EQUAL_INPUT_PREFERENCE\", then the active input will stay active as long as it is healthy. If \"PRIMARY_INPUT_PREFERRED\", then always switch back to the primary input when it is healthy.

EQUAL_INPUT_PREFERENCE

PRIMARY_INPUT_PREFERRED

InputResolution

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

Live Event input parameters. There can be multiple inputs in a single Live Event.

audioSelectors

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

Type: Array of type AudioSelector
Required: False

captionSelectors

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

Type: Array of type CaptionSelector
Required: False

deblockFilter

Enable or disable the deblock filter when filtering.

Type: InputDeblockFilter
Required: False

denoiseFilter

Enable or disable the denoise filter when filtering.

Type: InputDenoiseFilter
Required: False

filterStrength

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

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

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: InputFilter
Required: False

networkInputSettings

Input settings.

Type: NetworkInputSettings
Required: False

smpte2038DataPreference

Specifies whether to extract applicable ancillary data from a SMPTE-2038 source in this input. Applicable data types are captions, timecode, AFD, and SCTE-104 messages. - PREFER: Extract from SMPTE-2038 if present in this input, otherwise extract from another source (if any). - IGNORE: Never extract any ancillary data from SMPTE-2038.

Type: Smpte2038DataPreference
Required: False

sourceEndBehavior

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

Type: InputSourceEndBehavior
Required: False

videoSelector

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

Type: VideoSelector
Required: False

InputSourceEndBehavior

Input Source End Behavior

CONTINUE
LOOP

InputSpecification

codec

Input codec

Type: InputCodec
Required: False

maximumBitrate

Maximum input bitrate, categorized coarsely

Type: InputMaximumBitrate
Required: False

resolution

Input resolution, categorized coarsely

Type: InputResolution
Required: False

InternalServiceError

message

Type: string
Required: False

InvalidRequest

message

Type: string
Required: False

KeyProviderSettings

Key Provider Settings

staticKeySettings

Type: StaticKeySettings
Required: False

LimitExceeded

message

Type: string
Required: False

LogLevel

The log level the user wants for their channel.

```
ERROR
WARNING
INFO
DEBUG
DISABLED
```

M2tsAbsentInputAudioBehavior

M2ts Absent Input Audio Behavior

```
DROP
ENCODE_SILENCE
```

M2tsArib

M2ts Arib

```
DISABLED
ENABLED
```

M2tsAribCaptionsPidControl

M2ts Arib Captions Pid Control

```
AUTO
USE_CONFIGURED
```

M2tsAudioBufferModel

M2ts Audio Buffer Model

```
ATSC
DVB
```

M2tsAudioInterval

M2ts Audio Interval

VIDEO_AND_FIXED_INTERVALS
VIDEO_INTERVAL

M2tsAudioStreamType

M2ts Audio Stream Type

ATSC
DVB

M2tsBufferModel

M2ts Buffer Model

MULTIPLEX
NONE

M2tsCcDescriptor

M2ts Cc Descriptor

DISABLED
ENABLED

M2tsEbifControl

M2ts Ebif Control

NONE
PASSTHROUGH

M2tsEbpPlacement

M2ts Ebp Placement

VIDEO_AND_AUDIO_PIDS
VIDEO_PID

M2tsEsRateInPes

M2ts Es Rate In Pes

EXCLUDE
INCLUDE

M2tsKlv

M2ts Klv

NONE
PASSTHROUGH

M2tsNielsenId3Behavior

M2ts Nielsen Id3 Behavior

NO_PASSTHROUGH
PASSTHROUGH

M2tsPcrControl

M2ts Pcr Control

CONFIGURED_PCR_PERIOD
PCR_EVERY_PES_PACKET

M2tsRateMode

M2ts Rate Mode

CBR
VBR

M2tsScte35Control

M2ts Scte35 Control

NONE
PASSTHROUGH

M2tsSegmentationMarkers

M2ts Segmentation Markers

EBP
EBP_LEGACY
NONE
PSI_SEGSTART
RAI_ADAPT
RAI_SEGSTART

M2tsSegmentationStyle

M2ts Segmentation Style

MAINTAIN_CADENCE
RESET_CADENCE

M2tsSettings

M2ts Settings

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: M2tsAbsentInputAudioBehavior

Required: False

arib

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

Type: M2tsArib
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

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: M2tsAribCaptionsPidControl
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: M2tsAudioBufferModel
Required: False

audioFramesPerPes

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

Type: integer

Required: False
Minimum: 0

audioPids

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

Type: string
Required: False

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: M2tsAudioStreamType
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

bufferModel

Controls the timing accuracy for output network traffic. Leave as MULTIPLEX to ensure accurate network packet timing. Or set to NONE, which might result in lower latency but will result in more variability in output network packet timing. This variability might cause interruptions, jitter, or bursty behavior in your playback or receiving devices.

Type: M2tsBufferModel

Required: False

ccDescriptor

When set to enabled, generates captionServiceDescriptor in PMT.

Type: M2tsCcDescriptor
Required: False

dvbNitSettings

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

Type: DvbNitSettings
Required: False

dvbSdtSettings

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

Type: DvbSdtSettings
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

dvbTdtSettings

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

Type: DvbTdtSettings
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

ebif

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

Type: M2tsEbifControl
Required: False

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: M2tsAudioInterval
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

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: M2tsEbpPlacement
Required: False

ecmPid

This field is unused and deprecated.

Type: string
Required: False

esRateInPes

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

Type: M2tsEsRateInPes
Required: False

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

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

fragmentTime

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

Type: number
Required: False
Minimum: 0

klv

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

Type: M2tsKlv
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

nielsenId3Behavior

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

Type: M2tsNielsenId3Behavior
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

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

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: M2tsPcrControl
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

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

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

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

programNum

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

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

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: M2tsRateMode
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

scte35Control

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

Type: M2tsScte35Control

Required: False

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

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: M2tsSegmentationMarkers

Required: False

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: M2tsSegmentationStyle
Required: False

segmentationTime

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

Type: number
Required: False
Minimum: 1

timedMetadataBehavior

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

Type: M2tsTimedMetadataBehavior
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

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. Valid values are 32 (or 0x20)..8182 (or 0x1ff6).

Type: string
Required: False

M2tsTimedMetadataBehavior

M2ts Timed Metadata Behavior

NO_PASSTHROUGH
PASSTHROUGH

M3u8NielsenId3Behavior

M3u8 Nielsen Id3 Behavior

NO_PASSTHROUGH
PASSTHROUGH

M3u8PcrControl

M3u8 Pcr Control

CONFIGURED_PCR_PERIOD
PCR_EVERY_PES_PACKET

M3u8Scte35Behavior

M3u8 Scte35 Behavior

NO_PASSTHROUGH
PASSTHROUGH

M3u8Settings

Settings information for the .m3u8 container

audioFramesPerPes

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

Type: integer
Required: False
Minimum: 0

audioPids

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

Type: string
Required: False

ecmPid

This parameter is unused and deprecated.

Type: string
Required: False

nielsenId3Behavior

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

Type: M3u8NielsenId3Behavior
Required: False

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

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: M3u8PcrControl
Required: False

pcrPeriod

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

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

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

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

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

programNum

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

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

scte35Behavior

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

Type: M3u8Scte35Behavior
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

timedMetadataBehavior

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

Type: M3u8TimedMetadataBehavior
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

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

M3u8TimedMetadataBehavior

M3u8 Timed Metadata Behavior

NO_PASSTHROUGH
PASSTHROUGH

MaintenanceDay

The currently selected maintenance day.

MONDAY
TUESDAY
WEDNESDAY
THURSDAY
FRIDAY
SATURDAY
SUNDAY

MaintenanceStatus

maintenanceDay

The currently selected maintenance day.

Type: MaintenanceDay
Required: False

maintenanceDeadline

Maintenance is required by the displayed date and time. Date and time is in ISO.

Type: string
Required: False
Format: string

maintenanceScheduledDate

The currently scheduled maintenance date and time. Date and time is in ISO.

Type: string
Required: False
Format: string

maintenanceStartTime

The currently selected maintenance start time. Time is in UTC.

Type: string
Required: False

MaintenanceUpdateSettings

maintenanceDay

Choose one day of the week for maintenance. The chosen day is used for all future maintenance windows.

Type: MaintenanceDay
Required: False

maintenanceScheduledDate

Choose a specific date for maintenance to occur. The chosen date is used for the next maintenance window only.

Type: string
Required: False
Format: string

maintenanceStartTime

Choose the hour that maintenance will start. The chosen time is used for all future maintenance windows.

Type: string
Required: False

Pattern: `^([0,1]?[0-9]|2[0-3]):00$`

MediaPackageGroupSettings

Media Package Group Settings

destination

MediaPackage channel destination.

Type: OutputLocationRef
Required: True

MediaPackageOutputDestinationSettings

MediaPackage Output Destination Settings

channelId

ID of the channel in MediaPackage that is the destination for this output group. You do not need to specify the individual inputs in MediaPackage; MediaLive will handle the connection of the two MediaLive pipelines to the two MediaPackage inputs. The MediaPackage channel and MediaLive channel must be in the same region.

Type: string
Required: False
MinLength: 1

MediaPackageOutputSettings

Media Package Output Settings

MotionGraphicsConfiguration

Motion Graphics Configuration

motionGraphicsInsertion

Type: MotionGraphicsInsertion
Required: False

motionGraphicsSettings

Motion Graphics Settings

Type: MotionGraphicsSettings
Required: True

MotionGraphicsInsertion

Motion Graphics Insertion

```
DISABLED
ENABLED
```

MotionGraphicsSettings

Motion Graphics Settings

htmlMotionGraphicsSettings

Type: HtmlMotionGraphicsSettings
Required: False

Mp2CodingMode

Mp2 Coding Mode

```
CODING_MODE_1_0
CODING_MODE_2_0
```

Mp2Settings

Mp2 Settings

bitrate

Average bitrate in bits/second.

Type: number
Required: False

codingMode

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

Type: Mp2CodingMode
Required: False

sampleRate

Sample rate in Hz.

Type: number
Required: False

Mpeg2AdaptiveQuantization

Mpeg2 Adaptive Quantization

```
AUTO
HIGH
LOW
MEDIUM
OFF
```

Mpeg2ColorMetadata

Mpeg2 Color Metadata

```
IGNORE
INSERT
```

Mpeg2ColorSpace

Mpeg2 Color Space

```
AUTO
PASSTHROUGH
```

Mpeg2DisplayRatio

Mpeg2 Display Ratio

```
DISPLAYRATIO16X9
DISPLAYRATIO4X3
```

Mpeg2FilterSettings

Mpeg2 Filter Settings

temporalFilterSettings

Type: TemporalFilterSettings
Required: False

Mpeg2GopSizeUnits

Mpeg2 Gop Size Units

```
FRAMES
SECONDS
```

Mpeg2ScanType

Mpeg2 Scan Type

```
INTERLACED
PROGRESSIVE
```

Mpeg2Settings

Mpeg2 Settings

adaptiveQuantization

Choose Off to disable adaptive quantization. Or choose another value to enable the quantizer and set its strength. The strengths are: Auto, Off, Low, Medium, High. When you enable this field, MediaLive allows intra-frame quantizers to vary, which might improve visual quality.

Type: Mpeg2AdaptiveQuantization
Required: False

afdSignaling

Indicates the AFD values that MediaLive will write into the video encode. If you do not know what AFD signaling is, or if your downstream system has not given you guidance, choose AUTO. AUTO: MediaLive will try to preserve the input AFD value (in cases where multiple AFD values are valid). FIXED: MediaLive will use the value you specify in fixedAFD.

Type: AfdSignaling
Required: False

colorMetadata

Specifies whether to include the color space metadata. The metadata describes the color space that applies to the video (the colorSpace field). We recommend that you insert the metadata.

Type: Mpeg2ColorMetadata
Required: False

colorSpace

Choose the type of color space conversion to apply to the output. For detailed information on setting up both the input and the output to obtain the desired color space in the output, see the section on \"MediaLive Features - Video - color space\" in the MediaLive User Guide. PASSTHROUGH: Keep the color space of the input content - do not convert it. AUTO:Convert all content that is SD to rec 601, and convert all content that is HD to rec 709.

Type: Mpeg2ColorSpace
Required: False

displayAspectRatio

Sets the pixel aspect ratio for the encode.

Type: Mpeg2DisplayRatio
Required: False

filterSettings

Optionally specify a noise reduction filter, which can improve quality of compressed content. If you do not choose a filter, no filter will be applied. TEMPORAL: This filter is useful for both source content that is noisy (when it has excessive digital artifacts) and source content that is clean. When the content is noisy, the filter cleans up the source content before the encoding phase, with these two effects: First, it improves the output video quality because the content has been cleaned up. Secondly, it decreases the bandwidth because MediaLive does not waste bits on encoding noise. When the content is reasonably clean, the filter tends to decrease the bitrate.

Type: Mpeg2FilterSettings
Required: False

fixedAfd

Complete this field only when afdSignaling is set to FIXED. Enter the AFD value (4 bits) to write on all frames of the video encode.

Type: FixedAfd
Required: False

framerateDenominator

description": "The framerate denominator. For example, 1001. The framerate is the numerator divided by the denominator. For example, 24000 / 1001 = 23.976 FPS.

Type: integer
Required: True
Minimum: 1

framerateNumerator

The framerate numerator. For example, 24000. The framerate is the numerator divided by the denominator. For example, 24000 / 1001 = 23.976 FPS.

Type: integer
Required: True
Minimum: 1

gopClosedCadence

MPEG2: default is open GOP.

Type: integer
Required: False
Minimum: 0

gopNumBFrames

Relates to the GOP structure. The number of B-frames between reference frames. If you do not know what a B-frame is, use the default.

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

gopSize

Relates to the GOP structure. The GOP size (keyframe interval) in the units specified in gopSizeUnits. If you do not know what GOP is, use the default. If gopSizeUnits is frames, then the gopSize must be an integer and must be greater than or equal to 1. If gopSizeUnits is seconds, the gopSize must be greater than 0, but does not need to be an integer.

Type: number
Required: False

gopSizeUnits

Relates to the GOP structure. Specifies whether the gopSize is specified in frames or seconds. If you do not plan to change the default gopSize, leave the default. If you specify SECONDS, MediaLive will internally convert the gop size to a frame count.

Type: Mpeg2GopSizeUnits
Required: False

scanType

Set the scan type of the output to PROGRESSIVE or INTERLACED (top field first).

Type: Mpeg2ScanType
Required: False

subgopLength

Relates to the GOP structure. If you do not know what GOP is, use the default. FIXED: Set the number of B-frames in each sub-GOP to the value in gopNumBFrames. DYNAMIC: Let MediaLive optimize the number of B-frames in each sub-GOP, to improve visual quality.

Type: Mpeg2SubGopLength
Required: False

timecodeInsertion

Determines how MediaLive inserts timecodes in the output video. For detailed information about setting up the input and the output for a timecode, see the section on \"MediaLive Features - Timecode configuration\" in the MediaLive User Guide. DISABLED: do not include timecodes. GOP_TIMECODE: Include timecode metadata in the GOP header.

Type: Mpeg2TimecodeInsertionBehavior
Required: False

Mpeg2SubGopLength

Mpeg2 Sub Gop Length

```
DYNAMIC
FIXED
```

Mpeg2TimecodeInsertionBehavior

Mpeg2 Timecode Insertion Behavior

```
DISABLED
```

GOP_TIMECODE

MsSmoothGroupSettings

Ms Smooth Group Settings

acquisitionPointId

The ID to include in each message in the sparse track. Ignored if sparseTrackType is NONE.

Type: string
Required: False

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: SmoothGroupAudioOnlyTimecodeControl
Required: False

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.

Type: SmoothGroupCertificateMode
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

destination

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

Type: OutputLocationRef
Required: True

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

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: SmoothGroupEventIdMode
Required: False

eventStopBehavior

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

Type: SmoothGroupEventStopBehavior
Required: False

filecacheDuration

Size in seconds of file cache for streaming outputs.

Type: integer
Required: False

Minimum: 0

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

inputLossAction

Parameter that control output group behavior on input loss.

Type: InputLossActionForMsSmoothOut
Required: False

numRetries

Number of retry attempts.

Type: integer
Required: False
Minimum: 0

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

segmentationMode

useInputSegmentation has been deprecated. The configured segment size is always used.

Type: SmoothGroupSegmentationMode
Required: False

sendDelayMs

Number of milliseconds to delay the output from the second pipeline.

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

sparseTrackType

Identifies the type of data to place in the sparse track: - SCTE35: Insert SCTE-35 messages from the source content. With each message, insert an IDR frame to start a new segment. - SCTE35_WITHOUT_SEGMENTATION: Insert SCTE-35 messages from the source content. With each message, insert an IDR frame but don't start a new segment. - NONE: Don't generate a sparse track for any outputs in this output group.

Type: SmoothGroupSparseTrackType
Required: False

streamManifestBehavior

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

Type: SmoothGroupStreamManifestBehavior
Required: False

timestampOffset

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

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: SmoothGroupTimestampOffsetMode
Required: False

MsSmoothH265PackagingType

Ms Smooth H265 Packaging Type

HEV1
HVC1

MsSmoothOutputSettings

Ms Smooth Output Settings

h265PackagingType

Only applicable when this output is referencing an H.265 video description. Specifies whether MP4 segments should be packaged as HEV1 or HVC1.

Type: MsSmoothH265PackagingType
Required: False

nameModifier

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

Type: string
Required: False

MultiplexGroupSettings

Multiplex Group Settings

MultiplexOutputSettings

Multiplex Output Settings

destination

Destination is a Multiplex.

Type: OutputLocationRef
Required: True

MultiplexProgramChannelDestinationSettings

Multiplex Program Input Destination Settings for outputting a Channel to a Multiplex

multiplexId

The ID of the Multiplex that the encoder is providing output to. You do not need to specify the individual inputs to the Multiplex; MediaLive will handle the connection of the two MediaLive pipelines to the two Multiplex instances. The Multiplex must be in the same region as the Channel.

Type: string
Required: False
MinLength: 1

programName

The program name of the Multiplex program that the encoder is providing output to.

Type: string
Required: False
MinLength: 1

NetworkInputServerValidation

Network Input Server Validation

CHECK_CRYPTOGRAPHY_AND_VALIDATE_NAME
CHECK_CRYPTOGRAPHY_ONLY

NetworkInputSettings

Network source to transcode. Must be accessible to the Elemental Live node that is running the live event through a network connection.

hlsInputSettings

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

Type: HlsInputSettings
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: NetworkInputServerValidation
Required: False

NielsenCBET

Nielsen CBET

cbetCheckDigitString

Enter the CBET check digits to use in the watermark.

Type: string
Required: True
MinLength: 2
MaxLength: 2

cbetStepaside

Determines the method of CBET insertion mode when prior encoding is detected on the same layer.

Type: NielsenWatermarksCbetStepaside
Required: True

csid

Enter the CBET Source ID (CSID) to use in the watermark

Type: string
Required: True
MinLength: 1
MaxLength: 7

NielsenConfiguration

Nielsen Configuration

distributorId

Enter the Distributor ID assigned to your organization by Nielsen.

Type: string
Required: False

nielsenPcmToId3Tagging

Enables Nielsen PCM to ID3 tagging

Type: NielsenPcmToId3TaggingState
Required: False

NielsenNaesIiNw

Nielsen Naes Ii Nw

checkDigitString

Enter the check digit string for the watermark

Type: string

Required: True
MinLength: 2
MaxLength: 2

sid

Enter the Nielsen Source ID (SID) to include in the watermark

Type: number
Required: True
Minimum: 1
Maximum: 65535

NielsenPcmToId3TaggingState

State of Nielsen PCM to ID3 tagging

DISABLED
ENABLED

NielsenWatermarksCbetStepaside

Nielsen Watermarks Cbet Stepaside

DISABLED
ENABLED

NielsenWatermarksDistributionTypes

Nielsen Watermarks Distribution Types

FINAL_DISTRIBUTOR
PROGRAM_CONTENT

NielsenWatermarksSettings

Nielsen Watermarks Settings

nielsenCbetSettings

Complete these fields only if you want to insert watermarks of type Nielsen CBET

Type: NielsenCBET
Required: False

nielsenDistributionType

Choose the distribution types that you want to assign to the watermarks: - PROGRAM_CONTENT - FINAL_DISTRIBUTOR

Type: NielsenWatermarksDistributionTypes
Required: False

nielsenNaesIiNwSettings

Complete these fields only if you want to insert watermarks of type Nielsen NAES II (N2) and Nielsen NAES VI (NW).

Type: NielsenNaesIiNw
Required: False

Output

Output settings. There can be multiple outputs within a group.

audioDescriptionNames

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

Type: Array of 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
MinLength: 1
MaxLength: 255

outputSettings

Output type-specific settings.

Type: OutputSettings
Required: True

videoDescriptionName

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

Type: string
Required: False

OutputDestination

id

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

Type: string
Required: False

mediaPackageSettings

Destination settings for a MediaPackage output; one destination for both encoders.

Type: Array of type MediaPackageOutputDestinationSettings

Required: False

multiplexSettings

Destination settings for a Multiplex output; one destination for both encoders.

Type: MultiplexProgramChannelDestinationSettings
Required: False

settings

Destination settings for a standard output; one destination for each redundant encoder.

Type: Array of type OutputDestinationSettings
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

Output groups for this Live Event. Output groups contain information about where streams should be distributed.

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
MaxLength: 32

outputGroupSettings

Settings associated with the output group.

Type: OutputGroupSettings
Required: True

outputs

Type: Array of type Output
Required: True

OutputGroupSettings

Output Group Settings

archiveGroupSettings

Type: ArchiveGroupSettings

Required: False

frameCaptureGroupSettings

Type: FrameCaptureGroupSettings
Required: False

hlsGroupSettings

Type: HlsGroupSettings
Required: False

mediaPackageGroupSettings

Type: MediaPackageGroupSettings
Required: False

msSmoothGroupSettings

Type: MsSmoothGroupSettings
Required: False

multiplexGroupSettings

Type: MultiplexGroupSettings
Required: False

rtmpGroupSettings

Type: RtmpGroupSettings
Required: False

udpGroupSettings

Type: UdpGroupSettings
Required: False

OutputLocationRef

Reference to an OutputDestination ID defined in the channel

destinationRefId

Type: string
Required: False

OutputSettings

Output Settings

archiveOutputSettings

Type: ArchiveOutputSettings
Required: False

frameCaptureOutputSettings

Type: FrameCaptureOutputSettings
Required: False

hlsOutputSettings

Type: HlsOutputSettings
Required: False

mediaPackageOutputSettings

Type: MediaPackageOutputSettings
Required: False

msSmoothOutputSettings

Type: MsSmoothOutputSettings
Required: False

multiplexOutputSettings

Type: MultiplexOutputSettings
Required: False

rtmpOutputSettings

Type: RtmpOutputSettings
Required: False

udpOutputSettings

Type: UdpOutputSettings
Required: False

PassThroughSettings

Pass Through Settings

PipelineDetail

Runtime details of a pipeline when a channel is running.

activeInputAttachmentName

The name of the active input attachment currently being ingested by this pipeline.

Type: string
Required: False

activeInputSwitchActionName

The name of the input switch schedule action that occurred most recently and that resulted in the switch to the current input attachment for this pipeline.

Type: string
Required: False

activeMotionGraphicsActionName

The name of the motion graphics activate action that occurred most recently and that resulted in the current graphics URI for this pipeline.

Type: string
Required: False

activeMotionGraphicsUri

The current URI being used for HTML5 motion graphics for this pipeline.

Type: string
Required: False

pipelineId

Pipeline ID

Type: string
Required: False

RawSettings

Raw Settings

Rec601Settings

Rec601 Settings

Rec709Settings

Rec709 Settings

RemixSettings

Remix Settings

channelMappings

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

Type: Array of type AudioChannelMapping
Required: True

channelsIn

Number of input channels to be used.

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

channelsOut

Number of output channels to be produced. Valid values: 1, 2, 4, 6, 8

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

ResourceConflict

message

Type: string
Required: False

ResourceNotFound

message

Type: string
Required: False

RtmpAdMarkers

Rtmp Ad Markers

ON_CUE_POINT_SCTE35

RtmpCacheFullBehavior

Rtmp Cache Full Behavior

DISCONNECT_IMMEDIATELY
WAIT_FOR_SERVER

RtmpCaptionData

Rtmp Caption Data

ALL
FIELD1_608
FIELD1_AND_FIELD2_608

RtmpCaptionInfoDestinationSettings

Rtmp Caption Info Destination Settings

RtmpGroupSettings

Rtmp Group Settings

adMarkers

Choose the ad marker type for this output group. MediaLive will create a message based on the content of each SCTE-35 message, format it for that marker type, and insert it in the datastream.

Type: Array of type RtmpAdMarkers
Required: False

authenticationScheme

Authentication scheme to use when connecting with CDN

Type: AuthenticationScheme

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: RtmpCacheFullBehavior
Required: False

cacheLength

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

Type: integer
Required: False
Minimum: 30

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: RtmpCaptionData
Required: False

inputLossAction

Controls the behavior of this RTMP group if input becomes unavailable. - emitOutput: Emit a slate until input returns. - pauseOutput: Stop transmitting data until input returns. This does not close the underlying RTMP connection.

Type: InputLossActionForRtmpOut
Required: False

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

RtmpOutputCertificateMode

Rtmp Output Certificate Mode

```
SELF_SIGNED
VERIFY_AUTHENTICITY
```

RtmpOutputSettings

Rtmp Output Settings

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: RtmpOutputCertificateMode
Required: False

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

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
Required: True

numRetries

Number of retry attempts.

Type: integer
Required: False
Minimum: 0

S3CannedAcl

S3 Canned Acl

```
AUTHENTICATED_READ
BUCKET_OWNER_FULL_CONTROL
BUCKET_OWNER_READ
PUBLIC_READ
```

Scte20Convert608To708

Scte20 Convert608 To708

```
DISABLED
UPCONVERT
```

Scte20PlusEmbeddedDestinationSettings

Scte20 Plus Embedded Destination Settings

Scte20SourceSettings

Scte20 Source Settings

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: Scte20Convert608To708
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

Scte27DestinationSettings

Scte27 Destination Settings

Scte27OcrLanguage

Scte27 Ocr Language

DEU
ENG
FRA
NLD
POR
SPA

Scte27SourceSettings

Scte27 Source Settings

ocrLanguage

If you will configure a WebVTT caption description that references this caption selector, use this field to provide the language to consider when translating the image-based source to text.

Type: Scte27OcrLanguage
Required: False

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

Scte35 Apos No Regional Blackout Behavior

```
FOLLOW
IGNORE
```

Scte35AposWebDeliveryAllowedBehavior

Scte35 Apos Web Delivery Allowed Behavior

```
FOLLOW
IGNORE
```

Scte35SpliceInsert

Scte35 Splice Insert

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

noRegionalBlackoutFlag

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

Type: Scte35SpliceInsertNoRegionalBlackoutBehavior
Required: False

webDeliveryAllowedFlag

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

Type: Scte35SpliceInsertWebDeliveryAllowedBehavior
Required: False

Scte35SpliceInsertNoRegionalBlackoutBehavior

Scte35 Splice Insert No Regional Blackout Behavior

```
FOLLOW
IGNORE
```

Scte35SpliceInsertWebDeliveryAllowedBehavior

Scte35 Splice Insert Web Delivery Allowed Behavior

```
FOLLOW
```

IGNORE

Scte35TimeSignalApos

Scte35 Time Signal Apos

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

noRegionalBlackoutFlag

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

Type: Scte35AposNoRegionalBlackoutBehavior
Required: False

webDeliveryAllowedFlag

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

Type: Scte35AposWebDeliveryAllowedBehavior
Required: False

SmoothGroupAudioOnlyTimecodeControl

Smooth Group Audio Only Timecode Control

PASSTHROUGH
USE_CONFIGURED_CLOCK

SmoothGroupCertificateMode

Smooth Group Certificate Mode

SELF_SIGNED
VERIFY_AUTHENTICITY

SmoothGroupEventIdMode

Smooth Group Event Id Mode

NO_EVENT_ID
USE_CONFIGURED
USE_TIMESTAMP

SmoothGroupEventStopBehavior

Smooth Group Event Stop Behavior

NONE
SEND_EOS

SmoothGroupSegmentationMode

Smooth Group Segmentation Mode

USE_INPUT_SEGMENTATION
USE_SEGMENT_DURATION

SmoothGroupSparseTrackType

Smooth Group Sparse Track Type

NONE
SCTE_35
SCTE_35_WITHOUT_SEGMENTATION

SmoothGroupStreamManifestBehavior

Smooth Group Stream Manifest Behavior

DO_NOT_SEND
SEND

SmoothGroupTimestampOffsetMode

Smooth Group Timestamp Offset Mode

USE_CONFIGURED_OFFSET
USE_EVENT_START_DATE

Smpte2038DataPreference

Smpte2038 Data Preference

IGNORE
PREFER

SmpteTtDestinationSettings

Smpte Tt Destination Settings

StandardHlsSettings

Standard Hls Settings

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

m3u8Settings

Type: M3u8Settings

Required: True

StaticKeySettings

Static Key Settings

keyProviderServer

The URL of the license server used for protecting content.

Type: InputLocation
Required: False

staticKeyValue

Static key value as a 32 character hexadecimal string.

Type: string
Required: True
MinLength: 32
MaxLength: 32

Tags

key-value pairs

Type: string

TeletextDestinationSettings

Teletext Destination Settings

TeletextSourceSettings

Teletext Source Settings

outputRectangle

Optionally defines a region where TTML style captions will be displayed

Type: CaptionRectangle
Required: False

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

TemporalFilterPostFilterSharpening

Temporal Filter Post Filter Sharpening

AUTO
DISABLED
ENABLED

TemporalFilterSettings

Temporal Filter Settings

postFilterSharpening

If you enable this filter, the results are the following: - If the source content is noisy (it contains excessive digital artifacts), the filter cleans up the source. - If the source content is already clean, the filter tends to decrease the bitrate, especially when the rate control mode is QVBR.

Type: TemporalFilterPostFilterSharpening
Required: False

strength

Choose a filter strength. We recommend a strength of 1 or 2. A higher strength might take out good information, resulting in an image that is overly soft.

Type: TemporalFilterStrength
Required: False

TemporalFilterStrength

Temporal Filter Strength

AUTO
STRENGTH_1
STRENGTH_2
STRENGTH_3
STRENGTH_4
STRENGTH_5
STRENGTH_6
STRENGTH_7
STRENGTH_8
STRENGTH_9
STRENGTH_10
STRENGTH_11
STRENGTH_12
STRENGTH_13
STRENGTH_14
STRENGTH_15
STRENGTH_16

TimecodeConfig

Timecode Config

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: TimecodeConfigSource
Required: True

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

TimecodeConfigSource

Timecode Config Source

```
EMBEDDED
SYSTEMCLOCK
ZEROBASED
```

TtmlDestinationSettings

Ttml Destination Settings

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: TtmlDestinationStyleControl
Required: False

TtmlDestinationStyleControl

Ttml Destination Style Control

PASSTHROUGH
USE_CONFIGURED

UdpContainerSettings

Udp Container Settings

m2tsSettings

Type: M2tsSettings
Required: False

UdpGroupSettings

Udp Group Settings

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: InputLossActionForUdpOut
Required: False

timedMetadataId3Frame

Indicates ID3 frame that has the timecode.

Type: UdpTimedMetadataId3Frame
Required: False

timedMetadataId3Period

Timed Metadata interval in seconds.

Type: integer
Required: False
Minimum: 0

UdpOutputSettings

Udp Output Settings

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

containerSettings

Type: UdpContainerSettings
Required: True

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
Required: True

fecOutputSettings

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

Type: FecOutputSettings
Required: False

UdpTimedMetadataId3Frame

Udp Timed Metadata Id3 Frame

NONE
PRIV
TDRL

UpdateChannel

cdiInputSpecification

Specification of CDI inputs for this channel

Type: CdiInputSpecification
Required: False

destinations

A list of output destinations for this channel.

Type: Array of type OutputDestination
Required: False

encoderSettings

The encoder settings for this channel.

Type: EncoderSettings
Required: False

inputAttachments

Type: Array of type InputAttachment
Required: False

inputSpecification

Specification of network and file inputs for this channel

Type: InputSpecification
Required: False

logLevel

The log level to write to CloudWatch Logs.

Type: LogLevel
Required: False

maintenance

Maintenance settings for this channel.

Type: MaintenanceUpdateSettings
Required: False

name

The name of the channel.

Type: string
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

UpdateChannelResultModel

The updated channel's description.

channel

Type: Channel

Required: False

ValidationError

elementPath

Path to the source of the error.

Type: string
Required: False

errorMessage

The error message.

Type: string
Required: False

VideoBlackFailoverSettings

blackDetectThreshold

A value used in calculating the threshold below which MediaLive considers a pixel to be 'black'. For the input to be considered black, every pixel in a frame must be below this threshold. The threshold is calculated as a percentage (expressed as a decimal) of white. Therefore .1 means 10% white (or 90% black). Note how the formula works for any color depth. For example, if you set this field to 0.1 in 10-bit color depth: (1023*0.1=102.3), which means a pixel value of 102 or less is 'black'. If you set this field to .1 in an 8-bit color depth: (255*0.1=25.5), which means a pixel value of 25 or less is 'black'. The range is 0.0 to 1.0, with any number of decimal places.

Type: number
Required: False
Minimum: 0
Maximum: 1

videoBlackThresholdMsec

The amount of time (in milliseconds) that the active input must be black before automatic input failover occurs.

Type: integer
Required: False
Minimum: 1000

VideoCodecSettings

Video Codec Settings

frameCaptureSettings

Type: FrameCaptureSettings
Required: False

h264Settings

Type: H264Settings
Required: False

h265Settings

Type: H265Settings
Required: False

mpeg2Settings

Type: Mpeg2Settings
Required: False

VideoDescription

Video settings for this stream.

codecSettings

Video codec settings.

Type: VideoCodecSettings
Required: False

height

Output video height, in pixels. Must be an even number. For most codecs, you can leave this field and width blank in order to use the height and width (resolution) from the source. Note, however, that leaving blank is not recommended. For the Frame Capture codec, height and width are required.

Type: integer
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

respondToAfd

Indicates how MediaLive will respond to the AFD values that might be in the input video. If you do not know what AFD signaling is, or if your downstream system has not given you guidance, choose PASSTHROUGH. RESPOND: MediaLive clips the input video using a formula that uses the AFD values (configured in afdSignaling), the input display aspect ratio, and the output display aspect ratio. MediaLive also includes the AFD values in the output, unless the codec for this encode is FRAME_CAPTURE. PASSTHROUGH: MediaLive ignores the AFD values and does not clip the video. But MediaLive does include the values in the output. NONE: MediaLive does not clip the input video and does not include the AFD values in the output

Type: VideoDescriptionRespondToAfd
Required: False

scalingBehavior

STRETCH_TO_OUTPUT configures the output position to stretch the video to the specified output resolution (height and width). This option will override any position value. DEFAULT may insert black boxes (pillar boxes or letter boxes) around the video to provide the specified output resolution.

Type: VideoDescriptionScalingBehavior
Required: False

sharpness

Changes the strength of the anti-alias filter used for scaling. 0 is the softest setting, 100 is the sharpest. A setting of 50 is recommended for most content.

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

width

Output video width, in pixels. Must be an even number. For most codecs, you can leave this field and height blank in order to use the height and width (resolution) from the source. Note, however, that leaving blank is not recommended. For the Frame Capture codec, height and width are required.

Type: integer
Required: False

VideoDescriptionRespondToAfd

Video Description Respond To Afd

NONE
PASSTHROUGH
RESPOND

VideoDescriptionScalingBehavior

Video Description Scaling Behavior

DEFAULT
STRETCH_TO_OUTPUT

VideoSelector

Specifies a particular video stream within an input source. An input may have only a single video selector.

colorSpace

Specifies the color space of an input. This setting works in tandem with colorSpaceUsage and a video description's colorSpaceSettingsChoice to determine if any conversion will be performed.

Type: VideoSelectorColorSpace
Required: False

colorSpaceSettings

Color space settings

Type: VideoSelectorColorSpaceSettings
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: VideoSelectorColorSpaceUsage
Required: False

selectorSettings

The video selector settings.

Type: VideoSelectorSettings
Required: False

VideoSelectorColorSpace

Video Selector Color Space

```
FOLLOW
HDR10
HLG_2020
REC_601
REC_709
```

VideoSelectorColorSpaceSettings

Video Selector Color Space Settings

hdr10Settings

Type: Hdr10Settings
Required: False

VideoSelectorColorSpaceUsage

Video Selector Color Space Usage

```
FALLBACK
FORCE
```

VideoSelectorPid

Video Selector Pid

pid

Selects a specific PID from within a video source.

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

VideoSelectorProgramId

Video Selector Program Id

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

Video Selector Settings

videoSelectorPid

Type: VideoSelectorPid
Required: False

videoSelectorProgramId

Type: VideoSelectorProgramId
Required: False

VpcOutputSettingsDescription

The properties for a private VPC Output

availabilityZones

The Availability Zones where the vpc subnets are located. The first Availability Zone applies to the first subnet in the list of subnets. The second Availability Zone applies to the second subnet.

Type: Array of type string
Required: False

networkInterfaceIds

A list of Elastic Network Interfaces created by MediaLive in the customer's VPC

Type: Array of type string
Required: False

securityGroupIds

A list of up EC2 VPC security group IDs attached to the Output VPC network interfaces.

Type: Array of type string
Required: False

subnetIds

A list of VPC subnet IDs from the same VPC. If STANDARD channel, subnet IDs must be mapped to two unique availability zones (AZ).

Type: Array of type string
Required: False

WavCodingMode

Wav Coding Mode

`CODING_MODE_1_0`
`CODING_MODE_2_0`
`CODING_MODE_4_0`
`CODING_MODE_8_0`

WavSettings

Wav Settings

bitDepth

Bits per sample.

Type: number
Required: False

codingMode

The audio coding mode for the WAV audio. The mode determines the number of channels in the audio.

Type: WavCodingMode
Required: False

sampleRate

Sample rate in Hz.

Type: number
Required: False

WebvttDestinationSettings

Webvtt Destination Settings

styleControl

Controls whether the color and position of the source captions is passed through to the WebVTT output captions. PASSTHROUGH - Valid only if the source captions are EMBEDDED or TELETEXT. NO_STYLE_DATA - Don't pass through the style. The output captions will not contain any font styling information.

Type: WebvttDestinationStyleControl
Required: False

WebvttDestinationStyleControl

Webvtt Destination Style Control

```
NO_STYLE_DATA
PASSTHROUGH
```

See also

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

DeleteChannel

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

DescribeChannel

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

UpdateChannel

- AWS Command Line Interface
- AWS SDK for .NET

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Channels: class

URI

`/prod/channels/channelId/channelClass`

HTTP methods

PUT

Operation ID: `UpdateChannelClass`

Path parameters

Name	Type	Required	Description
channelId	String	True	

Responses

Status code	Response model	Description
200	UpdateChannelResultModel	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response

Status code	Response model	Description
409	ResourceConflict	409 response
422	ChannelConfigurationValidationError	422 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

Schemas

Request bodies

PUT schema

```
{
  "channelClass": enum,
  "destinations": [
    {
      "id": "string",
      "mediaPackageSettings": [
        {
          "channelId": "string"
        }
      ],
      "multiplexSettings": {
        "multiplexId": "string",
        "programName": "string"
      },
      "settings": [
        {
          "passwordParam": "string",
          "streamName": "string",
          "url": "string",
```

```
          "username": "string"
        }
      ]
    }
  ]
}
```

Response bodies

UpdateChannelResultModel schema

```
{
  "channel": {
    "arn": "string",
    "cdiInputSpecification": {
      "resolution": enum
    },
    "channelClass": enum,
    "destinations": [
      {
        "id": "string",
        "mediaPackageSettings": [
          {
            "channelId": "string"
          }
        ],
        "multiplexSettings": {
          "multiplexId": "string",
          "programName": "string"
        },
        "settings": [
          {
            "passwordParam": "string",
            "streamName": "string",
            "url": "string",
            "username": "string"
          }
        ]
      }
    ],
    "egressEndpoints": [
      {
        "sourceIp": "string"
```

```
      }
    ],
    "encoderSettings": {
      "audioDescriptions": [
        {
          "audioNormalizationSettings": {
            "algorithm": enum,
            "algorithmControl": enum,
            "targetLkfs": number
          },
          "audioSelectorName": "string",
          "audioType": enum,
          "audioTypeControl": enum,
          "audioWatermarkingSettings": {
            "nielsenWatermarksSettings": {
              "nielsenCbetSettings": {
                "cbetCheckDigitString": "string",
                "cbetStepaside": enum,
                "csid": "string"
              },
              "nielsenDistributionType": enum,
              "nielsenNaesIiNwSettings": {
                "checkDigitString": "string",
                "sid": number
              }
            }
          },
          "codecSettings": {
            "aacSettings": {
              "bitrate": number,
              "codingMode": enum,
              "inputType": enum,
              "profile": enum,
              "rateControlMode": enum,
              "rawFormat": enum,
              "sampleRate": number,
              "spec": enum,
              "vbrQuality": enum
            },
            "ac3Settings": {
              "bitrate": number,
              "bitstreamMode": enum,
              "codingMode": enum,
              "dialnorm": integer,
```

```
            "drcProfile": enum,
            "lfeFilter": enum,
            "metadataControl": enum
          },
          "eac3Settings": {
            "attenuationControl": enum,
            "bitrate": number,
            "bitstreamMode": enum,
            "codingMode": enum,
            "dcFilter": enum,
            "dialnorm": integer,
            "drcLine": enum,
            "drcRf": enum,
            "lfeControl": enum,
            "lfeFilter": enum,
            "loRoCenterMixLevel": number,
            "loRoSurroundMixLevel": number,
            "ltRtCenterMixLevel": number,
            "ltRtSurroundMixLevel": number,
            "metadataControl": enum,
            "passthroughControl": enum,
            "phaseControl": enum,
            "stereoDownmix": enum,
            "surroundExMode": enum,
            "surroundMode": enum
          },
          "mp2Settings": {
            "bitrate": number,
            "codingMode": enum,
            "sampleRate": number
          },
          "passThroughSettings": {
          },
          "wavSettings": {
            "bitDepth": number,
            "codingMode": enum,
            "sampleRate": number
          }
        },
        "languageCode": "string",
        "languageCodeControl": enum,
        "name": "string",
        "remixSettings": {
          "channelMappings": [
```

```
            {
              "inputChannelLevels": [
                {
                  "gain": integer,
                  "inputChannel": integer
                }
              ],
              "outputChannel": integer
            }
          ],
          "channelsIn": integer,
          "channelsOut": integer
        },
        "streamName": "string"
      }
    ],
    "availBlanking": {
      "availBlankingImage": {
        "passwordParam": "string",
        "uri": "string",
        "username": "string"
      },
      "state": enum
    },
    "availConfiguration": {
      "availSettings": {
        "scte35SpliceInsert": {
          "adAvailOffset": integer,
          "noRegionalBlackoutFlag": enum,
          "webDeliveryAllowedFlag": enum
        },
        "scte35TimeSignalApos": {
          "adAvailOffset": integer,
          "noRegionalBlackoutFlag": enum,
          "webDeliveryAllowedFlag": enum
        }
      }
    },
    "blackoutSlate": {
      "blackoutSlateImage": {
        "passwordParam": "string",
        "uri": "string",
        "username": "string"
      },
```

```
        "networkEndBlackout": enum,
        "networkEndBlackoutImage": {
          "passwordParam": "string",
          "uri": "string",
          "username": "string"
        },
        "networkId": "string",
        "state": enum
      },
      "captionDescriptions": [
        {
          "captionSelectorName": "string",
          "destinationSettings": {
            "aribDestinationSettings": {
            },
            "burnInDestinationSettings": {
              "alignment": enum,
              "backgroundColor": enum,
              "backgroundOpacity": integer,
              "font": {
                "passwordParam": "string",
                "uri": "string",
                "username": "string"
              },
              "fontColor": enum,
              "fontOpacity": integer,
              "fontResolution": integer,
              "fontSize": "string",
              "outlineColor": enum,
              "outlineSize": integer,
              "shadowColor": enum,
              "shadowOpacity": integer,
              "shadowXOffset": integer,
              "shadowYOffset": integer,
              "teletextGridControl": enum,
              "xPosition": integer,
              "yPosition": integer
            },
            "dvbSubDestinationSettings": {
              "alignment": enum,
              "backgroundColor": enum,
              "backgroundOpacity": integer,
              "font": {
                "passwordParam": "string",
```

```
            "uri": "string",
            "username": "string"
          },
          "fontColor": enum,
          "fontOpacity": integer,
          "fontResolution": integer,
          "fontSize": "string",
          "outlineColor": enum,
          "outlineSize": integer,
          "shadowColor": enum,
          "shadowOpacity": integer,
          "shadowXOffset": integer,
          "shadowYOffset": integer,
          "teletextGridControl": enum,
          "xPosition": integer,
          "yPosition": integer
        },
        "ebuTtDDestinationSettings": {
          "copyrightHolder": "string",
          "fillLineGap": enum,
          "fontFamily": "string",
          "styleControl": enum
        },
        "embeddedDestinationSettings": {
        },
        "embeddedPlusScte20DestinationSettings": {
        },
        "rtmpCaptionInfoDestinationSettings": {
        },
        "scte20PlusEmbeddedDestinationSettings": {
        },
        "scte27DestinationSettings": {
        },
        "smpteTtDestinationSettings": {
        },
        "teletextDestinationSettings": {
        },
        "ttmlDestinationSettings": {
          "styleControl": enum
        },
        "webvttDestinationSettings": {
          "styleControl": enum
        }
      },
```

```
          "languageCode": "string",
          "languageDescription": "string",
          "name": "string"
        }
      ],
      "featureActivations": {
        "inputPrepareScheduleActions": enum
      },
      "globalConfiguration": {
        "initialAudioGain": integer,
        "inputEndAction": enum,
        "inputLossBehavior": {
          "blackFrameMsec": integer,
          "inputLossImageColor": "string",
          "inputLossImageSlate": {
            "passwordParam": "string",
            "uri": "string",
            "username": "string"
          },
          "inputLossImageType": enum,
          "repeatFrameMsec": integer
        },
        "outputLockingMode": enum,
        "outputTimingSource": enum,
        "supportLowFramerateInputs": enum
      },
      "motionGraphicsConfiguration": {
        "motionGraphicsInsertion": enum,
        "motionGraphicsSettings": {
          "htmlMotionGraphicsSettings": {
          }
        }
      },
      "nielsenConfiguration": {
        "distributorId": "string",
        "nielsenPcmToId3Tagging": enum
      },
      "outputGroups": [
        {
          "name": "string",
          "outputGroupSettings": {
            "archiveGroupSettings": {
              "archiveCdnSettings": {
                "archiveS3Settings": {
```

```
            "cannedAcl": enum,
            "logUploads": enum
          }
        },
        "destination": {
          "destinationRefId": "string"
        },
        "rolloverInterval": integer
      },
      "frameCaptureGroupSettings": {
        "destination": {
          "destinationRefId": "string"
        },
        "frameCaptureCdnSettings": {
          "frameCaptureS3Settings": {
            "cannedAcl": enum,
            "logUploads": enum
          }
        }
      },
      "hlsGroupSettings": {
        "adMarkers": [
          enum
        ],
        "baseUrlContent": "string",
        "baseUrlContent1": "string",
        "baseUrlManifest": "string",
        "baseUrlManifest1": "string",
        "captionLanguageMappings": [
          {
            "captionChannel": integer,
            "languageCode": "string",
            "languageDescription": "string"
          }
        ],
        "captionLanguageSetting": enum,
        "clientCache": enum,
        "codecSpecification": enum,
        "constantIv": "string",
        "destination": {
          "destinationRefId": "string"
        },
        "directoryStructure": enum,
        "discontinuityTags": enum,
```

```
            "encryptionType": enum,
            "hlsCdnSettings": {
              "hlsAkamaiSettings": {
                "connectionRetryInterval": integer,
                "filecacheDuration": integer,
                "httpTransferMode": enum,
                "numRetries": integer,
                "restartDelay": integer,
                "salt": "string",
                "token": "string"
              },
              "hlsBasicPutSettings": {
                "connectionRetryInterval": integer,
                "filecacheDuration": integer,
                "numRetries": integer,
                "restartDelay": integer
              },
              "hlsMediaStoreSettings": {
                "connectionRetryInterval": integer,
                "filecacheDuration": integer,
                "mediaStoreStorageClass": enum,
                "numRetries": integer,
                "restartDelay": integer
              },
              "hlsS3Settings": {
                "cannedAcl": enum,
                "logUploads": enum
              },
              "hlsWebdavSettings": {
                "connectionRetryInterval": integer,
                "filecacheDuration": integer,
                "httpTransferMode": enum,
                "numRetries": integer,
                "restartDelay": integer
              }
            },
            "hlsId3SegmentTagging": enum,
            "iFrameOnlyPlaylists": enum,
            "incompleteSegmentBehavior": enum,
            "indexNSegments": integer,
            "inputLossAction": enum,
            "ivInManifest": enum,
            "ivSource": enum,
            "keepSegments": integer,
```

```
            "keyFormat": "string",
            "keyFormatVersions": "string",
            "keyProviderSettings": {
              "staticKeySettings": {
                "keyProviderServer": {
                  "passwordParam": "string",
                  "uri": "string",
                  "username": "string"
                },
                "staticKeyValue": "string"
              }
            },
            "manifestCompression": enum,
            "manifestDurationFormat": enum,
            "minSegmentLength": integer,
            "mode": enum,
            "outputSelection": enum,
            "programDateTime": enum,
            "programDateTimeClock": enum,
            "programDateTimePeriod": integer,
            "redundantManifest": enum,
            "segmentLength": integer,
            "segmentationMode": enum,
            "segmentsPerSubdirectory": integer,
            "streamInfResolution": enum,
            "timedMetadataId3Frame": enum,
            "timedMetadataId3Period": integer,
            "timestampDeltaMilliseconds": integer,
            "tsFileMode": enum
          },
          "mediaPackageGroupSettings": {
            "destination": {
              "destinationRefId": "string"
            }
          },
          "msSmoothGroupSettings": {
            "acquisitionPointId": "string",
            "audioOnlyTimecodeControl": enum,
            "certificateMode": enum,
            "connectionRetryInterval": integer,
            "destination": {
              "destinationRefId": "string"
            },
            "eventId": "string",
```

```
            "eventIdMode": enum,
            "eventStopBehavior": enum,
            "filecacheDuration": integer,
            "fragmentLength": integer,
            "inputLossAction": enum,
            "numRetries": integer,
            "restartDelay": integer,
            "segmentationMode": enum,
            "sendDelayMs": integer,
            "sparseTrackType": enum,
            "streamManifestBehavior": enum,
            "timestampOffset": "string",
            "timestampOffsetMode": enum
          },
          "multiplexGroupSettings": {
          },
          "rtmpGroupSettings": {
            "adMarkers": [
              enum
            ],
            "authenticationScheme": enum,
            "cacheFullBehavior": enum,
            "cacheLength": integer,
            "captionData": enum,
            "inputLossAction": enum,
            "restartDelay": integer
          },
          "udpGroupSettings": {
            "inputLossAction": enum,
            "timedMetadataId3Frame": enum,
            "timedMetadataId3Period": integer
          }
        },
        "outputs": [
          {
            "audioDescriptionNames": [
              "string"
            ],
            "captionDescriptionNames": [
              "string"
            ],
            "outputName": "string",
            "outputSettings": {
              "archiveOutputSettings": {
```

```
      "containerSettings": {
        "m2tsSettings": {
          "absentInputAudioBehavior": enum,
          "arib": enum,
          "aribCaptionsPid": "string",
          "aribCaptionsPidControl": enum,
          "audioBufferModel": enum,
          "audioFramesPerPes": integer,
          "audioPids": "string",
          "audioStreamType": enum,
          "bitrate": integer,
          "bufferModel": enum,
          "ccDescriptor": enum,
          "dvbNitSettings": {
            "networkId": integer,
            "networkName": "string",
            "repInterval": integer
          },
          "dvbSdtSettings": {
            "outputSdt": enum,
            "repInterval": integer,
            "serviceName": "string",
            "serviceProviderName": "string"
          },
          "dvbSubPids": "string",
          "dvbTdtSettings": {
            "repInterval": integer
          },
          "dvbTeletextPid": "string",
          "ebif": enum,
          "ebpAudioInterval": enum,
          "ebpLookaheadMs": integer,
          "ebpPlacement": enum,
          "ecmPid": "string",
          "esRateInPes": enum,
          "etvPlatformPid": "string",
          "etvSignalPid": "string",
          "fragmentTime": number,
          "klv": enum,
          "klvDataPids": "string",
          "nielsenId3Behavior": enum,
          "nullPacketBitrate": number,
          "patInterval": integer,
          "pcrControl": enum,
```

```
                    "pcrPeriod": integer,
                    "pcrPid": "string",
                    "pmtInterval": integer,
                    "pmtPid": "string",
                    "programNum": integer,
                    "rateMode": enum,
                    "scte27Pids": "string",
                    "scte35Control": enum,
                    "scte35Pid": "string",
                    "segmentationMarkers": enum,
                    "segmentationStyle": enum,
                    "segmentationTime": number,
                    "timedMetadataBehavior": enum,
                    "timedMetadataPid": "string",
                    "transportStreamId": integer,
                    "videoPid": "string"
                  },
                  "rawSettings": {
                  }
                },
                "extension": "string",
                "nameModifier": "string"
              },
              "frameCaptureOutputSettings": {
                "nameModifier": "string"
              },
              "hlsOutputSettings": {
                "h265PackagingType": enum,
                "hlsSettings": {
                  "audioOnlyHlsSettings": {
                    "audioGroupId": "string",
                    "audioOnlyImage": {
                      "passwordParam": "string",
                      "uri": "string",
                      "username": "string"
                    },
                    "audioTrackType": enum,
                    "segmentType": enum
                  },
                  "fmp4HlsSettings": {
                    "audioRenditionSets": "string",
                    "nielsenId3Behavior": enum,
                    "timedMetadataBehavior": enum
                  },
```

```
            "frameCaptureHlsSettings": {
            },
            "standardHlsSettings": {
              "audioRenditionSets": "string",
              "m3u8Settings": {
                "audioFramesPerPes": integer,
                "audioPids": "string",
                "ecmPid": "string",
                "nielsenId3Behavior": enum,
                "patInterval": integer,
                "pcrControl": enum,
                "pcrPeriod": integer,
                "pcrPid": "string",
                "pmtInterval": integer,
                "pmtPid": "string",
                "programNum": integer,
                "scte35Behavior": enum,
                "scte35Pid": "string",
                "timedMetadataBehavior": enum,
                "timedMetadataPid": "string",
                "transportStreamId": integer,
                "videoPid": "string"
              }
            }
          },
          "nameModifier": "string",
          "segmentModifier": "string"
        },
        "mediaPackageOutputSettings": {
        },
        "msSmoothOutputSettings": {
          "h265PackagingType": enum,
          "nameModifier": "string"
        },
        "multiplexOutputSettings": {
          "destination": {
            "destinationRefId": "string"
          }
        },
        "rtmpOutputSettings": {
          "certificateMode": enum,
          "connectionRetryInterval": integer,
          "destination": {
            "destinationRefId": "string"
```

```
        },
        "numRetries": integer
      },
      "udpOutputSettings": {
        "bufferMsec": integer,
        "containerSettings": {
          "m2tsSettings": {
            "absentInputAudioBehavior": enum,
            "arib": enum,
            "aribCaptionsPid": "string",
            "aribCaptionsPidControl": enum,
            "audioBufferModel": enum,
            "audioFramesPerPes": integer,
            "audioPids": "string",
            "audioStreamType": enum,
            "bitrate": integer,
            "bufferModel": enum,
            "ccDescriptor": enum,
            "dvbNitSettings": {
              "networkId": integer,
              "networkName": "string",
              "repInterval": integer
            },
            "dvbSdtSettings": {
              "outputSdt": enum,
              "repInterval": integer,
              "serviceName": "string",
              "serviceProviderName": "string"
            },
            "dvbSubPids": "string",
            "dvbTdtSettings": {
              "repInterval": integer
            },
            "dvbTeletextPid": "string",
            "ebif": enum,
            "ebpAudioInterval": enum,
            "ebpLookaheadMs": integer,
            "ebpPlacement": enum,
            "ecmPid": "string",
            "esRateInPes": enum,
            "etvPlatformPid": "string",
            "etvSignalPid": "string",
            "fragmentTime": number,
            "klv": enum,
```

```
                    "klvDataPids": "string",
                    "nielsenId3Behavior": enum,
                    "nullPacketBitrate": number,
                    "patInterval": integer,
                    "pcrControl": enum,
                    "pcrPeriod": integer,
                    "pcrPid": "string",
                    "pmtInterval": integer,
                    "pmtPid": "string",
                    "programNum": integer,
                    "rateMode": enum,
                    "scte27Pids": "string",
                    "scte35Control": enum,
                    "scte35Pid": "string",
                    "segmentationMarkers": enum,
                    "segmentationStyle": enum,
                    "segmentationTime": number,
                    "timedMetadataBehavior": enum,
                    "timedMetadataPid": "string",
                    "transportStreamId": integer,
                    "videoPid": "string"
                  }
                },
                "destination": {
                  "destinationRefId": "string"
                },
                "fecOutputSettings": {
                  "columnDepth": integer,
                  "includeFec": enum,
                  "rowLength": integer
                }
              }
            },
            "videoDescriptionName": "string"
          }
        ]
      }
    ],
    "timecodeConfig": {
      "source": enum,
      "syncThreshold": integer
    },
    "videoDescriptions": [
      {
```

```
          "codecSettings": {
            "frameCaptureSettings": {
              "captureInterval": integer,
              "captureIntervalUnits": enum
            },
            "h264Settings": {
              "adaptiveQuantization": enum,
              "afdSignaling": enum,
              "bitrate": integer,
              "bufFillPct": integer,
              "bufSize": integer,
              "colorMetadata": enum,
              "colorSpaceSettings": {
                "colorSpacePassthroughSettings": {
                },
                "rec601Settings": {
                },
                "rec709Settings": {
                }
              },
              "entropyEncoding": enum,
              "filterSettings": {
                "temporalFilterSettings": {
                  "postFilterSharpening": enum,
                  "strength": enum
                }
              },
              "fixedAfd": enum,
              "flickerAq": enum,
              "forceFieldPictures": enum,
              "framerateControl": enum,
              "framerateDenominator": integer,
              "framerateNumerator": integer,
              "gopBReference": enum,
              "gopClosedCadence": integer,
              "gopNumBFrames": integer,
              "gopSize": number,
              "gopSizeUnits": enum,
              "level": enum,
              "lookAheadRateControl": enum,
              "maxBitrate": integer,
              "minIInterval": integer,
              "numRefFrames": integer,
              "parControl": enum,
```

```
            "parDenominator": integer,
            "parNumerator": integer,
            "profile": enum,
            "qualityLevel": enum,
            "qvbrQualityLevel": integer,
            "rateControlMode": enum,
            "scanType": enum,
            "sceneChangeDetect": enum,
            "slices": integer,
            "softness": integer,
            "spatialAq": enum,
            "subgopLength": enum,
            "syntax": enum,
            "temporalAq": enum,
            "timecodeInsertion": enum
          },
          "h265Settings": {
            "adaptiveQuantization": enum,
            "afdSignaling": enum,
            "alternativeTransferFunction": enum,
            "bitrate": integer,
            "bufSize": integer,
            "colorMetadata": enum,
            "colorSpaceSettings": {
              "colorSpacePassthroughSettings": {
              },
              "hdr10Settings": {
                "maxCll": integer,
                "maxFall": integer
              },
              "rec601Settings": {
              },
              "rec709Settings": {
              }
            },
            "filterSettings": {
              "temporalFilterSettings": {
                "postFilterSharpening": enum,
                "strength": enum
              }
            },
            "fixedAfd": enum,
            "flickerAq": enum,
            "framerateDenominator": integer,
```

```
            "framerateNumerator": integer,
            "gopClosedCadence": integer,
            "gopSize": number,
            "gopSizeUnits": enum,
            "level": enum,
            "lookAheadRateControl": enum,
            "maxBitrate": integer,
            "minIInterval": integer,
            "parDenominator": integer,
            "parNumerator": integer,
            "profile": enum,
            "qvbrQualityLevel": integer,
            "rateControlMode": enum,
            "scanType": enum,
            "sceneChangeDetect": enum,
            "slices": integer,
            "tier": enum,
            "timecodeInsertion": enum
          },
          "mpeg2Settings": {
            "adaptiveQuantization": enum,
            "afdSignaling": enum,
            "colorMetadata": enum,
            "colorSpace": enum,
            "displayAspectRatio": enum,
            "filterSettings": {
              "temporalFilterSettings": {
                "postFilterSharpening": enum,
                "strength": enum
              }
            },
            "fixedAfd": enum,
            "framerateDenominator": integer,
            "framerateNumerator": integer,
            "gopClosedCadence": integer,
            "gopNumBFrames": integer,
            "gopSize": number,
            "gopSizeUnits": enum,
            "scanType": enum,
            "subgopLength": enum,
            "timecodeInsertion": enum
          }
        },
        "height": integer,
```

```
          "name": "string",
          "respondToAfd": enum,
          "scalingBehavior": enum,
          "sharpness": integer,
          "width": integer
        }
      ]
    },
    "id": "string",
    "inputAttachments": [
      {
        "automaticInputFailoverSettings": {
          "errorClearTimeMsec": integer,
          "failoverConditions": [
            {
              "failoverConditionSettings": {
                "audioSilenceSettings": {
                  "audioSelectorName": "string",
                  "audioSilenceThresholdMsec": integer
                },
                "inputLossSettings": {
                  "inputLossThresholdMsec": integer
                },
                "videoBlackSettings": {
                  "blackDetectThreshold": number,
                  "videoBlackThresholdMsec": integer
                }
              }
            }
          ],
          "inputPreference": enum,
          "secondaryInputId": "string"
        },
        "inputAttachmentName": "string",
        "inputId": "string",
        "inputSettings": {
          "audioSelectors": [
            {
              "name": "string",
              "selectorSettings": {
                "audioHlsRenditionSelection": {
                  "groupId": "string",
                  "name": "string"
                },
```

```
            "audioLanguageSelection": {
              "languageCode": "string",
              "languageSelectionPolicy": enum
            },
            "audioPidSelection": {
              "pid": integer
            },
            "audioTrackSelection": {
              "tracks": [
                {
                  "track": integer
                }
              ]
            }
          }
        }
      ],
      "captionSelectors": [
        {
          "languageCode": "string",
          "name": "string",
          "selectorSettings": {
            "ancillarySourceSettings": {
              "sourceAncillaryChannelNumber": integer
            },
            "aribSourceSettings": {
            },
            "dvbSubSourceSettings": {
              "ocrLanguage": enum,
              "pid": integer
            },
            "embeddedSourceSettings": {
              "convert608To708": enum,
              "scte20Detection": enum,
              "source608ChannelNumber": integer,
              "source608TrackNumber": integer
            },
            "scte20SourceSettings": {
              "convert608To708": enum,
              "source608ChannelNumber": integer
            },
            "scte27SourceSettings": {
              "ocrLanguage": enum,
              "pid": integer
```

```
            },
            "teletextSourceSettings": {
              "outputRectangle": {
                "height": number,
                "leftOffset": number,
                "topOffset": number,
                "width": number
              },
              "pageNumber": "string"
            }
          }
        }
      ],
      "deblockFilter": enum,
      "denoiseFilter": enum,
      "filterStrength": integer,
      "inputFilter": enum,
      "networkInputSettings": {
        "hlsInputSettings": {
          "bandwidth": integer,
          "bufferSegments": integer,
          "retries": integer,
          "retryInterval": integer,
          "scte35Source": enum
        },
        "serverValidation": enum
      },
      "smpte2038DataPreference": enum,
      "sourceEndBehavior": enum,
      "videoSelector": {
        "colorSpace": enum,
        "colorSpaceSettings": {
          "hdr10Settings": {
            "maxCll": integer,
            "maxFall": integer
          }
        },
        "colorSpaceUsage": enum,
        "selectorSettings": {
          "videoSelectorPid": {
            "pid": integer
          },
          "videoSelectorProgramId": {
            "programId": integer
```

```
            }
          }
        }
      }
    }
  ],
  "inputSpecification": {
    "codec": enum,
    "maximumBitrate": enum,
    "resolution": enum
  },
  "logLevel": enum,
  "maintenance": {
    "maintenanceDay": enum,
    "maintenanceDeadline": "string",
    "maintenanceScheduledDate": "string",
    "maintenanceStartTime": "string"
  },
  "name": "string",
  "pipelineDetails": [
    {
      "activeInputAttachmentName": "string",
      "activeInputSwitchActionName": "string",
      "activeMotionGraphicsActionName": "string",
      "activeMotionGraphicsUri": "string",
      "pipelineId": "string"
    }
  ],
  "pipelinesRunningCount": integer,
  "roleArn": "string",
  "state": enum,
  "tags": {
  },
  "vpc": {
    "availabilityZones": [
      "string"
    ],
    "networkInterfaceIds": [
      "string"
    ],
    "securityGroupIds": [
      "string"
    ],
    "subnetIds": [
```

```
        "string"
      ]
    }
  }
}
```

InvalidRequest schema

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

AccessDenied schema

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

ResourceNotFound schema

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

ResourceConflict schema

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

ChannelConfigurationValidationError schema

```
{
  "message": "string",
  "validationErrors": [
    {
```

```
      "elementPath": "string",
      "errorMessage": "string"
    }
  ]
}
```

LimitExceeded schema

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

InternalServiceError schema

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

BadGatewayException schema

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

GatewayTimeoutException schema

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

Properties

AacCodingMode

Aac Coding Mode

```
AD_RECEIVER_MIX
```

CODING_MODE_1_0
CODING_MODE_1_1
CODING_MODE_2_0
CODING_MODE_5_1

AacInputType

Aac Input Type

BROADCASTER_MIXED_AD
NORMAL

AacProfile

Aac Profile

HEV1
HEV2
LC

AacRateControlMode

Aac Rate Control Mode

CBR
VBR

AacRawFormat

Aac Raw Format

LATM_LOAS
NONE

AacSettings

Aac Settings

bitrate

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

Type: number
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: AacCodingMode
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: AacInputType
Required: False

profile

AAC Profile.

Type: AacProfile
Required: False

rateControlMode

Rate Control Mode.

Type: AacRateControlMode
Required: False

rawFormat

Sets LATM / LOAS AAC output for raw containers.

Type: AacRawFormat
Required: False

sampleRate

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

Type: number
Required: False

spec

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

Type: AacSpec
Required: False

vbrQuality

VBR Quality Level - Only used if rateControlMode is VBR.

Type: AacVbrQuality
Required: False

AacSpec

Aac Spec

MPEG2
MPEG4

AacVbrQuality

Aac Vbr Quality

HIGH
LOW
MEDIUM_HIGH
MEDIUM_LOW

Ac3BitstreamMode

Ac3 Bitstream Mode

COMMENTARY
COMPLETE_MAIN
DIALOGUE
EMERGENCY
HEARING_IMPAIRED
MUSIC_AND_EFFECTS
VISUALLY_IMPAIRED
VOICE_OVER

Ac3CodingMode

Ac3 Coding Mode

CODING_MODE_1_0
CODING_MODE_1_1
CODING_MODE_2_0
CODING_MODE_3_2_LFE

Ac3DrcProfile

Ac3 Drc Profile

FILM_STANDARD

NONE

Ac3LfeFilter

Ac3 Lfe Filter

DISABLED
ENABLED

Ac3MetadataControl

Ac3 Metadata Control

FOLLOW_INPUT
USE_CONFIGURED

Ac3Settings

Ac3 Settings

bitrate

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

Type: number
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: Ac3BitstreamMode
Required: False

codingMode

Dolby Digital coding mode. Determines number of channels.

Type: Ac3CodingMode
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

drcProfile

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

Type: Ac3DrcProfile
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: Ac3LfeFilter
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: Ac3MetadataControl
Required: False

AccessDenied

message

Type: string
Required: False

AfdSignaling

Afd Signaling

AUTO
FIXED
NONE

AncillarySourceSettings

Ancillary Source Settings

sourceAncillaryChannelNumber

Specifies the number (1 to 4) of the captions channel you want to extract from the ancillary captions. If you plan to convert the ancillary captions to another format, complete this field. If you plan to choose Embedded as the captions destination in the output (to pass through all the channels in the ancillary captions), leave this field blank because MediaLive ignores the field.

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

ArchiveCdnSettings

Archive Cdn Settings

archiveS3Settings

Type: ArchiveS3Settings

Required: False

ArchiveContainerSettings

Archive Container Settings

m2tsSettings

Type: M2tsSettings
Required: False

rawSettings

Type: RawSettings
Required: False

ArchiveGroupSettings

Archive Group Settings

archiveCdnSettings

Parameters that control interactions with the CDN.

Type: ArchiveCdnSettings
Required: False

destination

A directory and base filename where archive files should be written.

Type: OutputLocationRef
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

Archive Output Settings

containerSettings

Settings specific to the container type of the file.

Type: ArchiveContainerSettings
Required: True

extension

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

Type: string
Required: False

nameModifier

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

Type: string
Required: False

ArchiveS3LogUploads

Archive S3 Log Uploads

```
DISABLED
ENABLED
```

ArchiveS3Settings

Archive S3 Settings

cannedAcl

Specify the canned ACL to apply to each S3 request. Defaults to none.

Type: S3CannedAcl
Required: False

logUploads

When set to enabled, each upload to CDN or server will be logged.

Type: ArchiveS3LogUploads
Required: False

AribDestinationSettings

Arib Destination Settings

AribSourceSettings

Arib Source Settings

AudioChannelMapping

Audio Channel Mapping

inputChannelLevels

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

Type: Array of type InputChannelLevel
Required: True

outputChannel

The index of the output channel being produced.

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

AudioCodecSettings

Audio Codec Settings

aacSettings

Type: AacSettings
Required: False

ac3Settings

Type: Ac3Settings
Required: False

eac3Settings

Type: Eac3Settings
Required: False

mp2Settings

Type: Mp2Settings
Required: False

passThroughSettings

Type: PassThroughSettings
Required: False

wavSettings

Type: WavSettings
Required: False

AudioDescription

Audio Description

audioNormalizationSettings

Advanced audio normalization settings.

Type: AudioNormalizationSettings
Required: False

audioSelectorName

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

Type: string
Required: True

audioType

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

Type: AudioType
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: AudioDescriptionAudioTypeControl
Required: False

audioWatermarkingSettings

Settings to configure one or more solutions that insert audio watermarks in the audio encode

Type: AudioWatermarkSettings
Required: False

codecSettings

Audio codec settings.

Type: AudioCodecSettings
Required: False

languageCode

RFC 5646 language code representing 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
MinLength: 1
MaxLength: 35

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: AudioDescriptionLanguageCodeControl
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

remixSettings

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

Type: RemixSettings

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

AudioDescriptionAudioTypeControl

Audio Description Audio Type Control

`FOLLOW_INPUT`

`USE_CONFIGURED`

AudioDescriptionLanguageCodeControl

Audio Description Language Code Control

`FOLLOW_INPUT`

`USE_CONFIGURED`

AudioHlsRenditionSelection

Audio Hls Rendition Selection

groupId

Specifies the GROUP-ID in the #EXT-X-MEDIA tag of the target HLS audio rendition.

Type: string

Required: True

MinLength: 1

name

Specifies the NAME in the #EXT-X-MEDIA tag of the target HLS audio rendition.

Type: string
Required: True
MinLength: 1

AudioLanguageSelection

Audio Language Selection

languageCode

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

Type: string
Required: True

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: AudioLanguageSelectionPolicy
Required: False

AudioLanguageSelectionPolicy

Audio Language Selection Policy

LOOSE
STRICT

AudioNormalizationAlgorithm

Audio Normalization Algorithm

```
ITU_1770_1
ITU_1770_2
```

AudioNormalizationAlgorithmControl

Audio Normalization Algorithm Control

```
CORRECT_AUDIO
```

AudioNormalizationSettings

Audio Normalization Settings

algorithm

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

Type: AudioNormalizationAlgorithm
Required: False

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: AudioNormalizationAlgorithmControl
Required: False

targetLkfs

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

Type: number
Required: False
Minimum: -59
Maximum: 0

AudioOnlyHlsSegmentType

Audio Only Hls Segment Type

AAC
FMP4

AudioOnlyHlsSettings

Audio Only Hls Settings

audioGroupId

Specifies the group to which the audio Rendition belongs.

Type: string
Required: False

audioOnlyImage

Optional. Specifies the .jpg or .png image to use as the cover art for an audio-only output. We recommend a low bit-size file because the image increases the output audio bandwidth. The image is attached to the audio as an ID3 tag, frame type APIC, picture type 0x10, as per the "ID3 tag version 2.4.0 - Native Frames" standard.

Type: InputLocation
Required: False

audioTrackType

Four types of audio-only tracks are supported: Audio-Only Variant Stream The client can play back this audio-only stream instead of video in low-bandwidth scenarios. Represented as an EXT-X-STREAM-INF in the HLS manifest. Alternate Audio, Auto Select, Default Alternate rendition that

the client should try to play back by default. Represented as an EXT-X-MEDIA in the HLS manifest with DEFAULT=YES, AUTOSELECT=YES Alternate Audio, Auto Select, Not Default Alternate rendition that the client may try to play back by default. Represented as an EXT-X-MEDIA in the HLS manifest with DEFAULT=NO, AUTOSELECT=YES Alternate Audio, not Auto Select Alternate rendition that the client will not try to play back by default. Represented as an EXT-X-MEDIA in the HLS manifest with DEFAULT=NO, AUTOSELECT=NO

Type: AudioOnlyHlsTrackType
Required: False

segmentType

Specifies the segment type.

Type: AudioOnlyHlsSegmentType
Required: False

AudioOnlyHlsTrackType

Audio Only Hls Track Type

```
ALTERNATE_AUDIO_AUTO_SELECT
ALTERNATE_AUDIO_AUTO_SELECT_DEFAULT
ALTERNATE_AUDIO_NOT_AUTO_SELECT
AUDIO_ONLY_VARIANT_STREAM
```

AudioPidSelection

Audio Pid Selection

pid

Selects a specific PID from within a source.

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

AudioSelector

Audio Selector

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
MinLength: 1

selectorSettings

The audio selector settings.

Type: AudioSelectorSettings
Required: False

AudioSelectorSettings

Audio Selector Settings

audioHlsRenditionSelection

Type: AudioHlsRenditionSelection
Required: False

audioLanguageSelection

Type: AudioLanguageSelection
Required: False

audioPidSelection

Type: AudioPidSelection
Required: False

audioTrackSelection

Type: AudioTrackSelection
Required: False

AudioSilenceFailoverSettings

audioSelectorName

The name of the audio selector in the input that MediaLive should monitor to detect silence. Select your most important rendition. If you didn't create an audio selector in this input, leave blank.

Type: string
Required: True

audioSilenceThresholdMsec

The amount of time (in milliseconds) that the active input must be silent before automatic input failover occurs. Silence is defined as audio loss or audio quieter than -50 dBFS.

Type: integer
Required: False
Minimum: 1000

AudioTrack

Audio Track

track

1-based integer value that maps to a specific audio track

Type: integer
Required: True
Minimum: 1

AudioTrackSelection

Audio Track Selection

tracks

Selects one or more unique audio tracks from within a source.

Type: Array of type AudioTrack
Required: True

AudioType

Audio Type

CLEAN_EFFECTS
HEARING_IMPAIRED
UNDEFINED
VISUAL_IMPAIRED_COMMENTARY

AudioWatermarkSettings

Audio Watermark Settings

nielsenWatermarksSettings

Settings to configure Nielsen Watermarks in the audio encode

Type: NielsenWatermarksSettings
Required: False

AuthenticationScheme

Authentication Scheme

AKAMAI
COMMON

AutomaticInputFailoverSettings

The settings for Automatic Input Failover.

errorClearTimeMsec

This clear time defines the requirement a recovered input must meet to be considered healthy. The input must have no failover conditions for this length of time. Enter a time in milliseconds. This value is particularly important if the input_preference for the failover pair is set to PRIMARY_INPUT_PREFERRED, because after this time, MediaLive will switch back to the primary input.

Type: integer
Required: False
Minimum: 1

failoverConditions

A list of failover conditions. If any of these conditions occur, MediaLive will perform a failover to the other input.

Type: Array of type FailoverCondition
Required: False

inputPreference

Input preference when deciding which input to make active when a previously failed input has recovered.

Type: InputPreference
Required: False

secondaryInputId

The input ID of the secondary input in the automatic input failover pair.

Type: string
Required: True

AvailBlanking

Avail Blanking

availBlankingImage

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

Type: InputLocation
Required: False

state

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

Type: AvailBlankingState
Required: False

AvailBlankingState

Avail Blanking State

```
DISABLED
ENABLED
```

AvailConfiguration

Avail Configuration

availSettings

Ad avail settings.

Type: AvailSettings
Required: False

AvailSettings

Avail Settings

scte35SpliceInsert

Type: Scte35SpliceInsert

Required: False

scte35TimeSignalApos

Type: Scte35TimeSignalApos
Required: False

BadGatewayException

message

Type: string
Required: False

BlackoutSlate

Blackout Slate

blackoutSlateImage

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

Type: InputLocation
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: BlackoutSlateNetworkEndBlackout
Required: False

networkEndBlackoutImage

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

Type: InputLocation
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
MinLength: 34
MaxLength: 34

state

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

Type: BlackoutSlateState
Required: False

BlackoutSlateNetworkEndBlackout

Blackout Slate Network End Blackout

```
DISABLED
ENABLED
```

BlackoutSlateState

Blackout Slate State

```
DISABLED
ENABLED
```

BurnInAlignment

Burn In Alignment

```
CENTERED
LEFT
SMART
```

BurnInBackgroundColor

Burn In Background Color

```
BLACK
NONE
WHITE
```

BurnInDestinationSettings

Burn In Destination Settings

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: BurnInAlignment
Required: False

backgroundColor

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

Type: BurnInBackgroundColor

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

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
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: BurnInFontColor
Required: False

fontOpacity

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

Type: integer

Required: False
Minimum: 0
Maximum: 255

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

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

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: BurnInOutlineColor
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

shadowColor

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

Type: BurnInShadowColor
Required: False

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

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

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

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: BurnInTeletextGridControl
Required: False

xPosition

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

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

BurnInFontColor

Burn In Font Color

BLACK
BLUE
GREEN
RED

WHITE
YELLOW

BurnInOutlineColor

Burn In Outline Color

BLACK
BLUE
GREEN
RED
WHITE
YELLOW

BurnInShadowColor

Burn In Shadow Color

BLACK
NONE
WHITE

BurnInTeletextGridControl

Burn In Teletext Grid Control

FIXED
SCALED

CaptionDescription

Caption Description

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

destinationSettings

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

Type: CaptionDestinationSettings
Required: False

languageCode

ISO 639-2 three-digit code: http://www.loc.gov/standards/iso639-2/

Type: string
Required: False

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

CaptionDestinationSettings

Caption Destination Settings

aribDestinationSettings

Type: AribDestinationSettings

Required: False

burnInDestinationSettings

Type: BurnInDestinationSettings
Required: False

dvbSubDestinationSettings

Type: DvbSubDestinationSettings
Required: False

ebuTtDDestinationSettings

Type: EbuTtDDestinationSettings
Required: False

embeddedDestinationSettings

Type: EmbeddedDestinationSettings
Required: False

embeddedPlusScte20DestinationSettings

Type: EmbeddedPlusScte20DestinationSettings
Required: False

rtmpCaptionInfoDestinationSettings

Type: RtmpCaptionInfoDestinationSettings
Required: False

scte20PlusEmbeddedDestinationSettings

Type: Scte20PlusEmbeddedDestinationSettings
Required: False

scte27DestinationSettings

Type: Scte27DestinationSettings
Required: False

smpteTtDestinationSettings

Type: SmpteTtDestinationSettings
Required: False

teletextDestinationSettings

Type: TeletextDestinationSettings
Required: False

ttmlDestinationSettings

Type: TtmlDestinationSettings
Required: False

webvttDestinationSettings

Type: WebvttDestinationSettings
Required: False

CaptionLanguageMapping

Maps a caption channel to an ISO 693-2 language code (http://www.loc.gov/standards/iso639-2), with an optional description.

captionChannel

The closed caption channel being described by this CaptionLanguageMapping. 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
MinLength: 3
MaxLength: 3

languageDescription

Textual description of language

Type: string
Required: True
MinLength: 1

CaptionRectangle

Caption Rectangle

height

See the description in leftOffset. For height, specify the entire height of the rectangle as a percentage of the underlying frame height. For example, \"80\" means the rectangle height is 80% of the underlying frame height. The topOffset and rectangleHeight must add up to 100% or less. This field corresponds to tts:extent - Y in the TTML standard.

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

leftOffset

Applies only if you plan to convert these source captions to EBU-TT-D or TTML in an output. (Make sure to leave the default if you don't have either of these formats in the output.) You can

define a display rectangle for the captions that is smaller than the underlying video frame. You define the rectangle by specifying the position of the left edge, top edge, bottom edge, and right edge of the rectangle, all within the underlying video frame. The units for the measurements are percentages. If you specify a value for one of these fields, you must specify a value for all of them. For leftOffset, specify the position of the left edge of the rectangle, as a percentage of the underlying frame width, and relative to the left edge of the frame. For example, \"10\" means the measurement is 10% of the underlying frame width. The rectangle left edge starts at that position from the left edge of the frame. This field corresponds to tts:origin - X in the TTML standard.

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

topOffset

See the description in leftOffset. For topOffset, specify the position of the top edge of the rectangle, as a percentage of the underlying frame height, and relative to the top edge of the frame. For example, \"10\" means the measurement is 10% of the underlying frame height. The rectangle top edge starts at that position from the top edge of the frame. This field corresponds to tts:origin - Y in the TTML standard.

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

width

See the description in leftOffset. For width, specify the entire width of the rectangle as a percentage of the underlying frame width. For example, \"80\" means the rectangle width is 80% of the underlying frame width. The leftOffset and rectangleWidth must add up to 100% or less. This field corresponds to tts:extent - X in the TTML standard.

Type: number
Required: True
Minimum: 0

Maximum: 100

CaptionSelector

Output groups for this Live Event. Output groups contain information about where streams should be distributed.

languageCode

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

Type: string
Required: False

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
MinLength: 1

selectorSettings

Caption selector settings.

Type: CaptionSelectorSettings
Required: False

CaptionSelectorSettings

Caption Selector Settings

ancillarySourceSettings

Type: AncillarySourceSettings

Required: False

aribSourceSettings

Type: AribSourceSettings
Required: False

dvbSubSourceSettings

Type: DvbSubSourceSettings
Required: False

embeddedSourceSettings

Type: EmbeddedSourceSettings
Required: False

scte20SourceSettings

Type: Scte20SourceSettings
Required: False

scte27SourceSettings

Type: Scte27SourceSettings
Required: False

teletextSourceSettings

Type: TeletextSourceSettings
Required: False

CdiInputResolution

Maximum CDI input resolution; SD is 480i and 576i up to 30 frames-per-second (fps), HD is 720p up to 60 fps / 1080i up to 30 fps, FHD is 1080p up to 60 fps, UHD is 2160p up to 60 fps

SD

HD

FHD

UHD

CdiInputSpecification

resolution

Maximum CDI input resolution

Type: CdiInputResolution
Required: False

Channel

arn

The unique arn of the channel.

Type: string
Required: False

cdiInputSpecification

Specification of CDI inputs for this channel

Type: CdiInputSpecification
Required: False

channelClass

The class for this channel. STANDARD for a channel with two pipelines or SINGLE_PIPELINE for a channel with one pipeline.

Type: ChannelClass
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
Required: False

egressEndpoints

The endpoints where outgoing connections initiate from

Type: Array of type ChannelEgressEndpoint
Required: False

encoderSettings

Type: EncoderSettings
Required: False

id

The unique ID of the channel.

Type: string
Required: False

inputAttachments

List of input attachments for channel.

Type: Array of type InputAttachment
Required: False

inputSpecification

Specification of network and file inputs for this channel

Type: InputSpecification

Required: False

logLevel

The log level being written to CloudWatch Logs.

Type: LogLevel
Required: False

maintenance

Maintenance settings for this channel.

Type: MaintenanceStatus
Required: False

name

The name of the channel. (user-mutable)

Type: string
Required: False

pipelineDetails

Runtime details for the pipelines of a running channel.

Type: Array of type PipelineDetail
Required: False

pipelinesRunningCount

The number of currently healthy pipelines.

Type: integer
Required: False

roleArn

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

Type: string
Required: False

state

Type: ChannelState
Required: False

tags

A collection of key-value pairs.

Type: Tags
Required: False

vpc

Settings for VPC output

Type: VpcOutputSettingsDescription
Required: False

ChannelClass

A standard channel has two encoding pipelines and a single pipeline channel only has one.

```
STANDARD
SINGLE_PIPELINE
```

ChannelConfigurationValidationError

message

Type: string
Required: False

validationErrors

A collection of validation error responses.

Type: Array of type ValidationError
Required: False

ChannelEgressEndpoint

sourceIp

Public IP of where a channel's output comes from

Type: string
Required: False

ChannelState

```
CREATING
CREATE_FAILED
IDLE
STARTING
RUNNING
RECOVERING
STOPPING
DELETING
DELETED
UPDATING
UPDATE_FAILED
```

ColorSpacePassthroughSettings

Passthrough applies no color space conversion to the output

DvbNitSettings

DVB Network Information Table (NIT)

networkId

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

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

networkName

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

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

repInterval

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

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

DvbSdtOutputSdt

Dvb Sdt Output Sdt

SDT_FOLLOW
SDT_FOLLOW_IF_PRESENT
SDT_MANUAL
SDT_NONE

DvbSdtSettings

DVB Service Description Table (SDT)

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: DvbSdtOutputSdt
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

serviceName

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

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

serviceProviderName

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

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

DvbSubDestinationAlignment

Dvb Sub Destination Alignment

CENTERED
LEFT
SMART

DvbSubDestinationBackgroundColor

Dvb Sub Destination Background Color

BLACK
NONE
WHITE

DvbSubDestinationFontColor

Dvb Sub Destination Font Color

BLACK
BLUE
GREEN
RED
WHITE
YELLOW

DvbSubDestinationOutlineColor

Dvb Sub Destination Outline Color

BLACK

BLUE
GREEN
RED
WHITE
YELLOW

DvbSubDestinationSettings

Dvb Sub Destination Settings

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: DvbSubDestinationAlignment
Required: False

backgroundColor

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

Type: DvbSubDestinationBackgroundColor
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

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
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: DvbSubDestinationFontColor
Required: False

fontOpacity

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

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

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

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

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: DvbSubDestinationOutlineColor
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

shadowColor

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

Type: DvbSubDestinationShadowColor
Required: False

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

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

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

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: DvbSubDestinationTeletextGridControl
Required: False

xPosition

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

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

DvbSubDestinationShadowColor

Dvb Sub Destination Shadow Color

```
BLACK
NONE
WHITE
```

DvbSubDestinationTeletextGridControl

Dvb Sub Destination Teletext Grid Control

```
FIXED
```

SCALED

DvbSubOcrLanguage

Dvb Sub Ocr Language

DEU
ENG
FRA
NLD
POR
SPA

DvbSubSourceSettings

Dvb Sub Source Settings

ocrLanguage

If you will configure a WebVTT caption description that references this caption selector, use this field to provide the language to consider when translating the image-based source to text.

Type: DvbSubOcrLanguage
Required: False

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

DVB Time and Date Table (SDT)

repInterval

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

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

Eac3AttenuationControl

Eac3 Attenuation Control

ATTENUATE_3_DB
NONE

Eac3BitstreamMode

Eac3 Bitstream Mode

COMMENTARY
COMPLETE_MAIN
EMERGENCY
HEARING_IMPAIRED
VISUALLY_IMPAIRED

Eac3CodingMode

Eac3 Coding Mode

CODING_MODE_1_0
CODING_MODE_2_0
CODING_MODE_3_2

Eac3DcFilter

Eac3 Dc Filter

DISABLED
ENABLED

Eac3DrcLine

Eac3 Drc Line

FILM_LIGHT
FILM_STANDARD
MUSIC_LIGHT
MUSIC_STANDARD
NONE
SPEECH

Eac3DrcRf

Eac3 Drc Rf

FILM_LIGHT
FILM_STANDARD
MUSIC_LIGHT
MUSIC_STANDARD
NONE
SPEECH

Eac3LfeControl

Eac3 Lfe Control

LFE
NO_LFE

Eac3LfeFilter

Eac3 Lfe Filter

DISABLED

ENABLED

Eac3MetadataControl

Eac3 Metadata Control

FOLLOW_INPUT
USE_CONFIGURED

Eac3PassthroughControl

Eac3 Passthrough Control

NO_PASSTHROUGH
WHEN_POSSIBLE

Eac3PhaseControl

Eac3 Phase Control

NO_SHIFT
SHIFT_90_DEGREES

Eac3Settings

Eac3 Settings

attenuationControl

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

Type: Eac3AttenuationControl
Required: False

bitrate

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

Type: number
Required: False

bitstreamMode

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

Type: Eac3BitstreamMode
Required: False

codingMode

Dolby Digital Plus coding mode. Determines number of channels.

Type: Eac3CodingMode
Required: False

dcFilter

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

Type: Eac3DcFilter
Required: False

dialnorm

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

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

drcLine

Sets the Dolby dynamic range compression profile.

Type: Eac3DrcLine
Required: False

drcRf

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

Type: Eac3DrcRf
Required: False

lfeControl

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

Type: Eac3LfeControl
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: Eac3LfeFilter
Required: False

loRoCenterMixLevel

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

Type: number
Required: False

loRoSurroundMixLevel

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

Type: number
Required: False

ltRtCenterMixLevel

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

Type: number
Required: False

ltRtSurroundMixLevel

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

Type: number
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: Eac3MetadataControl
Required: False

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: Eac3PassthroughControl
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: Eac3PhaseControl

Required: False

stereoDownmix

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

Type: Eac3StereoDownmix
Required: False

surroundExMode

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

Type: Eac3SurroundExMode
Required: False

surroundMode

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

Type: Eac3SurroundMode
Required: False

Eac3StereoDownmix

Eac3 Stereo Downmix

```
DPL2
LO_RO
LT_RT
NOT_INDICATED
```

Eac3SurroundExMode

Eac3 Surround Ex Mode

```
DISABLED
```

ENABLED
NOT_INDICATED

Eac3SurroundMode

Eac3 Surround Mode

DISABLED
ENABLED
NOT_INDICATED

EbuTtDDestinationSettings

Ebu Tt DDestination Settings

copyrightHolder

Applies only if you plan to convert these source captions to EBU-TT-D or TTML in an output. Complete this field if you want to include the name of the copyright holder in the copyright metadata tag in the TTML

Type: string
Required: False
MaxLength: 1000

fillLineGap

Specifies how to handle the gap between the lines (in multi-line captions). - enabled: Fill with the captions background color (as specified in the input captions). - disabled: Leave the gap unfilled.

Type: EbuTtDFillLineGapControl
Required: False

fontFamily

Specifies the font family to include in the font data attached to the EBU-TT captions. Valid only if styleControl is set to include. If you leave this field empty, the font family is set to "monospaced". (If styleControl is set to exclude, the font family is always set to "monospaced".) You specify only

the font family. All other style information (color, bold, position and so on) is copied from the input captions. The size is always set to 100% to allow the downstream player to choose the size. - Enter a list of font families, as a comma-separated list of font names, in order of preference. The name can be a font family (such as “Arial”), or a generic font family (such as “serif”), or “default” (to let the downstream player choose the font). - Leave blank to set the family to “monospace”.

Type: string
Required: False

styleControl

Specifies the style information (font color, font position, and so on) to include in the font data that is attached to the EBU-TT captions. - include: Take the style information (font color, font position, and so on) from the source captions and include that information in the font data attached to the EBU-TT captions. This option is valid only if the source captions are Embedded or Teletext. - exclude: In the font data attached to the EBU-TT captions, set the font family to "monospaced". Do not include any other style information.

Type: EbuTtDDestinationStyleControl
Required: False

EbuTtDDestinationStyleControl

Ebu Tt DDestination Style Control

```
EXCLUDE
INCLUDE
```

EbuTtDFillLineGapControl

Ebu Tt DFill Line Gap Control

```
DISABLED
ENABLED
```

EmbeddedConvert608To708

Embedded Convert608 To708

DISABLED

UPCONVERT

EmbeddedDestinationSettings

Embedded Destination Settings

EmbeddedPlusScte20DestinationSettings

Embedded Plus Scte20 Destination Settings

EmbeddedScte20Detection

Embedded Scte20 Detection

AUTO

OFF

EmbeddedSourceSettings

Embedded Source Settings

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: EmbeddedConvert608To708

Required: False

scte20Detection

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

Type: EmbeddedScte20Detection

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

source608TrackNumber

This field is unused and deprecated.

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

EncoderSettings

Encoder Settings

audioDescriptions

Type: Array of type AudioDescription
Required: True

availBlanking

Settings for ad avail blanking.

Type: AvailBlanking
Required: False

availConfiguration

Event-wide configuration settings for ad avail insertion.

Type: AvailConfiguration
Required: False

blackoutSlate

Settings for blackout slate.

Type: BlackoutSlate
Required: False

captionDescriptions

Settings for caption decriptions

Type: Array of type CaptionDescription
Required: False

featureActivations

Feature Activations

Type: FeatureActivations
Required: False

globalConfiguration

Configuration settings that apply to the event as a whole.

Type: GlobalConfiguration
Required: False

motionGraphicsConfiguration

Settings for motion graphics.

Type: MotionGraphicsConfiguration
Required: False

nielsenConfiguration

Nielsen configuration settings.

Type: NielsenConfiguration
Required: False

outputGroups

Type: Array of type OutputGroup
Required: True

timecodeConfig

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

Type: TimecodeConfig
Required: True

videoDescriptions

Type: Array of type VideoDescription
Required: True

FailoverCondition

Failover Condition settings. There can be multiple failover conditions inside AutomaticInputFailoverSettings.

failoverConditionSettings

Failover condition type-specific settings.

Type: FailoverConditionSettings
Required: False

FailoverConditionSettings

Settings for one failover condition.

audioSilenceSettings

MediaLive will perform a failover if the specified audio selector is silent for the specified period.

Type: AudioSilenceFailoverSettings
Required: False

inputLossSettings

MediaLive will perform a failover if content is not detected in this input for the specified period.

Type: InputLossFailoverSettings
Required: False

videoBlackSettings

MediaLive will perform a failover if content is considered black for the specified period.

Type: VideoBlackFailoverSettings
Required: False

FeatureActivations

Feature Activations

inputPrepareScheduleActions

Enables the Input Prepare feature. You can create Input Prepare actions in the schedule only if this feature is enabled. If you disable the feature on an existing schedule, make sure that you first delete all input prepare actions from the schedule.

Type: FeatureActivationsInputPrepareScheduleActions
Required: False

FeatureActivationsInputPrepareScheduleActions

Feature Activations Input Prepare Schedule Actions

```
DISABLED
ENABLED
```

FecOutputIncludeFec

Fec Output Include Fec

COLUMN
COLUMN_AND_ROW

FecOutputSettings

Fec Output Settings

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: FecOutputIncludeFec
Required: False

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

FixedAfd

Fixed Afd

AFD_0000
AFD_0010
AFD_0011
AFD_0100
AFD_1000
AFD_1001
AFD_1010
AFD_1011
AFD_1101
AFD_1110
AFD_1111

Fmp4HlsSettings

Fmp4 Hls Settings

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

nielsenId3Behavior

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

Type: Fmp4NielsenId3Behavior
Required: False

timedMetadataBehavior

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

Type: Fmp4TimedMetadataBehavior
Required: False

Fmp4NielsenId3Behavior

Fmp4 Nielsen Id3 Behavior

NO_PASSTHROUGH
PASSTHROUGH

Fmp4TimedMetadataBehavior

Fmp4 Timed Metadata Behavior

NO_PASSTHROUGH
PASSTHROUGH

FrameCaptureCdnSettings

Frame Capture Cdn Settings

frameCaptureS3Settings

Type: FrameCaptureS3Settings
Required: False

FrameCaptureGroupSettings

Frame Capture Group Settings

destination

The destination for the frame capture files. Either the URI for an Amazon S3 bucket and object, plus a file name prefix (for example, s3ssl://sportsDelivery/highlights/20180820/curling-) or the URI for a MediaStore container, plus a file name prefix (for example, mediastoressl://

sportsDelivery/20180820/curling-). The final file names consist of the prefix from the destination field (for example, "curling-") + name modifier + the counter (5 digits, starting from 00001) + extension (which is always .jpg). For example, curling-low.00001.jpg

Type: OutputLocationRef
Required: True

frameCaptureCdnSettings

Parameters that control interactions with the CDN.

Type: FrameCaptureCdnSettings
Required: False

FrameCaptureHlsSettings

Frame Capture Hls Settings

FrameCaptureIntervalUnit

Frame Capture Interval Unit

```
MILLISECONDS
SECONDS
```

FrameCaptureOutputSettings

Frame Capture Output Settings

nameModifier

Required if the output group contains more than one output. This modifier forms part of the output file name.

Type: string
Required: False

FrameCaptureS3LogUploads

Frame Capture S3 Log Uploads

DISABLED
ENABLED

FrameCaptureS3Settings

Frame Capture S3 Settings

cannedAcl

Specify the canned ACL to apply to each S3 request. Defaults to none.

Type: S3CannedAcl
Required: False

logUploads

When set to enabled, each upload to CDN or server will be logged.

Type: FrameCaptureS3LogUploads
Required: False

FrameCaptureSettings

Frame Capture Settings

captureInterval

The frequency at which to capture frames for inclusion in the output. May be specified in either seconds or milliseconds, as specified by captureIntervalUnits.

Type: integer
Required: False
Minimum: 1
Maximum: 3600000

captureIntervalUnits

Unit for the frame capture interval.

Type: FrameCaptureIntervalUnit

Required: False

GatewayTimeoutException

message

Type: string
Required: False

GlobalConfiguration

Global Configuration

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 the current input completes (e.g. end-of-file). When switchAndLoopInputs is configured the encoder will restart at the beginning of the first input. When "none" is configured the encoder will transcode either black, a solid color, or a user specified slate images per the "Input Loss Behavior" configuration until the next input switch occurs (which is controlled through the Channel Schedule API).

Type: GlobalConfigurationInputEndAction
Required: False

inputLossBehavior

Settings for system actions when input is lost.

Type: InputLossBehavior
Required: False

outputLockingMode

Indicates how MediaLive pipelines are synchronized. PIPELINE_LOCKING - MediaLive will attempt to synchronize the output of each pipeline to the other. EPOCH_LOCKING - MediaLive will attempt to synchronize the output of each pipeline to the Unix epoch.

Type: GlobalConfigurationOutputLockingMode
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: GlobalConfigurationOutputTimingSource
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: GlobalConfigurationLowFramerateInputs
Required: False

GlobalConfigurationInputEndAction

Global Configuration Input End Action

```
NONE
SWITCH_AND_LOOP_INPUTS
```

GlobalConfigurationLowFramerateInputs

Global Configuration Low Framerate Inputs

```
DISABLED
ENABLED
```

GlobalConfigurationOutputLockingMode

Global Configuration Output Locking Mode

EPOCH_LOCKING
PIPELINE_LOCKING

GlobalConfigurationOutputTimingSource

Global Configuration Output Timing Source

INPUT_CLOCK
SYSTEM_CLOCK

H264AdaptiveQuantization

H264 Adaptive Quantization

AUTO
HIGH
HIGHER
LOW
MAX
MEDIUM
OFF

H264ColorMetadata

H264 Color Metadata

IGNORE
INSERT

H264ColorSpaceSettings

H264 Color Space Settings

colorSpacePassthroughSettings

Type: ColorSpacePassthroughSettings
Required: False

rec601Settings

Type: Rec601Settings
Required: False

rec709Settings

Type: Rec709Settings
Required: False

H264EntropyEncoding

H264 Entropy Encoding

CABAC
CAVLC

H264FilterSettings

H264 Filter Settings

temporalFilterSettings

Type: TemporalFilterSettings
Required: False

H264FlickerAq

H264 Flicker Aq

DISABLED
ENABLED

H264ForceFieldPictures

H264 Force Field Pictures

DISABLED
ENABLED

H264FramerateControl

H264 Framerate Control

INITIALIZE_FROM_SOURCE
SPECIFIED

H264GopBReference

H264 Gop BReference

DISABLED
ENABLED

H264GopSizeUnits

H264 Gop Size Units

FRAMES
SECONDS

H264Level

H264 Level

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

H264 Look Ahead Rate Control

HIGH
LOW
MEDIUM

H264ParControl

H264 Par Control

INITIALIZE_FROM_SOURCE
SPECIFIED

H264Profile

H264 Profile

BASELINE
HIGH
HIGH_10BIT
HIGH_422

HIGH_422_10BIT

MAIN

H264QualityLevel

H264 Quality Level

ENHANCED_QUALITY

STANDARD_QUALITY

H264RateControlMode

H264 Rate Control Mode

CBR

MULTIPLEX

QVBR

VBR

H264ScanType

H264 Scan Type

INTERLACED

PROGRESSIVE

H264SceneChangeDetect

H264 Scene Change Detect

DISABLED

ENABLED

H264Settings

H264 Settings

adaptiveQuantization

Enables or disables adaptive quantization, which is a technique MediaLive can apply to video on a frame-by-frame basis to produce more compression without losing quality. There are three types of adaptive quantization: flicker, spatial, and temporal. Set the field in one of these ways: Set to Auto. Recommended. For each type of AQ, MediaLive will determine if AQ is needed, and if so, the appropriate strength. Set a strength (a value other than Auto or Disable). This strength will apply to any of the AQ fields that you choose to enable. Set to Disabled to disable all types of adaptive quantization.

Type: H264AdaptiveQuantization
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: AfdSignaling
Required: False

bitrate

Average bitrate in bits/second. Required when the rate control mode is VBR or CBR. Not used for QVBR. In an MS Smooth output group, each output must have a unique value when its bitrate is 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

bufSize

Size of buffer (HRD buffer model) in bits.

Type: integer
Required: False
Minimum: 0

colorMetadata

Includes colorspace metadata in the output.

Type: H264ColorMetadata
Required: False

colorSpaceSettings

Color Space settings

Type: H264ColorSpaceSettings
Required: False

entropyEncoding

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

Type: H264EntropyEncoding
Required: False

filterSettings

Optional filters that you can apply to an encode.

Type: H264FilterSettings
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: FixedAfd
Required: False

flickerAq

Flicker AQ makes adjustments within each frame to reduce flicker or 'pop' on I-frames. The value to enter in this field depends on the value in the Adaptive quantization field: If you have set the Adaptive quantization field to Auto, MediaLive ignores any value in this field. MediaLive will determine if flicker AQ is appropriate and will apply the appropriate strength. If you have set the Adaptive quantization field to a strength, you can set this field to Enabled or Disabled. Enabled: MediaLive will apply flicker AQ using the specified strength. Disabled: MediaLive won't apply flicker AQ. If you have set the Adaptive quantization to Disabled, MediaLive ignores any value in this field and doesn't apply flicker AQ.

Type: H264FlickerAq
Required: False

forceFieldPictures

This setting applies only when scan type is "interlaced." It controls whether coding is performed on a field basis or on a frame basis. (When the video is progressive, the coding is always performed on a frame basis.) enabled: Force MediaLive to code on a field basis, so that odd and even sets of fields are coded separately. disabled: Code the two sets of fields separately (on a field basis) or together (on a frame basis using PAFF), depending on what is most appropriate for the content.

Type: H264ForceFieldPictures
Required: False

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: H264FramerateControl
Required: False

framerateDenominator

Framerate denominator.

Type: integer
Required: False
Minimum: 1

framerateNumerator

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

Type: integer
Required: False
Minimum: 1

gopBReference

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

Type: H264GopBReference
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

gopNumBFrames

Number of B-frames between reference frames.

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

gopSize

GOP size (keyframe interval) in units of either frames or seconds per gopSizeUnits. If gopSizeUnits is frames, gopSize must be an integer and must be greater than or equal to 1. If gopSizeUnits is seconds, gopSize must be greater than 0, but need not be an integer.

Type: number
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: H264GopSizeUnits
Required: False

level

H.264 Level.

Type: H264Level
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: H264LookAheadRateControl

Required: False

maxBitrate

For QVBR: See the tooltip for Quality level For VBR: Set the maximum bitrate in order to accommodate expected spikes in the complexity of the video.

Type: integer
Required: False
Minimum: 1000

minIInterval

Only meaningful if sceneChangeDetect is set to enabled. Defaults to 5 if multiplex rate control is used. 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

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

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 pixsel aspect ratio will be set equal to the input video pixel aspect ratio of the first input.

Type: H264ParControl
Required: False

parDenominator

Pixel Aspect Ratio denominator.

Type: integer
Required: False
Minimum: 1

parNumerator

Pixel Aspect Ratio numerator.

Type: integer
Required: False
Minimum: 1

profile

H.264 Profile.

Type: H264Profile
Required: False

qualityLevel

Leave as STANDARD_QUALITY or choose a different value (which might result in additional costs to run the channel). - ENHANCED_QUALITY: Produces a slightly better video quality without an increase in the bitrate. Has an effect only when the Rate control mode is QVBR

or CBR. If this channel is in a MediaLive multiplex, the value must be ENHANCED_QUALITY. - STANDARD_QUALITY: Valid for any Rate control mode.

Type: H264QualityLevel
Required: False

qvbrQualityLevel

Controls the target quality for the video encode. Applies only when the rate control mode is QVBR. You can set a target quality or you can let MediaLive determine the best quality. To set a target quality, enter values in the QVBR quality level field and the Max bitrate field. Enter values that suit your most important viewing devices. Recommended values are: - Primary screen: Quality level: 8 to 10. Max bitrate: 4M - PC or tablet: Quality level: 7. Max bitrate: 1.5M to 3M - Smartphone: Quality level: 6. Max bitrate: 1M to 1.5M To let MediaLive decide, leave the QVBR quality level field empty, and in Max bitrate enter the maximum rate you want in the video. For more information, see the section called "Video - rate control mode" in the MediaLive user guide

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

rateControlMode

Rate control mode. QVBR: Quality will match the specified quality level except when it is constrained by the maximum bitrate. Recommended if you or your viewers pay for bandwidth. VBR: Quality and bitrate vary, depending on the video complexity. Recommended instead of QVBR if you want to maintain a specific average bitrate over the duration of the channel. CBR: Quality varies, depending on the video complexity. Recommended only if you distribute your assets to devices that cannot handle variable bitrates. Multiplex: This rate control mode is only supported (and is required) when the video is being delivered to a MediaLive Multiplex in which case the rate control configuration is controlled by the properties within the Multiplex Program.

Type: H264RateControlMode
Required: False

scanType

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

Type: H264ScanType
Required: False

sceneChangeDetect

Scene change detection. - On: inserts I-frames when scene change is detected. - Off: does not force an I-frame when scene change is detected.

Type: H264SceneChangeDetect
Required: False

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

softness

Softness. Selects quantizer matrix, larger values reduce high-frequency content in the encoded image. If not set to zero, must be greater than 15.

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

spatialAq

Spatial AQ makes adjustments within each frame based on spatial variation of content complexity. The value to enter in this field depends on the value in the Adaptive quantization field: If you have set the Adaptive quantization field to Auto, MediaLive ignores any value in this field. MediaLive will determine if spatial AQ is appropriate and will apply the appropriate strength. If you have set the Adaptive quantization field to a strength, you can set this field to Enabled or Disabled. Enabled: MediaLive will apply spatial AQ using the specified strength. Disabled: MediaLive won't apply spatial AQ. If you have set the Adaptive quantization to Disabled, MediaLive ignores any value in this field and doesn't apply spatial AQ.

Type: H264SpatialAq
Required: False

subgopLength

If set to fixed, use gopNumBFrames B-frames per sub-GOP. If set to dynamic, optimize the number of B-frames used for each sub-GOP to improve visual quality.

Type: H264SubGopLength
Required: False

syntax

Produces a bitstream compliant with SMPTE RP-2027.

Type: H264Syntax
Required: False

temporalAq

Temporal makes adjustments within each frame based on temporal variation of content complexity. The value to enter in this field depends on the value in the Adaptive quantization field: If you have set the Adaptive quantization field to Auto, MediaLive ignores any value in this field. MediaLive will determine if temporal AQ is appropriate and will apply the appropriate strength. If you have set the Adaptive quantization field to a strength, you can set this field to Enabled or Disabled. Enabled: MediaLive will apply temporal AQ using the specified strength. Disabled:

MediaLive won't apply temporal AQ. If you have set the Adaptive quantization to Disabled, MediaLive ignores any value in this field and doesn't apply temporal AQ.

Type: H264TemporalAq
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: H264TimecodeInsertionBehavior
Required: False

H264SpatialAq

H264 Spatial Aq

```
DISABLED
ENABLED
```

H264SubGopLength

H264 Sub Gop Length

```
DYNAMIC
FIXED
```

H264Syntax

H264 Syntax

```
DEFAULT
RP2027
```

H264TemporalAq

H264 Temporal Aq

```
DISABLED
ENABLED
```

H264TimecodeInsertionBehavior

H264 Timecode Insertion Behavior

```
DISABLED
PIC_TIMING_SEI
```

H265AdaptiveQuantization

H265 Adaptive Quantization

```
AUTO
HIGH
HIGHER
LOW
MAX
MEDIUM
OFF
```

H265AlternativeTransferFunction

H265 Alternative Transfer Function

```
INSERT
OMIT
```

H265ColorMetadata

H265 Color Metadata

```
IGNORE
```

INSERT

H265ColorSpaceSettings

H265 Color Space Settings

colorSpacePassthroughSettings

Type: ColorSpacePassthroughSettings
Required: False

hdr10Settings

Type: Hdr10Settings
Required: False

rec601Settings

Type: Rec601Settings
Required: False

rec709Settings

Type: Rec709Settings
Required: False

H265FilterSettings

H265 Filter Settings

temporalFilterSettings

Type: TemporalFilterSettings
Required: False

H265FlickerAq

H265 Flicker Aq

DISABLED
ENABLED

H265GopSizeUnits

H265 Gop Size Units

FRAMES
SECONDS

H265Level

H265 Level

H265_LEVEL_1
H265_LEVEL_2
H265_LEVEL_2_1
H265_LEVEL_3
H265_LEVEL_3_1
H265_LEVEL_4
H265_LEVEL_4_1
H265_LEVEL_5
H265_LEVEL_5_1
H265_LEVEL_5_2
H265_LEVEL_6
H265_LEVEL_6_1
H265_LEVEL_6_2
H265_LEVEL_AUTO

H265LookAheadRateControl

H265 Look Ahead Rate Control

HIGH
LOW
MEDIUM

H265Profile

H265 Profile

MAIN
MAIN_10BIT

H265RateControlMode

H265 Rate Control Mode

CBR
MULTIPLEX
QVBR

H265ScanType

H265 Scan Type

INTERLACED
PROGRESSIVE

H265SceneChangeDetect

H265 Scene Change Detect

DISABLED
ENABLED

H265Settings

H265 Settings

adaptiveQuantization

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

Type: H265AdaptiveQuantization

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: AfdSignaling
Required: False

alternativeTransferFunction

Whether or not EML should insert an Alternative Transfer Function SEI message to support backwards compatibility with non-HDR decoders and displays.

Type: H265AlternativeTransferFunction
Required: False

bitrate

Average bitrate in bits/second. Required when the rate control mode is VBR or CBR. Not used for QVBR. In an MS Smooth output group, each output must have a unique value when its bitrate is rounded down to the nearest multiple of 1000.

Type: integer
Required: False
Minimum: 100000
Maximum: 40000000

bufSize

Size of buffer (HRD buffer model) in bits.

Type: integer
Required: False
Minimum: 100000
Maximum: 80000000

colorMetadata

Includes colorspace metadata in the output.

Type: H265ColorMetadata
Required: False

colorSpaceSettings

Color Space settings

Type: H265ColorSpaceSettings
Required: False

filterSettings

Optional filters that you can apply to an encode.

Type: H265FilterSettings
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: FixedAfd
Required: False

flickerAq

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

Type: H265FlickerAq
Required: False

framerateDenominator

Framerate denominator.

Type: integer
Required: True
Minimum: 1
Maximum: 3003

framerateNumerator

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

Type: integer
Required: True
Minimum: 1

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

gopSize

GOP size (keyframe interval) in units of either frames or seconds per gopSizeUnits. If gopSizeUnits is frames, gopSize must be an integer and must be greater than or equal to 1. If gopSizeUnits is seconds, gopSize must be greater than 0, but need not be an integer.

Type: number
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: H265GopSizeUnits

Required: False

level

H.265 Level.

Type: H265Level
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: H265LookAheadRateControl
Required: False

maxBitrate

For QVBR: See the tooltip for Quality level

Type: integer
Required: False
Minimum: 100000
Maximum: 40000000

minIInterval

Only meaningful if sceneChangeDetect is set to enabled. Defaults to 5 if multiplex rate control is used. 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

parDenominator

Pixel Aspect Ratio denominator.

Type: integer
Required: False
Minimum: 1

parNumerator

Pixel Aspect Ratio numerator.

Type: integer
Required: False
Minimum: 1

profile

H.265 Profile.

Type: H265Profile
Required: False

qvbrQualityLevel

Controls the target quality for the video encode. Applies only when the rate control mode is QVBR. Set values for the QVBR quality level field and Max bitrate field that suit your most important viewing devices. Recommended values are: - Primary screen: Quality level: 8 to 10. Max bitrate: 4M - PC or tablet: Quality level: 7. Max bitrate: 1.5M to 3M - Smartphone: Quality level: 6. Max bitrate: 1M to 1.5M

Type: integer
Required: False
Minimum: 1

Maximum: 10

rateControlMode

Rate control mode. QVBR: Quality will match the specified quality level except when it is constrained by the maximum bitrate. Recommended if you or your viewers pay for bandwidth. CBR: Quality varies, depending on the video complexity. Recommended only if you distribute your assets to devices that cannot handle variable bitrates. Multiplex: This rate control mode is only supported (and is required) when the video is being delivered to a MediaLive Multiplex in which case the rate control configuration is controlled by the properties within the Multiplex Program.

Type: H265RateControlMode
Required: False

scanType

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

Type: H265ScanType
Required: False

sceneChangeDetect

Scene change detection.

Type: H265SceneChangeDetect
Required: False

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: 16

tier

H.265 Tier.

Type: H265Tier
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: H265TimecodeInsertionBehavior
Required: False

H265Tier

H265 Tier

```
HIGH
MAIN
```

H265TimecodeInsertionBehavior

H265 Timecode Insertion Behavior

```
DISABLED
PIC_TIMING_SEI
```

Hdr10Settings

Hdr10 Settings

maxCll

Maximum Content Light Level An integer metadata value defining the maximum light level, in nits, of any single pixel within an encoded HDR video stream or file.

Type: integer
Required: False
Minimum: 0
Maximum: 32768

maxFall

Maximum Frame Average Light Level An integer metadata value defining the maximum average light level, in nits, for any single frame within an encoded HDR video stream or file.

Type: integer
Required: False
Minimum: 0
Maximum: 32768

HlsAdMarkers

Hls Ad Markers

ADOBE
ELEMENTAL
ELEMENTAL_SCTE35

HlsAkamaiHttpTransferMode

Hls Akamai Http Transfer Mode

CHUNKED
NON_CHUNKED

HlsAkamaiSettings

Hls Akamai Settings

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

httpTransferMode

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

Type: HlsAkamaiHttpTransferMode
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

salt

Salt for authenticated Akamai.

Type: string
Required: False

token

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

Type: string
Required: False

HlsBasicPutSettings

Hls Basic Put Settings

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

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

HlsCaptionLanguageSetting

Hls Caption Language Setting

```
INSERT
NONE
OMIT
```

HlsCdnSettings

Hls Cdn Settings

hlsAkamaiSettings

Type: HlsAkamaiSettings
Required: False

hlsBasicPutSettings

Type: HlsBasicPutSettings

Required: False

hlsMediaStoreSettings

Type: HlsMediaStoreSettings
Required: False

hlsS3Settings

Type: HlsS3Settings
Required: False

hlsWebdavSettings

Type: HlsWebdavSettings
Required: False

HlsClientCache

Hls Client Cache

DISABLED
ENABLED

HlsCodecSpecification

Hls Codec Specification

RFC_4281
RFC_6381

HlsDirectoryStructure

Hls Directory Structure

SINGLE_DIRECTORY
SUBDIRECTORY_PER_STREAM

HlsDiscontinuityTags

Hls Discontinuity Tags

```
INSERT
NEVER_INSERT
```

HlsEncryptionType

Hls Encryption Type

```
AES128
SAMPLE_AES
```

HlsGroupSettings

Hls Group Settings

adMarkers

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

Type: Array of type HlsAdMarkers
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

baseUrlContent1

Optional. One value per output group. This field is required only if you are completing Base URL content A, and the downstream system has notified you that the media files for pipeline 1 of all outputs are in a location different from the media files for pipeline 0.

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

baseUrlManifest1

Optional. One value per output group. Complete this field only if you are completing Base URL manifest A, and the downstream system has notified you that the child manifest files for pipeline 1 of all outputs are in a location different from the child manifest files for pipeline 0.

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

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: HlsCaptionLanguageSetting
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: HlsClientCache
Required: False

codecSpecification

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

Type: HlsCodecSpecification
Required: False

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
MinLength: 32
MaxLength: 32

destination

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

Type: OutputLocationRef
Required: True

directoryStructure

Place segments in subdirectories.

Type: HlsDirectoryStructure
Required: False

discontinuityTags

Specifies whether to insert EXT-X-DISCONTINUITY tags in the HLS child manifests for this output group. Typically, choose Insert because these tags are required in the manifest (according to the HLS specification) and serve an important purpose. Choose Never Insert only if the downstream system is doing real-time failover (without using the MediaLive automatic failover feature) and only if that downstream system has advised you to exclude the tags.

Type: HlsDiscontinuityTags
Required: False

encryptionType

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

Type: HlsEncryptionType
Required: False

hlsCdnSettings

Parameters that control interactions with the CDN.

Type: HlsCdnSettings
Required: False

hlsId3SegmentTagging

State of HLS ID3 Segment Tagging

Type: HlsId3SegmentTaggingState
Required: False

iFrameOnlyPlaylists

DISABLED: Do not create an I-frame-only manifest, but do create the master and media manifests (according to the Output Selection field). STANDARD: Create an I-frame-only manifest for each output that contains video, as well as the other manifests (according to the Output Selection field). The I-frame manifest contains a #EXT-X-I-FRAMES-ONLY tag to indicate it is I-frame only, and one or more #EXT-X-BYTERANGE entries identifying the I-frame position. For example, #EXT-X-BYTERANGE:160364@1461888"

Type: IFrameOnlyPlaylistType
Required: False

incompleteSegmentBehavior

Specifies whether to include the final (incomplete) segment in the media output when the pipeline stops producing output because of a channel stop, a channel pause or a loss of input to the pipeline. Auto means that MediaLive decides whether to include the final segment, depending on the channel class and the types of output groups. Suppress means to never include the incomplete segment. We recommend you choose Auto and let MediaLive control the behavior.

Type: HlsIncompleteSegmentBehavior
Required: False

indexNSegments

Applies only if Mode field is LIVE. Specifies the maximum number of segments in the media manifest file. After this maximum, older segments are removed from the media manifest. This number must be smaller than the number in the Keep Segments field.

Type: integer
Required: False
Minimum: 3

inputLossAction

Parameter that control output group behavior on input loss.

Type: InputLossActionForHlsOut

Required: False

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: HlsIvInManifest
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: HlsIvSource
Required: False

keepSegments

Applies only if Mode field is LIVE. Specifies the number of media segments to retain in the destination directory. This number should be bigger than indexNSegments (Num segments). We recommend (value = (2 x indexNsegments) + 1). If this "keep segments" number is too low, the following might happen: the player is still reading a media manifest file that lists this segment, but that segment has been removed from the destination directory (as directed by indexNSegments). This situation would result in a 404 HTTP error on the player.

Type: integer
Required: False
Minimum: 1

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

keyFormatVersions

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

Type: string
Required: False

keyProviderSettings

The key provider settings.

Type: KeyProviderSettings
Required: False

manifestCompression

When set to gzip, compresses HLS playlist.

Type: HlsManifestCompression
Required: False

manifestDurationFormat

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

Type: HlsManifestDurationFormat
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 "vod", all segments are indexed and kept permanently in the destination and manifest. If "live", only the number segments specified in keepSegments and indexNSegments are kept; newer segments replace older segments, which may prevent players from rewinding all the way to the beginning of the event. VOD mode uses HLS EXT-X-PLAYLIST-TYPE of EVENT while the channel is running, converting it to a "VOD" type manifest on completion of the stream.

Type: HlsMode
Required: False

outputSelection

MANIFESTS_AND_SEGMENTS: Generates manifests (master manifest, if applicable, and media manifests) for this output group. VARIANT_MANIFESTS_AND_SEGMENTS: Generates media manifests for this output group, but not a master manifest. SEGMENTS_ONLY: Does not generate any manifests for this output group.

Type: HlsOutputSelection
Required: False

programDateTime

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

Type: HlsProgramDateTime
Required: False

programDateTimeClock

Specifies the algorithm used to drive the HLS EXT-X-PROGRAM-DATE-TIME clock. Options include: INITIALIZE_FROM_OUTPUT_TIMECODE: The PDT clock is initialized as a function of the first output timecode, then incremented by the EXTINF duration of each encoded segment. SYSTEM_CLOCK:

The PDT clock is initialized as a function of the UTC wall clock, then incremented by the EXTINF duration of each encoded segment. If the PDT clock diverges from the wall clock by more than 500ms, it is resynchronized to the wall clock.

Type: HlsProgramDateTimeClock
Required: False

programDateTimePeriod

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

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

redundantManifest

ENABLED: The master manifest (.m3u8 file) for each pipeline includes information about both pipelines: first its own media files, then the media files of the other pipeline. This feature allows playout device that support stale manifest detection to switch from one manifest to the other, when the current manifest seems to be stale. There are still two destinations and two master manifests, but both master manifests reference the media files from both pipelines. DISABLED: The master manifest (.m3u8 file) for each pipeline includes information about its own pipeline only. For an HLS output group with MediaPackage as the destination, the DISABLED behavior is always followed. MediaPackage regenerates the manifests it serves to players so a redundant manifest from MediaLive is irrelevant.

Type: HlsRedundantManifest
Required: False

segmentLength

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

Type: integer
Required: False

Minimum: 1

segmentationMode

useInputSegmentation has been deprecated. The configured segment size is always used.

Type: HlsSegmentationMode
Required: False

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

streamInfResolution

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

Type: HlsStreamInfResolution
Required: False

timedMetadataId3Frame

Indicates ID3 frame that has the timecode.

Type: HlsTimedMetadataId3Frame
Required: False

timedMetadataId3Period

Timed Metadata interval in seconds.

Type: integer
Required: False
Minimum: 0

timestampDeltaMilliseconds

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

Type: integer
Required: False
Minimum: 0

tsFileMode

SEGMENTED_FILES: Emit the program as segments - multiple .ts media files. SINGLE_FILE: Applies only if Mode field is VOD. Emit the program as a single .ts media file. The media manifest includes #EXT-X-BYTERANGE tags to index segments for playback. A typical use for this value is when sending the output to AWS Elemental MediaConvert, which can accept only a single media file. Playback while the channel is running is not guaranteed due to HTTP server caching.

Type: HlsTsFileMode
Required: False

HlsH265PackagingType

Hls H265 Packaging Type

HEV1
HVC1

HlsId3SegmentTaggingState

State of HLS ID3 Segment Tagging

DISABLED
ENABLED

HlsIncompleteSegmentBehavior

Hls Incomplete Segment Behavior

AUTO

SUPPRESS

HlsInputSettings

Hls Input Settings

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

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

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

retryInterval

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

Type: integer
Required: False
Minimum: 0

scte35Source

Identifies the source for the SCTE-35 messages that MediaLive will ingest. Messages can be ingested from the content segments (in the stream) or from tags in the playlist (the HLS manifest). MediaLive ignores SCTE-35 information in the source that is not selected.

Type: HlsScte35SourceType
Required: False

HlsIvInManifest

Hls Iv In Manifest

```
EXCLUDE
INCLUDE
```

HlsIvSource

Hls Iv Source

```
EXPLICIT
FOLLOWS_SEGMENT_NUMBER
```

HlsManifestCompression

Hls Manifest Compression

```
GZIP
NONE
```

HlsManifestDurationFormat

Hls Manifest Duration Format

FLOATING_POINT
INTEGER

HlsMediaStoreSettings

Hls Media Store Settings

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

mediaStoreStorageClass

When set to temporal, output files are stored in non-persistent memory for faster reading and writing.

Type: HlsMediaStoreStorageClass
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

HlsMediaStoreStorageClass

Hls Media Store Storage Class

```
TEMPORAL
```

HlsMode

Hls Mode

```
LIVE
VOD
```

HlsOutputSelection

Hls Output Selection

```
MANIFESTS_AND_SEGMENTS
SEGMENTS_ONLY
VARIANT_MANIFESTS_AND_SEGMENTS
```

HlsOutputSettings

Hls Output Settings

h265PackagingType

Only applicable when this output is referencing an H.265 video description. Specifies whether MP4 segments should be packaged as HEV1 or HVC1.

Type: HlsH265PackagingType
Required: False

hlsSettings

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

Type: HlsSettings
Required: True

nameModifier

String concatenated to the end of the destination filename. Accepts \"Format Identifiers \":#formatIdentifierParameters.

Type: string
Required: False
MinLength: 1

segmentModifier

String concatenated to end of segment filenames.

Type: string
Required: False

HlsProgramDateTime

Hls Program Date Time

EXCLUDE
INCLUDE

HlsProgramDateTimeClock

Hls Program Date Time Clock

INITIALIZE_FROM_OUTPUT_TIMECODE

SYSTEM_CLOCK

HlsRedundantManifest

Hls Redundant Manifest

DISABLED
ENABLED

HlsS3LogUploads

Hls S3 Log Uploads

DISABLED
ENABLED

HlsS3Settings

Hls S3 Settings

cannedAcl

Specify the canned ACL to apply to each S3 request. Defaults to none.

Type: S3CannedAcl
Required: False

logUploads

When set to enabled, each fragment upload to CDN or server will be logged.

Type: HlsS3LogUploads
Required: False

HlsScte35SourceType

Hls Scte35 Source Type

MANIFEST

SEGMENTS

HlsSegmentationMode

Hls Segmentation Mode

USE_INPUT_SEGMENTATION
USE_SEGMENT_DURATION

HlsSettings

Hls Settings

audioOnlyHlsSettings

Type: AudioOnlyHlsSettings
Required: False

fmp4HlsSettings

Type: Fmp4HlsSettings
Required: False

frameCaptureHlsSettings

Type: FrameCaptureHlsSettings
Required: False

standardHlsSettings

Type: StandardHlsSettings
Required: False

HlsStreamInfResolution

Hls Stream Inf Resolution

EXCLUDE

INCLUDE

HlsTimedMetadataId3Frame

Hls Timed Metadata Id3 Frame

NONE
PRIV
TDRL

HlsTsFileMode

Hls Ts File Mode

SEGMENTED_FILES
SINGLE_FILE

HlsWebdavHttpTransferMode

Hls Webdav Http Transfer Mode

CHUNKED
NON_CHUNKED

HlsWebdavSettings

Hls Webdav Settings

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

httpTransferMode

Specify whether or not to use chunked transfer encoding to WebDAV.

Type: HlsWebdavHttpTransferMode
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

HtmlMotionGraphicsSettings

Html Motion Graphics Settings

IFrameOnlyPlaylistType

When set to "standard", an I-Frame only playlist will be written out for each video output in the output group. This I-Frame only playlist will contain byte range offsets pointing to the I-frame(s) in each segment.

DISABLED
STANDARD

InputAttachment

automaticInputFailoverSettings

User-specified settings for defining what the conditions are for declaring the input unhealthy and failing over to a different input.

Type: AutomaticInputFailoverSettings
Required: False

inputAttachmentName

User-specified name for the attachment. This is required if the user wants to use this input in an input switch action.

Type: string
Required: False

inputId

The ID of the input

Type: string
Required: False

inputSettings

Settings of an input (caption selector, etc.)

Type: InputSettings
Required: False

InputChannelLevel

Input Channel Level

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

inputChannel

The index of the input channel used as a source.

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

InputCodec

codec in increasing order of complexity

MPEG2
AVC
HEVC

InputDeblockFilter

Input Deblock Filter

DISABLED
ENABLED

InputDenoiseFilter

Input Denoise Filter

DISABLED
ENABLED

InputFilter

Input Filter

AUTO
DISABLED
FORCED

InputLocation

Input Location

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 RTMP destination should have a uri simliar 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

Input Loss Action For Hls Out

```
EMIT_OUTPUT
PAUSE_OUTPUT
```

InputLossActionForMsSmoothOut

Input Loss Action For Ms Smooth Out

```
EMIT_OUTPUT
PAUSE_OUTPUT
```

InputLossActionForRtmpOut

Input Loss Action For Rtmp Out

```
EMIT_OUTPUT
PAUSE_OUTPUT
```

InputLossActionForUdpOut

Input Loss Action For Udp Out

```
DROP_PROGRAM
DROP_TS
EMIT_PROGRAM
```

InputLossBehavior

Input Loss Behavior

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

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
MinLength: 6
MaxLength: 6

inputLossImageSlate

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

Type: InputLocation
Required: False

inputLossImageType

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

Type: InputLossImageType
Required: False

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

InputLossFailoverSettings

MediaLive will perform a failover if content is not detected in this input for the specified period.

inputLossThresholdMsec

The amount of time (in milliseconds) that no input is detected. After that time, an input failover will occur.

Type: integer
Required: False
Minimum: 100

InputLossImageType

Input Loss Image Type

COLOR
SLATE

InputMaximumBitrate

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

MAX_10_MBPS
MAX_20_MBPS
MAX_50_MBPS

InputPreference

Input preference when deciding which input to make active when a previously failed input has recovered. If \"EQUAL_INPUT_PREFERENCE\", then the active input will stay active as long as it is healthy. If \"PRIMARY_INPUT_PREFERRED\", then always switch back to the primary input when it is healthy.

EQUAL_INPUT_PREFERENCE

PRIMARY_INPUT_PREFERRED

InputResolution

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

Live Event input parameters. There can be multiple inputs in a single Live Event.

audioSelectors

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

Type: Array of type AudioSelector
Required: False

captionSelectors

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

Type: Array of type CaptionSelector
Required: False

deblockFilter

Enable or disable the deblock filter when filtering.

Type: InputDeblockFilter
Required: False

denoiseFilter

Enable or disable the denoise filter when filtering.

Type: InputDenoiseFilter
Required: False

filterStrength

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

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

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: InputFilter
Required: False

networkInputSettings

Input settings.

Type: NetworkInputSettings
Required: False

smpte2038DataPreference

Specifies whether to extract applicable ancillary data from a SMPTE-2038 source in this input. Applicable data types are captions, timecode, AFD, and SCTE-104 messages. - PREFER: Extract from SMPTE-2038 if present in this input, otherwise extract from another source (if any). - IGNORE: Never extract any ancillary data from SMPTE-2038.

Type: Smpte2038DataPreference
Required: False

sourceEndBehavior

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

Type: InputSourceEndBehavior
Required: False

videoSelector

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

Type: VideoSelector
Required: False

InputSourceEndBehavior

Input Source End Behavior

CONTINUE
LOOP

InputSpecification

codec

Input codec

Type: InputCodec
Required: False

maximumBitrate

Maximum input bitrate, categorized coarsely

Type: InputMaximumBitrate
Required: False

resolution

Input resolution, categorized coarsely

Type: InputResolution
Required: False

InternalServiceError

message

Type: string
Required: False

InvalidRequest

message

Type: string
Required: False

KeyProviderSettings

Key Provider Settings

staticKeySettings

Type: StaticKeySettings
Required: False

LimitExceeded

message

Type: string
Required: False

LogLevel

The log level the user wants for their channel.

```
ERROR
WARNING
INFO
DEBUG
DISABLED
```

M2tsAbsentInputAudioBehavior

M2ts Absent Input Audio Behavior

```
DROP
ENCODE_SILENCE
```

M2tsArib

M2ts Arib

```
DISABLED
ENABLED
```

M2tsAribCaptionsPidControl

M2ts Arib Captions Pid Control

```
AUTO
USE_CONFIGURED
```

M2tsAudioBufferModel

M2ts Audio Buffer Model

```
ATSC
DVB
```

M2tsAudioInterval

M2ts Audio Interval

VIDEO_AND_FIXED_INTERVALS
VIDEO_INTERVAL

M2tsAudioStreamType

M2ts Audio Stream Type

ATSC
DVB

M2tsBufferModel

M2ts Buffer Model

MULTIPLEX
NONE

M2tsCcDescriptor

M2ts Cc Descriptor

DISABLED
ENABLED

M2tsEbifControl

M2ts Ebif Control

NONE
PASSTHROUGH

M2tsEbpPlacement

M2ts Ebp Placement

VIDEO_AND_AUDIO_PIDS
VIDEO_PID

M2tsEsRateInPes

M2ts Es Rate In Pes

EXCLUDE
INCLUDE

M2tsKlv

M2ts Klv

NONE
PASSTHROUGH

M2tsNielsenId3Behavior

M2ts Nielsen Id3 Behavior

NO_PASSTHROUGH
PASSTHROUGH

M2tsPcrControl

M2ts Pcr Control

CONFIGURED_PCR_PERIOD
PCR_EVERY_PES_PACKET

M2tsRateMode

M2ts Rate Mode

CBR
VBR

M2tsScte35Control

M2ts Scte35 Control

NONE
PASSTHROUGH

M2tsSegmentationMarkers

M2ts Segmentation Markers

EBP
EBP_LEGACY
NONE
PSI_SEGSTART
RAI_ADAPT
RAI_SEGSTART

M2tsSegmentationStyle

M2ts Segmentation Style

MAINTAIN_CADENCE
RESET_CADENCE

M2tsSettings

M2ts Settings

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: M2tsAbsentInputAudioBehavior

Required: False

arib

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

Type: M2tsArib
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

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: M2tsAribCaptionsPidControl
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: M2tsAudioBufferModel
Required: False

audioFramesPerPes

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

Type: integer

Required: False
Minimum: 0

audioPids

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

Type: string
Required: False

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: M2tsAudioStreamType
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

bufferModel

Controls the timing accuracy for output network traffic. Leave as MULTIPLEX to ensure accurate network packet timing. Or set to NONE, which might result in lower latency but will result in more variability in output network packet timing. This variability might cause interruptions, jitter, or bursty behavior in your playback or receiving devices.

Type: M2tsBufferModel

Required: False

ccDescriptor

When set to enabled, generates captionServiceDescriptor in PMT.

Type: M2tsCcDescriptor
Required: False

dvbNitSettings

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

Type: DvbNitSettings
Required: False

dvbSdtSettings

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

Type: DvbSdtSettings
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

dvbTdtSettings

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

Type: DvbTdtSettings
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

ebif

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

Type: M2tsEbifControl
Required: False

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: M2tsAudioInterval
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

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: M2tsEbpPlacement
Required: False

ecmPid

This field is unused and deprecated.

Type: string
Required: False

esRateInPes

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

Type: M2tsEsRateInPes
Required: False

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

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

fragmentTime

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

Type: number
Required: False
Minimum: 0

klv

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

Type: M2tsKlv
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

nielsenId3Behavior

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

Type: M2tsNielsenId3Behavior
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

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

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: M2tsPcrControl
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

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

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

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

programNum

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

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

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: M2tsRateMode
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

scte35Control

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

Type: M2tsScte35Control

Required: False

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

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: M2tsSegmentationMarkers

Required: False

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: M2tsSegmentationStyle
Required: False

segmentationTime

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

Type: number
Required: False
Minimum: 1

timedMetadataBehavior

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

Type: M2tsTimedMetadataBehavior
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

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. Valid values are 32 (or 0x20)..8182 (or 0x1ff6).

Type: string
Required: False

M2tsTimedMetadataBehavior

M2ts Timed Metadata Behavior

NO_PASSTHROUGH
PASSTHROUGH

M3u8NielsenId3Behavior

M3u8 Nielsen Id3 Behavior

NO_PASSTHROUGH
PASSTHROUGH

M3u8PcrControl

M3u8 Pcr Control

CONFIGURED_PCR_PERIOD
PCR_EVERY_PES_PACKET

M3u8Scte35Behavior

M3u8 Scte35 Behavior

NO_PASSTHROUGH
PASSTHROUGH

M3u8Settings

Settings information for the .m3u8 container

audioFramesPerPes

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

Type: integer
Required: False
Minimum: 0

audioPids

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

Type: string
Required: False

ecmPid

This parameter is unused and deprecated.

Type: string
Required: False

nielsenId3Behavior

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

Type: M3u8NielsenId3Behavior
Required: False

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

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: M3u8PcrControl
Required: False

pcrPeriod

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

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

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

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

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

programNum

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

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

scte35Behavior

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

Type: M3u8Scte35Behavior
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

timedMetadataBehavior

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

Type: M3u8TimedMetadataBehavior
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

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

M3u8TimedMetadataBehavior

M3u8 Timed Metadata Behavior

NO_PASSTHROUGH
PASSTHROUGH

MaintenanceDay

The currently selected maintenance day.

MONDAY
TUESDAY
WEDNESDAY
THURSDAY
FRIDAY
SATURDAY
SUNDAY

MaintenanceStatus

maintenanceDay

The currently selected maintenance day.

Type: MaintenanceDay
Required: False

maintenanceDeadline

Maintenance is required by the displayed date and time. Date and time is in ISO.

Type: string
Required: False
Format: string

maintenanceScheduledDate

The currently scheduled maintenance date and time. Date and time is in ISO.

Type: string
Required: False
Format: string

maintenanceStartTime

The currently selected maintenance start time. Time is in UTC.

Type: string
Required: False

MediaPackageGroupSettings

Media Package Group Settings

destination

MediaPackage channel destination.

Type: OutputLocationRef
Required: True

MediaPackageOutputDestinationSettings

MediaPackage Output Destination Settings

channelId

ID of the channel in MediaPackage that is the destination for this output group. You do not need to specify the individual inputs in MediaPackage; MediaLive will handle the connection of the two MediaLive pipelines to the two MediaPackage inputs. The MediaPackage channel and MediaLive channel must be in the same region.

Type: string
Required: False
MinLength: 1

MediaPackageOutputSettings

Media Package Output Settings

MotionGraphicsConfiguration

Motion Graphics Configuration

motionGraphicsInsertion

Type: MotionGraphicsInsertion
Required: False

motionGraphicsSettings

Motion Graphics Settings

Type: MotionGraphicsSettings
Required: True

MotionGraphicsInsertion

Motion Graphics Insertion

```
DISABLED
ENABLED
```

MotionGraphicsSettings

Motion Graphics Settings

htmlMotionGraphicsSettings

Type: HtmlMotionGraphicsSettings
Required: False

Mp2CodingMode

Mp2 Coding Mode

CODING_MODE_1_0

CODING_MODE_2_0

Mp2Settings

Mp2 Settings

bitrate

Average bitrate in bits/second.

Type: number

Required: False

codingMode

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

Type: Mp2CodingMode

Required: False

sampleRate

Sample rate in Hz.

Type: number

Required: False

Mpeg2AdaptiveQuantization

Mpeg2 Adaptive Quantization

AUTO

HIGH

LOW

MEDIUM

OFF

Mpeg2ColorMetadata

Mpeg2 Color Metadata

IGNORE
INSERT

Mpeg2ColorSpace

Mpeg2 Color Space

AUTO
PASSTHROUGH

Mpeg2DisplayRatio

Mpeg2 Display Ratio

DISPLAYRATIO16X9
DISPLAYRATIO4X3

Mpeg2FilterSettings

Mpeg2 Filter Settings

temporalFilterSettings

Type: TemporalFilterSettings
Required: False

Mpeg2GopSizeUnits

Mpeg2 Gop Size Units

FRAMES

SECONDS

Mpeg2ScanType

Mpeg2 Scan Type

INTERLACED
PROGRESSIVE

Mpeg2Settings

Mpeg2 Settings

adaptiveQuantization

Choose Off to disable adaptive quantization. Or choose another value to enable the quantizer and set its strength. The strengths are: Auto, Off, Low, Medium, High. When you enable this field, MediaLive allows intra-frame quantizers to vary, which might improve visual quality.

Type: Mpeg2AdaptiveQuantization
Required: False

afdSignaling

Indicates the AFD values that MediaLive will write into the video encode. If you do not know what AFD signaling is, or if your downstream system has not given you guidance, choose AUTO. AUTO: MediaLive will try to preserve the input AFD value (in cases where multiple AFD values are valid). FIXED: MediaLive will use the value you specify in fixedAFD.

Type: AfdSignaling
Required: False

colorMetadata

Specifies whether to include the color space metadata. The metadata describes the color space that applies to the video (the colorSpace field). We recommend that you insert the metadata.

Type: Mpeg2ColorMetadata
Required: False

colorSpace

Choose the type of color space conversion to apply to the output. For detailed information on setting up both the input and the output to obtain the desired color space in the output, see the section on \"MediaLive Features - Video - color space\" in the MediaLive User Guide. PASSTHROUGH: Keep the color space of the input content - do not convert it. AUTO:Convert all content that is SD to rec 601, and convert all content that is HD to rec 709.

Type: Mpeg2ColorSpace
Required: False

displayAspectRatio

Sets the pixel aspect ratio for the encode.

Type: Mpeg2DisplayRatio
Required: False

filterSettings

Optionally specify a noise reduction filter, which can improve quality of compressed content. If you do not choose a filter, no filter will be applied. TEMPORAL: This filter is useful for both source content that is noisy (when it has excessive digital artifacts) and source content that is clean. When the content is noisy, the filter cleans up the source content before the encoding phase, with these two effects: First, it improves the output video quality because the content has been cleaned up. Secondly, it decreases the bandwidth because MediaLive does not waste bits on encoding noise. When the content is reasonably clean, the filter tends to decrease the bitrate.

Type: Mpeg2FilterSettings
Required: False

fixedAfd

Complete this field only when afdSignaling is set to FIXED. Enter the AFD value (4 bits) to write on all frames of the video encode.

Type: FixedAfd
Required: False

framerateDenominator

description": "The framerate denominator. For example, 1001. The framerate is the numerator divided by the denominator. For example, 24000 / 1001 = 23.976 FPS.

Type: integer
Required: True
Minimum: 1

framerateNumerator

The framerate numerator. For example, 24000. The framerate is the numerator divided by the denominator. For example, 24000 / 1001 = 23.976 FPS.

Type: integer
Required: True
Minimum: 1

gopClosedCadence

MPEG2: default is open GOP.

Type: integer
Required: False
Minimum: 0

gopNumBFrames

Relates to the GOP structure. The number of B-frames between reference frames. If you do not know what a B-frame is, use the default.

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

gopSize

Relates to the GOP structure. The GOP size (keyframe interval) in the units specified in gopSizeUnits. If you do not know what GOP is, use the default. If gopSizeUnits is frames, then the gopSize must be an integer and must be greater than or equal to 1. If gopSizeUnits is seconds, the gopSize must be greater than 0, but does not need to be an integer.

Type: number
Required: False

gopSizeUnits

Relates to the GOP structure. Specifies whether the gopSize is specified in frames or seconds. If you do not plan to change the default gopSize, leave the default. If you specify SECONDS, MediaLive will internally convert the gop size to a frame count.

Type: Mpeg2GopSizeUnits
Required: False

scanType

Set the scan type of the output to PROGRESSIVE or INTERLACED (top field first).

Type: Mpeg2ScanType
Required: False

subgopLength

Relates to the GOP structure. If you do not know what GOP is, use the default. FIXED: Set the number of B-frames in each sub-GOP to the value in gopNumBFrames. DYNAMIC: Let MediaLive optimize the number of B-frames in each sub-GOP, to improve visual quality.

Type: Mpeg2SubGopLength
Required: False

timecodeInsertion

Determines how MediaLive inserts timecodes in the output video. For detailed information about setting up the input and the output for a timecode, see the section on \"MediaLive Features -

Timecode configuration\" in the MediaLive User Guide. DISABLED: do not include timecodes. GOP_TIMECODE: Include timecode metadata in the GOP header.

Type: Mpeg2TimecodeInsertionBehavior
Required: False

Mpeg2SubGopLength

Mpeg2 Sub Gop Length

```
DYNAMIC
FIXED
```

Mpeg2TimecodeInsertionBehavior

Mpeg2 Timecode Insertion Behavior

```
DISABLED
GOP_TIMECODE
```

MsSmoothGroupSettings

Ms Smooth Group Settings

acquisitionPointId

The ID to include in each message in the sparse track. Ignored if sparseTrackType is NONE.

Type: string
Required: False

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: SmoothGroupAudioOnlyTimecodeControl
Required: False

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.

Type: SmoothGroupCertificateMode
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

destination

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

Type: OutputLocationRef
Required: True

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

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: SmoothGroupEventIdMode
Required: False

eventStopBehavior

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

Type: SmoothGroupEventStopBehavior
Required: False

filecacheDuration

Size in seconds of file cache for streaming outputs.

Type: integer
Required: False
Minimum: 0

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

inputLossAction

Parameter that control output group behavior on input loss.

Type: InputLossActionForMsSmoothOut
Required: False

numRetries

Number of retry attempts.

Type: integer
Required: False
Minimum: 0

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

segmentationMode

useInputSegmentation has been deprecated. The configured segment size is always used.

Type: SmoothGroupSegmentationMode
Required: False

sendDelayMs

Number of milliseconds to delay the output from the second pipeline.

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

sparseTrackType

Identifies the type of data to place in the sparse track: - SCTE35: Insert SCTE-35 messages from the source content. With each message, insert an IDR frame to start a new segment. - SCTE35_WITHOUT_SEGMENTATION: Insert SCTE-35 messages from the source content. With each

message, insert an IDR frame but don't start a new segment. - NONE: Don't generate a sparse track for any outputs in this output group.

Type: SmoothGroupSparseTrackType
Required: False

streamManifestBehavior

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

Type: SmoothGroupStreamManifestBehavior
Required: False

timestampOffset

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

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: SmoothGroupTimestampOffsetMode
Required: False

MsSmoothH265PackagingType

Ms Smooth H265 Packaging Type

HEV1
HVC1

MsSmoothOutputSettings

Ms Smooth Output Settings

h265PackagingType

Only applicable when this output is referencing an H.265 video description. Specifies whether MP4 segments should be packaged as HEV1 or HVC1.

Type: MsSmoothH265PackagingType
Required: False

nameModifier

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

Type: string
Required: False

MultiplexGroupSettings

Multiplex Group Settings

MultiplexOutputSettings

Multiplex Output Settings

destination

Destination is a Multiplex.

Type: OutputLocationRef
Required: True

MultiplexProgramChannelDestinationSettings

Multiplex Program Input Destination Settings for outputting a Channel to a Multiplex

multiplexId

The ID of the Multiplex that the encoder is providing output to. You do not need to specify the individual inputs to the Multiplex; MediaLive will handle the connection of the two MediaLive pipelines to the two Multiplex instances. The Multiplex must be in the same region as the Channel.

Type: string
Required: False
MinLength: 1

programName

The program name of the Multiplex program that the encoder is providing output to.

Type: string
Required: False
MinLength: 1

NetworkInputServerValidation

Network Input Server Validation

```
CHECK_CRYPTOGRAPHY_AND_VALIDATE_NAME
CHECK_CRYPTOGRAPHY_ONLY
```

NetworkInputSettings

Network source to transcode. Must be accessible to the Elemental Live node that is running the live event through a network connection.

hlsInputSettings

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

Type: HlsInputSettings
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: NetworkInputServerValidation
Required: False

NielsenCBET

Nielsen CBET

cbetCheckDigitString

Enter the CBET check digits to use in the watermark.

Type: string
Required: True
MinLength: 2
MaxLength: 2

cbetStepaside

Determines the method of CBET insertion mode when prior encoding is detected on the same layer.

Type: NielsenWatermarksCbetStepaside
Required: True

csid

Enter the CBET Source ID (CSID) to use in the watermark

Type: string
Required: True
MinLength: 1
MaxLength: 7

NielsenConfiguration

Nielsen Configuration

distributorId

Enter the Distributor ID assigned to your organization by Nielsen.

Type: string
Required: False

nielsenPcmToId3Tagging

Enables Nielsen PCM to ID3 tagging

Type: NielsenPcmToId3TaggingState
Required: False

NielsenNaesIiNw

Nielsen Naes Ii Nw

checkDigitString

Enter the check digit string for the watermark

Type: string
Required: True
MinLength: 2
MaxLength: 2

sid

Enter the Nielsen Source ID (SID) to include in the watermark

Type: number
Required: True
Minimum: 1
Maximum: 65535

NielsenPcmToId3TaggingState

State of Nielsen PCM to ID3 tagging

DISABLED
ENABLED

NielsenWatermarksCbetStepaside

Nielsen Watermarks Cbet Stepaside

```
DISABLED
ENABLED
```

NielsenWatermarksDistributionTypes

Nielsen Watermarks Distribution Types

```
FINAL_DISTRIBUTOR
PROGRAM_CONTENT
```

NielsenWatermarksSettings

Nielsen Watermarks Settings

nielsenCbetSettings

Complete these fields only if you want to insert watermarks of type Nielsen CBET

Type: NielsenCBET
Required: False

nielsenDistributionType

Choose the distribution types that you want to assign to the watermarks: - PROGRAM_CONTENT - FINAL_DISTRIBUTOR

Type: NielsenWatermarksDistributionTypes
Required: False

nielsenNaesIiNwSettings

Complete these fields only if you want to insert watermarks of type Nielsen NAES II (N2) and Nielsen NAES VI (NW).

Type: NielsenNaesIiNw
Required: False

Output

Output settings. There can be multiple outputs within a group.

audioDescriptionNames

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

Type: Array of 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
MinLength: 1
MaxLength: 255

outputSettings

Output type-specific settings.

Type: OutputSettings
Required: True

videoDescriptionName

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

Type: string
Required: False

OutputDestination

id

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

Type: string
Required: False

mediaPackageSettings

Destination settings for a MediaPackage output; one destination for both encoders.

Type: Array of type MediaPackageOutputDestinationSettings
Required: False

multiplexSettings

Destination settings for a Multiplex output; one destination for both encoders.

Type: MultiplexProgramChannelDestinationSettings
Required: False

settings

Destination settings for a standard output; one destination for each redundant encoder.

Type: Array of type OutputDestinationSettings
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

Output groups for this Live Event. Output groups contain information about where streams should be distributed.

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
MaxLength: 32

outputGroupSettings

Settings associated with the output group.

Type: OutputGroupSettings

Required: True

outputs

Type: Array of type Output

Required: True

OutputGroupSettings

Output Group Settings

archiveGroupSettings

Type: ArchiveGroupSettings

Required: False

frameCaptureGroupSettings

Type: FrameCaptureGroupSettings

Required: False

hlsGroupSettings

Type: HlsGroupSettings

Required: False

mediaPackageGroupSettings

Type: MediaPackageGroupSettings

Required: False

msSmoothGroupSettings

Type: MsSmoothGroupSettings

Required: False

multiplexGroupSettings

Type: MultiplexGroupSettings
Required: False

rtmpGroupSettings

Type: RtmpGroupSettings
Required: False

udpGroupSettings

Type: UdpGroupSettings
Required: False

OutputLocationRef

Reference to an OutputDestination ID defined in the channel

destinationRefId

Type: string
Required: False

OutputSettings

Output Settings

archiveOutputSettings

Type: ArchiveOutputSettings
Required: False

frameCaptureOutputSettings

Type: FrameCaptureOutputSettings
Required: False

hlsOutputSettings

Type: HlsOutputSettings
Required: False

mediaPackageOutputSettings

Type: MediaPackageOutputSettings
Required: False

msSmoothOutputSettings

Type: MsSmoothOutputSettings
Required: False

multiplexOutputSettings

Type: MultiplexOutputSettings
Required: False

rtmpOutputSettings

Type: RtmpOutputSettings
Required: False

udpOutputSettings

Type: UdpOutputSettings
Required: False

PassThroughSettings

Pass Through Settings

PipelineDetail

Runtime details of a pipeline when a channel is running.

activeInputAttachmentName

The name of the active input attachment currently being ingested by this pipeline.

Type: string
Required: False

activeInputSwitchActionName

The name of the input switch schedule action that occurred most recently and that resulted in the switch to the current input attachment for this pipeline.

Type: string
Required: False

activeMotionGraphicsActionName

The name of the motion graphics activate action that occurred most recently and that resulted in the current graphics URI for this pipeline.

Type: string
Required: False

activeMotionGraphicsUri

The current URI being used for HTML5 motion graphics for this pipeline.

Type: string
Required: False

pipelineId

Pipeline ID

Type: string
Required: False

RawSettings

Raw Settings

Rec601Settings

Rec601 Settings

Rec709Settings

Rec709 Settings

RemixSettings

Remix Settings

channelMappings

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

Type: Array of type AudioChannelMapping
Required: True

channelsIn

Number of input channels to be used.

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

channelsOut

Number of output channels to be produced. Valid values: 1, 2, 4, 6, 8

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

ResourceConflict

message

Type: string
Required: False

ResourceNotFound

message

Type: string
Required: False

RtmpAdMarkers

Rtmp Ad Markers

```
ON_CUE_POINT_SCTE35
```

RtmpCacheFullBehavior

Rtmp Cache Full Behavior

```
DISCONNECT_IMMEDIATELY
WAIT_FOR_SERVER
```

RtmpCaptionData

Rtmp Caption Data

```
ALL
FIELD1_608
FIELD1_AND_FIELD2_608
```

RtmpCaptionInfoDestinationSettings

Rtmp Caption Info Destination Settings

RtmpGroupSettings

Rtmp Group Settings

adMarkers

Choose the ad marker type for this output group. MediaLive will create a message based on the content of each SCTE-35 message, format it for that marker type, and insert it in the datastream.

Type: Array of type RtmpAdMarkers
Required: False

authenticationScheme

Authentication scheme to use when connecting with CDN

Type: AuthenticationScheme
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: RtmpCacheFullBehavior
Required: False

cacheLength

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

Type: integer
Required: False
Minimum: 30

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: RtmpCaptionData
Required: False

inputLossAction

Controls the behavior of this RTMP group if input becomes unavailable. - emitOutput: Emit a slate until input returns. - pauseOutput: Stop transmitting data until input returns. This does not close the underlying RTMP connection.

Type: InputLossActionForRtmpOut
Required: False

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

RtmpOutputCertificateMode

Rtmp Output Certificate Mode

```
SELF_SIGNED
VERIFY_AUTHENTICITY
```

RtmpOutputSettings

Rtmp Output Settings

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: RtmpOutputCertificateMode
Required: False

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

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
Required: True

numRetries

Number of retry attempts.

Type: integer
Required: False
Minimum: 0

S3CannedAcl

S3 Canned Acl

```
AUTHENTICATED_READ
BUCKET_OWNER_FULL_CONTROL
```

BUCKET_OWNER_READ
PUBLIC_READ

Scte20Convert608To708

Scte20 Convert608 To708

DISABLED
UPCONVERT

Scte20PlusEmbeddedDestinationSettings

Scte20 Plus Embedded Destination Settings

Scte20SourceSettings

Scte20 Source Settings

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: Scte20Convert608To708
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

Scte27DestinationSettings

Scte27 Destination Settings

Scte27OcrLanguage

Scte27 Ocr Language

DEU
ENG
FRA
NLD
POR
SPA

Scte27SourceSettings

Scte27 Source Settings

ocrLanguage

If you will configure a WebVTT caption description that references this caption selector, use this field to provide the language to consider when translating the image-based source to text.

Type: Scte27OcrLanguage
Required: False

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

Scte35 Apos No Regional Blackout Behavior

FOLLOW
IGNORE

Scte35AposWebDeliveryAllowedBehavior

Scte35 Apos Web Delivery Allowed Behavior

FOLLOW
IGNORE

Scte35SpliceInsert

Scte35 Splice Insert

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

noRegionalBlackoutFlag

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

Type: Scte35SpliceInsertNoRegionalBlackoutBehavior
Required: False

webDeliveryAllowedFlag

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

Type: Scte35SpliceInsertWebDeliveryAllowedBehavior

Required: False

Scte35SpliceInsertNoRegionalBlackoutBehavior

Scte35 Splice Insert No Regional Blackout Behavior

FOLLOW
IGNORE

Scte35SpliceInsertWebDeliveryAllowedBehavior

Scte35 Splice Insert Web Delivery Allowed Behavior

FOLLOW
IGNORE

Scte35TimeSignalApos

Scte35 Time Signal Apos

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

noRegionalBlackoutFlag

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

Type: Scte35AposNoRegionalBlackoutBehavior
Required: False

webDeliveryAllowedFlag

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

Type: Scte35AposWebDeliveryAllowedBehavior

Required: False

SmoothGroupAudioOnlyTimecodeControl

Smooth Group Audio Only Timecode Control

PASSTHROUGH

USE_CONFIGURED_CLOCK

SmoothGroupCertificateMode

Smooth Group Certificate Mode

SELF_SIGNED

VERIFY_AUTHENTICITY

SmoothGroupEventIdMode

Smooth Group Event Id Mode

NO_EVENT_ID

USE_CONFIGURED

USE_TIMESTAMP

SmoothGroupEventStopBehavior

Smooth Group Event Stop Behavior

NONE

SEND_EOS

SmoothGroupSegmentationMode

Smooth Group Segmentation Mode

USE_INPUT_SEGMENTATION
USE_SEGMENT_DURATION

SmoothGroupSparseTrackType

Smooth Group Sparse Track Type

NONE
SCTE_35
SCTE_35_WITHOUT_SEGMENTATION

SmoothGroupStreamManifestBehavior

Smooth Group Stream Manifest Behavior

DO_NOT_SEND
SEND

SmoothGroupTimestampOffsetMode

Smooth Group Timestamp Offset Mode

USE_CONFIGURED_OFFSET
USE_EVENT_START_DATE

Smpte2038DataPreference

Smpte2038 Data Preference

IGNORE
PREFER

SmpteTtDestinationSettings

Smpte Tt Destination Settings

StandardHlsSettings

Standard Hls Settings

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

m3u8Settings

Type: M3u8Settings
Required: True

StaticKeySettings

Static Key Settings

keyProviderServer

The URL of the license server used for protecting content.

Type: InputLocation
Required: False

staticKeyValue

Static key value as a 32 character hexadecimal string.

Type: string
Required: True
MinLength: 32

MaxLength: 32

Tags

key-value pairs

Type: string

TeletextDestinationSettings

Teletext Destination Settings

TeletextSourceSettings

Teletext Source Settings

outputRectangle

Optionally defines a region where TTML style captions will be displayed

Type: CaptionRectangle
Required: False

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

TemporalFilterPostFilterSharpening

Temporal Filter Post Filter Sharpening

AUTO
DISABLED
ENABLED

TemporalFilterSettings

Temporal Filter Settings

postFilterSharpening

If you enable this filter, the results are the following: - If the source content is noisy (it contains excessive digital artifacts), the filter cleans up the source. - If the source content is already clean, the filter tends to decrease the bitrate, especially when the rate control mode is QVBR.

Type: TemporalFilterPostFilterSharpening
Required: False

strength

Choose a filter strength. We recommend a strength of 1 or 2. A higher strength might take out good information, resulting in an image that is overly soft.

Type: TemporalFilterStrength
Required: False

TemporalFilterStrength

Temporal Filter Strength

AUTO
STRENGTH_1
STRENGTH_2
STRENGTH_3
STRENGTH_4
STRENGTH_5
STRENGTH_6
STRENGTH_7
STRENGTH_8
STRENGTH_9
STRENGTH_10
STRENGTH_11
STRENGTH_12

STRENGTH_13
STRENGTH_14
STRENGTH_15
STRENGTH_16

TimecodeConfig

Timecode Config

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: TimecodeConfigSource
Required: True

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

TimecodeConfigSource

Timecode Config Source

EMBEDDED
SYSTEMCLOCK
ZEROBASED

TtmlDestinationSettings

Ttml Destination Settings

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: TtmlDestinationStyleControl
Required: False

TtmlDestinationStyleControl

Ttml Destination Style Control

```
PASSTHROUGH
USE_CONFIGURED
```

UdpContainerSettings

Udp Container Settings

m2tsSettings

Type: M2tsSettings
Required: False

UdpGroupSettings

Udp Group Settings

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: InputLossActionForUdpOut
Required: False

timedMetadataId3Frame

Indicates ID3 frame that has the timecode.

Type: UdpTimedMetadataId3Frame
Required: False

timedMetadataId3Period

Timed Metadata interval in seconds.

Type: integer
Required: False
Minimum: 0

UdpOutputSettings

Udp Output Settings

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

containerSettings

Type: UdpContainerSettings
Required: True

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
Required: True

fecOutputSettings

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

Type: FecOutputSettings
Required: False

UdpTimedMetadataId3Frame

Udp Timed Metadata Id3 Frame

```
NONE
PRIV
TDRL
```

UpdateChannelClass

channelClass

The channel class that you wish to update this channel to use.

Type: ChannelClass
Required: True

destinations

A list of output destinations for this channel.

Type: Array of type OutputDestination
Required: False

UpdateChannelResultModel

The updated channel's description.

channel

Type: Channel
Required: False

ValidationError

elementPath

Path to the source of the error.

Type: string
Required: False

errorMessage

The error message.

Type: string
Required: False

VideoBlackFailoverSettings

blackDetectThreshold

A value used in calculating the threshold below which MediaLive considers a pixel to be 'black'. For the input to be considered black, every pixel in a frame must be below this threshold. The threshold is calculated as a percentage (expressed as a decimal) of white. Therefore .1 means 10% white (or 90% black). Note how the formula works for any color depth. For example, if you set this field to 0.1 in 10-bit color depth: (1023*0.1=102.3), which means a pixel value of 102 or less is 'black'. If you set this field to .1 in an 8-bit color depth: (255*0.1=25.5), which means a pixel value of 25 or less is 'black'. The range is 0.0 to 1.0, with any number of decimal places.

Type: number
Required: False
Minimum: 0

Maximum: 1

videoBlackThresholdMsec

The amount of time (in milliseconds) that the active input must be black before automatic input failover occurs.

Type: integer
Required: False
Minimum: 1000

VideoCodecSettings

Video Codec Settings

frameCaptureSettings

Type: FrameCaptureSettings
Required: False

h264Settings

Type: H264Settings
Required: False

h265Settings

Type: H265Settings
Required: False

mpeg2Settings

Type: Mpeg2Settings
Required: False

VideoDescription

Video settings for this stream.

codecSettings

Video codec settings.

Type: VideoCodecSettings
Required: False

height

Output video height, in pixels. Must be an even number. For most codecs, you can leave this field and width blank in order to use the height and width (resolution) from the source. Note, however, that leaving blank is not recommended. For the Frame Capture codec, height and width are required.

Type: integer
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

respondToAfd

Indicates how MediaLive will respond to the AFD values that might be in the input video. If you do not know what AFD signaling is, or if your downstream system has not given you guidance, choose PASSTHROUGH. RESPOND: MediaLive clips the input video using a formula that uses the AFD values (configured in afdSignaling), the input display aspect ratio, and the output display aspect ratio. MediaLive also includes the AFD values in the output, unless the codec for this encode is FRAME_CAPTURE. PASSTHROUGH: MediaLive ignores the AFD values and does not clip the video. But MediaLive does include the values in the output. NONE: MediaLive does not clip the input video and does not include the AFD values in the output

Type: VideoDescriptionRespondToAfd
Required: False

scalingBehavior

STRETCH_TO_OUTPUT configures the output position to stretch the video to the specified output resolution (height and width). This option will override any position value. DEFAULT may insert black boxes (pillar boxes or letter boxes) around the video to provide the specified output resolution.

Type: VideoDescriptionScalingBehavior
Required: False

sharpness

Changes the strength of the anti-alias filter used for scaling. 0 is the softest setting, 100 is the sharpest. A setting of 50 is recommended for most content.

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

width

Output video width, in pixels. Must be an even number. For most codecs, you can leave this field and height blank in order to use the height and width (resolution) from the source. Note, however, that leaving blank is not recommended. For the Frame Capture codec, height and width are required.

Type: integer
Required: False

VideoDescriptionRespondToAfd

Video Description Respond To Afd

NONE
PASSTHROUGH
RESPOND

VideoDescriptionScalingBehavior

Video Description Scaling Behavior

DEFAULT
STRETCH_TO_OUTPUT

VideoSelector

Specifies a particular video stream within an input source. An input may have only a single video selector.

colorSpace

Specifies the color space of an input. This setting works in tandem with colorSpaceUsage and a video description's colorSpaceSettingsChoice to determine if any conversion will be performed.

Type: VideoSelectorColorSpace
Required: False

colorSpaceSettings

Color space settings

Type: VideoSelectorColorSpaceSettings
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: VideoSelectorColorSpaceUsage
Required: False

selectorSettings

The video selector settings.

Type: VideoSelectorSettings
Required: False

VideoSelectorColorSpace

Video Selector Color Space

FOLLOW
HDR10
HLG_2020
REC_601
REC_709

VideoSelectorColorSpaceSettings

Video Selector Color Space Settings

hdr10Settings

Type: Hdr10Settings
Required: False

VideoSelectorColorSpaceUsage

Video Selector Color Space Usage

FALLBACK
FORCE

VideoSelectorPid

Video Selector Pid

pid

Selects a specific PID from within a video source.

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

VideoSelectorProgramId

Video Selector Program Id

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

Video Selector Settings

videoSelectorPid

Type: VideoSelectorPid
Required: False

videoSelectorProgramId

Type: VideoSelectorProgramId
Required: False

VpcOutputSettingsDescription

The properties for a private VPC Output

availabilityZones

The Availability Zones where the vpc subnets are located. The first Availability Zone applies to the first subnet in the list of subnets. The second Availability Zone applies to the second subnet.

Type: Array of type string
Required: False

networkInterfaceIds

A list of Elastic Network Interfaces created by MediaLive in the customer's VPC

Type: Array of type string
Required: False

securityGroupIds

A list of up EC2 VPC security group IDs attached to the Output VPC network interfaces.

Type: Array of type string
Required: False

subnetIds

A list of VPC subnet IDs from the same VPC. If STANDARD channel, subnet IDs must be mapped to two unique availability zones (AZ).

Type: Array of type string
Required: False

WavCodingMode

Wav Coding Mode

```
CODING_MODE_1_0
```

CODING_MODE_2_0
CODING_MODE_4_0
CODING_MODE_8_0

WavSettings

Wav Settings

bitDepth

Bits per sample.

Type: number
Required: False

codingMode

The audio coding mode for the WAV audio. The mode determines the number of channels in the audio.

Type: WavCodingMode
Required: False

sampleRate

Sample rate in Hz.

Type: number
Required: False

WebvttDestinationSettings

Webvtt Destination Settings

styleControl

Controls whether the color and position of the source captions is passed through to the WebVTT output captions. PASSTHROUGH - Valid only if the source captions are EMBEDDED or TELETEXT.

NO_STYLE_DATA - Don't pass through the style. The output captions will not contain any font styling information.

Type: WebvttDestinationStyleControl
Required: False

WebvttDestinationStyleControl

Webvtt Destination Style Control

```
NO_STYLE_DATA
PASSTHROUGH
```

See also

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

UpdateChannelClass

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

Channels: schedule

URI

`/prod/channels/channelId/schedule`

HTTP methods

DELETE

Operation ID: `DeleteSchedule`

Path parameters

Name	Type	Required	Description
channelId	String	True	

Responses

Status code	Response model	Description
200	ScheduleDeleteResultModel	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

GET

Operation ID: `DescribeSchedule`

Path parameters

Name	Type	Required	Description
channelId	String	True	

Query parameters

Name	Type	Required	Description
nextToken	String	False	
maxResults	String	False	

Responses

Status code	Response model	Description
200	ScheduleDescribeResultModel	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

PUT

Operation ID: BatchUpdateSchedule

Path parameters

Name	Type	Required	Description
channelId	String	True	

Responses

Status code	Response model	Description
200	BatchUpdateScheduleResult	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
422	ChannelConfigurationValidationError	422 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

Schemas

Request bodies

PUT schema

```
{
  "creates": {
    "scheduleActions": [
```

```
{
  "actionName": "string",
  "scheduleActionSettings": {
    "hlsId3SegmentTaggingSettings": {
      "tag": "string"
    },
    "hlsTimedMetadataSettings": {
      "id3": "string"
    },
    "inputPrepareSettings": {
      "inputAttachmentNameReference": "string",
      "inputClippingSettings": {
        "inputTimecodeSource": enum,
        "startTimecode": {
          "timecode": "string"
        },
        "stopTimecode": {
          "lastFrameClippingBehavior": enum,
          "timecode": "string"
        }
      },
      "urlPath": [
        "string"
      ]
    },
    "inputSwitchSettings": {
      "inputAttachmentNameReference": "string",
      "inputClippingSettings": {
        "inputTimecodeSource": enum,
        "startTimecode": {
          "timecode": "string"
        },
        "stopTimecode": {
          "lastFrameClippingBehavior": enum,
          "timecode": "string"
        }
      },
      "urlPath": [
        "string"
      ]
    },
    "motionGraphicsImageActivateSettings": {
      "duration": integer,
      "passwordParam": "string",
```

```
            "url": "string",
            "username": "string"
          },
          "motionGraphicsImageDeactivateSettings": {
          },
          "pauseStateSettings": {
            "pipelines": [
              {
                "pipelineId": enum
              }
            ]
          },
          "scte35ReturnToNetworkSettings": {
            "spliceEventId": integer
          },
          "scte35SpliceInsertSettings": {
            "duration": integer,
            "spliceEventId": integer
          },
          "scte35TimeSignalSettings": {
            "scte35Descriptors": [
              {
                "scte35DescriptorSettings": {
                  "segmentationDescriptorScte35DescriptorSettings": {
                    "deliveryRestrictions": {
                      "archiveAllowedFlag": enum,
                      "deviceRestrictions": enum,
                      "noRegionalBlackoutFlag": enum,
                      "webDeliveryAllowedFlag": enum
                    },
                    "segmentNum": integer,
                    "segmentationCancelIndicator": enum,
                    "segmentationDuration": integer,
                    "segmentationEventId": integer,
                    "segmentationTypeId": integer,
                    "segmentationUpid": "string",
                    "segmentationUpidType": integer,
                    "segmentsExpected": integer,
                    "subSegmentNum": integer,
                    "subSegmentsExpected": integer
                  }
                }
              }
            ]
```

```
          },
          "staticImageActivateSettings": {
            "duration": integer,
            "fadeIn": integer,
            "fadeOut": integer,
            "height": integer,
            "image": {
              "passwordParam": "string",
              "uri": "string",
              "username": "string"
            },
            "imageX": integer,
            "imageY": integer,
            "layer": integer,
            "opacity": integer,
            "width": integer
          },
          "staticImageDeactivateSettings": {
            "fadeOut": integer,
            "layer": integer
          }
        },
        "scheduleActionStartSettings": {
          "fixedModeScheduleActionStartSettings": {
            "time": "string"
          },
          "followModeScheduleActionStartSettings": {
            "followPoint": enum,
            "referenceActionName": "string"
          },
          "immediateModeScheduleActionStartSettings": {
            "time": "string"
          }
        }
      }
    ]
  },
  "deletes": {
    "actionNames": [
      "string"
    ]
  }
}
```

Response bodies

ScheduleDeleteResultModel schema

```
{
}
```

ScheduleDescribeResultModel schema

```
{
  "nextToken": "string",
  "scheduleActions": [
    {
      "actionName": "string",
      "scheduleActionSettings": {
        "hlsId3SegmentTaggingSettings": {
          "tag": "string"
        },
        "hlsTimedMetadataSettings": {
          "id3": "string"
        },
        "inputPrepareSettings": {
          "inputAttachmentNameReference": "string",
          "inputClippingSettings": {
            "inputTimecodeSource": enum,
            "startTimecode": {
              "timecode": "string"
            },
            "stopTimecode": {
              "lastFrameClippingBehavior": enum,
              "timecode": "string"
            }
          },
          "urlPath": [
            "string"
          ]
        },
        "inputSwitchSettings": {
          "inputAttachmentNameReference": "string",
          "inputClippingSettings": {
            "inputTimecodeSource": enum,
            "startTimecode": {
              "timecode": "string"
```

```
            },
            "stopTimecode": {
              "lastFrameClippingBehavior": enum,
              "timecode": "string"
            }
          },
          "urlPath": [
            "string"
          ]
        },
        "motionGraphicsImageActivateSettings": {
          "duration": integer,
          "passwordParam": "string",
          "url": "string",
          "username": "string"
        },
        "motionGraphicsImageDeactivateSettings": {
        },
        "pauseStateSettings": {
          "pipelines": [
            {
              "pipelineId": enum
            }
          ]
        },
        "scte35ReturnToNetworkSettings": {
          "spliceEventId": integer
        },
        "scte35SpliceInsertSettings": {
          "duration": integer,
          "spliceEventId": integer
        },
        "scte35TimeSignalSettings": {
          "scte35Descriptors": [
            {
              "scte35DescriptorSettings": {
                "segmentationDescriptorScte35DescriptorSettings": {
                  "deliveryRestrictions": {
                    "archiveAllowedFlag": enum,
                    "deviceRestrictions": enum,
                    "noRegionalBlackoutFlag": enum,
                    "webDeliveryAllowedFlag": enum
                  },
                  "segmentNum": integer,
```

```
                "segmentationCancelIndicator": enum,
                "segmentationDuration": integer,
                "segmentationEventId": integer,
                "segmentationTypeId": integer,
                "segmentationUpid": "string",
                "segmentationUpidType": integer,
                "segmentsExpected": integer,
                "subSegmentNum": integer,
                "subSegmentsExpected": integer
              }
            }
          }
        ]
      },
      "staticImageActivateSettings": {
        "duration": integer,
        "fadeIn": integer,
        "fadeOut": integer,
        "height": integer,
        "image": {
          "passwordParam": "string",
          "uri": "string",
          "username": "string"
        },
        "imageX": integer,
        "imageY": integer,
        "layer": integer,
        "opacity": integer,
        "width": integer
      },
      "staticImageDeactivateSettings": {
        "fadeOut": integer,
        "layer": integer
      }
    },
    "scheduleActionStartSettings": {
      "fixedModeScheduleActionStartSettings": {
        "time": "string"
      },
      "followModeScheduleActionStartSettings": {
        "followPoint": enum,
        "referenceActionName": "string"
      },
      "immediateModeScheduleActionStartSettings": {
```

```
          "time": "string"
        }
      }
    }
  ]
}
```

BatchUpdateScheduleResult schema

```
{
  "creates": {
    "scheduleActions": [
      {
        "actionName": "string",
        "scheduleActionSettings": {
          "hlsId3SegmentTaggingSettings": {
            "tag": "string"
          },
          "hlsTimedMetadataSettings": {
            "id3": "string"
          },
          "inputPrepareSettings": {
            "inputAttachmentNameReference": "string",
            "inputClippingSettings": {
              "inputTimecodeSource": enum,
              "startTimecode": {
                "timecode": "string"
              },
              "stopTimecode": {
                "lastFrameClippingBehavior": enum,
                "timecode": "string"
              }
            },
            "urlPath": [
              "string"
            ]
          },
          "inputSwitchSettings": {
            "inputAttachmentNameReference": "string",
            "inputClippingSettings": {
              "inputTimecodeSource": enum,
              "startTimecode": {
                "timecode": "string"
```

```
        },
        "stopTimecode": {
          "lastFrameClippingBehavior": enum,
          "timecode": "string"
        }
      },
      "urlPath": [
        "string"
      ]
    },
    "motionGraphicsImageActivateSettings": {
      "duration": integer,
      "passwordParam": "string",
      "url": "string",
      "username": "string"
    },
    "motionGraphicsImageDeactivateSettings": {
    },
    "pauseStateSettings": {
      "pipelines": [
        {
          "pipelineId": enum
        }
      ]
    },
    "scte35ReturnToNetworkSettings": {
      "spliceEventId": integer
    },
    "scte35SpliceInsertSettings": {
      "duration": integer,
      "spliceEventId": integer
    },
    "scte35TimeSignalSettings": {
      "scte35Descriptors": [
        {
          "scte35DescriptorSettings": {
            "segmentationDescriptorScte35DescriptorSettings": {
              "deliveryRestrictions": {
                "archiveAllowedFlag": enum,
                "deviceRestrictions": enum,
                "noRegionalBlackoutFlag": enum,
                "webDeliveryAllowedFlag": enum
              },
              "segmentNum": integer,
```

```
                    "segmentationCancelIndicator": enum,
                    "segmentationDuration": integer,
                    "segmentationEventId": integer,
                    "segmentationTypeId": integer,
                    "segmentationUpid": "string",
                    "segmentationUpidType": integer,
                    "segmentsExpected": integer,
                    "subSegmentNum": integer,
                    "subSegmentsExpected": integer
                  }
                }
              }
            ]
          },
          "staticImageActivateSettings": {
            "duration": integer,
            "fadeIn": integer,
            "fadeOut": integer,
            "height": integer,
            "image": {
              "passwordParam": "string",
              "uri": "string",
              "username": "string"
            },
            "imageX": integer,
            "imageY": integer,
            "layer": integer,
            "opacity": integer,
            "width": integer
          },
          "staticImageDeactivateSettings": {
            "fadeOut": integer,
            "layer": integer
          }
        },
        "scheduleActionStartSettings": {
          "fixedModeScheduleActionStartSettings": {
            "time": "string"
          },
          "followModeScheduleActionStartSettings": {
            "followPoint": enum,
            "referenceActionName": "string"
          },
          "immediateModeScheduleActionStartSettings": {
```

```
            "time": "string"
          }
        }
      }
    ]
  },
  "deletes": {
    "scheduleActions": [
      {
        "actionName": "string",
        "scheduleActionSettings": {
          "hlsId3SegmentTaggingSettings": {
            "tag": "string"
          },
          "hlsTimedMetadataSettings": {
            "id3": "string"
          },
          "inputPrepareSettings": {
            "inputAttachmentNameReference": "string",
            "inputClippingSettings": {
              "inputTimecodeSource": enum,
              "startTimecode": {
                "timecode": "string"
              },
              "stopTimecode": {
                "lastFrameClippingBehavior": enum,
                "timecode": "string"
              }
            },
            "urlPath": [
              "string"
            ]
          },
          "inputSwitchSettings": {
            "inputAttachmentNameReference": "string",
            "inputClippingSettings": {
              "inputTimecodeSource": enum,
              "startTimecode": {
                "timecode": "string"
              },
              "stopTimecode": {
                "lastFrameClippingBehavior": enum,
                "timecode": "string"
              }
```

```
        },
        "urlPath": [
          "string"
        ]
      },
      "motionGraphicsImageActivateSettings": {
        "duration": integer,
        "passwordParam": "string",
        "url": "string",
        "username": "string"
      },
      "motionGraphicsImageDeactivateSettings": {
      },
      "pauseStateSettings": {
        "pipelines": [
          {
            "pipelineId": enum
          }
        ]
      },
      "scte35ReturnToNetworkSettings": {
        "spliceEventId": integer
      },
      "scte35SpliceInsertSettings": {
        "duration": integer,
        "spliceEventId": integer
      },
      "scte35TimeSignalSettings": {
        "scte35Descriptors": [
          {
            "scte35DescriptorSettings": {
              "segmentationDescriptorScte35DescriptorSettings": {
                "deliveryRestrictions": {
                  "archiveAllowedFlag": enum,
                  "deviceRestrictions": enum,
                  "noRegionalBlackoutFlag": enum,
                  "webDeliveryAllowedFlag": enum
                },
                "segmentNum": integer,
                "segmentationCancelIndicator": enum,
                "segmentationDuration": integer,
                "segmentationEventId": integer,
                "segmentationTypeId": integer,
                "segmentationUpid": "string",
```

```
                    "segmentationUpidType": integer,
                    "segmentsExpected": integer,
                    "subSegmentNum": integer,
                    "subSegmentsExpected": integer
                  }
                }
              }
            ]
          },
          "staticImageActivateSettings": {
            "duration": integer,
            "fadeIn": integer,
            "fadeOut": integer,
            "height": integer,
            "image": {
              "passwordParam": "string",
              "uri": "string",
              "username": "string"
            },
            "imageX": integer,
            "imageY": integer,
            "layer": integer,
            "opacity": integer,
            "width": integer
          },
          "staticImageDeactivateSettings": {
            "fadeOut": integer,
            "layer": integer
          }
        },
        "scheduleActionStartSettings": {
          "fixedModeScheduleActionStartSettings": {
            "time": "string"
          },
          "followModeScheduleActionStartSettings": {
            "followPoint": enum,
            "referenceActionName": "string"
          },
          "immediateModeScheduleActionStartSettings": {
            "time": "string"
          }
        }
      }
    ]
```

```
  }
}
```

InvalidRequest schema

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

AccessDenied schema

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

ResourceNotFound schema

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

ChannelConfigurationValidationError schema

```
{
  "message": "string",
  "validationErrors": [
    {
      "elementPath": "string",
      "errorMessage": "string"
    }
  ]
}
```

LimitExceeded schema

```
{
```

```
  "message": "string"
}
```

InternalServiceError schema

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

BadGatewayException schema

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

GatewayTimeoutException schema

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

Properties

AccessDenied

message

Type: string

Required: False

BadGatewayException

message

Type: string

Required: False

BatchScheduleActionCreateRequest

A list of schedule actions to create (in a request) or that have been created (in a response).

scheduleActions

A list of schedule actions to create.

Type: Array of type ScheduleAction
Required: True

BatchScheduleActionCreateResult

List of actions that have been created in the schedule.

scheduleActions

List of actions that have been created in the schedule.

Type: Array of type ScheduleAction
Required: True

BatchScheduleActionDeleteRequest

A list of schedule actions to delete.

actionNames

A list of schedule actions to delete.

Type: Array of type string
Required: True

BatchScheduleActionDeleteResult

List of actions that have been deleted from the schedule.

scheduleActions

List of actions that have been deleted from the schedule.

Type: Array of type ScheduleAction
Required: True

BatchUpdateScheduleRequest

A request to create actions (add actions to the schedule), delete actions (remove actions from the schedule), or both create and delete actions.

creates

Schedule actions to create in the schedule.

Type: BatchScheduleActionCreateRequest
Required: False

deletes

Schedule actions to delete from the schedule.

Type: BatchScheduleActionDeleteRequest
Required: False

BatchUpdateScheduleResult

Results of a batch schedule update.

creates

Schedule actions created in the schedule.

Type: BatchScheduleActionCreateResult
Required: False

deletes

Schedule actions deleted from the schedule.

Type: BatchScheduleActionDeleteResult
Required: False

ChannelConfigurationValidationError

message

Type: string
Required: False

validationErrors

A collection of validation error responses.

Type: Array of type ValidationError
Required: False

FixedModeScheduleActionStartSettings

Start time for the action.

time

Start time for the action to start in the channel. (Not the time for the action to be added to the schedule: actions are always added to the schedule immediately.) UTC format: yyyy-mm-ddThh:mm:ss.nnnZ. All the letters are digits (for example, mm might be 01) except for the two constants "T" for time and "Z" for "UTC format".

Type: string
Required: True

FollowModeScheduleActionStartSettings

Settings to specify if an action follows another.

followPoint

Identifies whether this action starts relative to the start or relative to the end of the reference action.

Type: FollowPoint
Required: True

referenceActionName

The action name of another action that this one refers to.

Type: string
Required: True

FollowPoint

Follow reference point.

END
START

GatewayTimeoutException

message

Type: string
Required: False

HlsId3SegmentTaggingScheduleActionSettings

Settings for the action to insert a user-defined ID3 tag in each HLS segment

tag

ID3 tag to insert into each segment. Supports special keyword identifiers to substitute in segment-related values.\nSupported keyword identifiers: https://docs.aws.amazon.com/medialive/latest/ug/variable-data-identifiers.html

Type: string
Required: False

HlsTimedMetadataScheduleActionSettings

Settings for the action to emit HLS metadata

id3

Base64 string formatted according to the ID3 specification: http://id3.org/id3v2.4.0-structure

Type: string
Required: True

ImmediateModeScheduleActionStartSettings

Settings to configure an action so that it occurs as soon as possible.

time

The time that the immediate mode action is scheduled to occur. (MediaLive calculates this time to be as soon as possible after receiving the schedule)

Type: string
Required: False

InputClippingSettings

Settings to let you create a clip of the file input, in order to set up the input to ingest only a portion of the file.

inputTimecodeSource

The source of the timecodes in the source being clipped.

Type: InputTimecodeSource
Required: True

startTimecode

Settings to identify the start of the clip.

Type: StartTimecode
Required: False

stopTimecode

Settings to identify the end of the clip.

Type: StopTimecode
Required: False

InputLocation

Input Location

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 RTMP destination should have a uri simliar 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

InputPrepareScheduleActionSettings

Action to prepare an input for a future immediate input switch.

inputAttachmentNameReference

The name of the input attachment that should be prepared by this action. If no name is provided, the action will stop the most recent prepare (if any) when activated.

Type: string
Required: False

inputClippingSettings

Settings to let you create a clip of the file input, in order to set up the input to ingest only a portion of the file.

Type: InputClippingSettings
Required: False

urlPath

The value for the variable portion of the URL for the dynamic input, for this instance of the input. Each time you use the same dynamic input in an input switch action, you can provide a different value, in order to connect the input to a different content source.

Type: Array of type string
Required: False

InputSwitchScheduleActionSettings

Settings for the "switch input" action: to switch from ingesting one input to ingesting another input.

inputAttachmentNameReference

The name of the input attachment (not the name of the input!) to switch to. The name is specified in the channel configuration.

Type: string
Required: True

inputClippingSettings

Settings to let you create a clip of the file input, in order to set up the input to ingest only a portion of the file.

Type: InputClippingSettings
Required: False

urlPath

The value for the variable portion of the URL for the dynamic input, for this instance of the input. Each time you use the same dynamic input in an input switch action, you can provide a different value, in order to connect the input to a different content source.

Type: Array of type string
Required: False

InputTimecodeSource

To clip the file, you must specify the timecode for the start and end of the clip.
Specify EMBEDDED to use the timecode embedded in the source content. The embedded timecode must exist in the source content, otherwise MediaLive will output black frames until it reaches the end of the source.
Specify ZEROBASED to use a timecode that assumes that the first frame in the file has the timestamp 00:00:00.00.
 There is no default for this field, you must specify a value.

ZEROBASED
EMBEDDED

InternalServiceError

message

Type: string
Required: False

InvalidRequest

message

Type: string
Required: False

LastFrameClippingBehavior

If you specify a StopTimecode in an input (in order to clip the file), you can specify if you want the clip to exclude (the default) or include the frame specified by the timecode.

`EXCLUDE_LAST_FRAME`
`INCLUDE_LAST_FRAME`

LimitExceeded

message

Type: string
Required: False

MotionGraphicsActivateScheduleActionSettings

Settings to specify the rendering of motion graphics into the video stream.

duration

Duration (in milliseconds) that motion graphics should render on to the video stream. Leaving out this property or setting to 0 will result in rendering continuing until a deactivate action is processed.

Type: integer
Required: False
Format: int64
Minimum: 0
Maximum: 86400000

passwordParam

Key used to extract the password from EC2 Parameter store

Type: string
Required: False

url

URI of the HTML5 content to be rendered into the live stream.

Type: string
Required: False

username

Username if credentials are required to access a file. This must be a reference to an AWS parameter store name from which the password can be retrieved. AWS Parameter store format: \"ssm://<parameter name>\"

Type: string
Required: False

MotionGraphicsDeactivateScheduleActionSettings

Settings to specify the ending of rendering motion graphics into the video stream.

PauseStateScheduleActionSettings

Settings for the action to set pause state of a channel.

pipelines

Type: Array of type PipelinePauseStateSettings
Required: False

PipelineId

Pipeline ID

PIPELINE_0
PIPELINE_1

PipelinePauseStateSettings

Settings for pausing a pipeline.

pipelineId

Pipeline ID to pause ("PIPELINE_0" or "PIPELINE_1").

Type: PipelineId
Required: True

ResourceNotFound

message

Type: string
Required: False

ScheduleAction

Contains information on a single schedule action.

actionName

The name of the action, must be unique within the schedule. This name provides the main reference to an action once it is added to the schedule. A name is unique if it is no longer in the schedule. The schedule is automatically cleaned up to remove actions with a start time of more than 1 hour ago (approximately) so at that point a name can be reused.

Type: string
Required: True

scheduleActionSettings

Settings for this schedule action.

Type: ScheduleActionSettings

Required: True

scheduleActionStartSettings

The time for the action to start in the channel.

Type: ScheduleActionStartSettings
Required: True

ScheduleActionSettings

Holds the settings for a single schedule action.

hlsId3SegmentTaggingSettings

Action to insert HLS ID3 segment tagging

Type: HlsId3SegmentTaggingScheduleActionSettings
Required: False

hlsTimedMetadataSettings

Action to insert HLS metadata

Type: HlsTimedMetadataScheduleActionSettings
Required: False

inputPrepareSettings

Action to prepare an input for a future immediate input switch

Type: InputPrepareScheduleActionSettings
Required: False

inputSwitchSettings

Action to switch the input

Type: InputSwitchScheduleActionSettings
Required: False

motionGraphicsImageActivateSettings

Action to activate a motion graphics image overlay

Type: MotionGraphicsActivateScheduleActionSettings
Required: False

motionGraphicsImageDeactivateSettings

Action to deactivate a motion graphics image overlay

Type: MotionGraphicsDeactivateScheduleActionSettings
Required: False

pauseStateSettings

Action to pause or unpause one or both channel pipelines

Type: PauseStateScheduleActionSettings
Required: False

scte35ReturnToNetworkSettings

Action to insert SCTE-35 return_to_network message

Type: Scte35ReturnToNetworkScheduleActionSettings
Required: False

scte35SpliceInsertSettings

Action to insert SCTE-35 splice_insert message

Type: Scte35SpliceInsertScheduleActionSettings
Required: False

scte35TimeSignalSettings

Action to insert SCTE-35 time_signal message

Type: Scte35TimeSignalScheduleActionSettings

Required: False

staticImageActivateSettings

Action to activate a static image overlay

Type: StaticImageActivateScheduleActionSettings
Required: False

staticImageDeactivateSettings

Action to deactivate a static image overlay

Type: StaticImageDeactivateScheduleActionSettings
Required: False

ScheduleActionStartSettings

Settings to specify when an action should occur. Only one of the options must be selected.

fixedModeScheduleActionStartSettings

Option for specifying the start time for an action.

Type: FixedModeScheduleActionStartSettings
Required: False

followModeScheduleActionStartSettings

Option for specifying an action as relative to another action.

Type: FollowModeScheduleActionStartSettings
Required: False

immediateModeScheduleActionStartSettings

Option for specifying an action that should be applied immediately.

Type: ImmediateModeScheduleActionStartSettings
Required: False

ScheduleDeleteResultModel

Result of a schedule deletion.

ScheduleDescribeResultModel

Results of a schedule describe.

nextToken

The next token; for use in pagination.

Type: string
Required: False

scheduleActions

The list of actions in the schedule.

Type: Array of type ScheduleAction
Required: True

Scte35ArchiveAllowedFlag

Corresponds to the archive_allowed parameter. A value of ARCHIVE_NOT_ALLOWED corresponds to 0 (false) in the SCTE-35 specification. If you include one of the "restriction" flags then you must include all four of them.

```
ARCHIVE_NOT_ALLOWED
ARCHIVE_ALLOWED
```

Scte35DeliveryRestrictions

Corresponds to SCTE-35 delivery_not_restricted_flag parameter. To declare delivery restrictions, include this element and its four "restriction" flags. To declare that there are no restrictions, omit this element.

archiveAllowedFlag

Corresponds to SCTE-35 archive_allowed_flag.

Type: Scte35ArchiveAllowedFlag
Required: True

deviceRestrictions

Corresponds to SCTE-35 device_restrictions parameter.

Type: Scte35DeviceRestrictions
Required: True

noRegionalBlackoutFlag

Corresponds to SCTE-35 no_regional_blackout_flag parameter.

Type: Scte35NoRegionalBlackoutFlag
Required: True

webDeliveryAllowedFlag

Corresponds to SCTE-35 web_delivery_allowed_flag parameter.

Type: Scte35WebDeliveryAllowedFlag
Required: True

Scte35Descriptor

Holds one set of SCTE-35 Descriptor Settings.

scte35DescriptorSettings

SCTE-35 Descriptor Settings.

Type: Scte35DescriptorSettings
Required: True

Scte35DescriptorSettings

SCTE-35 Descriptor settings.

segmentationDescriptorScte35DescriptorSettings

SCTE-35 Segmentation Descriptor.

Type: Scte35SegmentationDescriptor
Required: True

Scte35DeviceRestrictions

Corresponds to the device_restrictions parameter in a segmentation_descriptor. If you include one of the "restriction" flags then you must include all four of them.

```
NONE
RESTRICT_GROUP0
RESTRICT_GROUP1
RESTRICT_GROUP2
```

Scte35NoRegionalBlackoutFlag

Corresponds to the no_regional_blackout_flag parameter. A value of REGIONAL_BLACKOUT corresponds to 0 (false) in the SCTE-35 specification. If you include one of the "restriction" flags then you must include all four of them.

```
REGIONAL_BLACKOUT
NO_REGIONAL_BLACKOUT
```

Scte35ReturnToNetworkScheduleActionSettings

Settings for a SCTE-35 return_to_network message.

spliceEventId

The splice_event_id for the SCTE-35 splice_insert, as defined in SCTE-35.

Type: integer
Required: True
Format: int64
Minimum: 0
Maximum: 4294967295

Scte35SegmentationCancelIndicator

Corresponds to SCTE-35 segmentation_event_cancel_indicator. SEGMENTATION_EVENT_NOT_CANCELED corresponds to 0 in the SCTE-35 specification and indicates that this is an insertion request. SEGMENTATION_EVENT_CANCELED corresponds to 1 in the SCTE-35 specification and indicates that this is a cancelation request, in which case complete this field and the existing event ID to cancel.

`SEGMENTATION_EVENT_NOT_CANCELED`
`SEGMENTATION_EVENT_CANCELED`

Scte35SegmentationDescriptor

Corresponds to SCTE-35 segmentation_descriptor.

deliveryRestrictions

Holds the four SCTE-35 delivery restriction parameters.

Type: Scte35DeliveryRestrictions
Required: False

segmentNum

Corresponds to SCTE-35 segment_num. A value that is valid for the specified segmentation_type_id.

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

segmentationCancelIndicator

Corresponds to SCTE-35 segmentation_event_cancel_indicator.

Type: Scte35SegmentationCancelIndicator
Required: True

segmentationDuration

Corresponds to SCTE-35 segmentation_duration. Optional. The duration for the time_signal, in 90 KHz ticks. To convert seconds to ticks, multiple the seconds by 90,000. Enter time in 90 KHz clock ticks. If you do not enter a duration, the time_signal will continue until you insert a cancellation message.

Type: integer
Required: False
Format: int64
Minimum: 0
Maximum: 1099511627775

segmentationEventId

Corresponds to SCTE-35 segmentation_event_id.

Type: integer
Required: True
Format: int64
Minimum: 0
Maximum: 4294967295

segmentationTypeId

Corresponds to SCTE-35 segmentation_type_id. One of the segmentation_type_id values listed in the SCTE-35 specification. On the console, enter the ID in decimal (for example, "52"). In the CLI, API, or an SDK, enter the ID in hex (for example, "0x34") or decimal (for example, "52").

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

segmentationUpid

Corresponds to SCTE-35 segmentation_upid. Enter a string containing the hexadecimal representation of the characters that make up the SCTE-35 segmentation_upid value. Must contain

an even number of hex characters. Do not include spaces between each hex pair. For example, the ASCII "ADS Information" becomes hex "41445320496e666f726d6174696f6e.

Type: string
Required: False

segmentationUpidType

Corresponds to SCTE-35 segmentation_upid_type. On the console, enter one of the types listed in the SCTE-35 specification, converted to a decimal. For example, "0x0C" hex from the specification is "12" in decimal. In the CLI, API, or an SDK, enter one of the types listed in the SCTE-35 specification, in either hex (for example, "0x0C") or in decimal (for example, "12").

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

segmentsExpected

Corresponds to SCTE-35 segments_expected. A value that is valid for the specified segmentation_type_id.

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

subSegmentNum

Corresponds to SCTE-35 sub_segment_num. A value that is valid for the specified segmentation_type_id.

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

subSegmentsExpected

Corresponds to SCTE-35 sub_segments_expected. A value that is valid for the specified segmentation_type_id.

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

Scte35SpliceInsertScheduleActionSettings

Settings for a SCTE-35 splice_insert message.

duration

Optional, the duration for the splice_insert, in 90 KHz ticks. To convert seconds to ticks, multiple the seconds by 90,000. If you enter a duration, there is an expectation that the downstream system can read the duration and cue in at that time. If you do not enter a duration, the splice_insert will continue indefinitely and there is an expectation that you will enter a return_to_network to end the splice_insert at the appropriate time.

Type: integer
Required: False
Format: int64
Minimum: 0
Maximum: 8589934591

spliceEventId

The splice_event_id for the SCTE-35 splice_insert, as defined in SCTE-35.

Type: integer
Required: True
Format: int64
Minimum: 0
Maximum: 4294967295

Scte35TimeSignalScheduleActionSettings

Settings for a SCTE-35 time_signal.

scte35Descriptors

The list of SCTE-35 descriptors accompanying the SCTE-35 time_signal.

Type: Array of type Scte35Descriptor
Required: True

Scte35WebDeliveryAllowedFlag

Corresponds to the web_delivery_allowed_flag parameter. A value of WEB_DELIVERY_NOT_ALLOWED corresponds to 0 (false) in the SCTE-35 specification. If you include one of the "restriction" flags then you must include all four of them.

```
WEB_DELIVERY_NOT_ALLOWED
WEB_DELIVERY_ALLOWED
```

StartTimecode

Settings to identify the start of the clip.

timecode

The timecode for the frame where you want to start the clip. Optional; if not specified, the clip starts at first frame in the file. Enter the timecode as HH:MM:SS:FF or HH:MM:SS;FF.

Type: string
Required: False

StaticImageActivateScheduleActionSettings

Settings for the action to activate a static image.

duration

The duration in milliseconds for the image to remain on the video. If omitted or set to 0 the duration is unlimited and the image will remain until it is explicitly deactivated.

Type: integer
Required: False
Minimum: 0

fadeIn

The time in milliseconds for the image to fade in. The fade-in starts at the start time of the overlay. Default is 0 (no fade-in).

Type: integer
Required: False
Minimum: 0

fadeOut

Applies only if a duration is specified. The time in milliseconds for the image to fade out. The fade-out starts when the duration time is hit, so it effectively extends the duration. Default is 0 (no fade-out).

Type: integer
Required: False
Minimum: 0

height

The height of the image when inserted into the video, in pixels. The overlay will be scaled up or down to the specified height. Leave blank to use the native height of the overlay.

Type: integer
Required: False
Minimum: 1

image

The location and filename of the image file to overlay on the video. The file must be a 32-bit BMP, PNG, or TGA file, and must not be larger (in pixels) than the input video.

Type: InputLocation

Required: True

imageX

Placement of the left edge of the overlay relative to the left edge of the video frame, in pixels. 0 (the default) is the left edge of the frame. If the placement causes the overlay to extend beyond the right edge of the underlying video, then the overlay is cropped on the right.

Type: integer
Required: False
Minimum: 0

imageY

Placement of the top edge of the overlay relative to the top edge of the video frame, in pixels. 0 (the default) is the top edge of the frame. If the placement causes the overlay to extend beyond the bottom edge of the underlying video, then the overlay is cropped on the bottom.

Type: integer
Required: False
Minimum: 0

layer

The number of the layer, 0 to 7. There are 8 layers that can be overlaid on the video, each layer with a different image. The layers are in Z order, which means that overlays with higher values of layer are inserted on top of overlays with lower values of layer. Default is 0.

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

opacity

Opacity of image where 0 is transparent and 100 is fully opaque. Default is 100.

Type: integer

Required: False
Minimum: 0
Maximum: 100

width

The width of the image when inserted into the video, in pixels. The overlay will be scaled up or down to the specified width. Leave blank to use the native width of the overlay.

Type: integer
Required: False
Minimum: 1

StaticImageDeactivateScheduleActionSettings

Settings for the action to deactivate the image in a specific layer.

fadeOut

The time in milliseconds for the image to fade out. Default is 0 (no fade-out).

Type: integer
Required: False
Minimum: 0

layer

The image overlay layer to deactivate, 0 to 7. Default is 0.

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

StopTimecode

Settings to identify the end of the clip.

lastFrameClippingBehavior

If you specify a StopTimecode in an input (in order to clip the file), you can specify if you want the clip to exclude (the default) or include the frame specified by the timecode.

Type: LastFrameClippingBehavior
Required: False

timecode

The timecode for the frame where you want to stop the clip. Optional; if not specified, the clip continues to the end of the file. Enter the timecode as HH:MM:SS:FF or HH:MM:SS;FF.

Type: string
Required: False

ValidationError

elementPath

Path to the source of the error.

Type: string
Required: False

errorMessage

The error message.

Type: string
Required: False

See also

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

DeleteSchedule

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

DescribeSchedule

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

BatchUpdateSchedule

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2

- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Channels: start

URI

`/prod/channels/channelId/start`

HTTP methods

POST

Operation ID: `StartChannel`

Path parameters

Name	Type	Required	Description
channelId	String	True	

Responses

Status code	Response model	Description
200	Channel	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
409	ResourceConflict	409 response
429	LimitExceeded	429 response

Status code	Response model	Description
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

Schemas

Response bodies

Channel schema

```
{
  "arn": "string",
  "cdiInputSpecification": {
    "resolution": enum
  },
  "channelClass": enum,
  "destinations": [
    {
      "id": "string",
      "mediaPackageSettings": [
        {
          "channelId": "string"
        }
      ],
      "multiplexSettings": {
        "multiplexId": "string",
        "programName": "string"
      },
      "settings": [
        {
          "passwordParam": "string",
          "streamName": "string",
          "url": "string",
          "username": "string"
        }
      ]
```

```
    }
  ],
  "egressEndpoints": [
    {
      "sourceIp": "string"
    }
  ],
  "encoderSettings": {
    "audioDescriptions": [
      {
        "audioNormalizationSettings": {
          "algorithm": enum,
          "algorithmControl": enum,
          "targetLkfs": number
        },
        "audioSelectorName": "string",
        "audioType": enum,
        "audioTypeControl": enum,
        "audioWatermarkingSettings": {
          "nielsenWatermarksSettings": {
            "nielsenCbetSettings": {
              "cbetCheckDigitString": "string",
              "cbetStepaside": enum,
              "csid": "string"
            },
            "nielsenDistributionType": enum,
            "nielsenNaesIiNwSettings": {
              "checkDigitString": "string",
              "sid": number
            }
          }
        },
        "codecSettings": {
          "aacSettings": {
            "bitrate": number,
            "codingMode": enum,
            "inputType": enum,
            "profile": enum,
            "rateControlMode": enum,
            "rawFormat": enum,
            "sampleRate": number,
            "spec": enum,
            "vbrQuality": enum
          },
```

```
        "ac3Settings": {
          "bitrate": number,
          "bitstreamMode": enum,
          "codingMode": enum,
          "dialnorm": integer,
          "drcProfile": enum,
          "lfeFilter": enum,
          "metadataControl": enum
        },
        "eac3Settings": {
          "attenuationControl": enum,
          "bitrate": number,
          "bitstreamMode": enum,
          "codingMode": enum,
          "dcFilter": enum,
          "dialnorm": integer,
          "drcLine": enum,
          "drcRf": enum,
          "lfeControl": enum,
          "lfeFilter": enum,
          "loRoCenterMixLevel": number,
          "loRoSurroundMixLevel": number,
          "ltRtCenterMixLevel": number,
          "ltRtSurroundMixLevel": number,
          "metadataControl": enum,
          "passthroughControl": enum,
          "phaseControl": enum,
          "stereoDownmix": enum,
          "surroundExMode": enum,
          "surroundMode": enum
        },
        "mp2Settings": {
          "bitrate": number,
          "codingMode": enum,
          "sampleRate": number
        },
        "passThroughSettings": {
        },
        "wavSettings": {
          "bitDepth": number,
          "codingMode": enum,
          "sampleRate": number
        }
      },
```

```
        "languageCode": "string",
        "languageCodeControl": enum,
        "name": "string",
        "remixSettings": {
          "channelMappings": [
            {
              "inputChannelLevels": [
                {
                  "gain": integer,
                  "inputChannel": integer
                }
              ],
              "outputChannel": integer
            }
          ],
          "channelsIn": integer,
          "channelsOut": integer
        },
        "streamName": "string"
      }
    ],
    "availBlanking": {
      "availBlankingImage": {
        "passwordParam": "string",
        "uri": "string",
        "username": "string"
      },
      "state": enum
    },
    "availConfiguration": {
      "availSettings": {
        "scte35SpliceInsert": {
          "adAvailOffset": integer,
          "noRegionalBlackoutFlag": enum,
          "webDeliveryAllowedFlag": enum
        },
        "scte35TimeSignalApos": {
          "adAvailOffset": integer,
          "noRegionalBlackoutFlag": enum,
          "webDeliveryAllowedFlag": enum
        }
      }
    },
    "blackoutSlate": {
```

```
      "blackoutSlateImage": {
        "passwordParam": "string",
        "uri": "string",
        "username": "string"
      },
      "networkEndBlackout": enum,
      "networkEndBlackoutImage": {
        "passwordParam": "string",
        "uri": "string",
        "username": "string"
      },
      "networkId": "string",
      "state": enum
    },
    "captionDescriptions": [
      {
        "captionSelectorName": "string",
        "destinationSettings": {
          "aribDestinationSettings": {
          },
          "burnInDestinationSettings": {
            "alignment": enum,
            "backgroundColor": enum,
            "backgroundOpacity": integer,
            "font": {
              "passwordParam": "string",
              "uri": "string",
              "username": "string"
            },
            "fontColor": enum,
            "fontOpacity": integer,
            "fontResolution": integer,
            "fontSize": "string",
            "outlineColor": enum,
            "outlineSize": integer,
            "shadowColor": enum,
            "shadowOpacity": integer,
            "shadowXOffset": integer,
            "shadowYOffset": integer,
            "teletextGridControl": enum,
            "xPosition": integer,
            "yPosition": integer
          },
          "dvbSubDestinationSettings": {
```

```
            "alignment": enum,
            "backgroundColor": enum,
            "backgroundOpacity": integer,
            "font": {
              "passwordParam": "string",
              "uri": "string",
              "username": "string"
            },
            "fontColor": enum,
            "fontOpacity": integer,
            "fontResolution": integer,
            "fontSize": "string",
            "outlineColor": enum,
            "outlineSize": integer,
            "shadowColor": enum,
            "shadowOpacity": integer,
            "shadowXOffset": integer,
            "shadowYOffset": integer,
            "teletextGridControl": enum,
            "xPosition": integer,
            "yPosition": integer
          },
          "ebuTtDDestinationSettings": {
            "copyrightHolder": "string",
            "fillLineGap": enum,
            "fontFamily": "string",
            "styleControl": enum
          },
          "embeddedDestinationSettings": {
          },
          "embeddedPlusScte20DestinationSettings": {
          },
          "rtmpCaptionInfoDestinationSettings": {
          },
          "scte20PlusEmbeddedDestinationSettings": {
          },
          "scte27DestinationSettings": {
          },
          "smpteTtDestinationSettings": {
          },
          "teletextDestinationSettings": {
          },
          "ttmlDestinationSettings": {
            "styleControl": enum
```

```
        },
        "webvttDestinationSettings": {
          "styleControl": enum
        }
      },
      "languageCode": "string",
      "languageDescription": "string",
      "name": "string"
    }
  ],
  "featureActivations": {
    "inputPrepareScheduleActions": enum
  },
  "globalConfiguration": {
    "initialAudioGain": integer,
    "inputEndAction": enum,
    "inputLossBehavior": {
      "blackFrameMsec": integer,
      "inputLossImageColor": "string",
      "inputLossImageSlate": {
        "passwordParam": "string",
        "uri": "string",
        "username": "string"
      },
      "inputLossImageType": enum,
      "repeatFrameMsec": integer
    },
    "outputLockingMode": enum,
    "outputTimingSource": enum,
    "supportLowFramerateInputs": enum
  },
  "motionGraphicsConfiguration": {
    "motionGraphicsInsertion": enum,
    "motionGraphicsSettings": {
      "htmlMotionGraphicsSettings": {
      }
    }
  },
  "nielsenConfiguration": {
    "distributorId": "string",
    "nielsenPcmToId3Tagging": enum
  },
  "outputGroups": [
    {
```

```
        "name": "string",
        "outputGroupSettings": {
          "archiveGroupSettings": {
            "archiveCdnSettings": {
              "archiveS3Settings": {
                "cannedAcl": enum,
                "logUploads": enum
              }
            },
            "destination": {
              "destinationRefId": "string"
            },
            "rolloverInterval": integer
          },
          "frameCaptureGroupSettings": {
            "destination": {
              "destinationRefId": "string"
            },
            "frameCaptureCdnSettings": {
              "frameCaptureS3Settings": {
                "cannedAcl": enum,
                "logUploads": enum
              }
            }
          },
          "hlsGroupSettings": {
            "adMarkers": [
              enum
            ],
            "baseUrlContent": "string",
            "baseUrlContent1": "string",
            "baseUrlManifest": "string",
            "baseUrlManifest1": "string",
            "captionLanguageMappings": [
              {
                "captionChannel": integer,
                "languageCode": "string",
                "languageDescription": "string"
              }
            ],
            "captionLanguageSetting": enum,
            "clientCache": enum,
            "codecSpecification": enum,
            "constantIv": "string",
```

```
        "destination": {
          "destinationRefId": "string"
        },
        "directoryStructure": enum,
        "discontinuityTags": enum,
        "encryptionType": enum,
        "hlsCdnSettings": {
          "hlsAkamaiSettings": {
            "connectionRetryInterval": integer,
            "filecacheDuration": integer,
            "httpTransferMode": enum,
            "numRetries": integer,
            "restartDelay": integer,
            "salt": "string",
            "token": "string"
          },
          "hlsBasicPutSettings": {
            "connectionRetryInterval": integer,
            "filecacheDuration": integer,
            "numRetries": integer,
            "restartDelay": integer
          },
          "hlsMediaStoreSettings": {
            "connectionRetryInterval": integer,
            "filecacheDuration": integer,
            "mediaStoreStorageClass": enum,
            "numRetries": integer,
            "restartDelay": integer
          },
          "hlsS3Settings": {
            "cannedAcl": enum,
            "logUploads": enum
          },
          "hlsWebdavSettings": {
            "connectionRetryInterval": integer,
            "filecacheDuration": integer,
            "httpTransferMode": enum,
            "numRetries": integer,
            "restartDelay": integer
          }
        },
        "hlsId3SegmentTagging": enum,
        "iFrameOnlyPlaylists": enum,
        "incompleteSegmentBehavior": enum,
```

```
            "indexNSegments": integer,
            "inputLossAction": enum,
            "ivInManifest": enum,
            "ivSource": enum,
            "keepSegments": integer,
            "keyFormat": "string",
            "keyFormatVersions": "string",
            "keyProviderSettings": {
              "staticKeySettings": {
                "keyProviderServer": {
                  "passwordParam": "string",
                  "uri": "string",
                  "username": "string"
                },
                "staticKeyValue": "string"
              }
            },
            "manifestCompression": enum,
            "manifestDurationFormat": enum,
            "minSegmentLength": integer,
            "mode": enum,
            "outputSelection": enum,
            "programDateTime": enum,
            "programDateTimeClock": enum,
            "programDateTimePeriod": integer,
            "redundantManifest": enum,
            "segmentLength": integer,
            "segmentationMode": enum,
            "segmentsPerSubdirectory": integer,
            "streamInfResolution": enum,
            "timedMetadataId3Frame": enum,
            "timedMetadataId3Period": integer,
            "timestampDeltaMilliseconds": integer,
            "tsFileMode": enum
          },
          "mediaPackageGroupSettings": {
            "destination": {
              "destinationRefId": "string"
            }
          },
          "msSmoothGroupSettings": {
            "acquisitionPointId": "string",
            "audioOnlyTimecodeControl": enum,
            "certificateMode": enum,
```

```
            "connectionRetryInterval": integer,
            "destination": {
              "destinationRefId": "string"
            },
            "eventId": "string",
            "eventIdMode": enum,
            "eventStopBehavior": enum,
            "filecacheDuration": integer,
            "fragmentLength": integer,
            "inputLossAction": enum,
            "numRetries": integer,
            "restartDelay": integer,
            "segmentationMode": enum,
            "sendDelayMs": integer,
            "sparseTrackType": enum,
            "streamManifestBehavior": enum,
            "timestampOffset": "string",
            "timestampOffsetMode": enum
          },
          "multiplexGroupSettings": {
          },
          "rtmpGroupSettings": {
            "adMarkers": [
              enum
            ],
            "authenticationScheme": enum,
            "cacheFullBehavior": enum,
            "cacheLength": integer,
            "captionData": enum,
            "inputLossAction": enum,
            "restartDelay": integer
          },
          "udpGroupSettings": {
            "inputLossAction": enum,
            "timedMetadataId3Frame": enum,
            "timedMetadataId3Period": integer
          }
        },
        "outputs": [
          {
            "audioDescriptionNames": [
              "string"
            ],
            "captionDescriptionNames": [
```

```
            "string"
          ],
          "outputName": "string",
          "outputSettings": {
            "archiveOutputSettings": {
              "containerSettings": {
                "m2tsSettings": {
                  "absentInputAudioBehavior": enum,
                  "arib": enum,
                  "aribCaptionsPid": "string",
                  "aribCaptionsPidControl": enum,
                  "audioBufferModel": enum,
                  "audioFramesPerPes": integer,
                  "audioPids": "string",
                  "audioStreamType": enum,
                  "bitrate": integer,
                  "bufferModel": enum,
                  "ccDescriptor": enum,
                  "dvbNitSettings": {
                    "networkId": integer,
                    "networkName": "string",
                    "repInterval": integer
                  },
                  "dvbSdtSettings": {
                    "outputSdt": enum,
                    "repInterval": integer,
                    "serviceName": "string",
                    "serviceProviderName": "string"
                  },
                  "dvbSubPids": "string",
                  "dvbTdtSettings": {
                    "repInterval": integer
                  },
                  "dvbTeletextPid": "string",
                  "ebif": enum,
                  "ebpAudioInterval": enum,
                  "ebpLookaheadMs": integer,
                  "ebpPlacement": enum,
                  "ecmPid": "string",
                  "esRateInPes": enum,
                  "etvPlatformPid": "string",
                  "etvSignalPid": "string",
                  "fragmentTime": number,
                  "klv": enum,
```

```
            "klvDataPids": "string",
            "nielsenId3Behavior": enum,
            "nullPacketBitrate": number,
            "patInterval": integer,
            "pcrControl": enum,
            "pcrPeriod": integer,
            "pcrPid": "string",
            "pmtInterval": integer,
            "pmtPid": "string",
            "programNum": integer,
            "rateMode": enum,
            "scte27Pids": "string",
            "scte35Control": enum,
            "scte35Pid": "string",
            "segmentationMarkers": enum,
            "segmentationStyle": enum,
            "segmentationTime": number,
            "timedMetadataBehavior": enum,
            "timedMetadataPid": "string",
            "transportStreamId": integer,
            "videoPid": "string"
          },
          "rawSettings": {
          }
        },
        "extension": "string",
        "nameModifier": "string"
      },
      "frameCaptureOutputSettings": {
        "nameModifier": "string"
      },
      "hlsOutputSettings": {
        "h265PackagingType": enum,
        "hlsSettings": {
          "audioOnlyHlsSettings": {
            "audioGroupId": "string",
            "audioOnlyImage": {
              "passwordParam": "string",
              "uri": "string",
              "username": "string"
            },
            "audioTrackType": enum,
            "segmentType": enum
          },
```

```
            "fmp4HlsSettings": {
              "audioRenditionSets": "string",
              "nielsenId3Behavior": enum,
              "timedMetadataBehavior": enum
            },
            "frameCaptureHlsSettings": {
            },
            "standardHlsSettings": {
              "audioRenditionSets": "string",
              "m3u8Settings": {
                "audioFramesPerPes": integer,
                "audioPids": "string",
                "ecmPid": "string",
                "nielsenId3Behavior": enum,
                "patInterval": integer,
                "pcrControl": enum,
                "pcrPeriod": integer,
                "pcrPid": "string",
                "pmtInterval": integer,
                "pmtPid": "string",
                "programNum": integer,
                "scte35Behavior": enum,
                "scte35Pid": "string",
                "timedMetadataBehavior": enum,
                "timedMetadataPid": "string",
                "transportStreamId": integer,
                "videoPid": "string"
              }
            }
          },
          "nameModifier": "string",
          "segmentModifier": "string"
        },
        "mediaPackageOutputSettings": {
        },
        "msSmoothOutputSettings": {
          "h265PackagingType": enum,
          "nameModifier": "string"
        },
        "multiplexOutputSettings": {
          "destination": {
            "destinationRefId": "string"
          }
        },
```

```
        "rtmpOutputSettings": {
          "certificateMode": enum,
          "connectionRetryInterval": integer,
          "destination": {
            "destinationRefId": "string"
          },
          "numRetries": integer
        },
        "udpOutputSettings": {
          "bufferMsec": integer,
          "containerSettings": {
            "m2tsSettings": {
              "absentInputAudioBehavior": enum,
              "arib": enum,
              "aribCaptionsPid": "string",
              "aribCaptionsPidControl": enum,
              "audioBufferModel": enum,
              "audioFramesPerPes": integer,
              "audioPids": "string",
              "audioStreamType": enum,
              "bitrate": integer,
              "bufferModel": enum,
              "ccDescriptor": enum,
              "dvbNitSettings": {
                "networkId": integer,
                "networkName": "string",
                "repInterval": integer
              },
              "dvbSdtSettings": {
                "outputSdt": enum,
                "repInterval": integer,
                "serviceName": "string",
                "serviceProviderName": "string"
              },
              "dvbSubPids": "string",
              "dvbTdtSettings": {
                "repInterval": integer
              },
              "dvbTeletextPid": "string",
              "ebif": enum,
              "ebpAudioInterval": enum,
              "ebpLookaheadMs": integer,
              "ebpPlacement": enum,
              "ecmPid": "string",
```

```
                    "esRateInPes": enum,
                    "etvPlatformPid": "string",
                    "etvSignalPid": "string",
                    "fragmentTime": number,
                    "klv": enum,
                    "klvDataPids": "string",
                    "nielsenId3Behavior": enum,
                    "nullPacketBitrate": number,
                    "patInterval": integer,
                    "pcrControl": enum,
                    "pcrPeriod": integer,
                    "pcrPid": "string",
                    "pmtInterval": integer,
                    "pmtPid": "string",
                    "programNum": integer,
                    "rateMode": enum,
                    "scte27Pids": "string",
                    "scte35Control": enum,
                    "scte35Pid": "string",
                    "segmentationMarkers": enum,
                    "segmentationStyle": enum,
                    "segmentationTime": number,
                    "timedMetadataBehavior": enum,
                    "timedMetadataPid": "string",
                    "transportStreamId": integer,
                    "videoPid": "string"
                  }
                },
                "destination": {
                  "destinationRefId": "string"
                },
                "fecOutputSettings": {
                  "columnDepth": integer,
                  "includeFec": enum,
                  "rowLength": integer
                }
              }
            },
            "videoDescriptionName": "string"
          }
        ]
      }
    ],
    "timecodeConfig": {
```

```
      "source": enum,
      "syncThreshold": integer
    },
    "videoDescriptions": [
      {
        "codecSettings": {
          "frameCaptureSettings": {
            "captureInterval": integer,
            "captureIntervalUnits": enum
          },
          "h264Settings": {
            "adaptiveQuantization": enum,
            "afdSignaling": enum,
            "bitrate": integer,
            "bufFillPct": integer,
            "bufSize": integer,
            "colorMetadata": enum,
            "colorSpaceSettings": {
              "colorSpacePassthroughSettings": {
              },
              "rec601Settings": {
              },
              "rec709Settings": {
              }
            },
            "entropyEncoding": enum,
            "filterSettings": {
              "temporalFilterSettings": {
                "postFilterSharpening": enum,
                "strength": enum
              }
            },
            "fixedAfd": enum,
            "flickerAq": enum,
            "forceFieldPictures": enum,
            "framerateControl": enum,
            "framerateDenominator": integer,
            "framerateNumerator": integer,
            "gopBReference": enum,
            "gopClosedCadence": integer,
            "gopNumBFrames": integer,
            "gopSize": number,
            "gopSizeUnits": enum,
            "level": enum,
```

```
            "lookAheadRateControl": enum,
            "maxBitrate": integer,
            "minIInterval": integer,
            "numRefFrames": integer,
            "parControl": enum,
            "parDenominator": integer,
            "parNumerator": integer,
            "profile": enum,
            "qualityLevel": enum,
            "qvbrQualityLevel": integer,
            "rateControlMode": enum,
            "scanType": enum,
            "sceneChangeDetect": enum,
            "slices": integer,
            "softness": integer,
            "spatialAq": enum,
            "subgopLength": enum,
            "syntax": enum,
            "temporalAq": enum,
            "timecodeInsertion": enum
          },
          "h265Settings": {
            "adaptiveQuantization": enum,
            "afdSignaling": enum,
            "alternativeTransferFunction": enum,
            "bitrate": integer,
            "bufSize": integer,
            "colorMetadata": enum,
            "colorSpaceSettings": {
              "colorSpacePassthroughSettings": {
              },
              "hdr10Settings": {
                "maxCll": integer,
                "maxFall": integer
              },
              "rec601Settings": {
              },
              "rec709Settings": {
              }
            },
            "filterSettings": {
              "temporalFilterSettings": {
                "postFilterSharpening": enum,
                "strength": enum
```

```
            }
          },
          "fixedAfd": enum,
          "flickerAq": enum,
          "framerateDenominator": integer,
          "framerateNumerator": integer,
          "gopClosedCadence": integer,
          "gopSize": number,
          "gopSizeUnits": enum,
          "level": enum,
          "lookAheadRateControl": enum,
          "maxBitrate": integer,
          "minIInterval": integer,
          "parDenominator": integer,
          "parNumerator": integer,
          "profile": enum,
          "qvbrQualityLevel": integer,
          "rateControlMode": enum,
          "scanType": enum,
          "sceneChangeDetect": enum,
          "slices": integer,
          "tier": enum,
          "timecodeInsertion": enum
        },
        "mpeg2Settings": {
          "adaptiveQuantization": enum,
          "afdSignaling": enum,
          "colorMetadata": enum,
          "colorSpace": enum,
          "displayAspectRatio": enum,
          "filterSettings": {
            "temporalFilterSettings": {
              "postFilterSharpening": enum,
              "strength": enum
            }
          },
          "fixedAfd": enum,
          "framerateDenominator": integer,
          "framerateNumerator": integer,
          "gopClosedCadence": integer,
          "gopNumBFrames": integer,
          "gopSize": number,
          "gopSizeUnits": enum,
          "scanType": enum,
```

```
            "subgopLength": enum,
            "timecodeInsertion": enum
          }
        },
        "height": integer,
        "name": "string",
        "respondToAfd": enum,
        "scalingBehavior": enum,
        "sharpness": integer,
        "width": integer
      }
    ]
  },
  "id": "string",
  "inputAttachments": [
    {
      "automaticInputFailoverSettings": {
        "errorClearTimeMsec": integer,
        "failoverConditions": [
          {
            "failoverConditionSettings": {
              "audioSilenceSettings": {
                "audioSelectorName": "string",
                "audioSilenceThresholdMsec": integer
              },
              "inputLossSettings": {
                "inputLossThresholdMsec": integer
              },
              "videoBlackSettings": {
                "blackDetectThreshold": number,
                "videoBlackThresholdMsec": integer
              }
            }
          }
        ],
        "inputPreference": enum,
        "secondaryInputId": "string"
      },
      "inputAttachmentName": "string",
      "inputId": "string",
      "inputSettings": {
        "audioSelectors": [
          {
            "name": "string",
```

```
            "selectorSettings": {
              "audioHlsRenditionSelection": {
                "groupId": "string",
                "name": "string"
              },
              "audioLanguageSelection": {
                "languageCode": "string",
                "languageSelectionPolicy": enum
              },
              "audioPidSelection": {
                "pid": integer
              },
              "audioTrackSelection": {
                "tracks": [
                  {
                    "track": integer
                  }
                ]
              }
            }
          }
        ],
        "captionSelectors": [
          {
            "languageCode": "string",
            "name": "string",
            "selectorSettings": {
              "ancillarySourceSettings": {
                "sourceAncillaryChannelNumber": integer
              },
              "aribSourceSettings": {
              },
              "dvbSubSourceSettings": {
                "ocrLanguage": enum,
                "pid": integer
              },
              "embeddedSourceSettings": {
                "convert608To708": enum,
                "scte20Detection": enum,
                "source608ChannelNumber": integer,
                "source608TrackNumber": integer
              },
              "scte20SourceSettings": {
                "convert608To708": enum,
```

```
            "source608ChannelNumber": integer
          },
          "scte27SourceSettings": {
            "ocrLanguage": enum,
            "pid": integer
          },
          "teletextSourceSettings": {
            "outputRectangle": {
              "height": number,
              "leftOffset": number,
              "topOffset": number,
              "width": number
            },
            "pageNumber": "string"
          }
        }
      }
    ],
    "deblockFilter": enum,
    "denoiseFilter": enum,
    "filterStrength": integer,
    "inputFilter": enum,
    "networkInputSettings": {
      "hlsInputSettings": {
        "bandwidth": integer,
        "bufferSegments": integer,
        "retries": integer,
        "retryInterval": integer,
        "scte35Source": enum
      },
      "serverValidation": enum
    },
    "smpte2038DataPreference": enum,
    "sourceEndBehavior": enum,
    "videoSelector": {
      "colorSpace": enum,
      "colorSpaceSettings": {
        "hdr10Settings": {
          "maxCll": integer,
          "maxFall": integer
        }
      },
      "colorSpaceUsage": enum,
      "selectorSettings": {
```

```
            "videoSelectorPid": {
              "pid": integer
            },
            "videoSelectorProgramId": {
              "programId": integer
            }
          }
        }
      }
    }
  ],
  "inputSpecification": {
    "codec": enum,
    "maximumBitrate": enum,
    "resolution": enum
  },
  "logLevel": enum,
  "maintenance": {
    "maintenanceDay": enum,
    "maintenanceDeadline": "string",
    "maintenanceScheduledDate": "string",
    "maintenanceStartTime": "string"
  },
  "name": "string",
  "pipelineDetails": [
    {
      "activeInputAttachmentName": "string",
      "activeInputSwitchActionName": "string",
      "activeMotionGraphicsActionName": "string",
      "activeMotionGraphicsUri": "string",
      "pipelineId": "string"
    }
  ],
  "pipelinesRunningCount": integer,
  "roleArn": "string",
  "state": enum,
  "tags": {
  },
  "vpc": {
    "availabilityZones": [
      "string"
    ],
    "networkInterfaceIds": [
      "string"
```

```
    ],
    "securityGroupIds": [
      "string"
    ],
    "subnetIds": [
      "string"
    ]
  }
}
```

InvalidRequest schema

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

AccessDenied schema

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

ResourceNotFound schema

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

ResourceConflict schema

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

LimitExceeded schema

```
{
  "message": "string"
```

```
}
```

InternalServiceError schema

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

BadGatewayException schema

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

GatewayTimeoutException schema

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

Properties

AacCodingMode

Aac Coding Mode

```
AD_RECEIVER_MIX
CODING_MODE_1_0
CODING_MODE_1_1
CODING_MODE_2_0
CODING_MODE_5_1
```

AacInputType

Aac Input Type

```
BROADCASTER_MIXED_AD
```

NORMAL

AacProfile

Aac Profile

HEV1
HEV2
LC

AacRateControlMode

Aac Rate Control Mode

CBR
VBR

AacRawFormat

Aac Raw Format

LATM_LOAS
NONE

AacSettings

Aac Settings

bitrate

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

Type: number
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: AacCodingMode
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: AacInputType
Required: False

profile

AAC Profile.

Type: AacProfile
Required: False

rateControlMode

Rate Control Mode.

Type: AacRateControlMode
Required: False

rawFormat

Sets LATM / LOAS AAC output for raw containers.

Type: AacRawFormat
Required: False

sampleRate

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

Type: number
Required: False

spec

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

Type: AacSpec
Required: False

vbrQuality

VBR Quality Level - Only used if rateControlMode is VBR.

Type: AacVbrQuality
Required: False

AacSpec

Aac Spec

`MPEG2`
`MPEG4`

AacVbrQuality

Aac Vbr Quality

`HIGH`
`LOW`
`MEDIUM_HIGH`
`MEDIUM_LOW`

Ac3BitstreamMode

Ac3 Bitstream Mode

COMMENTARY
COMPLETE_MAIN
DIALOGUE
EMERGENCY
HEARING_IMPAIRED
MUSIC_AND_EFFECTS
VISUALLY_IMPAIRED
VOICE_OVER

Ac3CodingMode

Ac3 Coding Mode

CODING_MODE_1_0
CODING_MODE_1_1
CODING_MODE_2_0
CODING_MODE_3_2_LFE

Ac3DrcProfile

Ac3 Drc Profile

FILM_STANDARD
NONE

Ac3LfeFilter

Ac3 Lfe Filter

DISABLED
ENABLED

Ac3MetadataControl

Ac3 Metadata Control

```
FOLLOW_INPUT
USE_CONFIGURED
```

Ac3Settings

Ac3 Settings

bitrate

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

Type: number
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: Ac3BitstreamMode
Required: False

codingMode

Dolby Digital coding mode. Determines number of channels.

Type: Ac3CodingMode
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

drcProfile

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

Type: Ac3DrcProfile
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: Ac3LfeFilter
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: Ac3MetadataControl
Required: False

AccessDenied

message

Type: string
Required: False

AfdSignaling

Afd Signaling

AUTO

FIXED

NONE

AncillarySourceSettings

Ancillary Source Settings

sourceAncillaryChannelNumber

Specifies the number (1 to 4) of the captions channel you want to extract from the ancillary captions. If you plan to convert the ancillary captions to another format, complete this field. If you plan to choose Embedded as the captions destination in the output (to pass through all the channels in the ancillary captions), leave this field blank because MediaLive ignores the field.

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

ArchiveCdnSettings

Archive Cdn Settings

archiveS3Settings

Type: ArchiveS3Settings
Required: False

ArchiveContainerSettings

Archive Container Settings

m2tsSettings

Type: M2tsSettings
Required: False

rawSettings

Type: RawSettings
Required: False

ArchiveGroupSettings

Archive Group Settings

archiveCdnSettings

Parameters that control interactions with the CDN.

Type: ArchiveCdnSettings
Required: False

destination

A directory and base filename where archive files should be written.

Type: OutputLocationRef
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

Archive Output Settings

containerSettings

Settings specific to the container type of the file.

Type: ArchiveContainerSettings

Required: True

extension

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

Type: string
Required: False

nameModifier

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

Type: string
Required: False

ArchiveS3LogUploads

Archive S3 Log Uploads

`DISABLED`
`ENABLED`

ArchiveS3Settings

Archive S3 Settings

cannedAcl

Specify the canned ACL to apply to each S3 request. Defaults to none.

Type: S3CannedAcl
Required: False

logUploads

When set to enabled, each upload to CDN or server will be logged.

Type: ArchiveS3LogUploads
Required: False

AribDestinationSettings

Arib Destination Settings

AribSourceSettings

Arib Source Settings

AudioChannelMapping

Audio Channel Mapping

inputChannelLevels

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

Type: Array of type InputChannelLevel
Required: True

outputChannel

The index of the output channel being produced.

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

AudioCodecSettings

Audio Codec Settings

aacSettings

Type: AacSettings

Required: False

ac3Settings

Type: Ac3Settings
Required: False

eac3Settings

Type: Eac3Settings
Required: False

mp2Settings

Type: Mp2Settings
Required: False

passThroughSettings

Type: PassThroughSettings
Required: False

wavSettings

Type: WavSettings
Required: False

AudioDescription

Audio Description

audioNormalizationSettings

Advanced audio normalization settings.

Type: AudioNormalizationSettings
Required: False

audioSelectorName

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

Type: string
Required: True

audioType

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

Type: AudioType
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: AudioDescriptionAudioTypeControl
Required: False

audioWatermarkingSettings

Settings to configure one or more solutions that insert audio watermarks in the audio encode

Type: AudioWatermarkSettings
Required: False

codecSettings

Audio codec settings.

Type: AudioCodecSettings
Required: False

languageCode

RFC 5646 language code representing 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
MinLength: 1
MaxLength: 35

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: AudioDescriptionLanguageCodeControl
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

remixSettings

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

Type: RemixSettings
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

AudioDescriptionAudioTypeControl

Audio Description Audio Type Control

`FOLLOW_INPUT`

`USE_CONFIGURED`

AudioDescriptionLanguageCodeControl

Audio Description Language Code Control

`FOLLOW_INPUT`

`USE_CONFIGURED`

AudioHlsRenditionSelection

Audio Hls Rendition Selection

groupId

Specifies the GROUP-ID in the #EXT-X-MEDIA tag of the target HLS audio rendition.

Type: string

Required: True

MinLength: 1

name

Specifies the NAME in the #EXT-X-MEDIA tag of the target HLS audio rendition.

Type: string

Required: True

MinLength: 1

AudioLanguageSelection

Audio Language Selection

languageCode

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

Type: string
Required: True

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: AudioLanguageSelectionPolicy
Required: False

AudioLanguageSelectionPolicy

Audio Language Selection Policy

```
LOOSE
STRICT
```

AudioNormalizationAlgorithm

Audio Normalization Algorithm

```
ITU_1770_1
ITU_1770_2
```

AudioNormalizationAlgorithmControl

Audio Normalization Algorithm Control

CORRECT_AUDIO

AudioNormalizationSettings

Audio Normalization Settings

algorithm

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

Type: AudioNormalizationAlgorithm
Required: False

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: AudioNormalizationAlgorithmControl
Required: False

targetLkfs

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

Type: number
Required: False
Minimum: -59
Maximum: 0

AudioOnlyHlsSegmentType

Audio Only Hls Segment Type

AAC
FMP4

AudioOnlyHlsSettings

Audio Only Hls Settings

audioGroupId

Specifies the group to which the audio Rendition belongs.

Type: string
Required: False

audioOnlyImage

Optional. Specifies the .jpg or .png image to use as the cover art for an audio-only output. We recommend a low bit-size file because the image increases the output audio bandwidth. The image is attached to the audio as an ID3 tag, frame type APIC, picture type 0x10, as per the "ID3 tag version 2.4.0 - Native Frames" standard.

Type: InputLocation
Required: False

audioTrackType

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

Type: AudioOnlyHlsTrackType
Required: False

segmentType

Specifies the segment type.

Type: AudioOnlyHlsSegmentType
Required: False

AudioOnlyHlsTrackType

Audio Only Hls Track Type

`ALTERNATE_AUDIO_AUTO_SELECT`
`ALTERNATE_AUDIO_AUTO_SELECT_DEFAULT`
`ALTERNATE_AUDIO_NOT_AUTO_SELECT`
`AUDIO_ONLY_VARIANT_STREAM`

AudioPidSelection

Audio Pid Selection

pid

Selects a specific PID from within a source.

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

AudioSelector

Audio Selector

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
MinLength: 1

selectorSettings

The audio selector settings.

Type: AudioSelectorSettings
Required: False

AudioSelectorSettings

Audio Selector Settings

audioHlsRenditionSelection

Type: AudioHlsRenditionSelection
Required: False

audioLanguageSelection

Type: AudioLanguageSelection
Required: False

audioPidSelection

Type: AudioPidSelection
Required: False

audioTrackSelection

Type: AudioTrackSelection
Required: False

AudioSilenceFailoverSettings

audioSelectorName

The name of the audio selector in the input that MediaLive should monitor to detect silence. Select your most important rendition. If you didn't create an audio selector in this input, leave blank.

Type: string
Required: True

audioSilenceThresholdMsec

The amount of time (in milliseconds) that the active input must be silent before automatic input failover occurs. Silence is defined as audio loss or audio quieter than -50 dBFS.

Type: integer
Required: False
Minimum: 1000

AudioTrack

Audio Track

track

1-based integer value that maps to a specific audio track

Type: integer
Required: True
Minimum: 1

AudioTrackSelection

Audio Track Selection

tracks

Selects one or more unique audio tracks from within a source.

Type: Array of type AudioTrack
Required: True

AudioType

Audio Type

CLEAN_EFFECTS
HEARING_IMPAIRED
UNDEFINED
VISUAL_IMPAIRED_COMMENTARY

AudioWatermarkSettings

Audio Watermark Settings

nielsenWatermarksSettings

Settings to configure Nielsen Watermarks in the audio encode

Type: NielsenWatermarksSettings
Required: False

AuthenticationScheme

Authentication Scheme

AKAMAI
COMMON

AutomaticInputFailoverSettings

The settings for Automatic Input Failover.

errorClearTimeMsec

This clear time defines the requirement a recovered input must meet to be considered healthy. The input must have no failover conditions for this length of time. Enter a time in milliseconds. This value is particularly important if the input_preference for the failover pair is set to PRIMARY_INPUT_PREFERRED, because after this time, MediaLive will switch back to the primary input.

Type: integer
Required: False
Minimum: 1

failoverConditions

A list of failover conditions. If any of these conditions occur, MediaLive will perform a failover to the other input.

Type: Array of type FailoverCondition
Required: False

inputPreference

Input preference when deciding which input to make active when a previously failed input has recovered.

Type: InputPreference
Required: False

secondaryInputId

The input ID of the secondary input in the automatic input failover pair.

Type: string
Required: True

AvailBlanking

Avail Blanking

availBlankingImage

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

Type: InputLocation
Required: False

state

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

Type: AvailBlankingState

Required: False

AvailBlankingState

Avail Blanking State

DISABLED
ENABLED

AvailConfiguration

Avail Configuration

availSettings

Ad avail settings.

Type: AvailSettings
Required: False

AvailSettings

Avail Settings

scte35SpliceInsert

Type: Scte35SpliceInsert
Required: False

scte35TimeSignalApos

Type: Scte35TimeSignalApos
Required: False

BadGatewayException

message

Type: string

Required: False

BlackoutSlate

Blackout Slate

blackoutSlateImage

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

Type: InputLocation
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: BlackoutSlateNetworkEndBlackout
Required: False

networkEndBlackoutImage

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

Type: InputLocation
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
MinLength: 34
MaxLength: 34

state

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

Type: BlackoutSlateState
Required: False

BlackoutSlateNetworkEndBlackout

Blackout Slate Network End Blackout

DISABLED
ENABLED

BlackoutSlateState

Blackout Slate State

DISABLED
ENABLED

BurnInAlignment

Burn In Alignment

CENTERED
LEFT
SMART

BurnInBackgroundColor

Burn In Background Color

BLACK

NONE

WHITE

BurnInDestinationSettings

Burn In Destination Settings

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: BurnInAlignment
Required: False

backgroundColor

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

Type: BurnInBackgroundColor
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

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
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: BurnInFontColor
Required: False

fontOpacity

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

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

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

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

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: BurnInOutlineColor
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

shadowColor

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

Type: BurnInShadowColor
Required: False

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

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

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

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: BurnInTeletextGridControl
Required: False

xPosition

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

explicit 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

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

BurnInFontColor

Burn In Font Color

BLACK
BLUE
GREEN
RED
WHITE
YELLOW

BurnInOutlineColor

Burn In Outline Color

BLACK
BLUE
GREEN

RED
WHITE
YELLOW

BurnInShadowColor

Burn In Shadow Color

BLACK
NONE
WHITE

BurnInTeletextGridControl

Burn In Teletext Grid Control

FIXED
SCALED

CaptionDescription

Caption Description

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

destinationSettings

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

Type: CaptionDestinationSettings
Required: False

languageCode

ISO 639-2 three-digit code: http://www.loc.gov/standards/iso639-2/

Type: string
Required: False

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

CaptionDestinationSettings

Caption Destination Settings

aribDestinationSettings

Type: AribDestinationSettings
Required: False

burnInDestinationSettings

Type: BurnInDestinationSettings
Required: False

dvbSubDestinationSettings

Type: DvbSubDestinationSettings

Required: False

ebuTtDDestinationSettings

Type: EbuTtDDestinationSettings
Required: False

embeddedDestinationSettings

Type: EmbeddedDestinationSettings
Required: False

embeddedPlusScte20DestinationSettings

Type: EmbeddedPlusScte20DestinationSettings
Required: False

rtmpCaptionInfoDestinationSettings

Type: RtmpCaptionInfoDestinationSettings
Required: False

scte20PlusEmbeddedDestinationSettings

Type: Scte20PlusEmbeddedDestinationSettings
Required: False

scte27DestinationSettings

Type: Scte27DestinationSettings
Required: False

smpteTtDestinationSettings

Type: SmpteTtDestinationSettings
Required: False

teletextDestinationSettings

Type: TeletextDestinationSettings
Required: False

ttmlDestinationSettings

Type: TtmlDestinationSettings
Required: False

webvttDestinationSettings

Type: WebvttDestinationSettings
Required: False

CaptionLanguageMapping

Maps a caption channel to an ISO 693-2 language code (http://www.loc.gov/standards/iso639-2), with an optional description.

captionChannel

The closed caption channel being described by this CaptionLanguageMapping. 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
MinLength: 3
MaxLength: 3

languageDescription

Textual description of language

Type: string
Required: True
MinLength: 1

CaptionRectangle

Caption Rectangle

height

See the description in leftOffset. For height, specify the entire height of the rectangle as a percentage of the underlying frame height. For example, \"80\" means the rectangle height is 80% of the underlying frame height. The topOffset and rectangleHeight must add up to 100% or less. This field corresponds to tts:extent - Y in the TTML standard.

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

leftOffset

Applies only if you plan to convert these source captions to EBU-TT-D or TTML in an output. (Make sure to leave the default if you don't have either of these formats in the output.) You can define a display rectangle for the captions that is smaller than the underlying video frame. You define the rectangle by specifying the position of the left edge, top edge, bottom edge, and right edge of the rectangle, all within the underlying video frame. The units for the measurements are percentages. If you specify a value for one of these fields, you must specify a value for all of them. For leftOffset, specify the position of the left edge of the rectangle, as a percentage of the underlying frame width, and relative to the left edge of the frame. For example, \"10\" means the measurement is 10% of the underlying frame width. The rectangle left edge starts at that position from the left edge of the frame. This field corresponds to tts:origin - X in the TTML standard.

Type: number

Required: True
Minimum: 0
Maximum: 100

topOffset

See the description in leftOffset. For topOffset, specify the position of the top edge of the rectangle, as a percentage of the underlying frame height, and relative to the top edge of the frame. For example, \"10\" means the measurement is 10% of the underlying frame height. The rectangle top edge starts at that position from the top edge of the frame. This field corresponds to tts:origin - Y in the TTML standard.

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

width

See the description in leftOffset. For width, specify the entire width of the rectangle as a percentage of the underlying frame width. For example, \"80\" means the rectangle width is 80% of the underlying frame width. The leftOffset and rectangleWidth must add up to 100% or less. This field corresponds to tts:extent - X in the TTML standard.

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

CaptionSelector

Output groups for this Live Event. Output groups contain information about where streams should be distributed.

languageCode

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

Type: string
Required: False

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
MinLength: 1

selectorSettings

Caption selector settings.

Type: CaptionSelectorSettings
Required: False

CaptionSelectorSettings

Caption Selector Settings

ancillarySourceSettings

Type: AncillarySourceSettings
Required: False

aribSourceSettings

Type: AribSourceSettings
Required: False

dvbSubSourceSettings

Type: DvbSubSourceSettings
Required: False

embeddedSourceSettings

Type: EmbeddedSourceSettings
Required: False

scte20SourceSettings

Type: Scte20SourceSettings
Required: False

scte27SourceSettings

Type: Scte27SourceSettings
Required: False

teletextSourceSettings

Type: TeletextSourceSettings
Required: False

CdiInputResolution

Maximum CDI input resolution; SD is 480i and 576i up to 30 frames-per-second (fps), HD is 720p up to 60 fps / 1080i up to 30 fps, FHD is 1080p up to 60 fps, UHD is 2160p up to 60 fps

SD
HD
FHD
UHD

CdiInputSpecification

resolution

Maximum CDI input resolution

Type: CdiInputResolution
Required: False

Channel

arn

The unique arn of the channel.

Type: string
Required: False

cdiInputSpecification

Specification of CDI inputs for this channel

Type: CdiInputSpecification
Required: False

channelClass

The class for this channel. STANDARD for a channel with two pipelines or SINGLE_PIPELINE for a channel with one pipeline.

Type: ChannelClass
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
Required: False

egressEndpoints

The endpoints where outgoing connections initiate from

Type: Array of type ChannelEgressEndpoint
Required: False

encoderSettings

Type: EncoderSettings
Required: False

id

The unique ID of the channel.

Type: string
Required: False

inputAttachments

List of input attachments for channel.

Type: Array of type InputAttachment
Required: False

inputSpecification

Specification of network and file inputs for this channel

Type: InputSpecification
Required: False

logLevel

The log level being written to CloudWatch Logs.

Type: LogLevel
Required: False

maintenance

Maintenance settings for this channel.

Type: MaintenanceStatus

Required: False

name

The name of the channel. (user-mutable)

Type: string
Required: False

pipelineDetails

Runtime details for the pipelines of a running channel.

Type: Array of type PipelineDetail
Required: False

pipelinesRunningCount

The number of currently healthy pipelines.

Type: integer
Required: False

roleArn

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

Type: string
Required: False

state

Type: ChannelState
Required: False

tags

A collection of key-value pairs.

Type: Tags
Required: False

vpc

Settings for VPC output

Type: VpcOutputSettingsDescription
Required: False

ChannelClass

A standard channel has two encoding pipelines and a single pipeline channel only has one.

```
STANDARD
SINGLE_PIPELINE
```

ChannelEgressEndpoint

sourceIp

Public IP of where a channel's output comes from

Type: string
Required: False

ChannelState

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

UPDATING
UPDATE_FAILED

ColorSpacePassthroughSettings

Passthrough applies no color space conversion to the output

DvbNitSettings

DVB Network Information Table (NIT)

networkId

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

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

networkName

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

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

repInterval

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

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

DvbSdtOutputSdt

Dvb Sdt Output Sdt

SDT_FOLLOW
SDT_FOLLOW_IF_PRESENT
SDT_MANUAL
SDT_NONE

DvbSdtSettings

DVB Service Description Table (SDT)

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: DvbSdtOutputSdt
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

serviceName

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

Type: string

Required: False
MinLength: 1
MaxLength: 256

serviceProviderName

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

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

DvbSubDestinationAlignment

Dvb Sub Destination Alignment

CENTERED
LEFT
SMART

DvbSubDestinationBackgroundColor

Dvb Sub Destination Background Color

BLACK
NONE
WHITE

DvbSubDestinationFontColor

Dvb Sub Destination Font Color

BLACK
BLUE
GREEN

RED
WHITE
YELLOW

DvbSubDestinationOutlineColor

Dvb Sub Destination Outline Color

BLACK
BLUE
GREEN
RED
WHITE
YELLOW

DvbSubDestinationSettings

Dvb Sub Destination Settings

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: DvbSubDestinationAlignment
Required: False

backgroundColor

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

Type: DvbSubDestinationBackgroundColor

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

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
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: DvbSubDestinationFontColor
Required: False

fontOpacity

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

Type: integer

Required: False
Minimum: 0
Maximum: 255

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

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

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: DvbSubDestinationOutlineColor
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

shadowColor

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

Type: DvbSubDestinationShadowColor
Required: False

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

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

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

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: DvbSubDestinationTeletextGridControl
Required: False

xPosition

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

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

DvbSubDestinationShadowColor

Dvb Sub Destination Shadow Color

BLACK

NONE
WHITE

DvbSubDestinationTeletextGridControl

Dvb Sub Destination Teletext Grid Control

FIXED
SCALED

DvbSubOcrLanguage

Dvb Sub Ocr Language

DEU
ENG
FRA
NLD
POR
SPA

DvbSubSourceSettings

Dvb Sub Source Settings

ocrLanguage

If you will configure a WebVTT caption description that references this caption selector, use this field to provide the language to consider when translating the image-based source to text.

Type: DvbSubOcrLanguage
Required: False

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

DVB Time and Date Table (SDT)

repInterval

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

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

Eac3AttenuationControl

Eac3 Attenuation Control

```
ATTENUATE_3_DB
NONE
```

Eac3BitstreamMode

Eac3 Bitstream Mode

```
COMMENTARY
COMPLETE_MAIN
EMERGENCY
HEARING_IMPAIRED
VISUALLY_IMPAIRED
```

Eac3CodingMode

Eac3 Coding Mode

```
CODING_MODE_1_0
```

CODING_MODE_2_0
CODING_MODE_3_2

Eac3DcFilter

Eac3 Dc Filter

DISABLED
ENABLED

Eac3DrcLine

Eac3 Drc Line

FILM_LIGHT
FILM_STANDARD
MUSIC_LIGHT
MUSIC_STANDARD
NONE
SPEECH

Eac3DrcRf

Eac3 Drc Rf

FILM_LIGHT
FILM_STANDARD
MUSIC_LIGHT
MUSIC_STANDARD
NONE
SPEECH

Eac3LfeControl

Eac3 Lfe Control

LFE

NO_LFE

Eac3LfeFilter

Eac3 Lfe Filter

DISABLED
ENABLED

Eac3MetadataControl

Eac3 Metadata Control

FOLLOW_INPUT
USE_CONFIGURED

Eac3PassthroughControl

Eac3 Passthrough Control

NO_PASSTHROUGH
WHEN_POSSIBLE

Eac3PhaseControl

Eac3 Phase Control

NO_SHIFT
SHIFT_90_DEGREES

Eac3Settings

Eac3 Settings

attenuationControl

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

Type: Eac3AttenuationControl
Required: False

bitrate

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

Type: number
Required: False

bitstreamMode

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

Type: Eac3BitstreamMode
Required: False

codingMode

Dolby Digital Plus coding mode. Determines number of channels.

Type: Eac3CodingMode
Required: False

dcFilter

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

Type: Eac3DcFilter
Required: False

dialnorm

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

Type: integer
Required: False

Minimum: 1
Maximum: 31

drcLine

Sets the Dolby dynamic range compression profile.

Type: Eac3DrcLine
Required: False

drcRf

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

Type: Eac3DrcRf
Required: False

lfeControl

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

Type: Eac3LfeControl
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: Eac3LfeFilter
Required: False

loRoCenterMixLevel

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

Type: number
Required: False

loRoSurroundMixLevel

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

Type: number
Required: False

ltRtCenterMixLevel

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

Type: number
Required: False

ltRtSurroundMixLevel

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

Type: number
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: Eac3MetadataControl
Required: False

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: Eac3PassthroughControl

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: Eac3PhaseControl
Required: False

stereoDownmix

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

Type: Eac3StereoDownmix
Required: False

surroundExMode

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

Type: Eac3SurroundExMode
Required: False

surroundMode

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

Type: Eac3SurroundMode
Required: False

Eac3StereoDownmix

Eac3 Stereo Downmix

DPL2
LO_RO
LT_RT

NOT_INDICATED

Eac3SurroundExMode

Eac3 Surround Ex Mode

DISABLED
ENABLED
NOT_INDICATED

Eac3SurroundMode

Eac3 Surround Mode

DISABLED
ENABLED
NOT_INDICATED

EbuTtDDestinationSettings

Ebu Tt DDestination Settings

copyrightHolder

Applies only if you plan to convert these source captions to EBU-TT-D or TTML in an output. Complete this field if you want to include the name of the copyright holder in the copyright metadata tag in the TTML

Type: string
Required: False
MaxLength: 1000

fillLineGap

Specifies how to handle the gap between the lines (in multi-line captions). - enabled: Fill with the captions background color (as specified in the input captions). - disabled: Leave the gap unfilled.

Type: EbuTtDFillLineGapControl
Required: False

fontFamily

Specifies the font family to include in the font data attached to the EBU-TT captions. Valid only if styleControl is set to include. If you leave this field empty, the font family is set to "monospaced". (If styleControl is set to exclude, the font family is always set to "monospaced".) You specify only the font family. All other style information (color, bold, position and so on) is copied from the input captions. The size is always set to 100% to allow the downstream player to choose the size. - Enter a list of font families, as a comma-separated list of font names, in order of preference. The name can be a font family (such as “Arial”), or a generic font family (such as “serif”), or “default” (to let the downstream player choose the font). - Leave blank to set the family to “monospace”.

Type: string
Required: False

styleControl

Specifies the style information (font color, font position, and so on) to include in the font data that is attached to the EBU-TT captions. - include: Take the style information (font color, font position, and so on) from the source captions and include that information in the font data attached to the EBU-TT captions. This option is valid only if the source captions are Embedded or Teletext. - exclude: In the font data attached to the EBU-TT captions, set the font family to "monospaced". Do not include any other style information.

Type: EbuTtDDestinationStyleControl
Required: False

EbuTtDDestinationStyleControl

Ebu Tt DDestination Style Control

```
EXCLUDE
INCLUDE
```

EbuTtDFillLineGapControl

Ebu Tt DFill Line Gap Control

```
DISABLED
```

ENABLED

EmbeddedConvert608To708

Embedded Convert608 To708

DISABLED
UPCONVERT

EmbeddedDestinationSettings

Embedded Destination Settings

EmbeddedPlusScte20DestinationSettings

Embedded Plus Scte20 Destination Settings

EmbeddedScte20Detection

Embedded Scte20 Detection

AUTO
OFF

EmbeddedSourceSettings

Embedded Source Settings

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: EmbeddedConvert608To708
Required: False

scte20Detection

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

Type: EmbeddedScte20Detection
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

source608TrackNumber

This field is unused and deprecated.

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

EncoderSettings

Encoder Settings

audioDescriptions

Type: Array of type AudioDescription
Required: True

availBlanking

Settings for ad avail blanking.

Type: AvailBlanking
Required: False

availConfiguration

Event-wide configuration settings for ad avail insertion.

Type: AvailConfiguration
Required: False

blackoutSlate

Settings for blackout slate.

Type: BlackoutSlate
Required: False

captionDescriptions

Settings for caption decriptions

Type: Array of type CaptionDescription
Required: False

featureActivations

Feature Activations

Type: FeatureActivations
Required: False

globalConfiguration

Configuration settings that apply to the event as a whole.

Type: GlobalConfiguration
Required: False

motionGraphicsConfiguration

Settings for motion graphics.

Type: MotionGraphicsConfiguration
Required: False

nielsenConfiguration

Nielsen configuration settings.

Type: NielsenConfiguration
Required: False

outputGroups

Type: Array of type OutputGroup
Required: True

timecodeConfig

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

Type: TimecodeConfig
Required: True

videoDescriptions

Type: Array of type VideoDescription
Required: True

FailoverCondition

Failover Condition settings. There can be multiple failover conditions inside AutomaticInputFailoverSettings.

failoverConditionSettings

Failover condition type-specific settings.

Type: FailoverConditionSettings
Required: False

FailoverConditionSettings

Settings for one failover condition.

audioSilenceSettings

MediaLive will perform a failover if the specified audio selector is silent for the specified period.

Type: AudioSilenceFailoverSettings
Required: False

inputLossSettings

MediaLive will perform a failover if content is not detected in this input for the specified period.

Type: InputLossFailoverSettings
Required: False

videoBlackSettings

MediaLive will perform a failover if content is considered black for the specified period.

Type: VideoBlackFailoverSettings
Required: False

FeatureActivations

Feature Activations

inputPrepareScheduleActions

Enables the Input Prepare feature. You can create Input Prepare actions in the schedule only if this feature is enabled. If you disable the feature on an existing schedule, make sure that you first delete all input prepare actions from the schedule.

Type: FeatureActivationsInputPrepareScheduleActions
Required: False

FeatureActivationsInputPrepareScheduleActions

Feature Activations Input Prepare Schedule Actions

```
DISABLED
ENABLED
```

FecOutputIncludeFec

Fec Output Include Fec

```
COLUMN
COLUMN_AND_ROW
```

FecOutputSettings

Fec Output Settings

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: FecOutputIncludeFec
Required: False

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

FixedAfd

Fixed Afd

AFD_0000
AFD_0010
AFD_0011
AFD_0100
AFD_1000
AFD_1001
AFD_1010
AFD_1011
AFD_1101
AFD_1110
AFD_1111

Fmp4HlsSettings

Fmp4 Hls Settings

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

nielsenId3Behavior

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

Type: Fmp4NielsenId3Behavior
Required: False

timedMetadataBehavior

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

Type: Fmp4TimedMetadataBehavior
Required: False

Fmp4NielsenId3Behavior

Fmp4 Nielsen Id3 Behavior

NO_PASSTHROUGH
PASSTHROUGH

Fmp4TimedMetadataBehavior

Fmp4 Timed Metadata Behavior

NO_PASSTHROUGH
PASSTHROUGH

FrameCaptureCdnSettings

Frame Capture Cdn Settings

frameCaptureS3Settings

Type: FrameCaptureS3Settings
Required: False

FrameCaptureGroupSettings

Frame Capture Group Settings

destination

The destination for the frame capture files. Either the URI for an Amazon S3 bucket and object, plus a file name prefix (for example, s3ssl://sportsDelivery/highlights/20180820/curling-) or the URI for a MediaStore container, plus a file name prefix (for example, mediastoressl://sportsDelivery/20180820/curling-). The final file names consist of the prefix from the destination field (for example, "curling-") + name modifier + the counter (5 digits, starting from 00001) + extension (which is always .jpg). For example, curling-low.00001.jpg

Type: OutputLocationRef
Required: True

frameCaptureCdnSettings

Parameters that control interactions with the CDN.

Type: FrameCaptureCdnSettings
Required: False

FrameCaptureHlsSettings

Frame Capture Hls Settings

FrameCaptureIntervalUnit

Frame Capture Interval Unit

```
MILLISECONDS
SECONDS
```

FrameCaptureOutputSettings

Frame Capture Output Settings

nameModifier

Required if the output group contains more than one output. This modifier forms part of the output file name.

Type: string
Required: False

FrameCaptureS3LogUploads

Frame Capture S3 Log Uploads

`DISABLED`
`ENABLED`

FrameCaptureS3Settings

Frame Capture S3 Settings

cannedAcl

Specify the canned ACL to apply to each S3 request. Defaults to none.

Type: S3CannedAcl
Required: False

logUploads

When set to enabled, each upload to CDN or server will be logged.

Type: FrameCaptureS3LogUploads
Required: False

FrameCaptureSettings

Frame Capture Settings

captureInterval

The frequency at which to capture frames for inclusion in the output. May be specified in either seconds or milliseconds, as specified by captureIntervalUnits.

Type: integer
Required: False
Minimum: 1
Maximum: 3600000

captureIntervalUnits

Unit for the frame capture interval.

Type: FrameCaptureIntervalUnit
Required: False

GatewayTimeoutException

message

Type: string
Required: False

GlobalConfiguration

Global Configuration

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 the current input completes (e.g. end-of-file). When switchAndLoopInputs is configured the encoder will restart at the beginning of the first input. When "none" is configured the encoder will transcode either black, a solid color, or a user specified slate images per the "Input Loss Behavior" configuration until the next input switch occurs (which is controlled through the Channel Schedule API).

Type: GlobalConfigurationInputEndAction
Required: False

inputLossBehavior

Settings for system actions when input is lost.

Type: InputLossBehavior
Required: False

outputLockingMode

Indicates how MediaLive pipelines are synchronized. PIPELINE_LOCKING - MediaLive will attempt to synchronize the output of each pipeline to the other. EPOCH_LOCKING - MediaLive will attempt to synchronize the output of each pipeline to the Unix epoch.

Type: GlobalConfigurationOutputLockingMode
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: GlobalConfigurationOutputTimingSource
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: GlobalConfigurationLowFramerateInputs

Required: False

GlobalConfigurationInputEndAction

Global Configuration Input End Action

```
NONE
SWITCH_AND_LOOP_INPUTS
```

GlobalConfigurationLowFramerateInputs

Global Configuration Low Framerate Inputs

```
DISABLED
ENABLED
```

GlobalConfigurationOutputLockingMode

Global Configuration Output Locking Mode

```
EPOCH_LOCKING
PIPELINE_LOCKING
```

GlobalConfigurationOutputTimingSource

Global Configuration Output Timing Source

```
INPUT_CLOCK
SYSTEM_CLOCK
```

H264AdaptiveQuantization

H264 Adaptive Quantization

```
AUTO
HIGH
HIGHER
LOW
MAX
MEDIUM
OFF
```

H264ColorMetadata

H264 Color Metadata

```
IGNORE
INSERT
```

H264ColorSpaceSettings

H264 Color Space Settings

colorSpacePassthroughSettings

Type: ColorSpacePassthroughSettings
Required: False

rec601Settings

Type: Rec601Settings
Required: False

rec709Settings

Type: Rec709Settings
Required: False

H264EntropyEncoding

H264 Entropy Encoding

CABAC

CAVLC

H264FilterSettings

H264 Filter Settings

temporalFilterSettings

Type: TemporalFilterSettings

Required: False

H264FlickerAq

H264 Flicker Aq

DISABLED

ENABLED

H264ForceFieldPictures

H264 Force Field Pictures

DISABLED

ENABLED

H264FramerateControl

H264 Framerate Control

INITIALIZE_FROM_SOURCE

SPECIFIED

H264GopBReference

H264 Gop BReference

DISABLED

ENABLED

H264GopSizeUnits

H264 Gop Size Units

FRAMES

SECONDS

H264Level

H264 Level

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

H264 Look Ahead Rate Control

HIGH
LOW
MEDIUM

H264ParControl

H264 Par Control

INITIALIZE_FROM_SOURCE
SPECIFIED

H264Profile

H264 Profile

BASELINE
HIGH
HIGH_10BIT
HIGH_422
HIGH_422_10BIT
MAIN

H264QualityLevel

H264 Quality Level

ENHANCED_QUALITY
STANDARD_QUALITY

H264RateControlMode

H264 Rate Control Mode

CBR

MULTIPLEX

QVBR

VBR

H264ScanType

H264 Scan Type

INTERLACED

PROGRESSIVE

H264SceneChangeDetect

H264 Scene Change Detect

DISABLED

ENABLED

H264Settings

H264 Settings

adaptiveQuantization

Enables or disables adaptive quantization, which is a technique MediaLive can apply to video on a frame-by-frame basis to produce more compression without losing quality. There are three types of adaptive quantization: flicker, spatial, and temporal. Set the field in one of these ways: Set to Auto. Recommended. For each type of AQ, MediaLive will determine if AQ is needed, and if so, the appropriate strength. Set a strength (a value other than Auto or Disable). This strength will apply to any of the AQ fields that you choose to enable. Set to Disabled to disable all types of adaptive quantization.

Type: H264AdaptiveQuantization

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: AfdSignaling
Required: False

bitrate

Average bitrate in bits/second. Required when the rate control mode is VBR or CBR. Not used for QVBR. In an MS Smooth output group, each output must have a unique value when its bitrate is 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

bufSize

Size of buffer (HRD buffer model) in bits.

Type: integer
Required: False
Minimum: 0

colorMetadata

Includes colorspace metadata in the output.

Type: H264ColorMetadata
Required: False

colorSpaceSettings

Color Space settings

Type: H264ColorSpaceSettings
Required: False

entropyEncoding

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

Type: H264EntropyEncoding
Required: False

filterSettings

Optional filters that you can apply to an encode.

Type: H264FilterSettings
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: FixedAfd
Required: False

flickerAq

Flicker AQ makes adjustments within each frame to reduce flicker or 'pop' on I-frames. The value to enter in this field depends on the value in the Adaptive quantization field: If you have set the Adaptive quantization field to Auto, MediaLive ignores any value in this field. MediaLive will determine if flicker AQ is appropriate and will apply the appropriate strength. If you have set the

Adaptive quantization field to a strength, you can set this field to Enabled or Disabled. Enabled: MediaLive will apply flicker AQ using the specified strength. Disabled: MediaLive won't apply flicker AQ. If you have set the Adaptive quantization to Disabled, MediaLive ignores any value in this field and doesn't apply flicker AQ.

Type: H264FlickerAq
Required: False

forceFieldPictures

This setting applies only when scan type is "interlaced." It controls whether coding is performed on a field basis or on a frame basis. (When the video is progressive, the coding is always performed on a frame basis.) enabled: Force MediaLive to code on a field basis, so that odd and even sets of fields are coded separately. disabled: Code the two sets of fields separately (on a field basis) or together (on a frame basis using PAFF), depending on what is most appropriate for the content.

Type: H264ForceFieldPictures
Required: False

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: H264FramerateControl
Required: False

framerateDenominator

Framerate denominator.

Type: integer
Required: False
Minimum: 1

framerateNumerator

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

Type: integer
Required: False
Minimum: 1

gopBReference

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

Type: H264GopBReference
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

gopNumBFrames

Number of B-frames between reference frames.

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

gopSize

GOP size (keyframe interval) in units of either frames or seconds per gopSizeUnits. If gopSizeUnits is frames, gopSize must be an integer and must be greater than or equal to 1. If gopSizeUnits is seconds, gopSize must be greater than 0, but need not be an integer.

Type: number
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: H264GopSizeUnits
Required: False

level

H.264 Level.

Type: H264Level
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: H264LookAheadRateControl
Required: False

maxBitrate

For QVBR: See the tooltip for Quality level For VBR: Set the maximum bitrate in order to accommodate expected spikes in the complexity of the video.

Type: integer
Required: False
Minimum: 1000

minIInterval

Only meaningful if sceneChangeDetect is set to enabled. Defaults to 5 if multiplex rate control is used. 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

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

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 pixsel aspect ratio will be set equal to the input video pixel aspect ratio of the first input.

Type: H264ParControl
Required: False

parDenominator

Pixel Aspect Ratio denominator.

Type: integer

Required: False
Minimum: 1

parNumerator

Pixel Aspect Ratio numerator.

Type: integer
Required: False
Minimum: 1

profile

H.264 Profile.

Type: H264Profile
Required: False

qualityLevel

Leave as STANDARD_QUALITY or choose a different value (which might result in additional costs to run the channel). - ENHANCED_QUALITY: Produces a slightly better video quality without an increase in the bitrate. Has an effect only when the Rate control mode is QVBR or CBR. If this channel is in a MediaLive multiplex, the value must be ENHANCED_QUALITY. - STANDARD_QUALITY: Valid for any Rate control mode.

Type: H264QualityLevel
Required: False

qvbrQualityLevel

Controls the target quality for the video encode. Applies only when the rate control mode is QVBR. You can set a target quality or you can let MediaLive determine the best quality. To set a target quality, enter values in the QVBR quality level field and the Max bitrate field. Enter values that suit your most important viewing devices. Recommended values are: - Primary screen: Quality level: 8 to 10. Max bitrate: 4M - PC or tablet: Quality level: 7. Max bitrate: 1.5M to 3M - Smartphone: Quality level: 6. Max bitrate: 1M to 1.5M To let MediaLive decide, leave the QVBR quality level field

empty, and in Max bitrate enter the maximum rate you want in the video. For more information, see the section called "Video - rate control mode" in the MediaLive user guide

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

rateControlMode

Rate control mode. QVBR: Quality will match the specified quality level except when it is constrained by the maximum bitrate. Recommended if you or your viewers pay for bandwidth. VBR: Quality and bitrate vary, depending on the video complexity. Recommended instead of QVBR if you want to maintain a specific average bitrate over the duration of the channel. CBR: Quality varies, depending on the video complexity. Recommended only if you distribute your assets to devices that cannot handle variable bitrates. Multiplex: This rate control mode is only supported (and is required) when the video is being delivered to a MediaLive Multiplex in which case the rate control configuration is controlled by the properties within the Multiplex Program.

Type: H264RateControlMode
Required: False

scanType

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

Type: H264ScanType
Required: False

sceneChangeDetect

Scene change detection. - On: inserts I-frames when scene change is detected. - Off: does not force an I-frame when scene change is detected.

Type: H264SceneChangeDetect
Required: False

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

softness

Softness. Selects quantizer matrix, larger values reduce high-frequency content in the encoded image. If not set to zero, must be greater than 15.

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

spatialAq

Spatial AQ makes adjustments within each frame based on spatial variation of content complexity. The value to enter in this field depends on the value in the Adaptive quantization field: If you have set the Adaptive quantization field to Auto, MediaLive ignores any value in this field. MediaLive will determine if spatial AQ is appropriate and will apply the appropriate strength. If you have set the Adaptive quantization field to a strength, you can set this field to Enabled or Disabled. Enabled: MediaLive will apply spatial AQ using the specified strength. Disabled: MediaLive won't apply spatial AQ. If you have set the Adaptive quantization to Disabled, MediaLive ignores any value in this field and doesn't apply spatial AQ.

Type: H264SpatialAq
Required: False

subgopLength

If set to fixed, use gopNumBFrames B-frames per sub-GOP. If set to dynamic, optimize the number of B-frames used for each sub-GOP to improve visual quality.

Type: H264SubGopLength
Required: False

syntax

Produces a bitstream compliant with SMPTE RP-2027.

Type: H264Syntax
Required: False

temporalAq

Temporal makes adjustments within each frame based on temporal variation of content complexity. The value to enter in this field depends on the value in the Adaptive quantization field: If you have set the Adaptive quantization field to Auto, MediaLive ignores any value in this field. MediaLive will determine if temporal AQ is appropriate and will apply the appropriate strength. If you have set the Adaptive quantization field to a strength, you can set this field to Enabled or Disabled. Enabled: MediaLive will apply temporal AQ using the specified strength. Disabled: MediaLive won't apply temporal AQ. If you have set the Adaptive quantization to Disabled, MediaLive ignores any value in this field and doesn't apply temporal AQ.

Type: H264TemporalAq
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: H264TimecodeInsertionBehavior
Required: False

H264SpatialAq

H264 Spatial Aq

DISABLED
ENABLED

H264SubGopLength

H264 Sub Gop Length

DYNAMIC
FIXED

H264Syntax

H264 Syntax

DEFAULT
RP2027

H264TemporalAq

H264 Temporal Aq

DISABLED
ENABLED

H264TimecodeInsertionBehavior

H264 Timecode Insertion Behavior

DISABLED
PIC_TIMING_SEI

H265AdaptiveQuantization

H265 Adaptive Quantization

AUTO

HIGH

HIGHER

LOW

MAX

MEDIUM

OFF

H265AlternativeTransferFunction

H265 Alternative Transfer Function

INSERT

OMIT

H265ColorMetadata

H265 Color Metadata

IGNORE

INSERT

H265ColorSpaceSettings

H265 Color Space Settings

colorSpacePassthroughSettings

Type: ColorSpacePassthroughSettings

Required: False

hdr10Settings

Type: Hdr10Settings

Required: False

rec601Settings

Type: Rec601Settings
Required: False

rec709Settings

Type: Rec709Settings
Required: False

H265FilterSettings

H265 Filter Settings

temporalFilterSettings

Type: TemporalFilterSettings
Required: False

H265FlickerAq

H265 Flicker Aq

```
DISABLED
ENABLED
```

H265GopSizeUnits

H265 Gop Size Units

```
FRAMES
SECONDS
```

H265Level

H265 Level

```
H265_LEVEL_1
```

H265_LEVEL_2
H265_LEVEL_2_1
H265_LEVEL_3
H265_LEVEL_3_1
H265_LEVEL_4
H265_LEVEL_4_1
H265_LEVEL_5
H265_LEVEL_5_1
H265_LEVEL_5_2
H265_LEVEL_6
H265_LEVEL_6_1
H265_LEVEL_6_2
H265_LEVEL_AUTO

H265LookAheadRateControl

H265 Look Ahead Rate Control

HIGH
LOW
MEDIUM

H265Profile

H265 Profile

MAIN
MAIN_10BIT

H265RateControlMode

H265 Rate Control Mode

CBR
MULTIPLEX
QVBR

H265ScanType

H265 Scan Type

```
INTERLACED
PROGRESSIVE
```

H265SceneChangeDetect

H265 Scene Change Detect

```
DISABLED
ENABLED
```

H265Settings

H265 Settings

adaptiveQuantization

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

Type: H265AdaptiveQuantization
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: AfdSignaling
Required: False

alternativeTransferFunction

Whether or not EML should insert an Alternative Transfer Function SEI message to support backwards compatibility with non-HDR decoders and displays.

Type: H265AlternativeTransferFunction
Required: False

bitrate

Average bitrate in bits/second. Required when the rate control mode is VBR or CBR. Not used for QVBR. In an MS Smooth output group, each output must have a unique value when its bitrate is rounded down to the nearest multiple of 1000.

Type: integer
Required: False
Minimum: 100000
Maximum: 40000000

bufSize

Size of buffer (HRD buffer model) in bits.

Type: integer
Required: False
Minimum: 100000
Maximum: 80000000

colorMetadata

Includes colorspace metadata in the output.

Type: H265ColorMetadata
Required: False

colorSpaceSettings

Color Space settings

Type: H265ColorSpaceSettings
Required: False

filterSettings

Optional filters that you can apply to an encode.

Type: H265FilterSettings
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: FixedAfd
Required: False

flickerAq

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

Type: H265FlickerAq
Required: False

framerateDenominator

Framerate denominator.

Type: integer
Required: True
Minimum: 1
Maximum: 3003

framerateNumerator

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

Type: integer
Required: True
Minimum: 1

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

gopSize

GOP size (keyframe interval) in units of either frames or seconds per gopSizeUnits. If gopSizeUnits is frames, gopSize must be an integer and must be greater than or equal to 1. If gopSizeUnits is seconds, gopSize must be greater than 0, but need not be an integer.

Type: number
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: H265GopSizeUnits
Required: False

level

H.265 Level.

Type: H265Level
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: H265LookAheadRateControl
Required: False

maxBitrate

For QVBR: See the tooltip for Quality level

Type: integer
Required: False
Minimum: 100000
Maximum: 40000000

minIInterval

Only meaningful if sceneChangeDetect is set to enabled. Defaults to 5 if multiplex rate control is used. 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

parDenominator

Pixel Aspect Ratio denominator.

Type: integer
Required: False
Minimum: 1

parNumerator

Pixel Aspect Ratio numerator.

Type: integer

Required: False

Minimum: 1

profile

H.265 Profile.

Type: H265Profile

Required: False

qvbrQualityLevel

Controls the target quality for the video encode. Applies only when the rate control mode is QVBR. Set values for the QVBR quality level field and Max bitrate field that suit your most important viewing devices. Recommended values are: - Primary screen: Quality level: 8 to 10. Max bitrate: 4M - PC or tablet: Quality level: 7. Max bitrate: 1.5M to 3M - Smartphone: Quality level: 6. Max bitrate: 1M to 1.5M

Type: integer

Required: False

Minimum: 1

Maximum: 10

rateControlMode

Rate control mode. QVBR: Quality will match the specified quality level except when it is constrained by the maximum bitrate. Recommended if you or your viewers pay for bandwidth. CBR: Quality varies, depending on the video complexity. Recommended only if you distribute your assets to devices that cannot handle variable bitrates. Multiplex: This rate control mode is only supported (and is required) when the video is being delivered to a MediaLive Multiplex in which case the rate control configuration is controlled by the properties within the Multiplex Program.

Type: H265RateControlMode

Required: False

scanType

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

Type: H265ScanType
Required: False

sceneChangeDetect

Scene change detection.

Type: H265SceneChangeDetect
Required: False

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: 16

tier

H.265 Tier.

Type: H265Tier
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: H265TimecodeInsertionBehavior
Required: False

H265Tier

H265 Tier

```
HIGH
MAIN
```

H265TimecodeInsertionBehavior

H265 Timecode Insertion Behavior

```
DISABLED
PIC_TIMING_SEI
```

Hdr10Settings

Hdr10 Settings

maxCll

Maximum Content Light Level An integer metadata value defining the maximum light level, in nits, of any single pixel within an encoded HDR video stream or file.

Type: integer
Required: False
Minimum: 0
Maximum: 32768

maxFall

Maximum Frame Average Light Level An integer metadata value defining the maximum average light level, in nits, for any single frame within an encoded HDR video stream or file.

Type: integer
Required: False

Minimum: 0
Maximum: 32768

HlsAdMarkers

Hls Ad Markers

ADOBE
ELEMENTAL
ELEMENTAL_SCTE35

HlsAkamaiHttpTransferMode

Hls Akamai Http Transfer Mode

CHUNKED
NON_CHUNKED

HlsAkamaiSettings

Hls Akamai Settings

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

httpTransferMode

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

Type: HlsAkamaiHttpTransferMode
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

salt

Salt for authenticated Akamai.

Type: string
Required: False

token

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

Type: string

Required: False

HlsBasicPutSettings

Hls Basic Put Settings

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

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

HlsCaptionLanguageSetting

Hls Caption Language Setting

INSERT
NONE
OMIT

HlsCdnSettings

Hls Cdn Settings

hlsAkamaiSettings

Type: HlsAkamaiSettings
Required: False

hlsBasicPutSettings

Type: HlsBasicPutSettings
Required: False

hlsMediaStoreSettings

Type: HlsMediaStoreSettings
Required: False

hlsS3Settings

Type: HlsS3Settings
Required: False

hlsWebdavSettings

Type: HlsWebdavSettings

Required: False

HlsClientCache

Hls Client Cache

DISABLED

ENABLED

HlsCodecSpecification

Hls Codec Specification

RFC_4281

RFC_6381

HlsDirectoryStructure

Hls Directory Structure

SINGLE_DIRECTORY

SUBDIRECTORY_PER_STREAM

HlsDiscontinuityTags

Hls Discontinuity Tags

INSERT

NEVER_INSERT

HlsEncryptionType

Hls Encryption Type

AES128

SAMPLE_AES

HlsGroupSettings

Hls Group Settings

adMarkers

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

Type: Array of type HlsAdMarkers
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

baseUrlContent1

Optional. One value per output group. This field is required only if you are completing Base URL content A, and the downstream system has notified you that the media files for pipeline 1 of all outputs are in a location different from the media files for pipeline 0.

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

baseUrlManifest1

Optional. One value per output group. Complete this field only if you are completing Base URL manifest A, and the downstream system has notified you that the child manifest files for pipeline 1 of all outputs are in a location different from the child manifest files for pipeline 0.

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

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: HlsCaptionLanguageSetting

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: HlsClientCache

Required: False

codecSpecification

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

Type: HlsCodecSpecification
Required: False

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
MinLength: 32
MaxLength: 32

destination

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

Type: OutputLocationRef
Required: True

directoryStructure

Place segments in subdirectories.

Type: HlsDirectoryStructure
Required: False

discontinuityTags

Specifies whether to insert EXT-X-DISCONTINUITY tags in the HLS child manifests for this output group. Typically, choose Insert because these tags are required in the manifest (according to the HLS specification) and serve an important purpose. Choose Never Insert only if the downstream

system is doing real-time failover (without using the MediaLive automatic failover feature) and only if that downstream system has advised you to exclude the tags.

Type: HlsDiscontinuityTags
Required: False

encryptionType

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

Type: HlsEncryptionType
Required: False

hlsCdnSettings

Parameters that control interactions with the CDN.

Type: HlsCdnSettings
Required: False

hlsId3SegmentTagging

State of HLS ID3 Segment Tagging

Type: HlsId3SegmentTaggingState
Required: False

iFrameOnlyPlaylists

DISABLED: Do not create an I-frame-only manifest, but do create the master and media manifests (according to the Output Selection field). STANDARD: Create an I-frame-only manifest for each output that contains video, as well as the other manifests (according to the Output Selection field). The I-frame manifest contains a #EXT-X-I-FRAMES-ONLY tag to indicate it is I-frame only, and one or more #EXT-X-BYTERANGE entries identifying the I-frame position. For example, #EXT-X-BYTERANGE:160364@1461888"

Type: IFrameOnlyPlaylistType

Required: False

incompleteSegmentBehavior

Specifies whether to include the final (incomplete) segment in the media output when the pipeline stops producing output because of a channel stop, a channel pause or a loss of input to the pipeline. Auto means that MediaLive decides whether to include the final segment, depending on the channel class and the types of output groups. Suppress means to never include the incomplete segment. We recommend you choose Auto and let MediaLive control the behavior.

Type: HlsIncompleteSegmentBehavior
Required: False

indexNSegments

Applies only if Mode field is LIVE. Specifies the maximum number of segments in the media manifest file. After this maximum, older segments are removed from the media manifest. This number must be smaller than the number in the Keep Segments field.

Type: integer
Required: False
Minimum: 3

inputLossAction

Parameter that control output group behavior on input loss.

Type: InputLossActionForHlsOut
Required: False

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: HlsIvInManifest
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: HlsIvSource
Required: False

keepSegments

Applies only if Mode field is LIVE. Specifies the number of media segments to retain in the destination directory. This number should be bigger than indexNSegments (Num segments). We recommend (value = (2 x indexNsegments) + 1). If this "keep segments" number is too low, the following might happen: the player is still reading a media manifest file that lists this segment, but that segment has been removed from the destination directory (as directed by indexNSegments). This situation would result in a 404 HTTP error on the player.

Type: integer
Required: False
Minimum: 1

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

keyFormatVersions

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

Type: string
Required: False

keyProviderSettings

The key provider settings.

Type: KeyProviderSettings
Required: False

manifestCompression

When set to gzip, compresses HLS playlist.

Type: HlsManifestCompression
Required: False

manifestDurationFormat

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

Type: HlsManifestDurationFormat
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 "vod", all segments are indexed and kept permanently in the destination and manifest. If "live", only the number segments specified in keepSegments and indexNSegments are kept; newer segments replace older segments, which may prevent players from rewinding all the way to the beginning of the event. VOD mode uses HLS EXT-X-PLAYLIST-TYPE of EVENT while the channel is running, converting it to a "VOD" type manifest on completion of the stream.

Type: HlsMode
Required: False

outputSelection

MANIFESTS_AND_SEGMENTS: Generates manifests (master manifest, if applicable, and media manifests) for this output group. VARIANT_MANIFESTS_AND_SEGMENTS: Generates media manifests for this output group, but not a master manifest. SEGMENTS_ONLY: Does not generate any manifests for this output group.

Type: HlsOutputSelection
Required: False

programDateTime

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

Type: HlsProgramDateTime
Required: False

programDateTimeClock

Specifies the algorithm used to drive the HLS EXT-X-PROGRAM-DATE-TIME clock. Options include: INITIALIZE_FROM_OUTPUT_TIMECODE: The PDT clock is initialized as a function of the first output timecode, then incremented by the EXTINF duration of each encoded segment. SYSTEM_CLOCK: The PDT clock is initialized as a function of the UTC wall clock, then incremented by the EXTINF duration of each encoded segment. If the PDT clock diverges from the wall clock by more than 500ms, it is resynchronized to the wall clock.

Type: HlsProgramDateTimeClock
Required: False

programDateTimePeriod

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

Type: integer

Required: False

Minimum: 0

Maximum: 3600

redundantManifest

ENABLED: The master manifest (.m3u8 file) for each pipeline includes information about both pipelines: first its own media files, then the media files of the other pipeline. This feature allows playout device that support stale manifest detection to switch from one manifest to the other, when the current manifest seems to be stale. There are still two destinations and two master manifests, but both master manifests reference the media files from both pipelines. DISABLED: The master manifest (.m3u8 file) for each pipeline includes information about its own pipeline only. For an HLS output group with MediaPackage as the destination, the DISABLED behavior is always followed. MediaPackage regenerates the manifests it serves to players so a redundant manifest from MediaLive is irrelevant.

Type: HlsRedundantManifest

Required: False

segmentLength

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

Type: integer

Required: False

Minimum: 1

segmentationMode

useInputSegmentation has been deprecated. The configured segment size is always used.

Type: HlsSegmentationMode

Required: False

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

streamInfResolution

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

Type: HlsStreamInfResolution
Required: False

timedMetadataId3Frame

Indicates ID3 frame that has the timecode.

Type: HlsTimedMetadataId3Frame
Required: False

timedMetadataId3Period

Timed Metadata interval in seconds.

Type: integer
Required: False
Minimum: 0

timestampDeltaMilliseconds

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

Type: integer
Required: False
Minimum: 0

tsFileMode

SEGMENTED_FILES: Emit the program as segments - multiple .ts media files. SINGLE_FILE: Applies only if Mode field is VOD. Emit the program as a single .ts media file. The media manifest includes #EXT-X-BYTERANGE tags to index segments for playback. A typical use for this value is when sending the output to AWS Elemental MediaConvert, which can accept only a single media file. Playback while the channel is running is not guaranteed due to HTTP server caching.

Type: HlsTsFileMode

Required: False

HlsH265PackagingType

Hls H265 Packaging Type

HEV1

HVC1

HlsId3SegmentTaggingState

State of HLS ID3 Segment Tagging

DISABLED

ENABLED

HlsIncompleteSegmentBehavior

Hls Incomplete Segment Behavior

AUTO

SUPPRESS

HlsInputSettings

Hls Input Settings

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

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

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

retryInterval

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

Type: integer
Required: False
Minimum: 0

scte35Source

Identifies the source for the SCTE-35 messages that MediaLive will ingest. Messages can be ingested from the content segments (in the stream) or from tags in the playlist (the HLS manifest). MediaLive ignores SCTE-35 information in the source that is not selected.

Type: HlsScte35SourceType

Required: False

HlsIvInManifest

Hls Iv In Manifest

```
EXCLUDE
INCLUDE
```

HlsIvSource

Hls Iv Source

```
EXPLICIT
FOLLOWS_SEGMENT_NUMBER
```

HlsManifestCompression

Hls Manifest Compression

```
GZIP
NONE
```

HlsManifestDurationFormat

Hls Manifest Duration Format

```
FLOATING_POINT
INTEGER
```

HlsMediaStoreSettings

Hls Media Store Settings

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

mediaStoreStorageClass

When set to temporal, output files are stored in non-persistent memory for faster reading and writing.

Type: HlsMediaStoreStorageClass
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

HlsMediaStoreStorageClass

Hls Media Store Storage Class

```
TEMPORAL
```

HlsMode

Hls Mode

```
LIVE
VOD
```

HlsOutputSelection

Hls Output Selection

```
MANIFESTS_AND_SEGMENTS
SEGMENTS_ONLY
VARIANT_MANIFESTS_AND_SEGMENTS
```

HlsOutputSettings

Hls Output Settings

h265PackagingType

Only applicable when this output is referencing an H.265 video description. Specifies whether MP4 segments should be packaged as HEV1 or HVC1.

Type: HlsH265PackagingType
Required: False

hlsSettings

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

Type: HlsSettings
Required: True

nameModifier

String concatenated to the end of the destination filename. Accepts \"Format Identifiers \":#formatIdentifierParameters.

Type: string
Required: False
MinLength: 1

segmentModifier

String concatenated to end of segment filenames.

Type: string
Required: False

HlsProgramDateTime

Hls Program Date Time

EXCLUDE
INCLUDE

HlsProgramDateTimeClock

Hls Program Date Time Clock

INITIALIZE_FROM_OUTPUT_TIMECODE

SYSTEM_CLOCK

HlsRedundantManifest

Hls Redundant Manifest

DISABLED
ENABLED

HlsS3LogUploads

Hls S3 Log Uploads

DISABLED
ENABLED

HlsS3Settings

Hls S3 Settings

cannedAcl

Specify the canned ACL to apply to each S3 request. Defaults to none.

Type: S3CannedAcl
Required: False

logUploads

When set to enabled, each fragment upload to CDN or server will be logged.

Type: HlsS3LogUploads
Required: False

HlsScte35SourceType

Hls Scte35 Source Type

MANIFEST

SEGMENTS

HlsSegmentationMode

Hls Segmentation Mode

USE_INPUT_SEGMENTATION
USE_SEGMENT_DURATION

HlsSettings

Hls Settings

audioOnlyHlsSettings

Type: AudioOnlyHlsSettings
Required: False

fmp4HlsSettings

Type: Fmp4HlsSettings
Required: False

frameCaptureHlsSettings

Type: FrameCaptureHlsSettings
Required: False

standardHlsSettings

Type: StandardHlsSettings
Required: False

HlsStreamInfResolution

Hls Stream Inf Resolution

EXCLUDE

INCLUDE

HlsTimedMetadataId3Frame

Hls Timed Metadata Id3 Frame

NONE
PRIV
TDRL

HlsTsFileMode

Hls Ts File Mode

SEGMENTED_FILES
SINGLE_FILE

HlsWebdavHttpTransferMode

Hls Webdav Http Transfer Mode

CHUNKED
NON_CHUNKED

HlsWebdavSettings

Hls Webdav Settings

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

httpTransferMode

Specify whether or not to use chunked transfer encoding to WebDAV.

Type: HlsWebdavHttpTransferMode
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

HtmlMotionGraphicsSettings

Html Motion Graphics Settings

IFrameOnlyPlaylistType

When set to "standard", an I-Frame only playlist will be written out for each video output in the output group. This I-Frame only playlist will contain byte range offsets pointing to the I-frame(s) in each segment.

DISABLED
STANDARD

InputAttachment

automaticInputFailoverSettings

User-specified settings for defining what the conditions are for declaring the input unhealthy and failing over to a different input.

Type: AutomaticInputFailoverSettings
Required: False

inputAttachmentName

User-specified name for the attachment. This is required if the user wants to use this input in an input switch action.

Type: string
Required: False

inputId

The ID of the input

Type: string
Required: False

inputSettings

Settings of an input (caption selector, etc.)

Type: InputSettings
Required: False

InputChannelLevel

Input Channel Level

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

inputChannel

The index of the input channel used as a source.

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

InputCodec

codec in increasing order of complexity

MPEG2
AVC
HEVC

InputDeblockFilter

Input Deblock Filter

DISABLED
ENABLED

InputDenoiseFilter

Input Denoise Filter

DISABLED
ENABLED

InputFilter

Input Filter

AUTO
DISABLED
FORCED

InputLocation

Input Location

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 RTMP destination should have a uri simliar 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

Input Loss Action For Hls Out

```
EMIT_OUTPUT
PAUSE_OUTPUT
```

InputLossActionForMsSmoothOut

Input Loss Action For Ms Smooth Out

```
EMIT_OUTPUT
PAUSE_OUTPUT
```

InputLossActionForRtmpOut

Input Loss Action For Rtmp Out

```
EMIT_OUTPUT
PAUSE_OUTPUT
```

InputLossActionForUdpOut

Input Loss Action For Udp Out

```
DROP_PROGRAM
DROP_TS
EMIT_PROGRAM
```

InputLossBehavior

Input Loss Behavior

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

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
MinLength: 6
MaxLength: 6

inputLossImageSlate

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

Type: InputLocation
Required: False

inputLossImageType

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

Type: InputLossImageType
Required: False

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

InputLossFailoverSettings

MediaLive will perform a failover if content is not detected in this input for the specified period.

inputLossThresholdMsec

The amount of time (in milliseconds) that no input is detected. After that time, an input failover will occur.

Type: integer
Required: False
Minimum: 100

InputLossImageType

Input Loss Image Type

COLOR
SLATE

InputMaximumBitrate

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

MAX_10_MBPS
MAX_20_MBPS
MAX_50_MBPS

InputPreference

Input preference when deciding which input to make active when a previously failed input has recovered. If \"EQUAL_INPUT_PREFERENCE\", then the active input will stay active as long as it is healthy. If \"PRIMARY_INPUT_PREFERRED\", then always switch back to the primary input when it is healthy.

EQUAL_INPUT_PREFERENCE

PRIMARY_INPUT_PREFERRED

InputResolution

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

Live Event input parameters. There can be multiple inputs in a single Live Event.

audioSelectors

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

Type: Array of type AudioSelector
Required: False

captionSelectors

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

Type: Array of type CaptionSelector
Required: False

deblockFilter

Enable or disable the deblock filter when filtering.

Type: InputDeblockFilter
Required: False

denoiseFilter

Enable or disable the denoise filter when filtering.

Type: InputDenoiseFilter
Required: False

filterStrength

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

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

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: InputFilter
Required: False

networkInputSettings

Input settings.

Type: NetworkInputSettings
Required: False

smpte2038DataPreference

Specifies whether to extract applicable ancillary data from a SMPTE-2038 source in this input. Applicable data types are captions, timecode, AFD, and SCTE-104 messages. - PREFER: Extract from SMPTE-2038 if present in this input, otherwise extract from another source (if any). - IGNORE: Never extract any ancillary data from SMPTE-2038.

Type: Smpte2038DataPreference
Required: False

sourceEndBehavior

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

Type: InputSourceEndBehavior
Required: False

videoSelector

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

Type: VideoSelector
Required: False

InputSourceEndBehavior

Input Source End Behavior

CONTINUE
LOOP

InputSpecification

codec

Input codec

Type: InputCodec
Required: False

maximumBitrate

Maximum input bitrate, categorized coarsely

Type: InputMaximumBitrate
Required: False

resolution

Input resolution, categorized coarsely

Type: InputResolution
Required: False

InternalServiceError

message

Type: string
Required: False

InvalidRequest

message

Type: string
Required: False

KeyProviderSettings

Key Provider Settings

staticKeySettings

Type: StaticKeySettings
Required: False

LimitExceeded

message

Type: string
Required: False

LogLevel

The log level the user wants for their channel.

ERROR
WARNING
INFO
DEBUG
DISABLED

M2tsAbsentInputAudioBehavior

M2ts Absent Input Audio Behavior

DROP
ENCODE_SILENCE

M2tsArib

M2ts Arib

DISABLED
ENABLED

M2tsAribCaptionsPidControl

M2ts Arib Captions Pid Control

AUTO
USE_CONFIGURED

M2tsAudioBufferModel

M2ts Audio Buffer Model

ATSC
DVB

M2tsAudioInterval

M2ts Audio Interval

VIDEO_AND_FIXED_INTERVALS
VIDEO_INTERVAL

M2tsAudioStreamType

M2ts Audio Stream Type

ATSC
DVB

M2tsBufferModel

M2ts Buffer Model

MULTIPLEX
NONE

M2tsCcDescriptor

M2ts Cc Descriptor

DISABLED
ENABLED

M2tsEbifControl

M2ts Ebif Control

NONE
PASSTHROUGH

M2tsEbpPlacement

M2ts Ebp Placement

VIDEO_AND_AUDIO_PIDS
VIDEO_PID

M2tsEsRateInPes

M2ts Es Rate In Pes

EXCLUDE
INCLUDE

M2tsKlv

M2ts Klv

NONE
PASSTHROUGH

M2tsNielsenId3Behavior

M2ts Nielsen Id3 Behavior

NO_PASSTHROUGH
PASSTHROUGH

M2tsPcrControl

M2ts Pcr Control

CONFIGURED_PCR_PERIOD
PCR_EVERY_PES_PACKET

M2tsRateMode

M2ts Rate Mode

CBR
VBR

M2tsScte35Control

M2ts Scte35 Control

NONE
PASSTHROUGH

M2tsSegmentationMarkers

M2ts Segmentation Markers

EBP
EBP_LEGACY
NONE
PSI_SEGSTART
RAI_ADAPT
RAI_SEGSTART

M2tsSegmentationStyle

M2ts Segmentation Style

MAINTAIN_CADENCE
RESET_CADENCE

M2tsSettings

M2ts Settings

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: M2tsAbsentInputAudioBehavior

Required: False

arib

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

Type: M2tsArib
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

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: M2tsAribCaptionsPidControl
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: M2tsAudioBufferModel
Required: False

audioFramesPerPes

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

Type: integer

Required: False
Minimum: 0

audioPids

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

Type: string
Required: False

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: M2tsAudioStreamType
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

bufferModel

Controls the timing accuracy for output network traffic. Leave as MULTIPLEX to ensure accurate network packet timing. Or set to NONE, which might result in lower latency but will result in more variability in output network packet timing. This variability might cause interruptions, jitter, or bursty behavior in your playback or receiving devices.

Type: M2tsBufferModel

Required: False

ccDescriptor

When set to enabled, generates captionServiceDescriptor in PMT.

Type: M2tsCcDescriptor
Required: False

dvbNitSettings

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

Type: DvbNitSettings
Required: False

dvbSdtSettings

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

Type: DvbSdtSettings
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

dvbTdtSettings

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

Type: DvbTdtSettings
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

ebif

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

Type: M2tsEbifControl
Required: False

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: M2tsAudioInterval
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

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: M2tsEbpPlacement
Required: False

ecmPid

This field is unused and deprecated.

Type: string
Required: False

esRateInPes

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

Type: M2tsEsRateInPes
Required: False

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

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

fragmentTime

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

Type: number
Required: False
Minimum: 0

klv

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

Type: M2tsKlv
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

nielsenId3Behavior

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

Type: M2tsNielsenId3Behavior
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

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

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: M2tsPcrControl
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

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

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

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

programNum

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

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

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: M2tsRateMode
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

scte35Control

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

Type: M2tsScte35Control

Required: False

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

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: M2tsSegmentationMarkers

Required: False

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: M2tsSegmentationStyle
Required: False

segmentationTime

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

Type: number
Required: False
Minimum: 1

timedMetadataBehavior

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

Type: M2tsTimedMetadataBehavior
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

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. Valid values are 32 (or 0x20)..8182 (or 0x1ff6).

Type: string
Required: False

M2tsTimedMetadataBehavior

M2ts Timed Metadata Behavior

NO_PASSTHROUGH
PASSTHROUGH

M3u8NielsenId3Behavior

M3u8 Nielsen Id3 Behavior

NO_PASSTHROUGH
PASSTHROUGH

M3u8PcrControl

M3u8 Pcr Control

CONFIGURED_PCR_PERIOD
PCR_EVERY_PES_PACKET

M3u8Scte35Behavior

M3u8 Scte35 Behavior

NO_PASSTHROUGH
PASSTHROUGH

M3u8Settings

Settings information for the .m3u8 container

audioFramesPerPes

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

Type: integer
Required: False
Minimum: 0

audioPids

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

Type: string
Required: False

ecmPid

This parameter is unused and deprecated.

Type: string
Required: False

nielsenId3Behavior

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

Type: M3u8NielsenId3Behavior
Required: False

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

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: M3u8PcrControl
Required: False

pcrPeriod

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

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

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

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

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

programNum

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

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

scte35Behavior

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

Type: M3u8Scte35Behavior
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

timedMetadataBehavior

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

Type: M3u8TimedMetadataBehavior
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

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

M3u8TimedMetadataBehavior

M3u8 Timed Metadata Behavior

NO_PASSTHROUGH
PASSTHROUGH

MaintenanceDay

The currently selected maintenance day.

MONDAY
TUESDAY
WEDNESDAY
THURSDAY
FRIDAY
SATURDAY
SUNDAY

MaintenanceStatus

maintenanceDay

The currently selected maintenance day.

Type: MaintenanceDay
Required: False

maintenanceDeadline

Maintenance is required by the displayed date and time. Date and time is in ISO.

Type: string
Required: False
Format: string

maintenanceScheduledDate

The currently scheduled maintenance date and time. Date and time is in ISO.

Type: string
Required: False
Format: string

maintenanceStartTime

The currently selected maintenance start time. Time is in UTC.

Type: string
Required: False

MediaPackageGroupSettings

Media Package Group Settings

destination

MediaPackage channel destination.

Type: OutputLocationRef
Required: True

MediaPackageOutputDestinationSettings

MediaPackage Output Destination Settings

channelId

ID of the channel in MediaPackage that is the destination for this output group. You do not need to specify the individual inputs in MediaPackage; MediaLive will handle the connection of the two MediaLive pipelines to the two MediaPackage inputs. The MediaPackage channel and MediaLive channel must be in the same region.

Type: string
Required: False
MinLength: 1

MediaPackageOutputSettings

Media Package Output Settings

MotionGraphicsConfiguration

Motion Graphics Configuration

motionGraphicsInsertion

Type: MotionGraphicsInsertion
Required: False

motionGraphicsSettings

Motion Graphics Settings

Type: MotionGraphicsSettings
Required: True

MotionGraphicsInsertion

Motion Graphics Insertion

```
DISABLED
ENABLED
```

MotionGraphicsSettings

Motion Graphics Settings

htmlMotionGraphicsSettings

Type: HtmlMotionGraphicsSettings
Required: False

Mp2CodingMode

Mp2 Coding Mode

CODING_MODE_1_0

CODING_MODE_2_0

Mp2Settings

Mp2 Settings

bitrate

Average bitrate in bits/second.

Type: number

Required: False

codingMode

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

Type: Mp2CodingMode

Required: False

sampleRate

Sample rate in Hz.

Type: number

Required: False

Mpeg2AdaptiveQuantization

Mpeg2 Adaptive Quantization

AUTO

HIGH

LOW

MEDIUM

OFF

Mpeg2ColorMetadata

Mpeg2 Color Metadata

IGNORE
INSERT

Mpeg2ColorSpace

Mpeg2 Color Space

AUTO
PASSTHROUGH

Mpeg2DisplayRatio

Mpeg2 Display Ratio

DISPLAYRATIO16X9
DISPLAYRATIO4X3

Mpeg2FilterSettings

Mpeg2 Filter Settings

temporalFilterSettings

Type: TemporalFilterSettings
Required: False

Mpeg2GopSizeUnits

Mpeg2 Gop Size Units

FRAMES

SECONDS

Mpeg2ScanType

Mpeg2 Scan Type

INTERLACED
PROGRESSIVE

Mpeg2Settings

Mpeg2 Settings

adaptiveQuantization

Choose Off to disable adaptive quantization. Or choose another value to enable the quantizer and set its strength. The strengths are: Auto, Off, Low, Medium, High. When you enable this field, MediaLive allows intra-frame quantizers to vary, which might improve visual quality.

Type: Mpeg2AdaptiveQuantization
Required: False

afdSignaling

Indicates the AFD values that MediaLive will write into the video encode. If you do not know what AFD signaling is, or if your downstream system has not given you guidance, choose AUTO. AUTO: MediaLive will try to preserve the input AFD value (in cases where multiple AFD values are valid). FIXED: MediaLive will use the value you specify in fixedAFD.

Type: AfdSignaling
Required: False

colorMetadata

Specifies whether to include the color space metadata. The metadata describes the color space that applies to the video (the colorSpace field). We recommend that you insert the metadata.

Type: Mpeg2ColorMetadata
Required: False

colorSpace

Choose the type of color space conversion to apply to the output. For detailed information on setting up both the input and the output to obtain the desired color space in the output, see the section on \"MediaLive Features - Video - color space\" in the MediaLive User Guide. PASSTHROUGH: Keep the color space of the input content - do not convert it. AUTO:Convert all content that is SD to rec 601, and convert all content that is HD to rec 709.

Type: Mpeg2ColorSpace
Required: False

displayAspectRatio

Sets the pixel aspect ratio for the encode.

Type: Mpeg2DisplayRatio
Required: False

filterSettings

Optionally specify a noise reduction filter, which can improve quality of compressed content. If you do not choose a filter, no filter will be applied. TEMPORAL: This filter is useful for both source content that is noisy (when it has excessive digital artifacts) and source content that is clean. When the content is noisy, the filter cleans up the source content before the encoding phase, with these two effects: First, it improves the output video quality because the content has been cleaned up. Secondly, it decreases the bandwidth because MediaLive does not waste bits on encoding noise. When the content is reasonably clean, the filter tends to decrease the bitrate.

Type: Mpeg2FilterSettings
Required: False

fixedAfd

Complete this field only when afdSignaling is set to FIXED. Enter the AFD value (4 bits) to write on all frames of the video encode.

Type: FixedAfd
Required: False

framerateDenominator

description": "The framerate denominator. For example, 1001. The framerate is the numerator divided by the denominator. For example, 24000 / 1001 = 23.976 FPS.

Type: integer
Required: True
Minimum: 1

framerateNumerator

The framerate numerator. For example, 24000. The framerate is the numerator divided by the denominator. For example, 24000 / 1001 = 23.976 FPS.

Type: integer
Required: True
Minimum: 1

gopClosedCadence

MPEG2: default is open GOP.

Type: integer
Required: False
Minimum: 0

gopNumBFrames

Relates to the GOP structure. The number of B-frames between reference frames. If you do not know what a B-frame is, use the default.

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

gopSize

Relates to the GOP structure. The GOP size (keyframe interval) in the units specified in gopSizeUnits. If you do not know what GOP is, use the default. If gopSizeUnits is frames, then the gopSize must be an integer and must be greater than or equal to 1. If gopSizeUnits is seconds, the gopSize must be greater than 0, but does not need to be an integer.

Type: number
Required: False

gopSizeUnits

Relates to the GOP structure. Specifies whether the gopSize is specified in frames or seconds. If you do not plan to change the default gopSize, leave the default. If you specify SECONDS, MediaLive will internally convert the gop size to a frame count.

Type: Mpeg2GopSizeUnits
Required: False

scanType

Set the scan type of the output to PROGRESSIVE or INTERLACED (top field first).

Type: Mpeg2ScanType
Required: False

subgopLength

Relates to the GOP structure. If you do not know what GOP is, use the default. FIXED: Set the number of B-frames in each sub-GOP to the value in gopNumBFrames. DYNAMIC: Let MediaLive optimize the number of B-frames in each sub-GOP, to improve visual quality.

Type: Mpeg2SubGopLength
Required: False

timecodeInsertion

Determines how MediaLive inserts timecodes in the output video. For detailed information about setting up the input and the output for a timecode, see the section on \"MediaLive Features -

Timecode configuration\" in the MediaLive User Guide. DISABLED: do not include timecodes. GOP_TIMECODE: Include timecode metadata in the GOP header.

Type: Mpeg2TimecodeInsertionBehavior
Required: False

Mpeg2SubGopLength

Mpeg2 Sub Gop Length

```
DYNAMIC
FIXED
```

Mpeg2TimecodeInsertionBehavior

Mpeg2 Timecode Insertion Behavior

```
DISABLED
GOP_TIMECODE
```

MsSmoothGroupSettings

Ms Smooth Group Settings

acquisitionPointId

The ID to include in each message in the sparse track. Ignored if sparseTrackType is NONE.

Type: string
Required: False

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: SmoothGroupAudioOnlyTimecodeControl
Required: False

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.

Type: SmoothGroupCertificateMode
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

destination

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

Type: OutputLocationRef
Required: True

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

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: SmoothGroupEventIdMode
Required: False

eventStopBehavior

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

Type: SmoothGroupEventStopBehavior
Required: False

filecacheDuration

Size in seconds of file cache for streaming outputs.

Type: integer
Required: False
Minimum: 0

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

inputLossAction

Parameter that control output group behavior on input loss.

Type: InputLossActionForMsSmoothOut
Required: False

numRetries

Number of retry attempts.

Type: integer
Required: False
Minimum: 0

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

segmentationMode

useInputSegmentation has been deprecated. The configured segment size is always used.

Type: SmoothGroupSegmentationMode
Required: False

sendDelayMs

Number of milliseconds to delay the output from the second pipeline.

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

sparseTrackType

Identifies the type of data to place in the sparse track: - SCTE35: Insert SCTE-35 messages from the source content. With each message, insert an IDR frame to start a new segment. - SCTE35_WITHOUT_SEGMENTATION: Insert SCTE-35 messages from the source content. With each

message, insert an IDR frame but don't start a new segment. - NONE: Don't generate a sparse track for any outputs in this output group.

Type: SmoothGroupSparseTrackType
Required: False

streamManifestBehavior

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

Type: SmoothGroupStreamManifestBehavior
Required: False

timestampOffset

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

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: SmoothGroupTimestampOffsetMode
Required: False

MsSmoothH265PackagingType

Ms Smooth H265 Packaging Type

HEV1
HVC1

MsSmoothOutputSettings

Ms Smooth Output Settings

h265PackagingType

Only applicable when this output is referencing an H.265 video description. Specifies whether MP4 segments should be packaged as HEV1 or HVC1.

Type: MsSmoothH265PackagingType
Required: False

nameModifier

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

Type: string
Required: False

MultiplexGroupSettings

Multiplex Group Settings

MultiplexOutputSettings

Multiplex Output Settings

destination

Destination is a Multiplex.

Type: OutputLocationRef
Required: True

MultiplexProgramChannelDestinationSettings

Multiplex Program Input Destination Settings for outputting a Channel to a Multiplex

multiplexId

The ID of the Multiplex that the encoder is providing output to. You do not need to specify the individual inputs to the Multiplex; MediaLive will handle the connection of the two MediaLive pipelines to the two Multiplex instances. The Multiplex must be in the same region as the Channel.

Type: string
Required: False
MinLength: 1

programName

The program name of the Multiplex program that the encoder is providing output to.

Type: string
Required: False
MinLength: 1

NetworkInputServerValidation

Network Input Server Validation

```
CHECK_CRYPTOGRAPHY_AND_VALIDATE_NAME
CHECK_CRYPTOGRAPHY_ONLY
```

NetworkInputSettings

Network source to transcode. Must be accessible to the Elemental Live node that is running the live event through a network connection.

hlsInputSettings

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

Type: HlsInputSettings
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: NetworkInputServerValidation
Required: False

NielsenCBET

Nielsen CBET

cbetCheckDigitString

Enter the CBET check digits to use in the watermark.

Type: string
Required: True
MinLength: 2
MaxLength: 2

cbetStepaside

Determines the method of CBET insertion mode when prior encoding is detected on the same layer.

Type: NielsenWatermarksCbetStepaside
Required: True

csid

Enter the CBET Source ID (CSID) to use in the watermark

Type: string
Required: True
MinLength: 1
MaxLength: 7

NielsenConfiguration

Nielsen Configuration

distributorId

Enter the Distributor ID assigned to your organization by Nielsen.

Type: string
Required: False

nielsenPcmToId3Tagging

Enables Nielsen PCM to ID3 tagging

Type: NielsenPcmToId3TaggingState
Required: False

NielsenNaesIiNw

Nielsen Naes Ii Nw

checkDigitString

Enter the check digit string for the watermark

Type: string
Required: True
MinLength: 2
MaxLength: 2

sid

Enter the Nielsen Source ID (SID) to include in the watermark

Type: number
Required: True
Minimum: 1
Maximum: 65535

NielsenPcmToId3TaggingState

State of Nielsen PCM to ID3 tagging

DISABLED
ENABLED

NielsenWatermarksCbetStepaside

Nielsen Watermarks Cbet Stepaside

```
DISABLED
ENABLED
```

NielsenWatermarksDistributionTypes

Nielsen Watermarks Distribution Types

```
FINAL_DISTRIBUTOR
PROGRAM_CONTENT
```

NielsenWatermarksSettings

Nielsen Watermarks Settings

nielsenCbetSettings

Complete these fields only if you want to insert watermarks of type Nielsen CBET

Type: NielsenCBET
Required: False

nielsenDistributionType

Choose the distribution types that you want to assign to the watermarks: - PROGRAM_CONTENT - FINAL_DISTRIBUTOR

Type: NielsenWatermarksDistributionTypes
Required: False

nielsenNaesIiNwSettings

Complete these fields only if you want to insert watermarks of type Nielsen NAES II (N2) and Nielsen NAES VI (NW).

Type: NielsenNaesIiNw
Required: False

Output

Output settings. There can be multiple outputs within a group.

audioDescriptionNames

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

Type: Array of 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
MinLength: 1
MaxLength: 255

outputSettings

Output type-specific settings.

Type: OutputSettings
Required: True

videoDescriptionName

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

Type: string
Required: False

OutputDestination

id

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

Type: string
Required: False

mediaPackageSettings

Destination settings for a MediaPackage output; one destination for both encoders.

Type: Array of type MediaPackageOutputDestinationSettings
Required: False

multiplexSettings

Destination settings for a Multiplex output; one destination for both encoders.

Type: MultiplexProgramChannelDestinationSettings
Required: False

settings

Destination settings for a standard output; one destination for each redundant encoder.

Type: Array of type OutputDestinationSettings
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

Output groups for this Live Event. Output groups contain information about where streams should be distributed.

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
MaxLength: 32

outputGroupSettings

Settings associated with the output group.

Type: OutputGroupSettings

Required: True

outputs

Type: Array of type Output

Required: True

OutputGroupSettings

Output Group Settings

archiveGroupSettings

Type: ArchiveGroupSettings

Required: False

frameCaptureGroupSettings

Type: FrameCaptureGroupSettings

Required: False

hlsGroupSettings

Type: HlsGroupSettings

Required: False

mediaPackageGroupSettings

Type: MediaPackageGroupSettings

Required: False

msSmoothGroupSettings

Type: MsSmoothGroupSettings

Required: False

multiplexGroupSettings

Type: MultiplexGroupSettings
Required: False

rtmpGroupSettings

Type: RtmpGroupSettings
Required: False

udpGroupSettings

Type: UdpGroupSettings
Required: False

OutputLocationRef

Reference to an OutputDestination ID defined in the channel

destinationRefId

Type: string
Required: False

OutputSettings

Output Settings

archiveOutputSettings

Type: ArchiveOutputSettings
Required: False

frameCaptureOutputSettings

Type: FrameCaptureOutputSettings
Required: False

hlsOutputSettings

Type: HlsOutputSettings
Required: False

mediaPackageOutputSettings

Type: MediaPackageOutputSettings
Required: False

msSmoothOutputSettings

Type: MsSmoothOutputSettings
Required: False

multiplexOutputSettings

Type: MultiplexOutputSettings
Required: False

rtmpOutputSettings

Type: RtmpOutputSettings
Required: False

udpOutputSettings

Type: UdpOutputSettings
Required: False

PassThroughSettings

Pass Through Settings

PipelineDetail

Runtime details of a pipeline when a channel is running.

activeInputAttachmentName

The name of the active input attachment currently being ingested by this pipeline.

Type: string
Required: False

activeInputSwitchActionName

The name of the input switch schedule action that occurred most recently and that resulted in the switch to the current input attachment for this pipeline.

Type: string
Required: False

activeMotionGraphicsActionName

The name of the motion graphics activate action that occurred most recently and that resulted in the current graphics URI for this pipeline.

Type: string
Required: False

activeMotionGraphicsUri

The current URI being used for HTML5 motion graphics for this pipeline.

Type: string
Required: False

pipelineId

Pipeline ID

Type: string
Required: False

RawSettings

Raw Settings

Rec601Settings

Rec601 Settings

Rec709Settings

Rec709 Settings

RemixSettings

Remix Settings

channelMappings

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

Type: Array of type AudioChannelMapping
Required: True

channelsIn

Number of input channels to be used.

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

channelsOut

Number of output channels to be produced. Valid values: 1, 2, 4, 6, 8

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

ResourceConflict

message

Type: string
Required: False

ResourceNotFound

message

Type: string
Required: False

RtmpAdMarkers

Rtmp Ad Markers

```
ON_CUE_POINT_SCTE35
```

RtmpCacheFullBehavior

Rtmp Cache Full Behavior

```
DISCONNECT_IMMEDIATELY
WAIT_FOR_SERVER
```

RtmpCaptionData

Rtmp Caption Data

```
ALL
FIELD1_608
FIELD1_AND_FIELD2_608
```

RtmpCaptionInfoDestinationSettings

Rtmp Caption Info Destination Settings

RtmpGroupSettings

Rtmp Group Settings

adMarkers

Choose the ad marker type for this output group. MediaLive will create a message based on the content of each SCTE-35 message, format it for that marker type, and insert it in the datastream.

Type: Array of type RtmpAdMarkers
Required: False

authenticationScheme

Authentication scheme to use when connecting with CDN

Type: AuthenticationScheme
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: RtmpCacheFullBehavior
Required: False

cacheLength

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

Type: integer
Required: False
Minimum: 30

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: RtmpCaptionData
Required: False

inputLossAction

Controls the behavior of this RTMP group if input becomes unavailable. - emitOutput: Emit a slate until input returns. - pauseOutput: Stop transmitting data until input returns. This does not close the underlying RTMP connection.

Type: InputLossActionForRtmpOut
Required: False

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

RtmpOutputCertificateMode

Rtmp Output Certificate Mode

```
SELF_SIGNED
VERIFY_AUTHENTICITY
```

RtmpOutputSettings

Rtmp Output Settings

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: RtmpOutputCertificateMode
Required: False

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

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
Required: True

numRetries

Number of retry attempts.

Type: integer
Required: False
Minimum: 0

S3CannedAcl

S3 Canned Acl

```
AUTHENTICATED_READ
BUCKET_OWNER_FULL_CONTROL
```

BUCKET_OWNER_READ
PUBLIC_READ

Scte20Convert608To708

Scte20 Convert608 To708

DISABLED
UPCONVERT

Scte20PlusEmbeddedDestinationSettings

Scte20 Plus Embedded Destination Settings

Scte20SourceSettings

Scte20 Source Settings

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: Scte20Convert608To708
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

Scte27DestinationSettings

Scte27 Destination Settings

Scte27OcrLanguage

Scte27 Ocr Language

DEU
ENG
FRA
NLD
POR
SPA

Scte27SourceSettings

Scte27 Source Settings

ocrLanguage

If you will configure a WebVTT caption description that references this caption selector, use this field to provide the language to consider when translating the image-based source to text.

Type: Scte27OcrLanguage
Required: False

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

Scte35 Apos No Regional Blackout Behavior

FOLLOW
IGNORE

Scte35AposWebDeliveryAllowedBehavior

Scte35 Apos Web Delivery Allowed Behavior

FOLLOW
IGNORE

Scte35SpliceInsert

Scte35 Splice Insert

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

noRegionalBlackoutFlag

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

Type: Scte35SpliceInsertNoRegionalBlackoutBehavior
Required: False

webDeliveryAllowedFlag

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

Type: Scte35SpliceInsertWebDeliveryAllowedBehavior

Required: False

Scte35SpliceInsertNoRegionalBlackoutBehavior

Scte35 Splice Insert No Regional Blackout Behavior

FOLLOW
IGNORE

Scte35SpliceInsertWebDeliveryAllowedBehavior

Scte35 Splice Insert Web Delivery Allowed Behavior

FOLLOW
IGNORE

Scte35TimeSignalApos

Scte35 Time Signal Apos

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

noRegionalBlackoutFlag

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

Type: Scte35AposNoRegionalBlackoutBehavior
Required: False

webDeliveryAllowedFlag

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

Type: Scte35AposWebDeliveryAllowedBehavior

Required: False

SmoothGroupAudioOnlyTimecodeControl

Smooth Group Audio Only Timecode Control

PASSTHROUGH

USE_CONFIGURED_CLOCK

SmoothGroupCertificateMode

Smooth Group Certificate Mode

SELF_SIGNED

VERIFY_AUTHENTICITY

SmoothGroupEventIdMode

Smooth Group Event Id Mode

NO_EVENT_ID

USE_CONFIGURED

USE_TIMESTAMP

SmoothGroupEventStopBehavior

Smooth Group Event Stop Behavior

NONE

SEND_EOS

SmoothGroupSegmentationMode

Smooth Group Segmentation Mode

USE_INPUT_SEGMENTATION
USE_SEGMENT_DURATION

SmoothGroupSparseTrackType

Smooth Group Sparse Track Type

NONE
SCTE_35
SCTE_35_WITHOUT_SEGMENTATION

SmoothGroupStreamManifestBehavior

Smooth Group Stream Manifest Behavior

DO_NOT_SEND
SEND

SmoothGroupTimestampOffsetMode

Smooth Group Timestamp Offset Mode

USE_CONFIGURED_OFFSET
USE_EVENT_START_DATE

Smpte2038DataPreference

Smpte2038 Data Preference

IGNORE
PREFER

SmpteTtDestinationSettings

Smpte Tt Destination Settings

StandardHlsSettings

Standard Hls Settings

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

m3u8Settings

Type: M3u8Settings
Required: True

StaticKeySettings

Static Key Settings

keyProviderServer

The URL of the license server used for protecting content.

Type: InputLocation
Required: False

staticKeyValue

Static key value as a 32 character hexadecimal string.

Type: string
Required: True
MinLength: 32

MaxLength: 32

Tags

key-value pairs

Type: string

TeletextDestinationSettings

Teletext Destination Settings

TeletextSourceSettings

Teletext Source Settings

outputRectangle

Optionally defines a region where TTML style captions will be displayed

Type: CaptionRectangle
Required: False

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

TemporalFilterPostFilterSharpening

Temporal Filter Post Filter Sharpening

AUTO
DISABLED
ENABLED

TemporalFilterSettings

Temporal Filter Settings

postFilterSharpening

If you enable this filter, the results are the following: - If the source content is noisy (it contains excessive digital artifacts), the filter cleans up the source. - If the source content is already clean, the filter tends to decrease the bitrate, especially when the rate control mode is QVBR.

Type: TemporalFilterPostFilterSharpening
Required: False

strength

Choose a filter strength. We recommend a strength of 1 or 2. A higher strength might take out good information, resulting in an image that is overly soft.

Type: TemporalFilterStrength
Required: False

TemporalFilterStrength

Temporal Filter Strength

AUTO
STRENGTH_1
STRENGTH_2
STRENGTH_3
STRENGTH_4
STRENGTH_5
STRENGTH_6
STRENGTH_7
STRENGTH_8
STRENGTH_9
STRENGTH_10
STRENGTH_11
STRENGTH_12

STRENGTH_13
STRENGTH_14
STRENGTH_15
STRENGTH_16

TimecodeConfig

Timecode Config

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: TimecodeConfigSource
Required: True

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

TimecodeConfigSource

Timecode Config Source

EMBEDDED
SYSTEMCLOCK
ZEROBASED

TtmlDestinationSettings

Ttml Destination Settings

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: TtmlDestinationStyleControl
Required: False

TtmlDestinationStyleControl

Ttml Destination Style Control

```
PASSTHROUGH
USE_CONFIGURED
```

UdpContainerSettings

Udp Container Settings

m2tsSettings

Type: M2tsSettings
Required: False

UdpGroupSettings

Udp Group Settings

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: InputLossActionForUdpOut
Required: False

timedMetadataId3Frame

Indicates ID3 frame that has the timecode.

Type: UdpTimedMetadataId3Frame
Required: False

timedMetadataId3Period

Timed Metadata interval in seconds.

Type: integer
Required: False
Minimum: 0

UdpOutputSettings

Udp Output Settings

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

containerSettings

Type: UdpContainerSettings
Required: True

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
Required: True

fecOutputSettings

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

Type: FecOutputSettings
Required: False

UdpTimedMetadataId3Frame

Udp Timed Metadata Id3 Frame

```
NONE
PRIV
TDRL
```

VideoBlackFailoverSettings

blackDetectThreshold

A value used in calculating the threshold below which MediaLive considers a pixel to be 'black'. For the input to be considered black, every pixel in a frame must be below this threshold. The threshold is calculated as a percentage (expressed as a decimal) of white. Therefore .1 means 10% white (or 90% black). Note how the formula works for any color depth. For example, if you set this field to 0.1 in 10-bit color depth: (1023*0.1=102.3), which means a pixel value of 102 or less is 'black'. If you set this field to .1 in an 8-bit color depth: (255*0.1=25.5), which means a pixel value of 25 or less is 'black'. The range is 0.0 to 1.0, with any number of decimal places.

Type: number
Required: False
Minimum: 0

Maximum: 1

videoBlackThresholdMsec

The amount of time (in milliseconds) that the active input must be black before automatic input failover occurs.

Type: integer
Required: False
Minimum: 1000

VideoCodecSettings

Video Codec Settings

frameCaptureSettings

Type: FrameCaptureSettings
Required: False

h264Settings

Type: H264Settings
Required: False

h265Settings

Type: H265Settings
Required: False

mpeg2Settings

Type: Mpeg2Settings
Required: False

VideoDescription

Video settings for this stream.

codecSettings

Video codec settings.

Type: VideoCodecSettings
Required: False

height

Output video height, in pixels. Must be an even number. For most codecs, you can leave this field and width blank in order to use the height and width (resolution) from the source. Note, however, that leaving blank is not recommended. For the Frame Capture codec, height and width are required.

Type: integer
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

respondToAfd

Indicates how MediaLive will respond to the AFD values that might be in the input video. If you do not know what AFD signaling is, or if your downstream system has not given you guidance, choose PASSTHROUGH. RESPOND: MediaLive clips the input video using a formula that uses the AFD values (configured in afdSignaling), the input display aspect ratio, and the output display aspect ratio. MediaLive also includes the AFD values in the output, unless the codec for this encode is FRAME_CAPTURE. PASSTHROUGH: MediaLive ignores the AFD values and does not clip the video. But MediaLive does include the values in the output. NONE: MediaLive does not clip the input video and does not include the AFD values in the output

Type: VideoDescriptionRespondToAfd
Required: False

scalingBehavior

STRETCH_TO_OUTPUT configures the output position to stretch the video to the specified output resolution (height and width). This option will override any position value. DEFAULT may insert black boxes (pillar boxes or letter boxes) around the video to provide the specified output resolution.

Type: VideoDescriptionScalingBehavior
Required: False

sharpness

Changes the strength of the anti-alias filter used for scaling. 0 is the softest setting, 100 is the sharpest. A setting of 50 is recommended for most content.

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

width

Output video width, in pixels. Must be an even number. For most codecs, you can leave this field and height blank in order to use the height and width (resolution) from the source. Note, however, that leaving blank is not recommended. For the Frame Capture codec, height and width are required.

Type: integer
Required: False

VideoDescriptionRespondToAfd

Video Description Respond To Afd

NONE
PASSTHROUGH
RESPOND

VideoDescriptionScalingBehavior

Video Description Scaling Behavior

DEFAULT
STRETCH_TO_OUTPUT

VideoSelector

Specifies a particular video stream within an input source. An input may have only a single video selector.

colorSpace

Specifies the color space of an input. This setting works in tandem with colorSpaceUsage and a video description's colorSpaceSettingsChoice to determine if any conversion will be performed.

Type: VideoSelectorColorSpace
Required: False

colorSpaceSettings

Color space settings

Type: VideoSelectorColorSpaceSettings
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: VideoSelectorColorSpaceUsage
Required: False

selectorSettings

The video selector settings.

Type: VideoSelectorSettings
Required: False

VideoSelectorColorSpace

Video Selector Color Space

FOLLOW
HDR10
HLG_2020
REC_601
REC_709

VideoSelectorColorSpaceSettings

Video Selector Color Space Settings

hdr10Settings

Type: Hdr10Settings
Required: False

VideoSelectorColorSpaceUsage

Video Selector Color Space Usage

FALLBACK
FORCE

VideoSelectorPid

Video Selector Pid

pid

Selects a specific PID from within a video source.

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

VideoSelectorProgramId

Video Selector Program Id

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

Video Selector Settings

videoSelectorPid

Type: VideoSelectorPid
Required: False

videoSelectorProgramId

Type: VideoSelectorProgramId
Required: False

VpcOutputSettingsDescription

The properties for a private VPC Output

availabilityZones

The Availability Zones where the vpc subnets are located. The first Availability Zone applies to the first subnet in the list of subnets. The second Availability Zone applies to the second subnet.

Type: Array of type string
Required: False

networkInterfaceIds

A list of Elastic Network Interfaces created by MediaLive in the customer's VPC

Type: Array of type string
Required: False

securityGroupIds

A list of up EC2 VPC security group IDs attached to the Output VPC network interfaces.

Type: Array of type string
Required: False

subnetIds

A list of VPC subnet IDs from the same VPC. If STANDARD channel, subnet IDs must be mapped to two unique availability zones (AZ).

Type: Array of type string
Required: False

WavCodingMode

Wav Coding Mode

```
CODING_MODE_1_0
```

CODING_MODE_2_0
CODING_MODE_4_0
CODING_MODE_8_0

WavSettings

Wav Settings

bitDepth

Bits per sample.

Type: number
Required: False

codingMode

The audio coding mode for the WAV audio. The mode determines the number of channels in the audio.

Type: WavCodingMode
Required: False

sampleRate

Sample rate in Hz.

Type: number
Required: False

WebvttDestinationSettings

Webvtt Destination Settings

styleControl

Controls whether the color and position of the source captions is passed through to the WebVTT output captions. PASSTHROUGH - Valid only if the source captions are EMBEDDED or TELETEXT.

NO_STYLE_DATA - Don't pass through the style. The output captions will not contain any font styling information.

Type: WebvttDestinationStyleControl
Required: False

WebvttDestinationStyleControl

Webvtt Destination Style Control

```
NO_STYLE_DATA
PASSTHROUGH
```

See also

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

StartChannel

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

Channels: stop

URI

`/prod/channels/channelId/stop`

HTTP methods

POST

Operation ID: `StopChannel`

Path parameters

Name	Type	Required	Description
channelId	String	True	

Responses

Status code	Response model	Description
200	Channel	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
409	ResourceConflict	409 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

Schemas

Response bodies

Channel schema

```
{
  "arn": "string",
  "cdiInputSpecification": {
    "resolution": enum
  },
  "channelClass": enum,
  "destinations": [
    {
      "id": "string",
      "mediaPackageSettings": [
        {
          "channelId": "string"
        }
      ],
      "multiplexSettings": {
        "multiplexId": "string",
        "programName": "string"
      },
      "settings": [
        {
          "passwordParam": "string",
          "streamName": "string",
          "url": "string",
          "username": "string"
        }
      ]
    }
  ],
  "egressEndpoints": [
    {
      "sourceIp": "string"
    }
  ],
  "encoderSettings": {
    "audioDescriptions": [
      {
        "audioNormalizationSettings": {
          "algorithm": enum,
          "algorithmControl": enum,
          "targetLkfs": number
        },
        "audioSelectorName": "string",
```

```
        "audioType": enum,
        "audioTypeControl": enum,
        "audioWatermarkingSettings": {
          "nielsenWatermarksSettings": {
            "nielsenCbetSettings": {
              "cbetCheckDigitString": "string",
              "cbetStepaside": enum,
              "csid": "string"
            },
            "nielsenDistributionType": enum,
            "nielsenNaesIiNwSettings": {
              "checkDigitString": "string",
              "sid": number
            }
          }
        },
        "codecSettings": {
          "aacSettings": {
            "bitrate": number,
            "codingMode": enum,
            "inputType": enum,
            "profile": enum,
            "rateControlMode": enum,
            "rawFormat": enum,
            "sampleRate": number,
            "spec": enum,
            "vbrQuality": enum
          },
          "ac3Settings": {
            "bitrate": number,
            "bitstreamMode": enum,
            "codingMode": enum,
            "dialnorm": integer,
            "drcProfile": enum,
            "lfeFilter": enum,
            "metadataControl": enum
          },
          "eac3Settings": {
            "attenuationControl": enum,
            "bitrate": number,
            "bitstreamMode": enum,
            "codingMode": enum,
            "dcFilter": enum,
            "dialnorm": integer,
```

```
            "drcLine": enum,
            "drcRf": enum,
            "lfeControl": enum,
            "lfeFilter": enum,
            "loRoCenterMixLevel": number,
            "loRoSurroundMixLevel": number,
            "ltRtCenterMixLevel": number,
            "ltRtSurroundMixLevel": number,
            "metadataControl": enum,
            "passthroughControl": enum,
            "phaseControl": enum,
            "stereoDownmix": enum,
            "surroundExMode": enum,
            "surroundMode": enum
          },
          "mp2Settings": {
            "bitrate": number,
            "codingMode": enum,
            "sampleRate": number
          },
          "passThroughSettings": {
          },
          "wavSettings": {
            "bitDepth": number,
            "codingMode": enum,
            "sampleRate": number
          }
        },
        "languageCode": "string",
        "languageCodeControl": enum,
        "name": "string",
        "remixSettings": {
          "channelMappings": [
            {
              "inputChannelLevels": [
                {
                  "gain": integer,
                  "inputChannel": integer
                }
              ],
              "outputChannel": integer
            }
          ],
          "channelsIn": integer,
```

```
          "channelsOut": integer
        },
        "streamName": "string"
      }
    ],
    "availBlanking": {
      "availBlankingImage": {
        "passwordParam": "string",
        "uri": "string",
        "username": "string"
      },
      "state": enum
    },
    "availConfiguration": {
      "availSettings": {
        "scte35SpliceInsert": {
          "adAvailOffset": integer,
          "noRegionalBlackoutFlag": enum,
          "webDeliveryAllowedFlag": enum
        },
        "scte35TimeSignalApos": {
          "adAvailOffset": integer,
          "noRegionalBlackoutFlag": enum,
          "webDeliveryAllowedFlag": enum
        }
      }
    },
    "blackoutSlate": {
      "blackoutSlateImage": {
        "passwordParam": "string",
        "uri": "string",
        "username": "string"
      },
      "networkEndBlackout": enum,
      "networkEndBlackoutImage": {
        "passwordParam": "string",
        "uri": "string",
        "username": "string"
      },
      "networkId": "string",
      "state": enum
    },
    "captionDescriptions": [
      {
```

```
        "captionSelectorName": "string",
        "destinationSettings": {
          "aribDestinationSettings": {
          },
          "burnInDestinationSettings": {
            "alignment": enum,
            "backgroundColor": enum,
            "backgroundOpacity": integer,
            "font": {
              "passwordParam": "string",
              "uri": "string",
              "username": "string"
            },
            "fontColor": enum,
            "fontOpacity": integer,
            "fontResolution": integer,
            "fontSize": "string",
            "outlineColor": enum,
            "outlineSize": integer,
            "shadowColor": enum,
            "shadowOpacity": integer,
            "shadowXOffset": integer,
            "shadowYOffset": integer,
            "teletextGridControl": enum,
            "xPosition": integer,
            "yPosition": integer
          },
          "dvbSubDestinationSettings": {
            "alignment": enum,
            "backgroundColor": enum,
            "backgroundOpacity": integer,
            "font": {
              "passwordParam": "string",
              "uri": "string",
              "username": "string"
            },
            "fontColor": enum,
            "fontOpacity": integer,
            "fontResolution": integer,
            "fontSize": "string",
            "outlineColor": enum,
            "outlineSize": integer,
            "shadowColor": enum,
            "shadowOpacity": integer,
```

```
            "shadowXOffset": integer,
            "shadowYOffset": integer,
            "teletextGridControl": enum,
            "xPosition": integer,
            "yPosition": integer
          },
          "ebuTtDDestinationSettings": {
            "copyrightHolder": "string",
            "fillLineGap": enum,
            "fontFamily": "string",
            "styleControl": enum
          },
          "embeddedDestinationSettings": {
          },
          "embeddedPlusScte20DestinationSettings": {
          },
          "rtmpCaptionInfoDestinationSettings": {
          },
          "scte20PlusEmbeddedDestinationSettings": {
          },
          "scte27DestinationSettings": {
          },
          "smpteTtDestinationSettings": {
          },
          "teletextDestinationSettings": {
          },
          "ttmlDestinationSettings": {
            "styleControl": enum
          },
          "webvttDestinationSettings": {
            "styleControl": enum
          }
        },
        "languageCode": "string",
        "languageDescription": "string",
        "name": "string"
      }
    ],
    "featureActivations": {
      "inputPrepareScheduleActions": enum
    },
    "globalConfiguration": {
      "initialAudioGain": integer,
      "inputEndAction": enum,
```

```
      "inputLossBehavior": {
        "blackFrameMsec": integer,
        "inputLossImageColor": "string",
        "inputLossImageSlate": {
          "passwordParam": "string",
          "uri": "string",
          "username": "string"
        },
        "inputLossImageType": enum,
        "repeatFrameMsec": integer
      },
      "outputLockingMode": enum,
      "outputTimingSource": enum,
      "supportLowFramerateInputs": enum
    },
    "motionGraphicsConfiguration": {
      "motionGraphicsInsertion": enum,
      "motionGraphicsSettings": {
        "htmlMotionGraphicsSettings": {
        }
      }
    },
    "nielsenConfiguration": {
      "distributorId": "string",
      "nielsenPcmToId3Tagging": enum
    },
    "outputGroups": [
      {
        "name": "string",
        "outputGroupSettings": {
          "archiveGroupSettings": {
            "archiveCdnSettings": {
              "archiveS3Settings": {
                "cannedAcl": enum,
                "logUploads": enum
              }
            },
            "destination": {
              "destinationRefId": "string"
            },
            "rolloverInterval": integer
          },
          "frameCaptureGroupSettings": {
            "destination": {
```

```
            "destinationRefId": "string"
          },
          "frameCaptureCdnSettings": {
            "frameCaptureS3Settings": {
              "cannedAcl": enum,
              "logUploads": enum
            }
          }
        },
        "hlsGroupSettings": {
          "adMarkers": [
            enum
          ],
          "baseUrlContent": "string",
          "baseUrlContent1": "string",
          "baseUrlManifest": "string",
          "baseUrlManifest1": "string",
          "captionLanguageMappings": [
            {
              "captionChannel": integer,
              "languageCode": "string",
              "languageDescription": "string"
            }
          ],
          "captionLanguageSetting": enum,
          "clientCache": enum,
          "codecSpecification": enum,
          "constantIv": "string",
          "destination": {
            "destinationRefId": "string"
          },
          "directoryStructure": enum,
          "discontinuityTags": enum,
          "encryptionType": enum,
          "hlsCdnSettings": {
            "hlsAkamaiSettings": {
              "connectionRetryInterval": integer,
              "filecacheDuration": integer,
              "httpTransferMode": enum,
              "numRetries": integer,
              "restartDelay": integer,
              "salt": "string",
              "token": "string"
            },
```

```
        "hlsBasicPutSettings": {
          "connectionRetryInterval": integer,
          "filecacheDuration": integer,
          "numRetries": integer,
          "restartDelay": integer
        },
        "hlsMediaStoreSettings": {
          "connectionRetryInterval": integer,
          "filecacheDuration": integer,
          "mediaStoreStorageClass": enum,
          "numRetries": integer,
          "restartDelay": integer
        },
        "hlsS3Settings": {
          "cannedAcl": enum,
          "logUploads": enum
        },
        "hlsWebdavSettings": {
          "connectionRetryInterval": integer,
          "filecacheDuration": integer,
          "httpTransferMode": enum,
          "numRetries": integer,
          "restartDelay": integer
        }
      },
      "hlsId3SegmentTagging": enum,
      "iFrameOnlyPlaylists": enum,
      "incompleteSegmentBehavior": enum,
      "indexNSegments": integer,
      "inputLossAction": enum,
      "ivInManifest": enum,
      "ivSource": enum,
      "keepSegments": integer,
      "keyFormat": "string",
      "keyFormatVersions": "string",
      "keyProviderSettings": {
        "staticKeySettings": {
          "keyProviderServer": {
            "passwordParam": "string",
            "uri": "string",
            "username": "string"
          },
          "staticKeyValue": "string"
        }
```

```
      },
      "manifestCompression": enum,
      "manifestDurationFormat": enum,
      "minSegmentLength": integer,
      "mode": enum,
      "outputSelection": enum,
      "programDateTime": enum,
      "programDateTimeClock": enum,
      "programDateTimePeriod": integer,
      "redundantManifest": enum,
      "segmentLength": integer,
      "segmentationMode": enum,
      "segmentsPerSubdirectory": integer,
      "streamInfResolution": enum,
      "timedMetadataId3Frame": enum,
      "timedMetadataId3Period": integer,
      "timestampDeltaMilliseconds": integer,
      "tsFileMode": enum
    },
    "mediaPackageGroupSettings": {
      "destination": {
        "destinationRefId": "string"
      }
    },
    "msSmoothGroupSettings": {
      "acquisitionPointId": "string",
      "audioOnlyTimecodeControl": enum,
      "certificateMode": enum,
      "connectionRetryInterval": integer,
      "destination": {
        "destinationRefId": "string"
      },
      "eventId": "string",
      "eventIdMode": enum,
      "eventStopBehavior": enum,
      "filecacheDuration": integer,
      "fragmentLength": integer,
      "inputLossAction": enum,
      "numRetries": integer,
      "restartDelay": integer,
      "segmentationMode": enum,
      "sendDelayMs": integer,
      "sparseTrackType": enum,
      "streamManifestBehavior": enum,
```

```
            "timestampOffset": "string",
            "timestampOffsetMode": enum
          },
          "multiplexGroupSettings": {
          },
          "rtmpGroupSettings": {
            "adMarkers": [
              enum
            ],
            "authenticationScheme": enum,
            "cacheFullBehavior": enum,
            "cacheLength": integer,
            "captionData": enum,
            "inputLossAction": enum,
            "restartDelay": integer
          },
          "udpGroupSettings": {
            "inputLossAction": enum,
            "timedMetadataId3Frame": enum,
            "timedMetadataId3Period": integer
          }
        },
        "outputs": [
          {
            "audioDescriptionNames": [
              "string"
            ],
            "captionDescriptionNames": [
              "string"
            ],
            "outputName": "string",
            "outputSettings": {
              "archiveOutputSettings": {
                "containerSettings": {
                  "m2tsSettings": {
                    "absentInputAudioBehavior": enum,
                    "arib": enum,
                    "aribCaptionsPid": "string",
                    "aribCaptionsPidControl": enum,
                    "audioBufferModel": enum,
                    "audioFramesPerPes": integer,
                    "audioPids": "string",
                    "audioStreamType": enum,
                    "bitrate": integer,
```

```
                "bufferModel": enum,
                "ccDescriptor": enum,
                "dvbNitSettings": {
                  "networkId": integer,
                  "networkName": "string",
                  "repInterval": integer
                },
                "dvbSdtSettings": {
                  "outputSdt": enum,
                  "repInterval": integer,
                  "serviceName": "string",
                  "serviceProviderName": "string"
                },
                "dvbSubPids": "string",
                "dvbTdtSettings": {
                  "repInterval": integer
                },
                "dvbTeletextPid": "string",
                "ebif": enum,
                "ebpAudioInterval": enum,
                "ebpLookaheadMs": integer,
                "ebpPlacement": enum,
                "ecmPid": "string",
                "esRateInPes": enum,
                "etvPlatformPid": "string",
                "etvSignalPid": "string",
                "fragmentTime": number,
                "klv": enum,
                "klvDataPids": "string",
                "nielsenId3Behavior": enum,
                "nullPacketBitrate": number,
                "patInterval": integer,
                "pcrControl": enum,
                "pcrPeriod": integer,
                "pcrPid": "string",
                "pmtInterval": integer,
                "pmtPid": "string",
                "programNum": integer,
                "rateMode": enum,
                "scte27Pids": "string",
                "scte35Control": enum,
                "scte35Pid": "string",
                "segmentationMarkers": enum,
                "segmentationStyle": enum,
```

```
            "segmentationTime": number,
            "timedMetadataBehavior": enum,
            "timedMetadataPid": "string",
            "transportStreamId": integer,
            "videoPid": "string"
          },
          "rawSettings": {
          }
        },
        "extension": "string",
        "nameModifier": "string"
      },
      "frameCaptureOutputSettings": {
        "nameModifier": "string"
      },
      "hlsOutputSettings": {
        "h265PackagingType": enum,
        "hlsSettings": {
          "audioOnlyHlsSettings": {
            "audioGroupId": "string",
            "audioOnlyImage": {
              "passwordParam": "string",
              "uri": "string",
              "username": "string"
            },
            "audioTrackType": enum,
            "segmentType": enum
          },
          "fmp4HlsSettings": {
            "audioRenditionSets": "string",
            "nielsenId3Behavior": enum,
            "timedMetadataBehavior": enum
          },
          "frameCaptureHlsSettings": {
          },
          "standardHlsSettings": {
            "audioRenditionSets": "string",
            "m3u8Settings": {
              "audioFramesPerPes": integer,
              "audioPids": "string",
              "ecmPid": "string",
              "nielsenId3Behavior": enum,
              "patInterval": integer,
              "pcrControl": enum,
```

```
                "pcrPeriod": integer,
                "pcrPid": "string",
                "pmtInterval": integer,
                "pmtPid": "string",
                "programNum": integer,
                "scte35Behavior": enum,
                "scte35Pid": "string",
                "timedMetadataBehavior": enum,
                "timedMetadataPid": "string",
                "transportStreamId": integer,
                "videoPid": "string"
              }
            }
          },
          "nameModifier": "string",
          "segmentModifier": "string"
        },
        "mediaPackageOutputSettings": {
        },
        "msSmoothOutputSettings": {
          "h265PackagingType": enum,
          "nameModifier": "string"
        },
        "multiplexOutputSettings": {
          "destination": {
            "destinationRefId": "string"
          }
        },
        "rtmpOutputSettings": {
          "certificateMode": enum,
          "connectionRetryInterval": integer,
          "destination": {
            "destinationRefId": "string"
          },
          "numRetries": integer
        },
        "udpOutputSettings": {
          "bufferMsec": integer,
          "containerSettings": {
            "m2tsSettings": {
              "absentInputAudioBehavior": enum,
              "arib": enum,
              "aribCaptionsPid": "string",
              "aribCaptionsPidControl": enum,
```

```
        "audioBufferModel": enum,
        "audioFramesPerPes": integer,
        "audioPids": "string",
        "audioStreamType": enum,
        "bitrate": integer,
        "bufferModel": enum,
        "ccDescriptor": enum,
        "dvbNitSettings": {
          "networkId": integer,
          "networkName": "string",
          "repInterval": integer
        },
        "dvbSdtSettings": {
          "outputSdt": enum,
          "repInterval": integer,
          "serviceName": "string",
          "serviceProviderName": "string"
        },
        "dvbSubPids": "string",
        "dvbTdtSettings": {
          "repInterval": integer
        },
        "dvbTeletextPid": "string",
        "ebif": enum,
        "ebpAudioInterval": enum,
        "ebpLookaheadMs": integer,
        "ebpPlacement": enum,
        "ecmPid": "string",
        "esRateInPes": enum,
        "etvPlatformPid": "string",
        "etvSignalPid": "string",
        "fragmentTime": number,
        "klv": enum,
        "klvDataPids": "string",
        "nielsenId3Behavior": enum,
        "nullPacketBitrate": number,
        "patInterval": integer,
        "pcrControl": enum,
        "pcrPeriod": integer,
        "pcrPid": "string",
        "pmtInterval": integer,
        "pmtPid": "string",
        "programNum": integer,
        "rateMode": enum,
```

```
                    "scte27Pids": "string",
                    "scte35Control": enum,
                    "scte35Pid": "string",
                    "segmentationMarkers": enum,
                    "segmentationStyle": enum,
                    "segmentationTime": number,
                    "timedMetadataBehavior": enum,
                    "timedMetadataPid": "string",
                    "transportStreamId": integer,
                    "videoPid": "string"
                  }
                },
                "destination": {
                  "destinationRefId": "string"
                },
                "fecOutputSettings": {
                  "columnDepth": integer,
                  "includeFec": enum,
                  "rowLength": integer
                }
              }
            },
            "videoDescriptionName": "string"
          }
        ]
      }
    ],
    "timecodeConfig": {
      "source": enum,
      "syncThreshold": integer
    },
    "videoDescriptions": [
      {
        "codecSettings": {
          "frameCaptureSettings": {
            "captureInterval": integer,
            "captureIntervalUnits": enum
          },
          "h264Settings": {
            "adaptiveQuantization": enum,
            "afdSignaling": enum,
            "bitrate": integer,
            "bufFillPct": integer,
            "bufSize": integer,
```

```
            "colorMetadata": enum,
            "colorSpaceSettings": {
              "colorSpacePassthroughSettings": {
              },
              "rec601Settings": {
              },
              "rec709Settings": {
              }
            },
            "entropyEncoding": enum,
            "filterSettings": {
              "temporalFilterSettings": {
                "postFilterSharpening": enum,
                "strength": enum
              }
            },
            "fixedAfd": enum,
            "flickerAq": enum,
            "forceFieldPictures": enum,
            "framerateControl": enum,
            "framerateDenominator": integer,
            "framerateNumerator": integer,
            "gopBReference": enum,
            "gopClosedCadence": integer,
            "gopNumBFrames": integer,
            "gopSize": number,
            "gopSizeUnits": enum,
            "level": enum,
            "lookAheadRateControl": enum,
            "maxBitrate": integer,
            "minIInterval": integer,
            "numRefFrames": integer,
            "parControl": enum,
            "parDenominator": integer,
            "parNumerator": integer,
            "profile": enum,
            "qualityLevel": enum,
            "qvbrQualityLevel": integer,
            "rateControlMode": enum,
            "scanType": enum,
            "sceneChangeDetect": enum,
            "slices": integer,
            "softness": integer,
            "spatialAq": enum,
```

```
            "subgopLength": enum,
            "syntax": enum,
            "temporalAq": enum,
            "timecodeInsertion": enum
          },
          "h265Settings": {
            "adaptiveQuantization": enum,
            "afdSignaling": enum,
            "alternativeTransferFunction": enum,
            "bitrate": integer,
            "bufSize": integer,
            "colorMetadata": enum,
            "colorSpaceSettings": {
              "colorSpacePassthroughSettings": {
              },
              "hdr10Settings": {
                "maxCll": integer,
                "maxFall": integer
              },
              "rec601Settings": {
              },
              "rec709Settings": {
              }
            },
            "filterSettings": {
              "temporalFilterSettings": {
                "postFilterSharpening": enum,
                "strength": enum
              }
            },
            "fixedAfd": enum,
            "flickerAq": enum,
            "framerateDenominator": integer,
            "framerateNumerator": integer,
            "gopClosedCadence": integer,
            "gopSize": number,
            "gopSizeUnits": enum,
            "level": enum,
            "lookAheadRateControl": enum,
            "maxBitrate": integer,
            "minIInterval": integer,
            "parDenominator": integer,
            "parNumerator": integer,
            "profile": enum,
```

```
            "qvbrQualityLevel": integer,
            "rateControlMode": enum,
            "scanType": enum,
            "sceneChangeDetect": enum,
            "slices": integer,
            "tier": enum,
            "timecodeInsertion": enum
          },
          "mpeg2Settings": {
            "adaptiveQuantization": enum,
            "afdSignaling": enum,
            "colorMetadata": enum,
            "colorSpace": enum,
            "displayAspectRatio": enum,
            "filterSettings": {
              "temporalFilterSettings": {
                "postFilterSharpening": enum,
                "strength": enum
              }
            },
            "fixedAfd": enum,
            "framerateDenominator": integer,
            "framerateNumerator": integer,
            "gopClosedCadence": integer,
            "gopNumBFrames": integer,
            "gopSize": number,
            "gopSizeUnits": enum,
            "scanType": enum,
            "subgopLength": enum,
            "timecodeInsertion": enum
          }
        },
        "height": integer,
        "name": "string",
        "respondToAfd": enum,
        "scalingBehavior": enum,
        "sharpness": integer,
        "width": integer
      }
    ]
  },
  "id": "string",
  "inputAttachments": [
    {
```

```
      "automaticInputFailoverSettings": {
        "errorClearTimeMsec": integer,
        "failoverConditions": [
          {
            "failoverConditionSettings": {
              "audioSilenceSettings": {
                "audioSelectorName": "string",
                "audioSilenceThresholdMsec": integer
              },
              "inputLossSettings": {
                "inputLossThresholdMsec": integer
              },
              "videoBlackSettings": {
                "blackDetectThreshold": number,
                "videoBlackThresholdMsec": integer
              }
            }
          }
        ],
        "inputPreference": enum,
        "secondaryInputId": "string"
      },
      "inputAttachmentName": "string",
      "inputId": "string",
      "inputSettings": {
        "audioSelectors": [
          {
            "name": "string",
            "selectorSettings": {
              "audioHlsRenditionSelection": {
                "groupId": "string",
                "name": "string"
              },
              "audioLanguageSelection": {
                "languageCode": "string",
                "languageSelectionPolicy": enum
              },
              "audioPidSelection": {
                "pid": integer
              },
              "audioTrackSelection": {
                "tracks": [
                  {
                    "track": integer
```

```
                }
              ]
            }
          }
        }
      ],
      "captionSelectors": [
        {
          "languageCode": "string",
          "name": "string",
          "selectorSettings": {
            "ancillarySourceSettings": {
              "sourceAncillaryChannelNumber": integer
            },
            "aribSourceSettings": {
            },
            "dvbSubSourceSettings": {
              "ocrLanguage": enum,
              "pid": integer
            },
            "embeddedSourceSettings": {
              "convert608To708": enum,
              "scte20Detection": enum,
              "source608ChannelNumber": integer,
              "source608TrackNumber": integer
            },
            "scte20SourceSettings": {
              "convert608To708": enum,
              "source608ChannelNumber": integer
            },
            "scte27SourceSettings": {
              "ocrLanguage": enum,
              "pid": integer
            },
            "teletextSourceSettings": {
              "outputRectangle": {
                "height": number,
                "leftOffset": number,
                "topOffset": number,
                "width": number
              },
              "pageNumber": "string"
            }
          }
```

```
          }
        ],
        "deblockFilter": enum,
        "denoiseFilter": enum,
        "filterStrength": integer,
        "inputFilter": enum,
        "networkInputSettings": {
          "hlsInputSettings": {
            "bandwidth": integer,
            "bufferSegments": integer,
            "retries": integer,
            "retryInterval": integer,
            "scte35Source": enum
          },
          "serverValidation": enum
        },
        "smpte2038DataPreference": enum,
        "sourceEndBehavior": enum,
        "videoSelector": {
          "colorSpace": enum,
          "colorSpaceSettings": {
            "hdr10Settings": {
              "maxCll": integer,
              "maxFall": integer
            }
          },
          "colorSpaceUsage": enum,
          "selectorSettings": {
            "videoSelectorPid": {
              "pid": integer
            },
            "videoSelectorProgramId": {
              "programId": integer
            }
          }
        }
      }
    }
  ],
  "inputSpecification": {
    "codec": enum,
    "maximumBitrate": enum,
    "resolution": enum
  },
```

```
  "logLevel": enum,
  "maintenance": {
    "maintenanceDay": enum,
    "maintenanceDeadline": "string",
    "maintenanceScheduledDate": "string",
    "maintenanceStartTime": "string"
  },
  "name": "string",
  "pipelineDetails": [
    {
      "activeInputAttachmentName": "string",
      "activeInputSwitchActionName": "string",
      "activeMotionGraphicsActionName": "string",
      "activeMotionGraphicsUri": "string",
      "pipelineId": "string"
    }
  ],
  "pipelinesRunningCount": integer,
  "roleArn": "string",
  "state": enum,
  "tags": {
  },
  "vpc": {
    "availabilityZones": [
      "string"
    ],
    "networkInterfaceIds": [
      "string"
    ],
    "securityGroupIds": [
      "string"
    ],
    "subnetIds": [
      "string"
    ]
  }
}
```

InvalidRequest schema

```
{
  "message": "string"
```

```
}
```

AccessDenied schema

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

ResourceNotFound schema

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

ResourceConflict schema

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

LimitExceeded schema

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

InternalServiceError schema

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

BadGatewayException schema

```
{
  "message": "string"
```

```
}
```

GatewayTimeoutException schema

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

Properties

AacCodingMode

Aac Coding Mode

AD_RECEIVER_MIX
CODING_MODE_1_0
CODING_MODE_1_1
CODING_MODE_2_0
CODING_MODE_5_1

AacInputType

Aac Input Type

BROADCASTER_MIXED_AD
NORMAL

AacProfile

Aac Profile

HEV1
HEV2
LC

AacRateControlMode

Aac Rate Control Mode

CBR

VBR

AacRawFormat

Aac Raw Format

LATM_LOAS

NONE

AacSettings

Aac Settings

bitrate

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

Type: number

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: AacCodingMode

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: AacInputType
Required: False

profile

AAC Profile.

Type: AacProfile
Required: False

rateControlMode

Rate Control Mode.

Type: AacRateControlMode
Required: False

rawFormat

Sets LATM / LOAS AAC output for raw containers.

Type: AacRawFormat
Required: False

sampleRate

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

Type: number
Required: False

spec

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

Type: AacSpec
Required: False

vbrQuality

VBR Quality Level - Only used if rateControlMode is VBR.

Type: AacVbrQuality
Required: False

AacSpec

Aac Spec

MPEG2
MPEG4

AacVbrQuality

Aac Vbr Quality

HIGH
LOW
MEDIUM_HIGH
MEDIUM_LOW

Ac3BitstreamMode

Ac3 Bitstream Mode

COMMENTARY
COMPLETE_MAIN
DIALOGUE
EMERGENCY
HEARING_IMPAIRED
MUSIC_AND_EFFECTS

VISUALLY_IMPAIRED

VOICE_OVER

Ac3CodingMode

Ac3 Coding Mode

CODING_MODE_1_0

CODING_MODE_1_1

CODING_MODE_2_0

CODING_MODE_3_2_LFE

Ac3DrcProfile

Ac3 Drc Profile

FILM_STANDARD

NONE

Ac3LfeFilter

Ac3 Lfe Filter

DISABLED

ENABLED

Ac3MetadataControl

Ac3 Metadata Control

FOLLOW_INPUT

USE_CONFIGURED

Ac3Settings

Ac3 Settings

bitrate

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

Type: number
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: Ac3BitstreamMode
Required: False

codingMode

Dolby Digital coding mode. Determines number of channels.

Type: Ac3CodingMode
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

drcProfile

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

Type: Ac3DrcProfile

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: Ac3LfeFilter
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: Ac3MetadataControl
Required: False

AccessDenied

message

Type: string
Required: False

AfdSignaling

Afd Signaling

AUTO
FIXED
NONE

AncillarySourceSettings

Ancillary Source Settings

sourceAncillaryChannelNumber

Specifies the number (1 to 4) of the captions channel you want to extract from the ancillary captions. If you plan to convert the ancillary captions to another format, complete this field. If you plan to choose Embedded as the captions destination in the output (to pass through all the channels in the ancillary captions), leave this field blank because MediaLive ignores the field.

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

ArchiveCdnSettings

Archive Cdn Settings

archiveS3Settings

Type: ArchiveS3Settings
Required: False

ArchiveContainerSettings

Archive Container Settings

m2tsSettings

Type: M2tsSettings
Required: False

rawSettings

Type: RawSettings
Required: False

ArchiveGroupSettings

Archive Group Settings

archiveCdnSettings

Parameters that control interactions with the CDN.

Type: ArchiveCdnSettings
Required: False

destination

A directory and base filename where archive files should be written.

Type: OutputLocationRef
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

Archive Output Settings

containerSettings

Settings specific to the container type of the file.

Type: ArchiveContainerSettings
Required: True

extension

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

Type: string
Required: False

nameModifier

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

Type: string
Required: False

ArchiveS3LogUploads

Archive S3 Log Uploads

```
DISABLED
ENABLED
```

ArchiveS3Settings

Archive S3 Settings

cannedAcl

Specify the canned ACL to apply to each S3 request. Defaults to none.

Type: S3CannedAcl
Required: False

logUploads

When set to enabled, each upload to CDN or server will be logged.

Type: ArchiveS3LogUploads
Required: False

AribDestinationSettings

Arib Destination Settings

AribSourceSettings

Arib Source Settings

AudioChannelMapping

Audio Channel Mapping

inputChannelLevels

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

Type: Array of type InputChannelLevel
Required: True

outputChannel

The index of the output channel being produced.

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

AudioCodecSettings

Audio Codec Settings

aacSettings

Type: AacSettings
Required: False

ac3Settings

Type: Ac3Settings
Required: False

eac3Settings

Type: Eac3Settings
Required: False

mp2Settings

Type: Mp2Settings
Required: False

passThroughSettings

Type: PassThroughSettings
Required: False

wavSettings

Type: WavSettings
Required: False

AudioDescription

Audio Description

audioNormalizationSettings

Advanced audio normalization settings.

Type: AudioNormalizationSettings
Required: False

audioSelectorName

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

Type: string
Required: True

audioType

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

Type: AudioType

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: AudioDescriptionAudioTypeControl
Required: False

audioWatermarkingSettings

Settings to configure one or more solutions that insert audio watermarks in the audio encode

Type: AudioWatermarkSettings
Required: False

codecSettings

Audio codec settings.

Type: AudioCodecSettings
Required: False

languageCode

RFC 5646 language code representing 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
MinLength: 1
MaxLength: 35

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: AudioDescriptionLanguageCodeControl
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

remixSettings

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

Type: RemixSettings
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

AudioDescriptionAudioTypeControl

Audio Description Audio Type Control

```
FOLLOW_INPUT
USE_CONFIGURED
```

AudioDescriptionLanguageCodeControl

Audio Description Language Code Control

FOLLOW_INPUT
USE_CONFIGURED

AudioHlsRenditionSelection

Audio Hls Rendition Selection

groupId

Specifies the GROUP-ID in the #EXT-X-MEDIA tag of the target HLS audio rendition.

Type: string
Required: True
MinLength: 1

name

Specifies the NAME in the #EXT-X-MEDIA tag of the target HLS audio rendition.

Type: string
Required: True
MinLength: 1

AudioLanguageSelection

Audio Language Selection

languageCode

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

Type: string
Required: True

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: AudioLanguageSelectionPolicy
Required: False

AudioLanguageSelectionPolicy

Audio Language Selection Policy

```
LOOSE
STRICT
```

AudioNormalizationAlgorithm

Audio Normalization Algorithm

```
ITU_1770_1
ITU_1770_2
```

AudioNormalizationAlgorithmControl

Audio Normalization Algorithm Control

```
CORRECT_AUDIO
```

AudioNormalizationSettings

Audio Normalization Settings

algorithm

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

Type: AudioNormalizationAlgorithm
Required: False

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: AudioNormalizationAlgorithmControl
Required: False

targetLkfs

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

Type: number
Required: False
Minimum: -59
Maximum: 0

AudioOnlyHlsSegmentType

Audio Only Hls Segment Type

AAC
FMP4

AudioOnlyHlsSettings

Audio Only Hls Settings

audioGroupId

Specifies the group to which the audio Rendition belongs.

Type: string
Required: False

audioOnlyImage

Optional. Specifies the .jpg or .png image to use as the cover art for an audio-only output. We recommend a low bit-size file because the image increases the output audio bandwidth. The image is attached to the audio as an ID3 tag, frame type APIC, picture type 0x10, as per the "ID3 tag version 2.4.0 - Native Frames" standard.

Type: InputLocation
Required: False

audioTrackType

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

Type: AudioOnlyHlsTrackType
Required: False

segmentType

Specifies the segment type.

Type: AudioOnlyHlsSegmentType
Required: False

AudioOnlyHlsTrackType

Audio Only Hls Track Type

```
ALTERNATE_AUDIO_AUTO_SELECT
ALTERNATE_AUDIO_AUTO_SELECT_DEFAULT
```

```
ALTERNATE_AUDIO_NOT_AUTO_SELECT
AUDIO_ONLY_VARIANT_STREAM
```

AudioPidSelection

Audio Pid Selection

pid

Selects a specific PID from within a source.

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

AudioSelector

Audio Selector

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
MinLength: 1

selectorSettings

The audio selector settings.

Type: AudioSelectorSettings
Required: False

AudioSelectorSettings

Audio Selector Settings

audioHlsRenditionSelection

Type: AudioHlsRenditionSelection
Required: False

audioLanguageSelection

Type: AudioLanguageSelection
Required: False

audioPidSelection

Type: AudioPidSelection
Required: False

audioTrackSelection

Type: AudioTrackSelection
Required: False

AudioSilenceFailoverSettings

audioSelectorName

The name of the audio selector in the input that MediaLive should monitor to detect silence. Select your most important rendition. If you didn't create an audio selector in this input, leave blank.

Type: string
Required: True

audioSilenceThresholdMsec

The amount of time (in milliseconds) that the active input must be silent before automatic input failover occurs. Silence is defined as audio loss or audio quieter than -50 dBFS.

Type: integer
Required: False
Minimum: 1000

AudioTrack

Audio Track

track

1-based integer value that maps to a specific audio track

Type: integer
Required: True
Minimum: 1

AudioTrackSelection

Audio Track Selection

tracks

Selects one or more unique audio tracks from within a source.

Type: Array of type AudioTrack
Required: True

AudioType

Audio Type

```
CLEAN_EFFECTS
HEARING_IMPAIRED
UNDEFINED
VISUAL_IMPAIRED_COMMENTARY
```

AudioWatermarkSettings

Audio Watermark Settings

nielsenWatermarksSettings

Settings to configure Nielsen Watermarks in the audio encode

Type: NielsenWatermarksSettings
Required: False

AuthenticationScheme

Authentication Scheme

AKAMAI
COMMON

AutomaticInputFailoverSettings

The settings for Automatic Input Failover.

errorClearTimeMsec

This clear time defines the requirement a recovered input must meet to be considered healthy. The input must have no failover conditions for this length of time. Enter a time in milliseconds. This value is particularly important if the input_preference for the failover pair is set to PRIMARY_INPUT_PREFERRED, because after this time, MediaLive will switch back to the primary input.

Type: integer
Required: False
Minimum: 1

failoverConditions

A list of failover conditions. If any of these conditions occur, MediaLive will perform a failover to the other input.

Type: Array of type FailoverCondition
Required: False

inputPreference

Input preference when deciding which input to make active when a previously failed input has recovered.

Type: InputPreference
Required: False

secondaryInputId

The input ID of the secondary input in the automatic input failover pair.

Type: string
Required: True

AvailBlanking

Avail Blanking

availBlankingImage

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

Type: InputLocation
Required: False

state

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

Type: AvailBlankingState
Required: False

AvailBlankingState

Avail Blanking State

```
DISABLED
ENABLED
```

AvailConfiguration

Avail Configuration

availSettings

Ad avail settings.

Type: AvailSettings
Required: False

AvailSettings

Avail Settings

scte35SpliceInsert

Type: Scte35SpliceInsert
Required: False

scte35TimeSignalApos

Type: Scte35TimeSignalApos
Required: False

BadGatewayException

message

Type: string
Required: False

BlackoutSlate

Blackout Slate

blackoutSlateImage

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

Type: InputLocation

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: BlackoutSlateNetworkEndBlackout
Required: False

networkEndBlackoutImage

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

Type: InputLocation
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
MinLength: 34
MaxLength: 34

state

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

Type: BlackoutSlateState
Required: False

BlackoutSlateNetworkEndBlackout

Blackout Slate Network End Blackout

DISABLED
ENABLED

BlackoutSlateState

Blackout Slate State

DISABLED
ENABLED

BurnInAlignment

Burn In Alignment

CENTERED
LEFT
SMART

BurnInBackgroundColor

Burn In Background Color

BLACK
NONE
WHITE

BurnInDestinationSettings

Burn In Destination Settings

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: BurnInAlignment
Required: False

backgroundColor

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

Type: BurnInBackgroundColor
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

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
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: BurnInFontColor
Required: False

fontOpacity

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

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

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

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

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: BurnInOutlineColor
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

shadowColor

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

Type: BurnInShadowColor
Required: False

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

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

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

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: BurnInTeletextGridControl
Required: False

xPosition

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

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

BurnInFontColor

Burn In Font Color

BLACK
BLUE
GREEN
RED
WHITE
YELLOW

BurnInOutlineColor

Burn In Outline Color

BLACK
BLUE
GREEN
RED
WHITE
YELLOW

BurnInShadowColor

Burn In Shadow Color

BLACK

NONE
WHITE

BurnInTeletextGridControl

Burn In Teletext Grid Control

FIXED
SCALED

CaptionDescription

Caption Description

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

destinationSettings

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

Type: CaptionDestinationSettings
Required: False

languageCode

ISO 639-2 three-digit code: http://www.loc.gov/standards/iso639-2/

Type: string
Required: False

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

CaptionDestinationSettings

Caption Destination Settings

aribDestinationSettings

Type: AribDestinationSettings
Required: False

burnInDestinationSettings

Type: BurnInDestinationSettings
Required: False

dvbSubDestinationSettings

Type: DvbSubDestinationSettings
Required: False

ebuTtDDestinationSettings

Type: EbuTtDDestinationSettings
Required: False

embeddedDestinationSettings

Type: EmbeddedDestinationSettings

Required: False

embeddedPlusScte20DestinationSettings

Type: EmbeddedPlusScte20DestinationSettings
Required: False

rtmpCaptionInfoDestinationSettings

Type: RtmpCaptionInfoDestinationSettings
Required: False

scte20PlusEmbeddedDestinationSettings

Type: Scte20PlusEmbeddedDestinationSettings
Required: False

scte27DestinationSettings

Type: Scte27DestinationSettings
Required: False

smpteTtDestinationSettings

Type: SmpteTtDestinationSettings
Required: False

teletextDestinationSettings

Type: TeletextDestinationSettings
Required: False

ttmlDestinationSettings

Type: TtmlDestinationSettings
Required: False

webvttDestinationSettings

Type: WebvttDestinationSettings
Required: False

CaptionLanguageMapping

Maps a caption channel to an ISO 693-2 language code (http://www.loc.gov/standards/iso639-2), with an optional description.

captionChannel

The closed caption channel being described by this CaptionLanguageMapping. 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
MinLength: 3
MaxLength: 3

languageDescription

Textual description of language

Type: string
Required: True
MinLength: 1

CaptionRectangle

Caption Rectangle

height

See the description in leftOffset. For height, specify the entire height of the rectangle as a percentage of the underlying frame height. For example, \"80\" means the rectangle height is 80% of the underlying frame height. The topOffset and rectangleHeight must add up to 100% or less. This field corresponds to tts:extent - Y in the TTML standard.

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

leftOffset

Applies only if you plan to convert these source captions to EBU-TT-D or TTML in an output. (Make sure to leave the default if you don't have either of these formats in the output.) You can define a display rectangle for the captions that is smaller than the underlying video frame. You define the rectangle by specifying the position of the left edge, top edge, bottom edge, and right edge of the rectangle, all within the underlying video frame. The units for the measurements are percentages. If you specify a value for one of these fields, you must specify a value for all of them. For leftOffset, specify the position of the left edge of the rectangle, as a percentage of the underlying frame width, and relative to the left edge of the frame. For example, \"10\" means the measurement is 10% of the underlying frame width. The rectangle left edge starts at that position from the left edge of the frame. This field corresponds to tts:origin - X in the TTML standard.

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

topOffset

See the description in leftOffset. For topOffset, specify the position of the top edge of the rectangle, as a percentage of the underlying frame height, and relative to the top edge of the frame. For example, \"10\" means the measurement is 10% of the underlying frame height. The

rectangle top edge starts at that position from the top edge of the frame. This field corresponds to tts:origin - Y in the TTML standard.

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

width

See the description in leftOffset. For width, specify the entire width of the rectangle as a percentage of the underlying frame width. For example, \"80\" means the rectangle width is 80% of the underlying frame width. The leftOffset and rectangleWidth must add up to 100% or less. This field corresponds to tts:extent - X in the TTML standard.

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

CaptionSelector

Output groups for this Live Event. Output groups contain information about where streams should be distributed.

languageCode

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

Type: string
Required: False

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
MinLength: 1

selectorSettings

Caption selector settings.

Type: CaptionSelectorSettings
Required: False

CaptionSelectorSettings

Caption Selector Settings

ancillarySourceSettings

Type: AncillarySourceSettings
Required: False

aribSourceSettings

Type: AribSourceSettings
Required: False

dvbSubSourceSettings

Type: DvbSubSourceSettings
Required: False

embeddedSourceSettings

Type: EmbeddedSourceSettings
Required: False

scte20SourceSettings

Type: Scte20SourceSettings
Required: False

scte27SourceSettings

Type: Scte27SourceSettings
Required: False

teletextSourceSettings

Type: TeletextSourceSettings
Required: False

CdiInputResolution

Maximum CDI input resolution; SD is 480i and 576i up to 30 frames-per-second (fps), HD is 720p up to 60 fps / 1080i up to 30 fps, FHD is 1080p up to 60 fps, UHD is 2160p up to 60 fps

SD

HD

FHD

UHD

CdiInputSpecification

resolution

Maximum CDI input resolution

Type: CdiInputResolution
Required: False

Channel

arn

The unique arn of the channel.

Type: string
Required: False

cdiInputSpecification

Specification of CDI inputs for this channel

Type: CdiInputSpecification
Required: False

channelClass

The class for this channel. STANDARD for a channel with two pipelines or SINGLE_PIPELINE for a channel with one pipeline.

Type: ChannelClass
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
Required: False

egressEndpoints

The endpoints where outgoing connections initiate from

Type: Array of type ChannelEgressEndpoint
Required: False

encoderSettings

Type: EncoderSettings
Required: False

id

The unique ID of the channel.

Type: string
Required: False

inputAttachments

List of input attachments for channel.

Type: Array of type InputAttachment
Required: False

inputSpecification

Specification of network and file inputs for this channel

Type: InputSpecification
Required: False

logLevel

The log level being written to CloudWatch Logs.

Type: LogLevel
Required: False

maintenance

Maintenance settings for this channel.

Type: MaintenanceStatus
Required: False

name

The name of the channel. (user-mutable)

Type: string
Required: False

pipelineDetails

Runtime details for the pipelines of a running channel.

Type: Array of type PipelineDetail
Required: False

pipelinesRunningCount

The number of currently healthy pipelines.

Type: integer
Required: False

roleArn

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

Type: string
Required: False

state

Type: ChannelState
Required: False

tags

A collection of key-value pairs.

Type: Tags
Required: False

vpc

Settings for VPC output

Type: VpcOutputSettingsDescription
Required: False

ChannelClass

A standard channel has two encoding pipelines and a single pipeline channel only has one.

```
STANDARD
SINGLE_PIPELINE
```

ChannelEgressEndpoint

sourceIp

Public IP of where a channel's output comes from

Type: string
Required: False

ChannelState

```
CREATING
CREATE_FAILED
IDLE
STARTING
RUNNING
RECOVERING
STOPPING
DELETING
DELETED
UPDATING
UPDATE_FAILED
```

ColorSpacePassthroughSettings

Passthrough applies no color space conversion to the output

DvbNitSettings

DVB Network Information Table (NIT)

networkId

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

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

networkName

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

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

repInterval

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

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

DvbSdtOutputSdt

Dvb Sdt Output Sdt

SDT_FOLLOW
SDT_FOLLOW_IF_PRESENT
SDT_MANUAL
SDT_NONE

DvbSdtSettings

DVB Service Description Table (SDT)

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: DvbSdtOutputSdt
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

serviceName

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

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

serviceProviderName

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

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

DvbSubDestinationAlignment

Dvb Sub Destination Alignment

CENTERED
LEFT
SMART

DvbSubDestinationBackgroundColor

Dvb Sub Destination Background Color

BLACK
NONE
WHITE

DvbSubDestinationFontColor

Dvb Sub Destination Font Color

BLACK
BLUE
GREEN
RED
WHITE
YELLOW

DvbSubDestinationOutlineColor

Dvb Sub Destination Outline Color

BLACK

BLUE
GREEN
RED
WHITE
YELLOW

DvbSubDestinationSettings

Dvb Sub Destination Settings

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: DvbSubDestinationAlignment
Required: False

backgroundColor

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

Type: DvbSubDestinationBackgroundColor
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

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
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: DvbSubDestinationFontColor
Required: False

fontOpacity

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

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

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

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

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: DvbSubDestinationOutlineColor
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

shadowColor

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

Type: DvbSubDestinationShadowColor
Required: False

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

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

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

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: DvbSubDestinationTeletextGridControl
Required: False

xPosition

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

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

DvbSubDestinationShadowColor

Dvb Sub Destination Shadow Color

```
BLACK
NONE
WHITE
```

DvbSubDestinationTeletextGridControl

Dvb Sub Destination Teletext Grid Control

```
FIXED
```

SCALED

DvbSubOcrLanguage

Dvb Sub Ocr Language

DEU
ENG
FRA
NLD
POR
SPA

DvbSubSourceSettings

Dvb Sub Source Settings

ocrLanguage

If you will configure a WebVTT caption description that references this caption selector, use this field to provide the language to consider when translating the image-based source to text.

Type: DvbSubOcrLanguage
Required: False

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

DVB Time and Date Table (SDT)

repInterval

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

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

Eac3AttenuationControl

Eac3 Attenuation Control

ATTENUATE_3_DB
NONE

Eac3BitstreamMode

Eac3 Bitstream Mode

COMMENTARY
COMPLETE_MAIN
EMERGENCY
HEARING_IMPAIRED
VISUALLY_IMPAIRED

Eac3CodingMode

Eac3 Coding Mode

CODING_MODE_1_0
CODING_MODE_2_0
CODING_MODE_3_2

Eac3DcFilter

Eac3 Dc Filter

DISABLED
ENABLED

Eac3DrcLine

Eac3 Drc Line

FILM_LIGHT
FILM_STANDARD
MUSIC_LIGHT
MUSIC_STANDARD
NONE
SPEECH

Eac3DrcRf

Eac3 Drc Rf

FILM_LIGHT
FILM_STANDARD
MUSIC_LIGHT
MUSIC_STANDARD
NONE
SPEECH

Eac3LfeControl

Eac3 Lfe Control

LFE
NO_LFE

Eac3LfeFilter

Eac3 Lfe Filter

DISABLED

ENABLED

Eac3MetadataControl

Eac3 Metadata Control

FOLLOW_INPUT
USE_CONFIGURED

Eac3PassthroughControl

Eac3 Passthrough Control

NO_PASSTHROUGH
WHEN_POSSIBLE

Eac3PhaseControl

Eac3 Phase Control

NO_SHIFT
SHIFT_90_DEGREES

Eac3Settings

Eac3 Settings

attenuationControl

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

Type: Eac3AttenuationControl
Required: False

bitrate

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

Type: number
Required: False

bitstreamMode

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

Type: Eac3BitstreamMode
Required: False

codingMode

Dolby Digital Plus coding mode. Determines number of channels.

Type: Eac3CodingMode
Required: False

dcFilter

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

Type: Eac3DcFilter
Required: False

dialnorm

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

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

drcLine

Sets the Dolby dynamic range compression profile.

Type: Eac3DrcLine
Required: False

drcRf

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

Type: Eac3DrcRf
Required: False

lfeControl

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

Type: Eac3LfeControl
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: Eac3LfeFilter
Required: False

loRoCenterMixLevel

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

Type: number
Required: False

loRoSurroundMixLevel

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

Type: number
Required: False

ltRtCenterMixLevel

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

Type: number
Required: False

ltRtSurroundMixLevel

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

Type: number
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: Eac3MetadataControl
Required: False

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: Eac3PassthroughControl
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: Eac3PhaseControl

Required: False

stereoDownmix

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

Type: Eac3StereoDownmix
Required: False

surroundExMode

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

Type: Eac3SurroundExMode
Required: False

surroundMode

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

Type: Eac3SurroundMode
Required: False

Eac3StereoDownmix

Eac3 Stereo Downmix

```
DPL2
LO_RO
LT_RT
NOT_INDICATED
```

Eac3SurroundExMode

Eac3 Surround Ex Mode

```
DISABLED
```

ENABLED
NOT_INDICATED

Eac3SurroundMode

Eac3 Surround Mode

DISABLED
ENABLED
NOT_INDICATED

EbuTtDDestinationSettings

Ebu Tt DDestination Settings

copyrightHolder

Applies only if you plan to convert these source captions to EBU-TT-D or TTML in an output. Complete this field if you want to include the name of the copyright holder in the copyright metadata tag in the TTML

Type: string
Required: False
MaxLength: 1000

fillLineGap

Specifies how to handle the gap between the lines (in multi-line captions). - enabled: Fill with the captions background color (as specified in the input captions). - disabled: Leave the gap unfilled.

Type: EbuTtDFillLineGapControl
Required: False

fontFamily

Specifies the font family to include in the font data attached to the EBU-TT captions. Valid only if styleControl is set to include. If you leave this field empty, the font family is set to "monospaced". (If styleControl is set to exclude, the font family is always set to "monospaced".) You specify only

the font family. All other style information (color, bold, position and so on) is copied from the input captions. The size is always set to 100% to allow the downstream player to choose the size. - Enter a list of font families, as a comma-separated list of font names, in order of preference. The name can be a font family (such as “Arial”), or a generic font family (such as “serif”), or “default” (to let the downstream player choose the font). - Leave blank to set the family to “monospace”.

Type: string
Required: False

styleControl

Specifies the style information (font color, font position, and so on) to include in the font data that is attached to the EBU-TT captions. - include: Take the style information (font color, font position, and so on) from the source captions and include that information in the font data attached to the EBU-TT captions. This option is valid only if the source captions are Embedded or Teletext. - exclude: In the font data attached to the EBU-TT captions, set the font family to "monospaced". Do not include any other style information.

Type: EbuTtDDestinationStyleControl
Required: False

EbuTtDDestinationStyleControl

Ebu Tt DDestination Style Control

```
EXCLUDE
INCLUDE
```

EbuTtDFillLineGapControl

Ebu Tt DFill Line Gap Control

```
DISABLED
ENABLED
```

EmbeddedConvert608To708

Embedded Convert608 To708

DISABLED

UPCONVERT

EmbeddedDestinationSettings

Embedded Destination Settings

EmbeddedPlusScte20DestinationSettings

Embedded Plus Scte20 Destination Settings

EmbeddedScte20Detection

Embedded Scte20 Detection

AUTO

OFF

EmbeddedSourceSettings

Embedded Source Settings

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: EmbeddedConvert608To708

Required: False

scte20Detection

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

Type: EmbeddedScte20Detection

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

source608TrackNumber

This field is unused and deprecated.

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

EncoderSettings

Encoder Settings

audioDescriptions

Type: Array of type AudioDescription
Required: True

availBlanking

Settings for ad avail blanking.

Type: AvailBlanking
Required: False

availConfiguration

Event-wide configuration settings for ad avail insertion.

Type: AvailConfiguration
Required: False

blackoutSlate

Settings for blackout slate.

Type: BlackoutSlate
Required: False

captionDescriptions

Settings for caption decriptions

Type: Array of type CaptionDescription
Required: False

featureActivations

Feature Activations

Type: FeatureActivations
Required: False

globalConfiguration

Configuration settings that apply to the event as a whole.

Type: GlobalConfiguration
Required: False

motionGraphicsConfiguration

Settings for motion graphics.

Type: MotionGraphicsConfiguration
Required: False

nielsenConfiguration

Nielsen configuration settings.

Type: NielsenConfiguration
Required: False

outputGroups

Type: Array of type OutputGroup
Required: True

timecodeConfig

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

Type: TimecodeConfig
Required: True

videoDescriptions

Type: Array of type VideoDescription
Required: True

FailoverCondition

Failover Condition settings. There can be multiple failover conditions inside AutomaticInputFailoverSettings.

failoverConditionSettings

Failover condition type-specific settings.

Type: FailoverConditionSettings
Required: False

FailoverConditionSettings

Settings for one failover condition.

audioSilenceSettings

MediaLive will perform a failover if the specified audio selector is silent for the specified period.

Type: AudioSilenceFailoverSettings
Required: False

inputLossSettings

MediaLive will perform a failover if content is not detected in this input for the specified period.

Type: InputLossFailoverSettings
Required: False

videoBlackSettings

MediaLive will perform a failover if content is considered black for the specified period.

Type: VideoBlackFailoverSettings
Required: False

FeatureActivations

Feature Activations

inputPrepareScheduleActions

Enables the Input Prepare feature. You can create Input Prepare actions in the schedule only if this feature is enabled. If you disable the feature on an existing schedule, make sure that you first delete all input prepare actions from the schedule.

Type: FeatureActivationsInputPrepareScheduleActions
Required: False

FeatureActivationsInputPrepareScheduleActions

Feature Activations Input Prepare Schedule Actions

```
DISABLED
ENABLED
```

FecOutputIncludeFec

Fec Output Include Fec

COLUMN
COLUMN_AND_ROW

FecOutputSettings

Fec Output Settings

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: FecOutputIncludeFec
Required: False

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

FixedAfd

Fixed Afd

AFD_0000
AFD_0010
AFD_0011
AFD_0100
AFD_1000
AFD_1001
AFD_1010
AFD_1011
AFD_1101
AFD_1110
AFD_1111

Fmp4HlsSettings

Fmp4 Hls Settings

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

nielsenId3Behavior

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

Type: Fmp4NielsenId3Behavior
Required: False

timedMetadataBehavior

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

Type: Fmp4TimedMetadataBehavior
Required: False

Fmp4NielsenId3Behavior

Fmp4 Nielsen Id3 Behavior

NO_PASSTHROUGH
PASSTHROUGH

Fmp4TimedMetadataBehavior

Fmp4 Timed Metadata Behavior

NO_PASSTHROUGH
PASSTHROUGH

FrameCaptureCdnSettings

Frame Capture Cdn Settings

frameCaptureS3Settings

Type: FrameCaptureS3Settings
Required: False

FrameCaptureGroupSettings

Frame Capture Group Settings

destination

The destination for the frame capture files. Either the URI for an Amazon S3 bucket and object, plus a file name prefix (for example, s3ssl://sportsDelivery/highlights/20180820/curling-) or the URI for a MediaStore container, plus a file name prefix (for example, mediastoressl://

sportsDelivery/20180820/curling-). The final file names consist of the prefix from the destination field (for example, "curling-") + name modifier + the counter (5 digits, starting from 00001) + extension (which is always .jpg). For example, curling-low.00001.jpg

Type: OutputLocationRef
Required: True

frameCaptureCdnSettings

Parameters that control interactions with the CDN.

Type: FrameCaptureCdnSettings
Required: False

FrameCaptureHlsSettings

Frame Capture Hls Settings

FrameCaptureIntervalUnit

Frame Capture Interval Unit

```
MILLISECONDS
SECONDS
```

FrameCaptureOutputSettings

Frame Capture Output Settings

nameModifier

Required if the output group contains more than one output. This modifier forms part of the output file name.

Type: string
Required: False

FrameCaptureS3LogUploads

Frame Capture S3 Log Uploads

DISABLED
ENABLED

FrameCaptureS3Settings

Frame Capture S3 Settings

cannedAcl

Specify the canned ACL to apply to each S3 request. Defaults to none.

Type: S3CannedAcl
Required: False

logUploads

When set to enabled, each upload to CDN or server will be logged.

Type: FrameCaptureS3LogUploads
Required: False

FrameCaptureSettings

Frame Capture Settings

captureInterval

The frequency at which to capture frames for inclusion in the output. May be specified in either seconds or milliseconds, as specified by captureIntervalUnits.

Type: integer
Required: False
Minimum: 1
Maximum: 3600000

captureIntervalUnits

Unit for the frame capture interval.

Type: FrameCaptureIntervalUnit

Required: False

GatewayTimeoutException

message

Type: string
Required: False

GlobalConfiguration

Global Configuration

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 the current input completes (e.g. end-of-file). When switchAndLoopInputs is configured the encoder will restart at the beginning of the first input. When "none" is configured the encoder will transcode either black, a solid color, or a user specified slate images per the "Input Loss Behavior" configuration until the next input switch occurs (which is controlled through the Channel Schedule API).

Type: GlobalConfigurationInputEndAction
Required: False

inputLossBehavior

Settings for system actions when input is lost.

Type: InputLossBehavior
Required: False

outputLockingMode

Indicates how MediaLive pipelines are synchronized. PIPELINE_LOCKING - MediaLive will attempt to synchronize the output of each pipeline to the other. EPOCH_LOCKING - MediaLive will attempt to synchronize the output of each pipeline to the Unix epoch.

Type: GlobalConfigurationOutputLockingMode
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: GlobalConfigurationOutputTimingSource
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: GlobalConfigurationLowFramerateInputs
Required: False

GlobalConfigurationInputEndAction

Global Configuration Input End Action

```
NONE
SWITCH_AND_LOOP_INPUTS
```

GlobalConfigurationLowFramerateInputs

Global Configuration Low Framerate Inputs

```
DISABLED
ENABLED
```

GlobalConfigurationOutputLockingMode

Global Configuration Output Locking Mode

EPOCH_LOCKING
PIPELINE_LOCKING

GlobalConfigurationOutputTimingSource

Global Configuration Output Timing Source

INPUT_CLOCK
SYSTEM_CLOCK

H264AdaptiveQuantization

H264 Adaptive Quantization

AUTO
HIGH
HIGHER
LOW
MAX
MEDIUM
OFF

H264ColorMetadata

H264 Color Metadata

IGNORE
INSERT

H264ColorSpaceSettings

H264 Color Space Settings

colorSpacePassthroughSettings

Type: ColorSpacePassthroughSettings
Required: False

rec601Settings

Type: Rec601Settings
Required: False

rec709Settings

Type: Rec709Settings
Required: False

H264EntropyEncoding

H264 Entropy Encoding

CABAC
CAVLC

H264FilterSettings

H264 Filter Settings

temporalFilterSettings

Type: TemporalFilterSettings
Required: False

H264FlickerAq

H264 Flicker Aq

DISABLED
ENABLED

H264ForceFieldPictures

H264 Force Field Pictures

DISABLED
ENABLED

H264FramerateControl

H264 Framerate Control

INITIALIZE_FROM_SOURCE
SPECIFIED

H264GopBReference

H264 Gop BReference

DISABLED
ENABLED

H264GopSizeUnits

H264 Gop Size Units

FRAMES
SECONDS

H264Level

H264 Level

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

H264 Look Ahead Rate Control

HIGH
LOW
MEDIUM

H264ParControl

H264 Par Control

INITIALIZE_FROM_SOURCE
SPECIFIED

H264Profile

H264 Profile

BASELINE
HIGH
HIGH_10BIT
HIGH_422

HIGH_422_10BIT

MAIN

H264QualityLevel

H264 Quality Level

ENHANCED_QUALITY

STANDARD_QUALITY

H264RateControlMode

H264 Rate Control Mode

CBR

MULTIPLEX

QVBR

VBR

H264ScanType

H264 Scan Type

INTERLACED

PROGRESSIVE

H264SceneChangeDetect

H264 Scene Change Detect

DISABLED

ENABLED

H264Settings

H264 Settings

adaptiveQuantization

Enables or disables adaptive quantization, which is a technique MediaLive can apply to video on a frame-by-frame basis to produce more compression without losing quality. There are three types of adaptive quantization: flicker, spatial, and temporal. Set the field in one of these ways: Set to Auto. Recommended. For each type of AQ, MediaLive will determine if AQ is needed, and if so, the appropriate strength. Set a strength (a value other than Auto or Disable). This strength will apply to any of the AQ fields that you choose to enable. Set to Disabled to disable all types of adaptive quantization.

Type: H264AdaptiveQuantization
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: AfdSignaling
Required: False

bitrate

Average bitrate in bits/second. Required when the rate control mode is VBR or CBR. Not used for QVBR. In an MS Smooth output group, each output must have a unique value when its bitrate is 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

bufSize

Size of buffer (HRD buffer model) in bits.

Type: integer
Required: False
Minimum: 0

colorMetadata

Includes colorspace metadata in the output.

Type: H264ColorMetadata
Required: False

colorSpaceSettings

Color Space settings

Type: H264ColorSpaceSettings
Required: False

entropyEncoding

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

Type: H264EntropyEncoding
Required: False

filterSettings

Optional filters that you can apply to an encode.

Type: H264FilterSettings
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: FixedAfd
Required: False

flickerAq

Flicker AQ makes adjustments within each frame to reduce flicker or 'pop' on I-frames. The value to enter in this field depends on the value in the Adaptive quantization field: If you have set the Adaptive quantization field to Auto, MediaLive ignores any value in this field. MediaLive will determine if flicker AQ is appropriate and will apply the appropriate strength. If you have set the Adaptive quantization field to a strength, you can set this field to Enabled or Disabled. Enabled: MediaLive will apply flicker AQ using the specified strength. Disabled: MediaLive won't apply flicker AQ. If you have set the Adaptive quantization to Disabled, MediaLive ignores any value in this field and doesn't apply flicker AQ.

Type: H264FlickerAq
Required: False

forceFieldPictures

This setting applies only when scan type is "interlaced." It controls whether coding is performed on a field basis or on a frame basis. (When the video is progressive, the coding is always performed on a frame basis.) enabled: Force MediaLive to code on a field basis, so that odd and even sets of fields are coded separately. disabled: Code the two sets of fields separately (on a field basis) or together (on a frame basis using PAFF), depending on what is most appropriate for the content.

Type: H264ForceFieldPictures
Required: False

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: H264FramerateControl
Required: False

framerateDenominator

Framerate denominator.

Type: integer
Required: False
Minimum: 1

framerateNumerator

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

Type: integer
Required: False
Minimum: 1

gopBReference

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

Type: H264GopBReference
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

gopNumBFrames

Number of B-frames between reference frames.

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

gopSize

GOP size (keyframe interval) in units of either frames or seconds per gopSizeUnits. If gopSizeUnits is frames, gopSize must be an integer and must be greater than or equal to 1. If gopSizeUnits is seconds, gopSize must be greater than 0, but need not be an integer.

Type: number
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: H264GopSizeUnits
Required: False

level

H.264 Level.

Type: H264Level
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: H264LookAheadRateControl

Required: False

maxBitrate

For QVBR: See the tooltip for Quality level For VBR: Set the maximum bitrate in order to accommodate expected spikes in the complexity of the video.

Type: integer
Required: False
Minimum: 1000

minIInterval

Only meaningful if sceneChangeDetect is set to enabled. Defaults to 5 if multiplex rate control is used. 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

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

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 pixsel aspect ratio will be set equal to the input video pixel aspect ratio of the first input.

Type: H264ParControl
Required: False

parDenominator

Pixel Aspect Ratio denominator.

Type: integer
Required: False
Minimum: 1

parNumerator

Pixel Aspect Ratio numerator.

Type: integer
Required: False
Minimum: 1

profile

H.264 Profile.

Type: H264Profile
Required: False

qualityLevel

Leave as STANDARD_QUALITY or choose a different value (which might result in additional costs to run the channel). - ENHANCED_QUALITY: Produces a slightly better video quality without an increase in the bitrate. Has an effect only when the Rate control mode is QVBR

or CBR. If this channel is in a MediaLive multiplex, the value must be ENHANCED_QUALITY. - STANDARD_QUALITY: Valid for any Rate control mode.

Type: H264QualityLevel
Required: False

qvbrQualityLevel

Controls the target quality for the video encode. Applies only when the rate control mode is QVBR. You can set a target quality or you can let MediaLive determine the best quality. To set a target quality, enter values in the QVBR quality level field and the Max bitrate field. Enter values that suit your most important viewing devices. Recommended values are: - Primary screen: Quality level: 8 to 10. Max bitrate: 4M - PC or tablet: Quality level: 7. Max bitrate: 1.5M to 3M - Smartphone: Quality level: 6. Max bitrate: 1M to 1.5M To let MediaLive decide, leave the QVBR quality level field empty, and in Max bitrate enter the maximum rate you want in the video. For more information, see the section called "Video - rate control mode" in the MediaLive user guide

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

rateControlMode

Rate control mode. QVBR: Quality will match the specified quality level except when it is constrained by the maximum bitrate. Recommended if you or your viewers pay for bandwidth. VBR: Quality and bitrate vary, depending on the video complexity. Recommended instead of QVBR if you want to maintain a specific average bitrate over the duration of the channel. CBR: Quality varies, depending on the video complexity. Recommended only if you distribute your assets to devices that cannot handle variable bitrates. Multiplex: This rate control mode is only supported (and is required) when the video is being delivered to a MediaLive Multiplex in which case the rate control configuration is controlled by the properties within the Multiplex Program.

Type: H264RateControlMode
Required: False

scanType

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

Type: H264ScanType
Required: False

sceneChangeDetect

Scene change detection. - On: inserts I-frames when scene change is detected. - Off: does not force an I-frame when scene change is detected.

Type: H264SceneChangeDetect
Required: False

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

softness

Softness. Selects quantizer matrix, larger values reduce high-frequency content in the encoded image. If not set to zero, must be greater than 15.

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

spatialAq

Spatial AQ makes adjustments within each frame based on spatial variation of content complexity. The value to enter in this field depends on the value in the Adaptive quantization field: If you have set the Adaptive quantization field to Auto, MediaLive ignores any value in this field. MediaLive will determine if spatial AQ is appropriate and will apply the appropriate strength. If you have set the Adaptive quantization field to a strength, you can set this field to Enabled or Disabled. Enabled: MediaLive will apply spatial AQ using the specified strength. Disabled: MediaLive won't apply spatial AQ. If you have set the Adaptive quantization to Disabled, MediaLive ignores any value in this field and doesn't apply spatial AQ.

Type: H264SpatialAq
Required: False

subgopLength

If set to fixed, use gopNumBFrames B-frames per sub-GOP. If set to dynamic, optimize the number of B-frames used for each sub-GOP to improve visual quality.

Type: H264SubGopLength
Required: False

syntax

Produces a bitstream compliant with SMPTE RP-2027.

Type: H264Syntax
Required: False

temporalAq

Temporal makes adjustments within each frame based on temporal variation of content complexity. The value to enter in this field depends on the value in the Adaptive quantization field: If you have set the Adaptive quantization field to Auto, MediaLive ignores any value in this field. MediaLive will determine if temporal AQ is appropriate and will apply the appropriate strength. If you have set the Adaptive quantization field to a strength, you can set this field to Enabled or Disabled. Enabled: MediaLive will apply temporal AQ using the specified strength. Disabled:

MediaLive won't apply temporal AQ. If you have set the Adaptive quantization to Disabled, MediaLive ignores any value in this field and doesn't apply temporal AQ.

Type: H264TemporalAq
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: H264TimecodeInsertionBehavior
Required: False

H264SpatialAq

H264 Spatial Aq

```
DISABLED
ENABLED
```

H264SubGopLength

H264 Sub Gop Length

```
DYNAMIC
FIXED
```

H264Syntax

H264 Syntax

```
DEFAULT
RP2027
```

H264TemporalAq

H264 Temporal Aq

```
DISABLED
ENABLED
```

H264TimecodeInsertionBehavior

H264 Timecode Insertion Behavior

```
DISABLED
PIC_TIMING_SEI
```

H265AdaptiveQuantization

H265 Adaptive Quantization

```
AUTO
HIGH
HIGHER
LOW
MAX
MEDIUM
OFF
```

H265AlternativeTransferFunction

H265 Alternative Transfer Function

```
INSERT
OMIT
```

H265ColorMetadata

H265 Color Metadata

```
IGNORE
```

INSERT

H265ColorSpaceSettings

H265 Color Space Settings

colorSpacePassthroughSettings

Type: ColorSpacePassthroughSettings
Required: False

hdr10Settings

Type: Hdr10Settings
Required: False

rec601Settings

Type: Rec601Settings
Required: False

rec709Settings

Type: Rec709Settings
Required: False

H265FilterSettings

H265 Filter Settings

temporalFilterSettings

Type: TemporalFilterSettings
Required: False

H265FlickerAq

H265 Flicker Aq

DISABLED
ENABLED

H265GopSizeUnits

H265 Gop Size Units

FRAMES
SECONDS

H265Level

H265 Level

H265_LEVEL_1
H265_LEVEL_2
H265_LEVEL_2_1
H265_LEVEL_3
H265_LEVEL_3_1
H265_LEVEL_4
H265_LEVEL_4_1
H265_LEVEL_5
H265_LEVEL_5_1
H265_LEVEL_5_2
H265_LEVEL_6
H265_LEVEL_6_1
H265_LEVEL_6_2
H265_LEVEL_AUTO

H265LookAheadRateControl

H265 Look Ahead Rate Control

HIGH
LOW
MEDIUM

H265Profile

H265 Profile

```
MAIN
MAIN_10BIT
```

H265RateControlMode

H265 Rate Control Mode

```
CBR
MULTIPLEX
QVBR
```

H265ScanType

H265 Scan Type

```
INTERLACED
PROGRESSIVE
```

H265SceneChangeDetect

H265 Scene Change Detect

```
DISABLED
ENABLED
```

H265Settings

H265 Settings

adaptiveQuantization

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

Type: H265AdaptiveQuantization

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: AfdSignaling
Required: False

alternativeTransferFunction

Whether or not EML should insert an Alternative Transfer Function SEI message to support backwards compatibility with non-HDR decoders and displays.

Type: H265AlternativeTransferFunction
Required: False

bitrate

Average bitrate in bits/second. Required when the rate control mode is VBR or CBR. Not used for QVBR. In an MS Smooth output group, each output must have a unique value when its bitrate is rounded down to the nearest multiple of 1000.

Type: integer
Required: False
Minimum: 100000
Maximum: 40000000

bufSize

Size of buffer (HRD buffer model) in bits.

Type: integer
Required: False
Minimum: 100000
Maximum: 80000000

colorMetadata

Includes colorspace metadata in the output.

Type: H265ColorMetadata
Required: False

colorSpaceSettings

Color Space settings

Type: H265ColorSpaceSettings
Required: False

filterSettings

Optional filters that you can apply to an encode.

Type: H265FilterSettings
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: FixedAfd
Required: False

flickerAq

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

Type: H265FlickerAq
Required: False

framerateDenominator

Framerate denominator.

Type: integer
Required: True
Minimum: 1
Maximum: 3003

framerateNumerator

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

Type: integer
Required: True
Minimum: 1

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

gopSize

GOP size (keyframe interval) in units of either frames or seconds per gopSizeUnits. If gopSizeUnits is frames, gopSize must be an integer and must be greater than or equal to 1. If gopSizeUnits is seconds, gopSize must be greater than 0, but need not be an integer.

Type: number
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: H265GopSizeUnits

Required: False

level

H.265 Level.

Type: H265Level
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: H265LookAheadRateControl
Required: False

maxBitrate

For QVBR: See the tooltip for Quality level

Type: integer
Required: False
Minimum: 100000
Maximum: 40000000

minIInterval

Only meaningful if sceneChangeDetect is set to enabled. Defaults to 5 if multiplex rate control is used. 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

parDenominator

Pixel Aspect Ratio denominator.

Type: integer
Required: False
Minimum: 1

parNumerator

Pixel Aspect Ratio numerator.

Type: integer
Required: False
Minimum: 1

profile

H.265 Profile.

Type: H265Profile
Required: False

qvbrQualityLevel

Controls the target quality for the video encode. Applies only when the rate control mode is QVBR. Set values for the QVBR quality level field and Max bitrate field that suit your most important viewing devices. Recommended values are: - Primary screen: Quality level: 8 to 10. Max bitrate: 4M - PC or tablet: Quality level: 7. Max bitrate: 1.5M to 3M - Smartphone: Quality level: 6. Max bitrate: 1M to 1.5M

Type: integer
Required: False
Minimum: 1

Maximum: 10

rateControlMode

Rate control mode. QVBR: Quality will match the specified quality level except when it is constrained by the maximum bitrate. Recommended if you or your viewers pay for bandwidth. CBR: Quality varies, depending on the video complexity. Recommended only if you distribute your assets to devices that cannot handle variable bitrates. Multiplex: This rate control mode is only supported (and is required) when the video is being delivered to a MediaLive Multiplex in which case the rate control configuration is controlled by the properties within the Multiplex Program.

Type: H265RateControlMode
Required: False

scanType

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

Type: H265ScanType
Required: False

sceneChangeDetect

Scene change detection.

Type: H265SceneChangeDetect
Required: False

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: 16

tier

H.265 Tier.

Type: H265Tier
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: H265TimecodeInsertionBehavior
Required: False

H265Tier

H265 Tier

`HIGH`
`MAIN`

H265TimecodeInsertionBehavior

H265 Timecode Insertion Behavior

`DISABLED`
`PIC_TIMING_SEI`

Hdr10Settings

Hdr10 Settings

maxCll

Maximum Content Light Level An integer metadata value defining the maximum light level, in nits, of any single pixel within an encoded HDR video stream or file.

Type: integer
Required: False
Minimum: 0
Maximum: 32768

maxFall

Maximum Frame Average Light Level An integer metadata value defining the maximum average light level, in nits, for any single frame within an encoded HDR video stream or file.

Type: integer
Required: False
Minimum: 0
Maximum: 32768

HlsAdMarkers

Hls Ad Markers

ADOBE
ELEMENTAL
ELEMENTAL_SCTE35

HlsAkamaiHttpTransferMode

Hls Akamai Http Transfer Mode

CHUNKED
NON_CHUNKED

HlsAkamaiSettings

Hls Akamai Settings

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

httpTransferMode

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

Type: HlsAkamaiHttpTransferMode
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

salt

Salt for authenticated Akamai.

Type: string
Required: False

token

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

Type: string
Required: False

HlsBasicPutSettings

Hls Basic Put Settings

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

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

HlsCaptionLanguageSetting

Hls Caption Language Setting

```
INSERT
NONE
OMIT
```

HlsCdnSettings

Hls Cdn Settings

hlsAkamaiSettings

Type: HlsAkamaiSettings
Required: False

hlsBasicPutSettings

Type: HlsBasicPutSettings

Required: False

hlsMediaStoreSettings

Type: HlsMediaStoreSettings
Required: False

hlsS3Settings

Type: HlsS3Settings
Required: False

hlsWebdavSettings

Type: HlsWebdavSettings
Required: False

HlsClientCache

Hls Client Cache

DISABLED
ENABLED

HlsCodecSpecification

Hls Codec Specification

RFC_4281
RFC_6381

HlsDirectoryStructure

Hls Directory Structure

SINGLE_DIRECTORY
SUBDIRECTORY_PER_STREAM

HlsDiscontinuityTags

Hls Discontinuity Tags

```
INSERT
NEVER_INSERT
```

HlsEncryptionType

Hls Encryption Type

```
AES128
SAMPLE_AES
```

HlsGroupSettings

Hls Group Settings

adMarkers

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

Type: Array of type HlsAdMarkers
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

baseUrlContent1

Optional. One value per output group. This field is required only if you are completing Base URL content A, and the downstream system has notified you that the media files for pipeline 1 of all outputs are in a location different from the media files for pipeline 0.

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

baseUrlManifest1

Optional. One value per output group. Complete this field only if you are completing Base URL manifest A, and the downstream system has notified you that the child manifest files for pipeline 1 of all outputs are in a location different from the child manifest files for pipeline 0.

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

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: HlsCaptionLanguageSetting
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: HlsClientCache
Required: False

codecSpecification

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

Type: HlsCodecSpecification
Required: False

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
MinLength: 32
MaxLength: 32

destination

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

Type: OutputLocationRef
Required: True

directoryStructure

Place segments in subdirectories.

Type: HlsDirectoryStructure
Required: False

discontinuityTags

Specifies whether to insert EXT-X-DISCONTINUITY tags in the HLS child manifests for this output group. Typically, choose Insert because these tags are required in the manifest (according to the HLS specification) and serve an important purpose. Choose Never Insert only if the downstream system is doing real-time failover (without using the MediaLive automatic failover feature) and only if that downstream system has advised you to exclude the tags.

Type: HlsDiscontinuityTags
Required: False

encryptionType

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

Type: HlsEncryptionType
Required: False

hlsCdnSettings

Parameters that control interactions with the CDN.

Type: HlsCdnSettings
Required: False

hlsId3SegmentTagging

State of HLS ID3 Segment Tagging

Type: HlsId3SegmentTaggingState
Required: False

iFrameOnlyPlaylists

DISABLED: Do not create an I-frame-only manifest, but do create the master and media manifests (according to the Output Selection field). STANDARD: Create an I-frame-only manifest for each output that contains video, as well as the other manifests (according to the Output Selection field). The I-frame manifest contains a #EXT-X-I-FRAMES-ONLY tag to indicate it is I-frame only, and one or more #EXT-X-BYTERANGE entries identifying the I-frame position. For example, #EXT-X-BYTERANGE:160364@1461888"

Type: IFrameOnlyPlaylistType
Required: False

incompleteSegmentBehavior

Specifies whether to include the final (incomplete) segment in the media output when the pipeline stops producing output because of a channel stop, a channel pause or a loss of input to the pipeline. Auto means that MediaLive decides whether to include the final segment, depending on the channel class and the types of output groups. Suppress means to never include the incomplete segment. We recommend you choose Auto and let MediaLive control the behavior.

Type: HlsIncompleteSegmentBehavior
Required: False

indexNSegments

Applies only if Mode field is LIVE. Specifies the maximum number of segments in the media manifest file. After this maximum, older segments are removed from the media manifest. This number must be smaller than the number in the Keep Segments field.

Type: integer
Required: False
Minimum: 3

inputLossAction

Parameter that control output group behavior on input loss.

Type: InputLossActionForHlsOut

Required: False

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: HlsIvInManifest
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: HlsIvSource
Required: False

keepSegments

Applies only if Mode field is LIVE. Specifies the number of media segments to retain in the destination directory. This number should be bigger than indexNSegments (Num segments). We recommend (value = (2 x indexNsegments) + 1). If this "keep segments" number is too low, the following might happen: the player is still reading a media manifest file that lists this segment, but that segment has been removed from the destination directory (as directed by indexNSegments). This situation would result in a 404 HTTP error on the player.

Type: integer
Required: False
Minimum: 1

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

keyFormatVersions

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

Type: string
Required: False

keyProviderSettings

The key provider settings.

Type: KeyProviderSettings
Required: False

manifestCompression

When set to gzip, compresses HLS playlist.

Type: HlsManifestCompression
Required: False

manifestDurationFormat

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

Type: HlsManifestDurationFormat
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 "vod", all segments are indexed and kept permanently in the destination and manifest. If "live", only the number segments specified in keepSegments and indexNSegments are kept; newer segments replace older segments, which may prevent players from rewinding all the way to the beginning of the event. VOD mode uses HLS EXT-X-PLAYLIST-TYPE of EVENT while the channel is running, converting it to a "VOD" type manifest on completion of the stream.

Type: HlsMode
Required: False

outputSelection

MANIFESTS_AND_SEGMENTS: Generates manifests (master manifest, if applicable, and media manifests) for this output group. VARIANT_MANIFESTS_AND_SEGMENTS: Generates media manifests for this output group, but not a master manifest. SEGMENTS_ONLY: Does not generate any manifests for this output group.

Type: HlsOutputSelection
Required: False

programDateTime

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

Type: HlsProgramDateTime
Required: False

programDateTimeClock

Specifies the algorithm used to drive the HLS EXT-X-PROGRAM-DATE-TIME clock. Options include: INITIALIZE_FROM_OUTPUT_TIMECODE: The PDT clock is initialized as a function of the first output timecode, then incremented by the EXTINF duration of each encoded segment. SYSTEM_CLOCK:

The PDT clock is initialized as a function of the UTC wall clock, then incremented by the EXTINF duration of each encoded segment. If the PDT clock diverges from the wall clock by more than 500ms, it is resynchronized to the wall clock.

Type: HlsProgramDateTimeClock
Required: False

programDateTimePeriod

Period of insertion of EXT-X-PROGRAM-DATE-TIME entry, in seconds.

Type: integer
Required: False
Minimum: 0
Maximum: 3600

redundantManifest

ENABLED: The master manifest (.m3u8 file) for each pipeline includes information about both pipelines: first its own media files, then the media files of the other pipeline. This feature allows playout device that support stale manifest detection to switch from one manifest to the other, when the current manifest seems to be stale. There are still two destinations and two master manifests, but both master manifests reference the media files from both pipelines. DISABLED: The master manifest (.m3u8 file) for each pipeline includes information about its own pipeline only. For an HLS output group with MediaPackage as the destination, the DISABLED behavior is always followed. MediaPackage regenerates the manifests it serves to players so a redundant manifest from MediaLive is irrelevant.

Type: HlsRedundantManifest
Required: False

segmentLength

Length of MPEG-2 Transport Stream segments to create (in seconds). Note that segments will end on the next keyframe after this number of seconds, so actual segment length may be longer.

Type: integer
Required: False

Minimum: 1

segmentationMode

useInputSegmentation has been deprecated. The configured segment size is always used.

Type: HlsSegmentationMode
Required: False

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

streamInfResolution

Include or exclude RESOLUTION attribute for video in EXT-X-STREAM-INF tag of variant manifest.

Type: HlsStreamInfResolution
Required: False

timedMetadataId3Frame

Indicates ID3 frame that has the timecode.

Type: HlsTimedMetadataId3Frame
Required: False

timedMetadataId3Period

Timed Metadata interval in seconds.

Type: integer
Required: False
Minimum: 0

timestampDeltaMilliseconds

Provides an extra millisecond delta offset to fine tune the timestamps.

Type: integer
Required: False
Minimum: 0

tsFileMode

SEGMENTED_FILES: Emit the program as segments - multiple .ts media files. SINGLE_FILE: Applies only if Mode field is VOD. Emit the program as a single .ts media file. The media manifest includes #EXT-X-BYTERANGE tags to index segments for playback. A typical use for this value is when sending the output to AWS Elemental MediaConvert, which can accept only a single media file. Playback while the channel is running is not guaranteed due to HTTP server caching.

Type: HlsTsFileMode
Required: False

HlsH265PackagingType

Hls H265 Packaging Type

HEV1
HVC1

HlsId3SegmentTaggingState

State of HLS ID3 Segment Tagging

DISABLED
ENABLED

HlsIncompleteSegmentBehavior

Hls Incomplete Segment Behavior

AUTO

SUPPRESS

HlsInputSettings

Hls Input Settings

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

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

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

retryInterval

The number of seconds between retries when an attempt to read a manifest or segment fails.

Type: integer
Required: False
Minimum: 0

scte35Source

Identifies the source for the SCTE-35 messages that MediaLive will ingest. Messages can be ingested from the content segments (in the stream) or from tags in the playlist (the HLS manifest). MediaLive ignores SCTE-35 information in the source that is not selected.

Type: HlsScte35SourceType
Required: False

HlsIvInManifest

Hls Iv In Manifest

```
EXCLUDE
INCLUDE
```

HlsIvSource

Hls Iv Source

```
EXPLICIT
FOLLOWS_SEGMENT_NUMBER
```

HlsManifestCompression

Hls Manifest Compression

```
GZIP
NONE
```

HlsManifestDurationFormat

Hls Manifest Duration Format

FLOATING_POINT
INTEGER

HlsMediaStoreSettings

Hls Media Store Settings

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

mediaStoreStorageClass

When set to temporal, output files are stored in non-persistent memory for faster reading and writing.

Type: HlsMediaStoreStorageClass
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

HlsMediaStoreStorageClass

Hls Media Store Storage Class

```
TEMPORAL
```

HlsMode

Hls Mode

```
LIVE
VOD
```

HlsOutputSelection

Hls Output Selection

```
MANIFESTS_AND_SEGMENTS
SEGMENTS_ONLY
VARIANT_MANIFESTS_AND_SEGMENTS
```

HlsOutputSettings

Hls Output Settings

h265PackagingType

Only applicable when this output is referencing an H.265 video description. Specifies whether MP4 segments should be packaged as HEV1 or HVC1.

Type: HlsH265PackagingType
Required: False

hlsSettings

Settings regarding the underlying stream. These settings are different for audio-only outputs.

Type: HlsSettings
Required: True

nameModifier

String concatenated to the end of the destination filename. Accepts \"Format Identifiers \":#formatIdentifierParameters.

Type: string
Required: False
MinLength: 1

segmentModifier

String concatenated to end of segment filenames.

Type: string
Required: False

HlsProgramDateTime

Hls Program Date Time

EXCLUDE
INCLUDE

HlsProgramDateTimeClock

Hls Program Date Time Clock

INITIALIZE_FROM_OUTPUT_TIMECODE

SYSTEM_CLOCK

HlsRedundantManifest

Hls Redundant Manifest

DISABLED
ENABLED

HlsS3LogUploads

Hls S3 Log Uploads

DISABLED
ENABLED

HlsS3Settings

Hls S3 Settings

cannedAcl

Specify the canned ACL to apply to each S3 request. Defaults to none.

Type: S3CannedAcl
Required: False

logUploads

When set to enabled, each fragment upload to CDN or server will be logged.

Type: HlsS3LogUploads
Required: False

HlsScte35SourceType

Hls Scte35 Source Type

MANIFEST

SEGMENTS

HlsSegmentationMode

Hls Segmentation Mode

USE_INPUT_SEGMENTATION
USE_SEGMENT_DURATION

HlsSettings

Hls Settings

audioOnlyHlsSettings

Type: AudioOnlyHlsSettings
Required: False

fmp4HlsSettings

Type: Fmp4HlsSettings
Required: False

frameCaptureHlsSettings

Type: FrameCaptureHlsSettings
Required: False

standardHlsSettings

Type: StandardHlsSettings
Required: False

HlsStreamInfResolution

Hls Stream Inf Resolution

EXCLUDE

INCLUDE

HlsTimedMetadataId3Frame

Hls Timed Metadata Id3 Frame

NONE
PRIV
TDRL

HlsTsFileMode

Hls Ts File Mode

SEGMENTED_FILES
SINGLE_FILE

HlsWebdavHttpTransferMode

Hls Webdav Http Transfer Mode

CHUNKED
NON_CHUNKED

HlsWebdavSettings

Hls Webdav Settings

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

httpTransferMode

Specify whether or not to use chunked transfer encoding to WebDAV.

Type: HlsWebdavHttpTransferMode
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

HtmlMotionGraphicsSettings

Html Motion Graphics Settings

IFrameOnlyPlaylistType

When set to "standard", an I-Frame only playlist will be written out for each video output in the output group. This I-Frame only playlist will contain byte range offsets pointing to the I-frame(s) in each segment.

DISABLED
STANDARD

InputAttachment

automaticInputFailoverSettings

User-specified settings for defining what the conditions are for declaring the input unhealthy and failing over to a different input.

Type: AutomaticInputFailoverSettings
Required: False

inputAttachmentName

User-specified name for the attachment. This is required if the user wants to use this input in an input switch action.

Type: string
Required: False

inputId

The ID of the input

Type: string
Required: False

inputSettings

Settings of an input (caption selector, etc.)

Type: InputSettings
Required: False

InputChannelLevel

Input Channel Level

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

inputChannel

The index of the input channel used as a source.

Type: integer
Required: True
Minimum: 0
Maximum: 15

InputCodec

codec in increasing order of complexity

MPEG2
AVC
HEVC

InputDeblockFilter

Input Deblock Filter

DISABLED
ENABLED

InputDenoiseFilter

Input Denoise Filter

DISABLED
ENABLED

InputFilter

Input Filter

AUTO
DISABLED
FORCED

InputLocation

Input Location

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 RTMP destination should have a uri simliar 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

Input Loss Action For Hls Out

```
EMIT_OUTPUT
PAUSE_OUTPUT
```

InputLossActionForMsSmoothOut

Input Loss Action For Ms Smooth Out

```
EMIT_OUTPUT
PAUSE_OUTPUT
```

InputLossActionForRtmpOut

Input Loss Action For Rtmp Out

```
EMIT_OUTPUT
PAUSE_OUTPUT
```

InputLossActionForUdpOut

Input Loss Action For Udp Out

```
DROP_PROGRAM
DROP_TS
EMIT_PROGRAM
```

InputLossBehavior

Input Loss Behavior

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

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
MinLength: 6
MaxLength: 6

inputLossImageSlate

When input loss image type is "slate" these fields specify the parameters for accessing the slate.

Type: InputLocation
Required: False

inputLossImageType

Indicates whether to substitute a solid color or a slate into the output after input loss exceeds blackFrameMsec.

Type: InputLossImageType
Required: False

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

InputLossFailoverSettings

MediaLive will perform a failover if content is not detected in this input for the specified period.

inputLossThresholdMsec

The amount of time (in milliseconds) that no input is detected. After that time, an input failover will occur.

Type: integer
Required: False
Minimum: 100

InputLossImageType

Input Loss Image Type

COLOR
SLATE

InputMaximumBitrate

Maximum input bitrate in megabits per second. Bitrates up to 50 Mbps are supported currently.

MAX_10_MBPS
MAX_20_MBPS
MAX_50_MBPS

InputPreference

Input preference when deciding which input to make active when a previously failed input has recovered. If \"EQUAL_INPUT_PREFERENCE\", then the active input will stay active as long as it is healthy. If \"PRIMARY_INPUT_PREFERRED\", then always switch back to the primary input when it is healthy.

EQUAL_INPUT_PREFERENCE

PRIMARY_INPUT_PREFERRED

InputResolution

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

Live Event input parameters. There can be multiple inputs in a single Live Event.

audioSelectors

Used to select the audio stream to decode for inputs that have multiple available.

Type: Array of type AudioSelector
Required: False

captionSelectors

Used to select the caption input to use for inputs that have multiple available.

Type: Array of type CaptionSelector
Required: False

deblockFilter

Enable or disable the deblock filter when filtering.

Type: InputDeblockFilter
Required: False

denoiseFilter

Enable or disable the denoise filter when filtering.

Type: InputDenoiseFilter
Required: False

filterStrength

Adjusts the magnitude of filtering from 1 (minimal) to 5 (strongest).

Type: integer
Required: False
Minimum: 1
Maximum: 5

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: InputFilter
Required: False

networkInputSettings

Input settings.

Type: NetworkInputSettings
Required: False

smpte2038DataPreference

Specifies whether to extract applicable ancillary data from a SMPTE-2038 source in this input. Applicable data types are captions, timecode, AFD, and SCTE-104 messages. - PREFER: Extract from SMPTE-2038 if present in this input, otherwise extract from another source (if any). - IGNORE: Never extract any ancillary data from SMPTE-2038.

Type: Smpte2038DataPreference
Required: False

sourceEndBehavior

Loop input if it is a file. This allows a file input to be streamed indefinitely.

Type: InputSourceEndBehavior
Required: False

videoSelector

Informs which video elementary stream to decode for input types that have multiple available.

Type: VideoSelector
Required: False

InputSourceEndBehavior

Input Source End Behavior

CONTINUE
LOOP

InputSpecification

codec

Input codec

Type: InputCodec
Required: False

maximumBitrate

Maximum input bitrate, categorized coarsely

Type: InputMaximumBitrate
Required: False

resolution

Input resolution, categorized coarsely

Type: InputResolution
Required: False

InternalServiceError

message

Type: string
Required: False

InvalidRequest

message

Type: string
Required: False

KeyProviderSettings

Key Provider Settings

staticKeySettings

Type: StaticKeySettings
Required: False

LimitExceeded

message

Type: string
Required: False

LogLevel

The log level the user wants for their channel.

```
ERROR
WARNING
INFO
DEBUG
DISABLED
```

M2tsAbsentInputAudioBehavior

M2ts Absent Input Audio Behavior

```
DROP
ENCODE_SILENCE
```

M2tsArib

M2ts Arib

```
DISABLED
ENABLED
```

M2tsAribCaptionsPidControl

M2ts Arib Captions Pid Control

```
AUTO
USE_CONFIGURED
```

M2tsAudioBufferModel

M2ts Audio Buffer Model

```
ATSC
DVB
```

M2tsAudioInterval

M2ts Audio Interval

VIDEO_AND_FIXED_INTERVALS
VIDEO_INTERVAL

M2tsAudioStreamType

M2ts Audio Stream Type

ATSC
DVB

M2tsBufferModel

M2ts Buffer Model

MULTIPLEX
NONE

M2tsCcDescriptor

M2ts Cc Descriptor

DISABLED
ENABLED

M2tsEbifControl

M2ts Ebif Control

NONE
PASSTHROUGH

M2tsEbpPlacement

M2ts Ebp Placement

VIDEO_AND_AUDIO_PIDS
VIDEO_PID

M2tsEsRateInPes

M2ts Es Rate In Pes

EXCLUDE
INCLUDE

M2tsKlv

M2ts Klv

NONE
PASSTHROUGH

M2tsNielsenId3Behavior

M2ts Nielsen Id3 Behavior

NO_PASSTHROUGH
PASSTHROUGH

M2tsPcrControl

M2ts Pcr Control

CONFIGURED_PCR_PERIOD
PCR_EVERY_PES_PACKET

M2tsRateMode

M2ts Rate Mode

CBR
VBR

M2tsScte35Control

M2ts Scte35 Control

NONE
PASSTHROUGH

M2tsSegmentationMarkers

M2ts Segmentation Markers

EBP
EBP_LEGACY
NONE
PSI_SEGSTART
RAI_ADAPT
RAI_SEGSTART

M2tsSegmentationStyle

M2ts Segmentation Style

MAINTAIN_CADENCE
RESET_CADENCE

M2tsSettings

M2ts Settings

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: M2tsAbsentInputAudioBehavior

Required: False

arib

When set to enabled, uses ARIB-compliant field muxing and removes video descriptor.

Type: M2tsArib
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

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: M2tsAribCaptionsPidControl
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: M2tsAudioBufferModel
Required: False

audioFramesPerPes

The number of audio frames to insert for each PES packet.

Type: integer

Required: False
Minimum: 0

audioPids

Packet Identifier (PID) of the elementary audio stream(s) in the transport stream. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values. Each PID specified must be in the range of 32 (or 0x20)..8182 (or 0x1ff6).

Type: string
Required: False

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: M2tsAudioStreamType
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

bufferModel

Controls the timing accuracy for output network traffic. Leave as MULTIPLEX to ensure accurate network packet timing. Or set to NONE, which might result in lower latency but will result in more variability in output network packet timing. This variability might cause interruptions, jitter, or bursty behavior in your playback or receiving devices.

Type: M2tsBufferModel

Required: False

ccDescriptor

When set to enabled, generates captionServiceDescriptor in PMT.

Type: M2tsCcDescriptor
Required: False

dvbNitSettings

Inserts DVB Network Information Table (NIT) at the specified table repetition interval.

Type: DvbNitSettings
Required: False

dvbSdtSettings

Inserts DVB Service Description Table (SDT) at the specified table repetition interval.

Type: DvbSdtSettings
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

dvbTdtSettings

Inserts DVB Time and Date Table (TDT) at the specified table repetition interval.

Type: DvbTdtSettings
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

ebif

If set to passthrough, passes any EBIF data from the input source to this output.

Type: M2tsEbifControl
Required: False

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: M2tsAudioInterval
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

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: M2tsEbpPlacement
Required: False

ecmPid

This field is unused and deprecated.

Type: string
Required: False

esRateInPes

Include or exclude the ES Rate field in the PES header.

Type: M2tsEsRateInPes
Required: False

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

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

fragmentTime

The length in seconds of each fragment. Only used with EBP markers.

Type: number
Required: False
Minimum: 0

klv

If set to passthrough, passes any KLV data from the input source to this output.

Type: M2tsKlv
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

nielsenId3Behavior

If set to passthrough, Nielsen inaudible tones for media tracking will be detected in the input audio and an equivalent ID3 tag will be inserted in the output.

Type: M2tsNielsenId3Behavior
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

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

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: M2tsPcrControl
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

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

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

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

programNum

The value of the program number field in the Program Map Table.

Type: integer
Required: False
Minimum: 0
Maximum: 65535

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: M2tsRateMode
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

scte35Control

Optionally pass SCTE-35 signals from the input source to this output.

Type: M2tsScte35Control

Required: False

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

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: M2tsSegmentationMarkers

Required: False

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: M2tsSegmentationStyle
Required: False

segmentationTime

The length in seconds of each segment. Required unless markers is set to _none_.

Type: number
Required: False
Minimum: 1

timedMetadataBehavior

When set to passthrough, timed metadata will be passed through from input to output.

Type: M2tsTimedMetadataBehavior
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

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. Valid values are 32 (or 0x20)..8182 (or 0x1ff6).

Type: string
Required: False

M2tsTimedMetadataBehavior

M2ts Timed Metadata Behavior

NO_PASSTHROUGH
PASSTHROUGH

M3u8NielsenId3Behavior

M3u8 Nielsen Id3 Behavior

NO_PASSTHROUGH
PASSTHROUGH

M3u8PcrControl

M3u8 Pcr Control

CONFIGURED_PCR_PERIOD
PCR_EVERY_PES_PACKET

M3u8Scte35Behavior

M3u8 Scte35 Behavior

NO_PASSTHROUGH
PASSTHROUGH

M3u8Settings

Settings information for the .m3u8 container

audioFramesPerPes

The number of audio frames to insert for each PES packet.

Type: integer
Required: False
Minimum: 0

audioPids

Packet Identifier (PID) of the elementary audio stream(s) in the transport stream. Multiple values are accepted, and can be entered in ranges and/or by comma separation. Can be entered as decimal or hexadecimal values.

Type: string
Required: False

ecmPid

This parameter is unused and deprecated.

Type: string
Required: False

nielsenId3Behavior

If set to passthrough, Nielsen inaudible tones for media tracking will be detected in the input audio and an equivalent ID3 tag will be inserted in the output.

Type: M3u8NielsenId3Behavior
Required: False

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

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: M3u8PcrControl
Required: False

pcrPeriod

Maximum time in milliseconds between Program Clock References (PCRs) inserted into the transport stream.

Type: integer
Required: False
Minimum: 0
Maximum: 500

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

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

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

programNum

The value of the program number field in the Program Map Table.

Type: integer
Required: False
Minimum: 0
Maximum: 65535

scte35Behavior

If set to passthrough, passes any SCTE-35 signals from the input source to this output.

Type: M3u8Scte35Behavior
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

timedMetadataBehavior

When set to passthrough, timed metadata is passed through from input to output.

Type: M3u8TimedMetadataBehavior
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

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

M3u8TimedMetadataBehavior

M3u8 Timed Metadata Behavior

NO_PASSTHROUGH
PASSTHROUGH

MaintenanceDay

The currently selected maintenance day.

MONDAY
TUESDAY
WEDNESDAY
THURSDAY
FRIDAY
SATURDAY
SUNDAY

MaintenanceStatus

maintenanceDay

The currently selected maintenance day.

Type: MaintenanceDay
Required: False

maintenanceDeadline

Maintenance is required by the displayed date and time. Date and time is in ISO.

Type: string
Required: False
Format: string

maintenanceScheduledDate

The currently scheduled maintenance date and time. Date and time is in ISO.

Type: string
Required: False
Format: string

maintenanceStartTime

The currently selected maintenance start time. Time is in UTC.

Type: string
Required: False

MediaPackageGroupSettings

Media Package Group Settings

destination

MediaPackage channel destination.

Type: OutputLocationRef
Required: True

MediaPackageOutputDestinationSettings

MediaPackage Output Destination Settings

channelId

ID of the channel in MediaPackage that is the destination for this output group. You do not need to specify the individual inputs in MediaPackage; MediaLive will handle the connection of the two MediaLive pipelines to the two MediaPackage inputs. The MediaPackage channel and MediaLive channel must be in the same region.

Type: string
Required: False
MinLength: 1

MediaPackageOutputSettings

Media Package Output Settings

MotionGraphicsConfiguration

Motion Graphics Configuration

motionGraphicsInsertion

Type: MotionGraphicsInsertion
Required: False

motionGraphicsSettings

Motion Graphics Settings

Type: MotionGraphicsSettings
Required: True

MotionGraphicsInsertion

Motion Graphics Insertion

```
DISABLED
ENABLED
```

MotionGraphicsSettings

Motion Graphics Settings

htmlMotionGraphicsSettings

Type: HtmlMotionGraphicsSettings
Required: False

Mp2CodingMode

Mp2 Coding Mode

CODING_MODE_1_0

CODING_MODE_2_0

Mp2Settings

Mp2 Settings

bitrate

Average bitrate in bits/second.

Type: number

Required: False

codingMode

The MPEG2 Audio coding mode. Valid values are codingMode10 (for mono) or codingMode20 (for stereo).

Type: Mp2CodingMode

Required: False

sampleRate

Sample rate in Hz.

Type: number

Required: False

Mpeg2AdaptiveQuantization

Mpeg2 Adaptive Quantization

AUTO

HIGH

LOW

MEDIUM

OFF

Mpeg2ColorMetadata

Mpeg2 Color Metadata

IGNORE
INSERT

Mpeg2ColorSpace

Mpeg2 Color Space

AUTO
PASSTHROUGH

Mpeg2DisplayRatio

Mpeg2 Display Ratio

DISPLAYRATIO16X9
DISPLAYRATIO4X3

Mpeg2FilterSettings

Mpeg2 Filter Settings

temporalFilterSettings

Type: TemporalFilterSettings
Required: False

Mpeg2GopSizeUnits

Mpeg2 Gop Size Units

FRAMES

SECONDS

Mpeg2ScanType

Mpeg2 Scan Type

INTERLACED
PROGRESSIVE

Mpeg2Settings

Mpeg2 Settings

adaptiveQuantization

Choose Off to disable adaptive quantization. Or choose another value to enable the quantizer and set its strength. The strengths are: Auto, Off, Low, Medium, High. When you enable this field, MediaLive allows intra-frame quantizers to vary, which might improve visual quality.

Type: Mpeg2AdaptiveQuantization
Required: False

afdSignaling

Indicates the AFD values that MediaLive will write into the video encode. If you do not know what AFD signaling is, or if your downstream system has not given you guidance, choose AUTO. AUTO: MediaLive will try to preserve the input AFD value (in cases where multiple AFD values are valid). FIXED: MediaLive will use the value you specify in fixedAFD.

Type: AfdSignaling
Required: False

colorMetadata

Specifies whether to include the color space metadata. The metadata describes the color space that applies to the video (the colorSpace field). We recommend that you insert the metadata.

Type: Mpeg2ColorMetadata
Required: False

colorSpace

Choose the type of color space conversion to apply to the output. For detailed information on setting up both the input and the output to obtain the desired color space in the output, see the section on \"MediaLive Features - Video - color space\" in the MediaLive User Guide. PASSTHROUGH: Keep the color space of the input content - do not convert it. AUTO:Convert all content that is SD to rec 601, and convert all content that is HD to rec 709.

Type: Mpeg2ColorSpace
Required: False

displayAspectRatio

Sets the pixel aspect ratio for the encode.

Type: Mpeg2DisplayRatio
Required: False

filterSettings

Optionally specify a noise reduction filter, which can improve quality of compressed content. If you do not choose a filter, no filter will be applied. TEMPORAL: This filter is useful for both source content that is noisy (when it has excessive digital artifacts) and source content that is clean. When the content is noisy, the filter cleans up the source content before the encoding phase, with these two effects: First, it improves the output video quality because the content has been cleaned up. Secondly, it decreases the bandwidth because MediaLive does not waste bits on encoding noise. When the content is reasonably clean, the filter tends to decrease the bitrate.

Type: Mpeg2FilterSettings
Required: False

fixedAfd

Complete this field only when afdSignaling is set to FIXED. Enter the AFD value (4 bits) to write on all frames of the video encode.

Type: FixedAfd
Required: False

framerateDenominator

description": "The framerate denominator. For example, 1001. The framerate is the numerator divided by the denominator. For example, 24000 / 1001 = 23.976 FPS.

Type: integer
Required: True
Minimum: 1

framerateNumerator

The framerate numerator. For example, 24000. The framerate is the numerator divided by the denominator. For example, 24000 / 1001 = 23.976 FPS.

Type: integer
Required: True
Minimum: 1

gopClosedCadence

MPEG2: default is open GOP.

Type: integer
Required: False
Minimum: 0

gopNumBFrames

Relates to the GOP structure. The number of B-frames between reference frames. If you do not know what a B-frame is, use the default.

Type: integer
Required: False
Minimum: 0
Maximum: 7

gopSize

Relates to the GOP structure. The GOP size (keyframe interval) in the units specified in gopSizeUnits. If you do not know what GOP is, use the default. If gopSizeUnits is frames, then the gopSize must be an integer and must be greater than or equal to 1. If gopSizeUnits is seconds, the gopSize must be greater than 0, but does not need to be an integer.

Type: number
Required: False

gopSizeUnits

Relates to the GOP structure. Specifies whether the gopSize is specified in frames or seconds. If you do not plan to change the default gopSize, leave the default. If you specify SECONDS, MediaLive will internally convert the gop size to a frame count.

Type: Mpeg2GopSizeUnits
Required: False

scanType

Set the scan type of the output to PROGRESSIVE or INTERLACED (top field first).

Type: Mpeg2ScanType
Required: False

subgopLength

Relates to the GOP structure. If you do not know what GOP is, use the default. FIXED: Set the number of B-frames in each sub-GOP to the value in gopNumBFrames. DYNAMIC: Let MediaLive optimize the number of B-frames in each sub-GOP, to improve visual quality.

Type: Mpeg2SubGopLength
Required: False

timecodeInsertion

Determines how MediaLive inserts timecodes in the output video. For detailed information about setting up the input and the output for a timecode, see the section on \"MediaLive Features -

Timecode configuration\" in the MediaLive User Guide. DISABLED: do not include timecodes. GOP_TIMECODE: Include timecode metadata in the GOP header.

Type: Mpeg2TimecodeInsertionBehavior
Required: False

Mpeg2SubGopLength

Mpeg2 Sub Gop Length

```
DYNAMIC
FIXED
```

Mpeg2TimecodeInsertionBehavior

Mpeg2 Timecode Insertion Behavior

```
DISABLED
GOP_TIMECODE
```

MsSmoothGroupSettings

Ms Smooth Group Settings

acquisitionPointId

The ID to include in each message in the sparse track. Ignored if sparseTrackType is NONE.

Type: string
Required: False

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: SmoothGroupAudioOnlyTimecodeControl
Required: False

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.

Type: SmoothGroupCertificateMode
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

destination

Smooth Streaming publish point on an IIS server. Elemental Live acts as a "Push" encoder to IIS.

Type: OutputLocationRef
Required: True

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

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: SmoothGroupEventIdMode
Required: False

eventStopBehavior

When set to sendEos, send EOS signal to IIS server when stopping the event

Type: SmoothGroupEventStopBehavior
Required: False

filecacheDuration

Size in seconds of file cache for streaming outputs.

Type: integer
Required: False
Minimum: 0

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

inputLossAction

Parameter that control output group behavior on input loss.

Type: InputLossActionForMsSmoothOut
Required: False

numRetries

Number of retry attempts.

Type: integer
Required: False
Minimum: 0

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

segmentationMode

useInputSegmentation has been deprecated. The configured segment size is always used.

Type: SmoothGroupSegmentationMode
Required: False

sendDelayMs

Number of milliseconds to delay the output from the second pipeline.

Type: integer
Required: False
Minimum: 0
Maximum: 10000

sparseTrackType

Identifies the type of data to place in the sparse track: - SCTE35: Insert SCTE-35 messages from the source content. With each message, insert an IDR frame to start a new segment. - SCTE35_WITHOUT_SEGMENTATION: Insert SCTE-35 messages from the source content. With each

message, insert an IDR frame but don't start a new segment. - NONE: Don't generate a sparse track for any outputs in this output group.

Type: SmoothGroupSparseTrackType
Required: False

streamManifestBehavior

When set to send, send stream manifest so publishing point doesn't start until all streams start.

Type: SmoothGroupStreamManifestBehavior
Required: False

timestampOffset

Timestamp offset for the event. Only used if timestampOffsetMode is set to useConfiguredOffset.

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: SmoothGroupTimestampOffsetMode
Required: False

MsSmoothH265PackagingType

Ms Smooth H265 Packaging Type

HEV1
HVC1

MsSmoothOutputSettings

Ms Smooth Output Settings

h265PackagingType

Only applicable when this output is referencing an H.265 video description. Specifies whether MP4 segments should be packaged as HEV1 or HVC1.

Type: MsSmoothH265PackagingType
Required: False

nameModifier

String concatenated to the end of the destination filename. Required for multiple outputs of the same type.

Type: string
Required: False

MultiplexGroupSettings

Multiplex Group Settings

MultiplexOutputSettings

Multiplex Output Settings

destination

Destination is a Multiplex.

Type: OutputLocationRef
Required: True

MultiplexProgramChannelDestinationSettings

Multiplex Program Input Destination Settings for outputting a Channel to a Multiplex

multiplexId

The ID of the Multiplex that the encoder is providing output to. You do not need to specify the individual inputs to the Multiplex; MediaLive will handle the connection of the two MediaLive pipelines to the two Multiplex instances. The Multiplex must be in the same region as the Channel.

Type: string
Required: False
MinLength: 1

programName

The program name of the Multiplex program that the encoder is providing output to.

Type: string
Required: False
MinLength: 1

NetworkInputServerValidation

Network Input Server Validation

```
CHECK_CRYPTOGRAPHY_AND_VALIDATE_NAME
CHECK_CRYPTOGRAPHY_ONLY
```

NetworkInputSettings

Network source to transcode. Must be accessible to the Elemental Live node that is running the live event through a network connection.

hlsInputSettings

Specifies HLS input settings when the uri is for a HLS manifest.

Type: HlsInputSettings
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: NetworkInputServerValidation
Required: False

NielsenCBET

Nielsen CBET

cbetCheckDigitString

Enter the CBET check digits to use in the watermark.

Type: string
Required: True
MinLength: 2
MaxLength: 2

cbetStepaside

Determines the method of CBET insertion mode when prior encoding is detected on the same layer.

Type: NielsenWatermarksCbetStepaside
Required: True

csid

Enter the CBET Source ID (CSID) to use in the watermark

Type: string
Required: True
MinLength: 1
MaxLength: 7

NielsenConfiguration

Nielsen Configuration

distributorId

Enter the Distributor ID assigned to your organization by Nielsen.

Type: string
Required: False

nielsenPcmToId3Tagging

Enables Nielsen PCM to ID3 tagging

Type: NielsenPcmToId3TaggingState
Required: False

NielsenNaesIiNw

Nielsen Naes Ii Nw

checkDigitString

Enter the check digit string for the watermark

Type: string
Required: True
MinLength: 2
MaxLength: 2

sid

Enter the Nielsen Source ID (SID) to include in the watermark

Type: number
Required: True
Minimum: 1
Maximum: 65535

NielsenPcmToId3TaggingState

State of Nielsen PCM to ID3 tagging

DISABLED
ENABLED

NielsenWatermarksCbetStepaside

Nielsen Watermarks Cbet Stepaside

```
DISABLED
ENABLED
```

NielsenWatermarksDistributionTypes

Nielsen Watermarks Distribution Types

```
FINAL_DISTRIBUTOR
PROGRAM_CONTENT
```

NielsenWatermarksSettings

Nielsen Watermarks Settings

nielsenCbetSettings

Complete these fields only if you want to insert watermarks of type Nielsen CBET

Type: NielsenCBET
Required: False

nielsenDistributionType

Choose the distribution types that you want to assign to the watermarks: - PROGRAM_CONTENT - FINAL_DISTRIBUTOR

Type: NielsenWatermarksDistributionTypes
Required: False

nielsenNaesIiNwSettings

Complete these fields only if you want to insert watermarks of type Nielsen NAES II (N2) and Nielsen NAES VI (NW).

Type: NielsenNaesIiNw
Required: False

Output

Output settings. There can be multiple outputs within a group.

audioDescriptionNames

The names of the AudioDescriptions used as audio sources for this output.

Type: Array of 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
MinLength: 1
MaxLength: 255

outputSettings

Output type-specific settings.

Type: OutputSettings
Required: True

videoDescriptionName

The name of the VideoDescription used as the source for this output.

Type: string
Required: False

OutputDestination

id

User-specified id. This is used in an output group or an output.

Type: string
Required: False

mediaPackageSettings

Destination settings for a MediaPackage output; one destination for both encoders.

Type: Array of type MediaPackageOutputDestinationSettings
Required: False

multiplexSettings

Destination settings for a Multiplex output; one destination for both encoders.

Type: MultiplexProgramChannelDestinationSettings
Required: False

settings

Destination settings for a standard output; one destination for each redundant encoder.

Type: Array of type OutputDestinationSettings
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

Output groups for this Live Event. Output groups contain information about where streams should be distributed.

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
MaxLength: 32

outputGroupSettings

Settings associated with the output group.

Type: OutputGroupSettings

Required: True

outputs

Type: Array of type Output

Required: True

OutputGroupSettings

Output Group Settings

archiveGroupSettings

Type: ArchiveGroupSettings

Required: False

frameCaptureGroupSettings

Type: FrameCaptureGroupSettings

Required: False

hlsGroupSettings

Type: HlsGroupSettings

Required: False

mediaPackageGroupSettings

Type: MediaPackageGroupSettings

Required: False

msSmoothGroupSettings

Type: MsSmoothGroupSettings

Required: False

multiplexGroupSettings

Type: MultiplexGroupSettings
Required: False

rtmpGroupSettings

Type: RtmpGroupSettings
Required: False

udpGroupSettings

Type: UdpGroupSettings
Required: False

OutputLocationRef

Reference to an OutputDestination ID defined in the channel

destinationRefId

Type: string
Required: False

OutputSettings

Output Settings

archiveOutputSettings

Type: ArchiveOutputSettings
Required: False

frameCaptureOutputSettings

Type: FrameCaptureOutputSettings
Required: False

hlsOutputSettings

Type: HlsOutputSettings
Required: False

mediaPackageOutputSettings

Type: MediaPackageOutputSettings
Required: False

msSmoothOutputSettings

Type: MsSmoothOutputSettings
Required: False

multiplexOutputSettings

Type: MultiplexOutputSettings
Required: False

rtmpOutputSettings

Type: RtmpOutputSettings
Required: False

udpOutputSettings

Type: UdpOutputSettings
Required: False

PassThroughSettings

Pass Through Settings

PipelineDetail

Runtime details of a pipeline when a channel is running.

activeInputAttachmentName

The name of the active input attachment currently being ingested by this pipeline.

Type: string
Required: False

activeInputSwitchActionName

The name of the input switch schedule action that occurred most recently and that resulted in the switch to the current input attachment for this pipeline.

Type: string
Required: False

activeMotionGraphicsActionName

The name of the motion graphics activate action that occurred most recently and that resulted in the current graphics URI for this pipeline.

Type: string
Required: False

activeMotionGraphicsUri

The current URI being used for HTML5 motion graphics for this pipeline.

Type: string
Required: False

pipelineId

Pipeline ID

Type: string
Required: False

RawSettings

Raw Settings

Rec601Settings

Rec601 Settings

Rec709Settings

Rec709 Settings

RemixSettings

Remix Settings

channelMappings

Mapping of input channels to output channels, with appropriate gain adjustments.

Type: Array of type AudioChannelMapping
Required: True

channelsIn

Number of input channels to be used.

Type: integer
Required: False
Minimum: 1
Maximum: 16

channelsOut

Number of output channels to be produced. Valid values: 1, 2, 4, 6, 8

Type: integer
Required: False
Minimum: 1
Maximum: 8

ResourceConflict

message

Type: string
Required: False

ResourceNotFound

message

Type: string
Required: False

RtmpAdMarkers

Rtmp Ad Markers

```
ON_CUE_POINT_SCTE35
```

RtmpCacheFullBehavior

Rtmp Cache Full Behavior

```
DISCONNECT_IMMEDIATELY
WAIT_FOR_SERVER
```

RtmpCaptionData

Rtmp Caption Data

```
ALL
FIELD1_608
FIELD1_AND_FIELD2_608
```

RtmpCaptionInfoDestinationSettings

Rtmp Caption Info Destination Settings

RtmpGroupSettings

Rtmp Group Settings

adMarkers

Choose the ad marker type for this output group. MediaLive will create a message based on the content of each SCTE-35 message, format it for that marker type, and insert it in the datastream.

Type: Array of type RtmpAdMarkers
Required: False

authenticationScheme

Authentication scheme to use when connecting with CDN

Type: AuthenticationScheme
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: RtmpCacheFullBehavior
Required: False

cacheLength

Cache length, in seconds, is used to calculate buffer size.

Type: integer
Required: False
Minimum: 30

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: RtmpCaptionData
Required: False

inputLossAction

Controls the behavior of this RTMP group if input becomes unavailable. - emitOutput: Emit a slate until input returns. - pauseOutput: Stop transmitting data until input returns. This does not close the underlying RTMP connection.

Type: InputLossActionForRtmpOut
Required: False

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

RtmpOutputCertificateMode

Rtmp Output Certificate Mode

```
SELF_SIGNED
VERIFY_AUTHENTICITY
```

RtmpOutputSettings

Rtmp Output Settings

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: RtmpOutputCertificateMode
Required: False

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

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
Required: True

numRetries

Number of retry attempts.

Type: integer
Required: False
Minimum: 0

S3CannedAcl

S3 Canned Acl

```
AUTHENTICATED_READ
BUCKET_OWNER_FULL_CONTROL
```

BUCKET_OWNER_READ
PUBLIC_READ

Scte20Convert608To708

Scte20 Convert608 To708

DISABLED
UPCONVERT

Scte20PlusEmbeddedDestinationSettings

Scte20 Plus Embedded Destination Settings

Scte20SourceSettings

Scte20 Source Settings

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: Scte20Convert608To708
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

Scte27DestinationSettings

Scte27 Destination Settings

Scte27OcrLanguage

Scte27 Ocr Language

DEU
ENG
FRA
NLD
POR
SPA

Scte27SourceSettings

Scte27 Source Settings

ocrLanguage

If you will configure a WebVTT caption description that references this caption selector, use this field to provide the language to consider when translating the image-based source to text.

Type: Scte27OcrLanguage
Required: False

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

Scte35 Apos No Regional Blackout Behavior

FOLLOW
IGNORE

Scte35AposWebDeliveryAllowedBehavior

Scte35 Apos Web Delivery Allowed Behavior

FOLLOW
IGNORE

Scte35SpliceInsert

Scte35 Splice Insert

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

noRegionalBlackoutFlag

When set to ignore, Segment Descriptors with noRegionalBlackoutFlag set to 0 will no longer trigger blackouts or Ad Avail slates

Type: Scte35SpliceInsertNoRegionalBlackoutBehavior
Required: False

webDeliveryAllowedFlag

When set to ignore, Segment Descriptors with webDeliveryAllowedFlag set to 0 will no longer trigger blackouts or Ad Avail slates

Type: Scte35SpliceInsertWebDeliveryAllowedBehavior

Required: False

Scte35SpliceInsertNoRegionalBlackoutBehavior

Scte35 Splice Insert No Regional Blackout Behavior

FOLLOW
IGNORE

Scte35SpliceInsertWebDeliveryAllowedBehavior

Scte35 Splice Insert Web Delivery Allowed Behavior

FOLLOW
IGNORE

Scte35TimeSignalApos

Scte35 Time Signal Apos

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

noRegionalBlackoutFlag

When set to ignore, Segment Descriptors with noRegionalBlackoutFlag set to 0 will no longer trigger blackouts or Ad Avail slates

Type: Scte35AposNoRegionalBlackoutBehavior
Required: False

webDeliveryAllowedFlag

When set to ignore, Segment Descriptors with webDeliveryAllowedFlag set to 0 will no longer trigger blackouts or Ad Avail slates

Type: Scte35AposWebDeliveryAllowedBehavior

Required: False

SmoothGroupAudioOnlyTimecodeControl

Smooth Group Audio Only Timecode Control

PASSTHROUGH

USE_CONFIGURED_CLOCK

SmoothGroupCertificateMode

Smooth Group Certificate Mode

SELF_SIGNED

VERIFY_AUTHENTICITY

SmoothGroupEventIdMode

Smooth Group Event Id Mode

NO_EVENT_ID

USE_CONFIGURED

USE_TIMESTAMP

SmoothGroupEventStopBehavior

Smooth Group Event Stop Behavior

NONE

SEND_EOS

SmoothGroupSegmentationMode

Smooth Group Segmentation Mode

USE_INPUT_SEGMENTATION
USE_SEGMENT_DURATION

SmoothGroupSparseTrackType

Smooth Group Sparse Track Type

NONE
SCTE_35
SCTE_35_WITHOUT_SEGMENTATION

SmoothGroupStreamManifestBehavior

Smooth Group Stream Manifest Behavior

DO_NOT_SEND
SEND

SmoothGroupTimestampOffsetMode

Smooth Group Timestamp Offset Mode

USE_CONFIGURED_OFFSET
USE_EVENT_START_DATE

Smpte2038DataPreference

Smpte2038 Data Preference

IGNORE
PREFER

SmpteTtDestinationSettings

Smpte Tt Destination Settings

StandardHlsSettings

Standard Hls Settings

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

m3u8Settings

Type: M3u8Settings
Required: True

StaticKeySettings

Static Key Settings

keyProviderServer

The URL of the license server used for protecting content.

Type: InputLocation
Required: False

staticKeyValue

Static key value as a 32 character hexadecimal string.

Type: string
Required: True
MinLength: 32

MaxLength: 32

Tags

key-value pairs

Type: string

TeletextDestinationSettings

Teletext Destination Settings

TeletextSourceSettings

Teletext Source Settings

outputRectangle

Optionally defines a region where TTML style captions will be displayed

Type: CaptionRectangle
Required: False

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

TemporalFilterPostFilterSharpening

Temporal Filter Post Filter Sharpening

AUTO
DISABLED
ENABLED

TemporalFilterSettings

Temporal Filter Settings

postFilterSharpening

If you enable this filter, the results are the following: - If the source content is noisy (it contains excessive digital artifacts), the filter cleans up the source. - If the source content is already clean, the filter tends to decrease the bitrate, especially when the rate control mode is QVBR.

Type: TemporalFilterPostFilterSharpening
Required: False

strength

Choose a filter strength. We recommend a strength of 1 or 2. A higher strength might take out good information, resulting in an image that is overly soft.

Type: TemporalFilterStrength
Required: False

TemporalFilterStrength

Temporal Filter Strength

AUTO
STRENGTH_1
STRENGTH_2
STRENGTH_3
STRENGTH_4
STRENGTH_5
STRENGTH_6
STRENGTH_7
STRENGTH_8
STRENGTH_9
STRENGTH_10
STRENGTH_11
STRENGTH_12

STRENGTH_13
STRENGTH_14
STRENGTH_15
STRENGTH_16

TimecodeConfig

Timecode Config

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: TimecodeConfigSource
Required: True

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

TimecodeConfigSource

Timecode Config Source

EMBEDDED
SYSTEMCLOCK
ZEROBASED

TtmlDestinationSettings

Ttml Destination Settings

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: TtmlDestinationStyleControl
Required: False

TtmlDestinationStyleControl

Ttml Destination Style Control

```
PASSTHROUGH
USE_CONFIGURED
```

UdpContainerSettings

Udp Container Settings

m2tsSettings

Type: M2tsSettings
Required: False

UdpGroupSettings

Udp Group Settings

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: InputLossActionForUdpOut
Required: False

timedMetadataId3Frame

Indicates ID3 frame that has the timecode.

Type: UdpTimedMetadataId3Frame
Required: False

timedMetadataId3Period

Timed Metadata interval in seconds.

Type: integer
Required: False
Minimum: 0

UdpOutputSettings

Udp Output Settings

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

containerSettings

Type: UdpContainerSettings
Required: True

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
Required: True

fecOutputSettings

Settings for enabling and adjusting Forward Error Correction on UDP outputs.

Type: FecOutputSettings
Required: False

UdpTimedMetadataId3Frame

Udp Timed Metadata Id3 Frame

```
NONE
PRIV
TDRL
```

VideoBlackFailoverSettings

blackDetectThreshold

A value used in calculating the threshold below which MediaLive considers a pixel to be 'black'. For the input to be considered black, every pixel in a frame must be below this threshold. The threshold is calculated as a percentage (expressed as a decimal) of white. Therefore .1 means 10% white (or 90% black). Note how the formula works for any color depth. For example, if you set this field to 0.1 in 10-bit color depth: (1023*0.1=102.3), which means a pixel value of 102 or less is 'black'. If you set this field to .1 in an 8-bit color depth: (255*0.1=25.5), which means a pixel value of 25 or less is 'black'. The range is 0.0 to 1.0, with any number of decimal places.

Type: number
Required: False
Minimum: 0

Maximum: 1

videoBlackThresholdMsec

The amount of time (in milliseconds) that the active input must be black before automatic input failover occurs.

Type: integer
Required: False
Minimum: 1000

VideoCodecSettings

Video Codec Settings

frameCaptureSettings

Type: FrameCaptureSettings
Required: False

h264Settings

Type: H264Settings
Required: False

h265Settings

Type: H265Settings
Required: False

mpeg2Settings

Type: Mpeg2Settings
Required: False

VideoDescription

Video settings for this stream.

codecSettings

Video codec settings.

Type: VideoCodecSettings
Required: False

height

Output video height, in pixels. Must be an even number. For most codecs, you can leave this field and width blank in order to use the height and width (resolution) from the source. Note, however, that leaving blank is not recommended. For the Frame Capture codec, height and width are required.

Type: integer
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

respondToAfd

Indicates how MediaLive will respond to the AFD values that might be in the input video. If you do not know what AFD signaling is, or if your downstream system has not given you guidance, choose PASSTHROUGH. RESPOND: MediaLive clips the input video using a formula that uses the AFD values (configured in afdSignaling), the input display aspect ratio, and the output display aspect ratio. MediaLive also includes the AFD values in the output, unless the codec for this encode is FRAME_CAPTURE. PASSTHROUGH: MediaLive ignores the AFD values and does not clip the video. But MediaLive does include the values in the output. NONE: MediaLive does not clip the input video and does not include the AFD values in the output

Type: VideoDescriptionRespondToAfd
Required: False

scalingBehavior

STRETCH_TO_OUTPUT configures the output position to stretch the video to the specified output resolution (height and width). This option will override any position value. DEFAULT may insert black boxes (pillar boxes or letter boxes) around the video to provide the specified output resolution.

Type: VideoDescriptionScalingBehavior
Required: False

sharpness

Changes the strength of the anti-alias filter used for scaling. 0 is the softest setting, 100 is the sharpest. A setting of 50 is recommended for most content.

Type: integer
Required: False
Minimum: 0
Maximum: 100

width

Output video width, in pixels. Must be an even number. For most codecs, you can leave this field and height blank in order to use the height and width (resolution) from the source. Note, however, that leaving blank is not recommended. For the Frame Capture codec, height and width are required.

Type: integer
Required: False

VideoDescriptionRespondToAfd

Video Description Respond To Afd

NONE
PASSTHROUGH
RESPOND

VideoDescriptionScalingBehavior

Video Description Scaling Behavior

DEFAULT
STRETCH_TO_OUTPUT

VideoSelector

Specifies a particular video stream within an input source. An input may have only a single video selector.

colorSpace

Specifies the color space of an input. This setting works in tandem with colorSpaceUsage and a video description's colorSpaceSettingsChoice to determine if any conversion will be performed.

Type: VideoSelectorColorSpace
Required: False

colorSpaceSettings

Color space settings

Type: VideoSelectorColorSpaceSettings
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: VideoSelectorColorSpaceUsage
Required: False

selectorSettings

The video selector settings.

Type: VideoSelectorSettings
Required: False

VideoSelectorColorSpace

Video Selector Color Space

FOLLOW
HDR10
HLG_2020
REC_601
REC_709

VideoSelectorColorSpaceSettings

Video Selector Color Space Settings

hdr10Settings

Type: Hdr10Settings
Required: False

VideoSelectorColorSpaceUsage

Video Selector Color Space Usage

FALLBACK
FORCE

VideoSelectorPid

Video Selector Pid

pid

Selects a specific PID from within a video source.

Type: integer
Required: False
Minimum: 0
Maximum: 8191

VideoSelectorProgramId

Video Selector Program Id

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

Video Selector Settings

videoSelectorPid

Type: VideoSelectorPid
Required: False

videoSelectorProgramId

Type: VideoSelectorProgramId
Required: False

VpcOutputSettingsDescription

The properties for a private VPC Output

availabilityZones

The Availability Zones where the vpc subnets are located. The first Availability Zone applies to the first subnet in the list of subnets. The second Availability Zone applies to the second subnet.

Type: Array of type string
Required: False

networkInterfaceIds

A list of Elastic Network Interfaces created by MediaLive in the customer's VPC

Type: Array of type string
Required: False

securityGroupIds

A list of up EC2 VPC security group IDs attached to the Output VPC network interfaces.

Type: Array of type string
Required: False

subnetIds

A list of VPC subnet IDs from the same VPC. If STANDARD channel, subnet IDs must be mapped to two unique availability zones (AZ).

Type: Array of type string
Required: False

WavCodingMode

Wav Coding Mode

```
CODING_MODE_1_0
```

CODING_MODE_2_0
CODING_MODE_4_0
CODING_MODE_8_0

WavSettings

Wav Settings

bitDepth

Bits per sample.

Type: number
Required: False

codingMode

The audio coding mode for the WAV audio. The mode determines the number of channels in the audio.

Type: WavCodingMode
Required: False

sampleRate

Sample rate in Hz.

Type: number
Required: False

WebvttDestinationSettings

Webvtt Destination Settings

styleControl

Controls whether the color and position of the source captions is passed through to the WebVTT output captions. PASSTHROUGH - Valid only if the source captions are EMBEDDED or TELETEXT.

NO_STYLE_DATA - Don't pass through the style. The output captions will not contain any font styling information.

Type: WebvttDestinationStyleControl
Required: False

WebvttDestinationStyleControl

Webvtt Destination Style Control

```
NO_STYLE_DATA
PASSTHROUGH
```

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

StopChannel

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Input devices

URI

```
/prod/inputDevices
```

HTTP methods

GET

Operation ID: `ListInputDevices`

Query parameters

Name	Type	Required	Description
`nextToken`	String	False	
`maxResults`	String	False	

Responses

Status code	Response model	Description
200	ListInputDevicesResultModel	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

Schemas

Response bodies

ListInputDevicesResultModel schema

```
{
  "inputDevices": [
    {
      "arn": "string",
      "availabilityZone": "string",
      "connectionState": enum,
      "deviceSettingsSyncState": enum,
      "deviceUpdateStatus": enum,
      "hdDeviceSettings": {
        "activeInput": enum,
        "configuredInput": enum,
        "deviceState": enum,
        "framerate": number,
        "height": integer,
        "maxBitrate": integer,
        "scanType": enum,
        "width": integer
      },
      "id": "string",
      "macAddress": "string",
      "name": "string",
      "networkSettings": {
        "dnsAddresses": [
          "string"
        ],
        "gateway": "string",
        "ipAddress": "string",
        "ipScheme": enum,
        "subnetMask": "string"
      },
      "serialNumber": "string",
      "type": enum,
      "uhdDeviceSettings": {
        "activeInput": enum,
        "configuredInput": enum,
        "deviceState": enum,
        "framerate": number,
        "height": integer,
        "maxBitrate": integer,
        "scanType": enum,
        "width": integer
      }
```

```
    }
  ],
  "nextToken": "string"
}
```

InvalidRequest schema

```
{
  "message": "string"
}
```

AccessDenied schema

```
{
  "message": "string"
}
```

LimitExceeded schema

```
{
  "message": "string"
}
```

InternalServiceError schema

```
{
  "message": "string"
}
```

BadGatewayException schema

```
{
  "message": "string"
}
```

GatewayTimeoutException schema

```
{
  "message": "string"
}
```

Properties

AccessDenied

message

Type: string
Required: False

BadGatewayException

message

Type: string
Required: False

DeviceSettingsSyncState

The status of the action to synchronize the device configuration. If you change the configuration of the input device (for example, the maximum bitrate), MediaLive sends the new data to the device. The device might not update itself immediately. SYNCED means the device has updated its configuration. SYNCING means that it has not updated its configuration.

```
SYNCED
SYNCING
```

DeviceUpdateStatus

The status of software on the input device.

```
UP_TO_DATE
NOT_UP_TO_DATE
UPDATING
```

GatewayTimeoutException

message

Type: string
Required: False

InputDeviceActiveInput

The source at the input device that is currently active.

```
HDMI
SDI
```

InputDeviceConfiguredInput

The source to activate (use) from the input device.

```
AUTO
HDMI
SDI
```

InputDeviceConnectionState

The state of the connection between the input device and AWS.

```
DISCONNECTED
CONNECTED
```

InputDeviceHdSettings

Settings that describe the active source from the input device, and the video characteristics of that source.

activeInput

If you specified Auto as the configured input, specifies which of the sources is currently active (SDI or HDMI).

Type: InputDeviceActiveInput
Required: False

configuredInput

The source at the input device that is currently active. You can specify this source.

Type: InputDeviceConfiguredInput
Required: False

deviceState

The state of the input device.

Type: InputDeviceState
Required: False

framerate

The frame rate of the video source.

Type: number
Required: False

height

The height of the video source, in pixels.

Type: integer
Required: False

maxBitrate

The current maximum bitrate for ingesting this source, in bits per second. You can specify this maximum.

Type: integer
Required: False

scanType

The scan type of the video source.

Type: InputDeviceScanType
Required: False

width

The width of the video source, in pixels.

Type: integer
Required: False

InputDeviceIpScheme

Specifies whether the input device has been configured (outside of MediaLive) to use a dynamic IP address assignment (DHCP) or a static IP address.

`STATIC`
`DHCP`

InputDeviceNetworkSettings

The network settings for the input device.

dnsAddresses

The DNS addresses of the input device.

Type: Array of type string
Required: False

gateway

The network gateway IP address.

Type: string

Required: False

ipAddress

The IP address of the input device.

Type: string
Required: False

ipScheme

Specifies whether the input device has been configured (outside of MediaLive) to use a dynamic IP address assignment (DHCP) or a static IP address.

Type: InputDeviceIpScheme
Required: False

subnetMask

The subnet mask of the input device.

Type: string
Required: False

InputDeviceScanType

The scan type of the video source.

```
INTERLACED
PROGRESSIVE
```

InputDeviceState

The state of the input device.

```
IDLE
STREAMING
```

InputDeviceSummary

Details of the input device.

arn

The unique ARN of the input device.

Type: string
Required: False

availabilityZone

The Availability Zone associated with this input device.

Type: string
Required: False

connectionState

The state of the connection between the input device and AWS.

Type: InputDeviceConnectionState
Required: False

deviceSettingsSyncState

The status of the action to synchronize the device configuration. If you change the configuration of the input device (for example, the maximum bitrate), MediaLive sends the new data to the device. The device might not update itself immediately. SYNCED means the device has updated its configuration. SYNCING means that it has not updated its configuration.

Type: DeviceSettingsSyncState
Required: False

deviceUpdateStatus

The status of software on the input device.

Type: DeviceUpdateStatus
Required: False

hdDeviceSettings

Settings that describe an input device that is type HD.

Type: InputDeviceHdSettings
Required: False

id

The unique ID of the input device.

Type: string
Required: False

macAddress

The network MAC address of the input device.

Type: string
Required: False

name

A name that you specify for the input device.

Type: string
Required: False

networkSettings

Network settings for the input device.

Type: InputDeviceNetworkSettings
Required: False

serialNumber

The unique serial number of the input device.

Type: string
Required: False

type

The type of the input device.

Type: InputDeviceType
Required: False

uhdDeviceSettings

Settings that describe an input device that is type UHD.

Type: InputDeviceUhdSettings
Required: False

InputDeviceType

The type of the input device. For an AWS Elemental Link device that outputs resolutions up to 1080, choose "HD".

HD

UHD

InputDeviceUhdSettings

Settings that describe the active source from the input device, and the video characteristics of that source.

activeInput

If you specified Auto as the configured input, specifies which of the sources is currently active (SDI or HDMI).

Type: InputDeviceActiveInput
Required: False

configuredInput

The source at the input device that is currently active. You can specify this source.

Type: InputDeviceConfiguredInput
Required: False

deviceState

The state of the input device.

Type: InputDeviceState
Required: False

framerate

The frame rate of the video source.

Type: number
Required: False

height

The height of the video source, in pixels.

Type: integer
Required: False

maxBitrate

The current maximum bitrate for ingesting this source, in bits per second. You can specify this maximum.

Type: integer
Required: False

scanType

The scan type of the video source.

Type: InputDeviceScanType
Required: False

width

The width of the video source, in pixels.

Type: integer
Required: False

InternalServiceError

message

Type: string
Required: False

InvalidRequest

message

Type: string
Required: False

LimitExceeded

message

Type: string
Required: False

ListInputDevicesResultModel

The list of input devices owned by the AWS account.

inputDevices

The list of input devices.

Type: Array of type InputDeviceSummary
Required: False

nextToken

A token to get additional list results.

Type: string
Required: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

ListInputDevices

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Input devices: accept device transfer

URI

`/prod/inputDevices/inputDeviceId/accept`

HTTP methods

POST

Operation ID: `AcceptInputDeviceTransfer`

Path parameters

Name	Type	Required	Description
inputDeviceId	String	True	

Responses

Status code	Response model	Description
200	None	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
409	ResourceConflict	409 response
422	ValidationError	422 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

Schemas

Response bodies

InvalidRequest schema

```
{
  "message": "string"
}
```

AccessDenied schema

```
{
  "message": "string"
}
```

ResourceNotFound schema

```
{
  "message": "string"
}
```

ResourceConflict schema

```
{
  "message": "string"
}
```

ValidationError schema

```
{
  "elementPath": "string",
  "errorMessage": "string"
}
```

LimitExceeded schema

```
{
  "message": "string"
}
```

InternalServiceError schema

```
{
  "message": "string"
}
```

BadGatewayException schema

```
{
  "message": "string"
}
```

GatewayTimeoutException schema

```
{
  "message": "string"
}
```

Properties

AccessDenied

message

Type: string
Required: False

BadGatewayException

message

Type: string
Required: False

GatewayTimeoutException

message

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

ValidationError

elementPath

Path to the source of the error.

Type: string

Required: False

errorMessage

The error message.

Type: string

Required: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

AcceptInputDeviceTransfer

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Input devices: cancel device transfer

URI

/prod/inputDevices/*inputDeviceId*/cancel

HTTP methods

POST

Operation ID: CancelInputDeviceTransfer

Path parameters

Name	Type	Required	Description
inputDeviceId	String	True	

Responses

Status code	Response model	Description
200	None	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
409	ResourceConflict	409 response
422	ValidationError	422 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response

Status code	Response model	Description
504	GatewayTimeoutException	504 response

Schemas

Response bodies

InvalidRequest schema

```
{
  "message": "string"
}
```

AccessDenied schema

```
{
  "message": "string"
}
```

ResourceNotFound schema

```
{
  "message": "string"
}
```

ResourceConflict schema

```
{
  "message": "string"
}
```

ValidationError schema

```
{
```

```
  "elementPath": "string",
  "errorMessage": "string"
}
```

LimitExceeded schema

```
{
  "message": "string"
}
```

InternalServiceError schema

```
{
  "message": "string"
}
```

BadGatewayException schema

```
{
  "message": "string"
}
```

GatewayTimeoutException schema

```
{
  "message": "string"
}
```

Properties

AccessDenied

message

Type: string

Required: False

BadGatewayException

message

Type: string
Required: False

GatewayTimeoutException

message

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

ValidationError

elementPath

Path to the source of the error.

Type: string
Required: False

errorMessage

The error message.

Type: string
Required: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

CancelInputDeviceTransfer

- AWS Command Line Interface
- AWS SDK for .NET

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Input devices: claim device

URI

`/prod/claimDevice`

HTTP methods

POST

Operation ID: `ClaimDevice`

Responses

Status code	Response model	Description
200	None	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
422	ValidationError	422 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response

Status code	Response model	Description
504	GatewayTimeoutException	504 response

Schemas

Request bodies

POST schema

```
{
  "id": "string"
}
```

Response bodies

InvalidRequest schema

```
{
  "message": "string"
}
```

AccessDenied schema

```
{
  "message": "string"
}
```

ResourceNotFound schema

```
{
  "message": "string"
}
```

ValidationError schema

```
{
```

```
  "elementPath": "string",
  "errorMessage": "string"
}
```

LimitExceeded schema

```
{
  "message": "string"
}
```

InternalServiceError schema

```
{
  "message": "string"
}
```

BadGatewayException schema

```
{
  "message": "string"
}
```

GatewayTimeoutException schema

```
{
  "message": "string"
}
```

Properties

AccessDenied

message

Type: string

Required: False

BadGatewayException

message

Type: string
Required: False

ClaimDeviceRequest

Request to claim an AWS Elemental device that you have purchased from a third-party vendor.

id

The id of the device you want to claim.

Type: string
Required: False

GatewayTimeoutException

message

Type: string
Required: False

InternalServiceError

message

Type: string
Required: False

InvalidRequest

message

Type: string
Required: False

LimitExceeded

message

Type: string
Required: False

ResourceNotFound

message

Type: string
Required: False

ValidationError

elementPath

Path to the source of the error.

Type: string
Required: False

errorMessage

The error message.

Type: string
Required: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

ClaimDevice

- AWS Command Line Interface
- AWS SDK for .NET

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Input devices: device ID

URI

`/prod/inputDevices/inputDeviceId`

HTTP methods

GET

Operation ID: `DescribeInputDevice`

Path parameters

Name	Type	Required	Description
inputDeviceId	String	True	

Responses

Status code	Response model	Description
200	InputDevice	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response

Status code	Response model	Description
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

PUT

Operation ID: UpdateInputDevice

Path parameters

Name	Type	Required	Description
inputDeviceId	String	True	

Responses

Status code	Response model	Description
200	InputDevice	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
422	InputDeviceConfigurationValidationError	422 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response

Status code	Response model	Description
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

Schemas

Request bodies

PUT schema

```
{
  "availabilityZone": "string",
  "hdDeviceSettings": {
    "configuredInput": enum,
    "maxBitrate": integer
  },
  "name": "string",
  "uhdDeviceSettings": {
    "configuredInput": enum,
    "maxBitrate": integer
  }
}
```

Response bodies

InputDevice schema

```
{
  "arn": "string",
  "availabilityZone": "string",
  "connectionState": enum,
  "deviceSettingsSyncState": enum,
  "deviceUpdateStatus": enum,
  "hdDeviceSettings": {
    "activeInput": enum,
    "configuredInput": enum,
    "deviceState": enum,
    "framerate": number,
```

```
    "height": integer,
    "maxBitrate": integer,
    "scanType": enum,
    "width": integer
  },
  "id": "string",
  "macAddress": "string",
  "name": "string",
  "networkSettings": {
    "dnsAddresses": [
      "string"
    ],
    "gateway": "string",
    "ipAddress": "string",
    "ipScheme": enum,
    "subnetMask": "string"
  },
  "serialNumber": "string",
  "type": enum,
  "uhdDeviceSettings": {
    "activeInput": enum,
    "configuredInput": enum,
    "deviceState": enum,
    "framerate": number,
    "height": integer,
    "maxBitrate": integer,
    "scanType": enum,
    "width": integer
  }
}
```

InvalidRequest schema

```
{
  "message": "string"
}
```

AccessDenied schema

```
{
  "message": "string"
```

```
}
```

ResourceNotFound schema

```
{
  "message": "string"
}
```

InputDeviceConfigurationValidationError schema

```
{
  "message": "string",
  "validationErrors": [
    {
      "elementPath": "string",
      "errorMessage": "string"
    }
  ]
}
```

LimitExceeded schema

```
{
  "message": "string"
}
```

InternalServiceError schema

```
{
  "message": "string"
}
```

BadGatewayException schema

```
{
  "message": "string"
}
```

GatewayTimeoutException schema

```
{
  "message": "string"
}
```

Properties

AccessDenied

message

Type: string
Required: False

BadGatewayException

message

Type: string
Required: False

DeviceSettingsSyncState

The status of the action to synchronize the device configuration. If you change the configuration of the input device (for example, the maximum bitrate), MediaLive sends the new data to the device. The device might not update itself immediately. SYNCED means the device has updated its configuration. SYNCING means that it has not updated its configuration.

```
SYNCED
SYNCING
```

DeviceUpdateStatus

The status of software on the input device.

```
UP_TO_DATE
NOT_UP_TO_DATE
UPDATING
```

GatewayTimeoutException

message

Type: string
Required: False

InputDevice

An input device.

arn

The unique ARN of the input device.

Type: string
Required: False

availabilityZone

The Availability Zone associated with this input device.

Type: string
Required: False

connectionState

The state of the connection between the input device and AWS.

Type: InputDeviceConnectionState
Required: False

deviceSettingsSyncState

The status of the action to synchronize the device configuration. If you change the configuration of the input device (for example, the maximum bitrate), MediaLive sends the new data to the device. The device might not update itself immediately. SYNCED means the device has updated its configuration. SYNCING means that it has not updated its configuration.

Type: DeviceSettingsSyncState
Required: False

deviceUpdateStatus

The status of software on the input device.

Type: DeviceUpdateStatus
Required: False

hdDeviceSettings

Settings that describe an input device that is type HD.

Type: InputDeviceHdSettings
Required: False

id

The unique ID of the input device.

Type: string
Required: False

macAddress

The network MAC address of the input device.

Type: string
Required: False

name

A name that you specify for the input device.

Type: string
Required: False

networkSettings

The network settings for the input device.

Type: InputDeviceNetworkSettings
Required: False

serialNumber

The unique serial number of the input device.

Type: string
Required: False

type

The type of the input device.

Type: InputDeviceType
Required: False

uhdDeviceSettings

Settings that describe an input device that is type UHD.

Type: InputDeviceUhdSettings
Required: False

InputDeviceActiveInput

The source at the input device that is currently active.

```
HDMI
SDI
```

InputDeviceConfigurableSettings

Configurable settings for the input device.

configuredInput

The input source that you want to use. If the device has a source connected to only one of its input ports, or if you don't care which source the device sends, specify Auto. If the device has sources connected to both its input ports, and you want to use a specific source, specify the source.

Type: InputDeviceConfiguredInput
Required: False

maxBitrate

The maximum bitrate in bits per second. Set a value here to throttle the bitrate of the source video.

Type: integer
Required: False

InputDeviceConfigurationValidationError

message

The error message.

Type: string
Required: False

validationErrors

A collection of validation error responses.

Type: Array of type ValidationError
Required: False

InputDeviceConfiguredInput

The source to activate (use) from the input device.

AUTO
HDMI

SDI

InputDeviceConnectionState

The state of the connection between the input device and AWS.

DISCONNECTED
CONNECTED

InputDeviceHdSettings

Settings that describe the active source from the input device, and the video characteristics of that source.

activeInput

If you specified Auto as the configured input, specifies which of the sources is currently active (SDI or HDMI).

Type: InputDeviceActiveInput
Required: False

configuredInput

The source at the input device that is currently active. You can specify this source.

Type: InputDeviceConfiguredInput
Required: False

deviceState

The state of the input device.

Type: InputDeviceState
Required: False

framerate

The frame rate of the video source.

Type: number
Required: False

height

The height of the video source, in pixels.

Type: integer
Required: False

maxBitrate

The current maximum bitrate for ingesting this source, in bits per second. You can specify this maximum.

Type: integer
Required: False

scanType

The scan type of the video source.

Type: InputDeviceScanType
Required: False

width

The width of the video source, in pixels.

Type: integer
Required: False

InputDeviceIpScheme

Specifies whether the input device has been configured (outside of MediaLive) to use a dynamic IP address assignment (DHCP) or a static IP address.

```
STATIC
```

DHCP

InputDeviceNetworkSettings

The network settings for the input device.

dnsAddresses

The DNS addresses of the input device.

Type: Array of type string
Required: False

gateway

The network gateway IP address.

Type: string
Required: False

ipAddress

The IP address of the input device.

Type: string
Required: False

ipScheme

Specifies whether the input device has been configured (outside of MediaLive) to use a dynamic IP address assignment (DHCP) or a static IP address.

Type: InputDeviceIpScheme
Required: False

subnetMask

The subnet mask of the input device.

Type: string

Required: False

InputDeviceScanType

The scan type of the video source.

`INTERLACED`

`PROGRESSIVE`

InputDeviceState

The state of the input device.

`IDLE`

`STREAMING`

InputDeviceType

The type of the input device. For an AWS Elemental Link device that outputs resolutions up to 1080, choose "HD".

`HD`

`UHD`

InputDeviceUhdSettings

Settings that describe the active source from the input device, and the video characteristics of that source.

activeInput

If you specified Auto as the configured input, specifies which of the sources is currently active (SDI or HDMI).

Type: InputDeviceActiveInput

Required: False

configuredInput

The source at the input device that is currently active. You can specify this source.

Type: InputDeviceConfiguredInput
Required: False

deviceState

The state of the input device.

Type: InputDeviceState
Required: False

framerate

The frame rate of the video source.

Type: number
Required: False

height

The height of the video source, in pixels.

Type: integer
Required: False

maxBitrate

The current maximum bitrate for ingesting this source, in bits per second. You can specify this maximum.

Type: integer
Required: False

scanType

The scan type of the video source.

Type: InputDeviceScanType
Required: False

width

The width of the video source, in pixels.

Type: integer
Required: False

InternalServiceError

message

Type: string
Required: False

InvalidRequest

message

Type: string
Required: False

LimitExceeded

message

Type: string
Required: False

ResourceNotFound

message

Type: string
Required: False

UpdateInputDevice

Updates an input device.

availabilityZone

The Availability Zone you want to associate with this input device.

Type: string
Required: False

hdDeviceSettings

The settings that you want to apply to the HD input device.

Type: InputDeviceConfigurableSettings
Required: False

name

The name that you assigned to this input device (not the unique ID).

Type: string
Required: False

uhdDeviceSettings

The settings that you want to apply to the UHD input device.

Type: InputDeviceConfigurableSettings
Required: False

ValidationError

elementPath

Path to the source of the error.

Type: string
Required: False

errorMessage

The error message.

Type: string
Required: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

DescribeInputDevice

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UpdateInputDevice

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3

- AWS SDK for Python
- AWS SDK for Ruby V3

Input devices: input device transfers

URI

`/prod/inputDeviceTransfers`

HTTP methods

GET

Operation ID: `ListInputDeviceTransfers`

Query parameters

Name	Type	Required	Description
`transferType`	String	True	
`nextToken`	String	False	
`maxResults`	String	False	

Responses

Status code	Response model	Description
200	`ListInputDeviceTransfersResultModel`	200 response
400	`InvalidRequest`	400 response
403	`AccessDenied`	403 response
422	`ValidationError`	422 response
429	`LimitExceeded`	429 response

Status code	Response model	Description
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

Schemas

Response bodies

ListInputDeviceTransfersResultModel schema

```
{
  "inputDeviceTransfers": [
    {
      "id": "string",
      "message": "string",
      "targetCustomerId": "string",
      "transferType": enum
    }
  ],
  "nextToken": "string"
}
```

InvalidRequest schema

```
{
  "message": "string"
}
```

AccessDenied schema

```
{
  "message": "string"
}
```

ValidationError schema

```
{
  "elementPath": "string",
  "errorMessage": "string"
}
```

LimitExceeded schema

```
{
  "message": "string"
}
```

InternalServiceError schema

```
{
  "message": "string"
}
```

BadGatewayException schema

```
{
  "message": "string"
}
```

GatewayTimeoutException schema

```
{
  "message": "string"
}
```

Properties

AccessDenied

message

Type: string

Required: False

BadGatewayException

message

Type: string
Required: False

GatewayTimeoutException

message

Type: string
Required: False

InputDeviceTransferType

The type of device transfer. INCOMING for an input device that is being transferred to you, OUTGOING for an input device that you are transferring to another AWS account.

`OUTGOING`
`INCOMING`

InternalServiceError

message

Type: string
Required: False

InvalidRequest

message

Type: string
Required: False

LimitExceeded

message

Type: string
Required: False

ListInputDeviceTransfersResultModel

The list of input devices in the transferred state. The recipient hasn't yet accepted or rejected the transfer.

inputDeviceTransfers

The list of devices that you are transferring or are being transferred to you.

Type: Array of type TransferringInputDeviceSummary
Required: False

nextToken

A token to get additional list results.

Type: string
Required: False

TransferringInputDeviceSummary

Details about the input device that is being transferred.

id

The unique ID of the input device.

Type: string
Required: False

message

The optional message that the sender has attached to the transfer.

Type: string
Required: False

targetCustomerId

The AWS account ID for the recipient of the input device transfer.

Type: string
Required: False

transferType

The type (direction) of the input device transfer.

Type: InputDeviceTransferType
Required: False

ValidationError

elementPath

Path to the source of the error.

Type: string
Required: False

errorMessage

The error message.

Type: string
Required: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

ListInputDeviceTransfers

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Input devices: reboot

URI

`/prod/inputDevices/inputDeviceId/reboot`

HTTP methods

POST

Operation ID: `RebootInputDevice`

Path parameters

Name	Type	Required	Description
inputDeviceId	String	True	

Responses

Status code	Response model	Description
200	None	200 response
400	InvalidRequest	400 response

Status code	Response model	Description
403	AccessDenied	403 response
404	ResourceNotFound	404 response
422	ValidationError	422 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

Schemas

Request bodies

POST schema

```
{
  "force": enum
}
```

Response bodies

InvalidRequest schema

```
{
  "message": "string"
}
```

AccessDenied schema

```
{
  "message": "string"
```

```
}
```

ResourceNotFound schema

```
{
  "message": "string"
}
```

ValidationError schema

```
{
  "elementPath": "string",
  "errorMessage": "string"
}
```

LimitExceeded schema

```
{
  "message": "string"
}
```

InternalServiceError schema

```
{
  "message": "string"
}
```

BadGatewayException schema

```
{
  "message": "string"
}
```

GatewayTimeoutException schema

```
{
```

```
  "message": "string"
}
```

Properties

AccessDenied

message

Type: string
Required: False

BadGatewayException

message

Type: string
Required: False

GatewayTimeoutException

message

Type: string
Required: False

InternalServiceError

message

Type: string
Required: False

InvalidRequest

message

Type: string

Required: False

LimitExceeded

message

Type: string
Required: False

RebootInputDevice

force

Force a reboot of an input device. If the device is streaming, it will stop streaming and begin rebooting within a few seconds of sending the command. If the device was streaming prior to the reboot, the device will resume streaming when the reboot completes.

Type: RebootInputDeviceForce
Required: False

RebootInputDeviceForce

Whether or not to force reboot the input device.

NO
YES

ResourceNotFound

message

Type: string
Required: False

ValidationError

elementPath

Path to the source of the error.

Type: string

Required: False

errorMessage

The error message.

Type: string

Required: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

RebootInputDevice

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Input devices: reject device transfer

URI

`/prod/inputDevices/inputDeviceId/reject`

HTTP methods

POST

Operation ID: `RejectInputDeviceTransfer`

Path parameters

Name	Type	Required	Description
inputDeviceId	String	True	

Responses

Status code	Response model	Description
200	None	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
409	ResourceConflict	409 response
422	ValidationError	422 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

Schemas

Response bodies

InvalidRequest schema

```
{
  "message": "string"
}
```

AccessDenied schema

```
{
  "message": "string"
}
```

ResourceNotFound schema

```
{
  "message": "string"
}
```

ResourceConflict schema

```
{
  "message": "string"
}
```

ValidationError schema

```
{
  "elementPath": "string",
  "errorMessage": "string"
}
```

LimitExceeded schema

```
{
  "message": "string"
}
```

InternalServiceError schema

```
{
  "message": "string"
}
```

BadGatewayException schema

```
{
  "message": "string"
}
```

GatewayTimeoutException schema

```
{
  "message": "string"
}
```

Properties

AccessDenied

message

Type: string
Required: False

BadGatewayException

message

Type: string
Required: False

GatewayTimeoutException

message

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

ValidationError

elementPath

Path to the source of the error.

Type: string

Required: False

errorMessage

The error message.

Type: string

Required: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

RejectInputDeviceTransfer

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Input devices: thumbnail data

URI

/prod/inputDevices/*inputDeviceId*/thumbnailData

HTTP methods

GET

Operation ID: DescribeInputDeviceThumbnail

Path parameters

Name	Type	Required	Description
inputDeviceId	String	True	

Header parameters

Name	Type	Required	Description
Accept	String	True	

Responses

Status code	Response model	Description
200	ThumbnailData	200 response
204	None	204 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
429	LimitExceeded	429 response

Status code	Response model	Description
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

Schemas

Response bodies

ThumbnailData schema

```
{
  "body": "string"
}
```

InvalidRequest schema

```
{
  "message": "string"
}
```

AccessDenied schema

```
{
  "message": "string"
}
```

ResourceNotFound schema

```
{
  "message": "string"
}
```

LimitExceeded schema

```
{
  "message": "string"
}
```

InternalServiceError schema

```
{
  "message": "string"
}
```

BadGatewayException schema

```
{
  "message": "string"
}
```

GatewayTimeoutException schema

```
{
  "message": "string"
}
```

Properties

AccessDenied

message

Type: string
Required: False

BadGatewayException

message

Type: string
Required: False

GatewayTimeoutException

message

Type: string
Required: False

InternalServiceError

message

Type: string
Required: False

InvalidRequest

message

Type: string
Required: False

LimitExceeded

message

Type: string
Required: False

ResourceNotFound

message

Type: string
Required: False

ThumbnailData

The binary data for the thumbnail that the Link device has most recently sent to MediaLive.

body

The binary data for the thumbnail that the Link device has most recently sent to MediaLive.

Type: string
Required: False
Format: byte

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

DescribeInputDeviceThumbnail

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Input devices: transfer device

URI

`/prod/inputDevices/inputDeviceId/transfer`

HTTP methods

POST

Operation ID: `TransferInputDevice`

Path parameters

Name	Type	Required	Description
inputDeviceId	String	True	

Responses

Status code	Response model	Description
200	None	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
409	ResourceConflict	409 response
422	ValidationError	422 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

Schemas

Request bodies

POST schema

```
{
  "targetCustomerId": "string",
  "targetRegion": "string",
  "transferMessage": "string"
```

```
}
```

Response bodies

InvalidRequest schema

```
{
  "message": "string"
}
```

AccessDenied schema

```
{
  "message": "string"
}
```

ResourceNotFound schema

```
{
  "message": "string"
}
```

ResourceConflict schema

```
{
  "message": "string"
}
```

ValidationError schema

```
{
  "elementPath": "string",
  "errorMessage": "string"
}
```

LimitExceeded schema

```
{
```

```
  "message": "string"
}
```

InternalServiceError schema

```
{
  "message": "string"
}
```

BadGatewayException schema

```
{
  "message": "string"
}
```

GatewayTimeoutException schema

```
{
  "message": "string"
}
```

Properties

AccessDenied

message

Type: string

Required: False

BadGatewayException

message

Type: string

Required: False

GatewayTimeoutException

message

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

TransferInputDevice

The transfer details of the input device.

targetCustomerId

The AWS account ID (12 digits) for the recipient of the device transfer.

Type: string
Required: False

targetRegion

The target AWS Region to transfer the device.

Type: string
Required: False

transferMessage

An optional message for the recipient. Maximum 280 characters.

Type: string
Required: False

ValidationError

elementPath

Path to the source of the error.

Type: string
Required: False

errorMessage

The error message.

Type: string
Required: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

TransferInputDevice

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Input devices: update

URI

`/prod/inputDevices/inputDeviceId/startInputDeviceMaintenanceWindow`

HTTP methods

POST

Operation ID: `StartInputDeviceMaintenanceWindow`

Path parameters

Name	Type	Required	Description
inputDeviceId	String	True	

Responses

Status code	Response model	Description
200	None	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
422	ValidationError	422 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

Schemas

Response bodies

InvalidRequest schema

```
{
  "message": "string"
}
```

AccessDenied schema

```
{
  "message": "string"
}
```

ResourceNotFound schema

```
{
  "message": "string"
}
```

ValidationError schema

```
{
  "elementPath": "string",
  "errorMessage": "string"
}
```

LimitExceeded schema

```
{
  "message": "string"
}
```

InternalServiceError schema

```
{
  "message": "string"
}
```

BadGatewayException schema

```
{
  "message": "string"
```

```
}
```

GatewayTimeoutException schema

```
{
  "message": "string"
}
```

Properties

AccessDenied

message

Type: string
Required: False

BadGatewayException

message

Type: string
Required: False

GatewayTimeoutException

message

Type: string
Required: False

InternalServiceError

message

Type: string
Required: False

InvalidRequest

message

Type: string
Required: False

LimitExceeded

message

Type: string
Required: False

ResourceNotFound

message

Type: string
Required: False

ValidationError

elementPath

Path to the source of the error.

Type: string
Required: False

errorMessage

The error message.

Type: string
Required: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

StartInputDeviceMaintenanceWindow

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Input security groups

URI

`/prod/inputSecurityGroups`

HTTP methods

GET

Operation ID: `ListInputSecurityGroups`

Query parameters

Name	Type	Required	Description
`nextToken`	String	False	
`maxResults`	String	False	

Responses

Status code	Response model	Description
200	ListInputSecurityGroupsResultModel	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

POST

Operation ID: `CreateInputSecurityGroup`

Responses

Status code	Response model	Description
200	CreateInputSecurityGroupResultModel	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response

Status code	Response model	Description
504	GatewayTimeoutException	504 response

Schemas

Request bodies

POST schema

```
{
  "tags": {
  },
  "whitelistRules": [
    {
      "cidr": "string"
    }
  ]
}
```

Response bodies

ListInputSecurityGroupsResultModel schema

```
{
  "inputSecurityGroups": [
    {
      "arn": "string",
      "id": "string",
      "inputs": [
        "string"
      ],
      "state": enum,
      "tags": {
      },
      "whitelistRules": [
        {
          "cidr": "string"
        }
      ]
```

```
    }
  ],
  "nextToken": "string"
}
```

CreateInputSecurityGroupResultModel schema

```
{
  "securityGroup": {
    "arn": "string",
    "id": "string",
    "inputs": [
      "string"
    ],
    "state": enum,
    "tags": {
    },
    "whitelistRules": [
      {
        "cidr": "string"
      }
    ]
  }
}
```

InvalidRequest schema

```
{
  "message": "string"
}
```

AccessDenied schema

```
{
  "message": "string"
}
```

LimitExceeded schema

```
{
```

```
  "message": "string"
}
```

InternalServiceError schema

```
{
  "message": "string"
}
```

BadGatewayException schema

```
{
  "message": "string"
}
```

GatewayTimeoutException schema

```
{
  "message": "string"
}
```

Properties

AccessDenied

message

Type: string

Required: False

BadGatewayException

message

Type: string

Required: False

CreateInputSecurityGroupResultModel

securityGroup

Type: InputSecurityGroup
Required: False

GatewayTimeoutException

message

Type: string
Required: False

InputSecurityGroup

An Input Security Group

arn

Unique ARN of Input Security Group

Type: string
Required: False

id

The Id of the Input Security Group

Type: string
Required: False

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: InputSecurityGroupState
Required: False

tags

A collection of key-value pairs.

Type: Tags
Required: False

whitelistRules

Whitelist rules and their sync status

Type: Array of type InputWhitelistRule
Required: False

InputSecurityGroupState

```
IDLE
IN_USE
UPDATING
DELETED
```

InputSecurityGroupWhitelistRequest

Request of IPv4 CIDR addresses to whitelist in a security group.

tags

A collection of key-value pairs.

Type: Tags
Required: False

whitelistRules

List of IPv4 CIDR addresses to whitelist

Type: Array of type InputWhitelistRuleCidr
Required: False

InputWhitelistRule

Whitelist rule

cidr

The IPv4 CIDR that's whitelisted.

Type: string
Required: False

InputWhitelistRuleCidr

An IPv4 CIDR to whitelist.

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

ListInputSecurityGroupsResultModel

Result of input security group list request

inputSecurityGroups

List of input security groups

Type: Array of type InputSecurityGroup
Required: False

nextToken

Type: string
Required: False

Tags

key-value pairs

Type: string

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

ListInputSecurityGroups

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

CreateInputSecurityGroup

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Input security groups: group ID

URI

`/prod/inputSecurityGroups/`*`inputSecurityGroupId`*

HTTP methods

DELETE

Operation ID: `DeleteInputSecurityGroup`

Path parameters

Name	Type	Required	Description
inputSecurityGroupId	String	True	

Responses

Status code	Response model	Description
200	Empty	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

GET

Operation ID: `DescribeInputSecurityGroup`

Path parameters

Name	Type	Required	Description
inputSecurityGroupId	String	True	

Responses

Status code	Response model	Description
200	InputSecurityGroup	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

PUT

Operation ID: UpdateInputSecurityGroup

Path parameters

Name	Type	Required	Description
inputSecurityGroupId	String	True	

Responses

Status code	Response model	Description
200	UpdateInputSecurityGroupResultModel	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
409	ResourceConflict	409 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

Schemas

Request bodies

PUT schema

```
{
  "tags": {
  },
  "whitelistRules": [
    {
      "cidr": "string"
    }
  ]
}
```

Response bodies

Empty schema

```
{
}
```

InputSecurityGroup schema

```
{
  "arn": "string",
  "id": "string",
  "inputs": [
    "string"
  ],
  "state": enum,
  "tags": {
  },
  "whitelistRules": [
    {
      "cidr": "string"
    }
  ]
}
```

UpdateInputSecurityGroupResultModel schema

```
{
  "securityGroup": {
    "arn": "string",
    "id": "string",
    "inputs": [
      "string"
    ],
    "state": enum,
    "tags": {
    },
    "whitelistRules": [
      {
        "cidr": "string"
      }
    ]
  }
```

```
}
```

InvalidRequest schema

```
{
  "message": "string"
}
```

AccessDenied schema

```
{
  "message": "string"
}
```

ResourceNotFound schema

```
{
  "message": "string"
}
```

ResourceConflict schema

```
{
  "message": "string"
}
```

LimitExceeded schema

```
{
  "message": "string"
}
```

InternalServiceError schema

```
{
```

```
  "message": "string"
}
```

BadGatewayException schema

```
{
  "message": "string"
}
```

GatewayTimeoutException schema

```
{
  "message": "string"
}
```

Properties

AccessDenied

message

Type: string
Required: False

BadGatewayException

message

Type: string
Required: False

Empty

GatewayTimeoutException

message

Type: string

Required: False

InputSecurityGroup

An Input Security Group

arn

Unique ARN of Input Security Group

Type: string
Required: False

id

The Id of the Input Security Group

Type: string
Required: False

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: InputSecurityGroupState
Required: False

tags

A collection of key-value pairs.

Type: Tags

Required: False

whitelistRules

Whitelist rules and their sync status

Type: Array of type InputWhitelistRule
Required: False

InputSecurityGroupState

```
IDLE
IN_USE
UPDATING
DELETED
```

InputSecurityGroupWhitelistRequest

Request of IPv4 CIDR addresses to whitelist in a security group.

tags

A collection of key-value pairs.

Type: Tags
Required: False

whitelistRules

List of IPv4 CIDR addresses to whitelist

Type: Array of type InputWhitelistRuleCidr
Required: False

InputWhitelistRule

Whitelist rule

cidr

The IPv4 CIDR that's whitelisted.

Type: string
Required: False

InputWhitelistRuleCidr

An IPv4 CIDR to whitelist.

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

Tags

key-value pairs

Type: string

UpdateInputSecurityGroupResultModel

securityGroup

Type: InputSecurityGroup
Required: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

DeleteInputSecurityGroup

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++

- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

DescribeInputSecurityGroup

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UpdateInputSecurityGroup

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Inputs

URI

`/prod/inputs`

HTTP methods

GET

Operation ID: `ListInputs`

Query parameters

Name	Type	Required	Description
`nextToken`	String	False	
`maxResults`	String	False	

Responses

Status code	Response model	Description
200	`ListInputsResultModel`	200 response
400	`InvalidRequest`	400 response
403	`AccessDenied`	403 response
429	`LimitExceeded`	429 response
500	`InternalServiceError`	500 response
502	`BadGatewayException`	502 response
504	`GatewayTimeoutException`	504 response

POST

Operation ID: `CreateInput`

Responses

Status code	Response model	Description
201	CreateInputResultModel	201 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

Schemas

Request bodies

POST schema

```
{
  "destinations": [
    {
      "streamName": "string"
    }
  ],
  "inputDevices": [
    {
      "id": "string"
    }
  ],
  "inputSecurityGroups": [
```

```
    "string"
  ],
  "mediaConnectFlows": [
    {
      "flowArn": "string"
    }
  ],
  "name": "string",
  "requestId": "string",
  "roleArn": "string",
  "sources": [
    {
      "passwordParam": "string",
      "url": "string",
      "username": "string"
    }
  ],
  "tags": {
  },
  "type": enum,
  "vpc": {
    "securityGroupIds": [
      "string"
    ],
    "subnetIds": [
      "string"
    ]
  }
}
```

Response bodies

ListInputsResultModel schema

```
{
  "inputs": [
    {
      "arn": "string",
      "attachedChannels": [
        "string"
      ],
      "destinations": [
        {
```

```
          "ip": "string",
          "port": "string",
          "url": "string",
          "vpc": {
            "availabilityZone": "string",
            "networkInterfaceId": "string"
          }
        }
      ],
      "id": "string",
      "inputClass": enum,
      "inputDevices": [
        {
          "id": "string"
        }
      ],
      "inputPartnerIds": [
        "string"
      ],
      "inputSourceType": enum,
      "mediaConnectFlows": [
        {
          "flowArn": "string"
        }
      ],
      "name": "string",
      "roleArn": "string",
      "securityGroups": [
        "string"
      ],
      "sources": [
        {
          "passwordParam": "string",
          "url": "string",
          "username": "string"
        }
      ],
      "state": enum,
      "tags": {
      },
      "type": enum
    }
  ],
  "nextToken": "string"
```

```
}
```

CreateInputResultModel schema

```
{
  "input": {
    "arn": "string",
    "attachedChannels": [
      "string"
    ],
    "destinations": [
      {
        "ip": "string",
        "port": "string",
        "url": "string",
        "vpc": {
          "availabilityZone": "string",
          "networkInterfaceId": "string"
        }
      }
    ],
    "id": "string",
    "inputClass": enum,
    "inputDevices": [
      {
        "id": "string"
      }
    ],
    "inputPartnerIds": [
      "string"
    ],
    "inputSourceType": enum,
    "mediaConnectFlows": [
      {
        "flowArn": "string"
      }
    ],
    "name": "string",
    "roleArn": "string",
    "securityGroups": [
      "string"
    ],
    "sources": [
```

```
      {
        "passwordParam": "string",
        "url": "string",
        "username": "string"
      }
    ],
    "state": enum,
    "tags": {
    },
    "type": enum
  }
}
```

InvalidRequest schema

```
{
  "message": "string"
}
```

AccessDenied schema

```
{
  "message": "string"
}
```

LimitExceeded schema

```
{
  "message": "string"
}
```

InternalServiceError schema

```
{
  "message": "string"
}
```

BadGatewayException schema

```
{
  "message": "string"
}
```

GatewayTimeoutException schema

```
{
  "message": "string"
}
```

Properties

AccessDenied

message

Type: string

Required: False

BadGatewayException

message

Type: string

Required: False

CreateInput

destinations

Destination settings for PUSH type inputs.

Type: Array of type InputDestinationRequest

Required: False

inputDevices

Settings for the devices.

Type: Array of type InputDeviceSettings
Required: False

inputSecurityGroups

A list of security groups referenced by IDs to attach to the input.

Type: Array of type string
Required: False

mediaConnectFlows

A list of the MediaConnect Flows that you want to use in this input. You can specify as few as one Flow and presently, as many as two. The only requirement is when you have more than one is that each Flow is in a separate Availability Zone as this ensures your EML input is redundant to AZ issues.

Type: Array of type MediaConnectFlowRequest
Required: False

name

Name of the input.

Type: string
Required: False

requestId

Unique identifier of the request to ensure the request is handled exactly once in case of retries.

Type: string
Required: False

roleArn

The Amazon Resource Name (ARN) of the role this input assumes during and after creation.

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
Required: False

tags

A collection of key-value pairs.

Type: Tags
Required: False

type

Type: InputType
Required: False

vpc

Type: InputVpcRequest
Required: False

CreateInputResultModel

input

Type: Input
Required: False

GatewayTimeoutException

message

Type: string
Required: False

Input

arn

The Unique ARN of the input (generated, immutable).

Type: string
Required: False

attachedChannels

A 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

destinations

A list of the destinations of the input (PUSH-type).

Type: Array of type InputDestination
Required: False

id

Read-only ID for the input. Unique in the AWS account.

Type: string
Required: False

inputClass

STANDARD - MediaLive expects two sources to be connected to this input. If the channel is also STANDARD, both sources will be ingested. If the channel is SINGLE_PIPELINE, only the first source will be ingested; the second source will always be ignored, even if the first source fails. SINGLE_PIPELINE - You can connect only one source to this input. If the ChannelClass is also SINGLE_PIPELINE, this value is valid. If the ChannelClass is STANDARD, this value is not valid because the channel requires two sources in the input.

Type: InputClass
Required: False

inputDevices

Settings for the input devices.

Type: Array of type InputDeviceSettings
Required: False

inputPartnerIds

A list of IDs for all Inputs which are partners of this one.

Type: Array of type string
Required: False

inputSourceType

Certain pull input sources can be dynamic, meaning that they can have their URL's dynamically changes during input switch actions. Presently, this functionality only works with MP4_FILE and TS_FILE inputs.

Type: InputSourceType
Required: False

mediaConnectFlows

A list of MediaConnect Flows for this input.

Type: Array of type MediaConnectFlow
Required: False

name

A modifiable ID for the input.

Type: string
Required: False

roleArn

The Amazon Resource Name (ARN) of the role this input assumes during and after creation.

Type: string
Required: False

securityGroups

A list of IDs for all the Input Security Groups attached to the input.

Type: Array of type string
Required: False

sources

A list of the sources of the input (PULL-type).

Type: Array of type InputSource
Required: False

state

Type: InputState
Required: False

tags

A collection of key-value pairs.

Type: Tags
Required: False

type

Type: InputType
Required: False

InputClass

A standard input has two sources and a single pipeline input only has one.

STANDARD
SINGLE_PIPELINE

InputDestination

The settings for a PUSH type input.

ip

The system-generated static IP address of endpoint. It remains fixed for the lifetime of the input.

Type: string
Required: False

port

The port number for the input.

Type: string
Required: False

url

This represents the endpoint that the customer stream will be pushed to.

Type: string
Required: False

vpc

Type: InputDestinationVpc
Required: False

InputDestinationRequest

Endpoint settings for a PUSH type input.

streamName

A unique name for the location the RTMP stream is being pushed to.

Type: string
Required: False

InputDestinationVpc

The properties for a VPC type input destination.

availabilityZone

The availability zone of the Input destination.

Type: string
Required: False

networkInterfaceId

The network interface ID of the Input destination in the VPC.

Type: string
Required: False

InputDeviceSettings

Settings for an input device.

id

The unique ID for the device.

Type: string
Required: False

InputSource

The settings for a PULL type input.

passwordParam

The 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

The username for the input source.

Type: string
Required: False

InputSourceRequest

Settings for for a PULL type input.

passwordParam

The 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

The username for the input source.

Type: string
Required: False

InputSourceType

There are two types of input sources, static and dynamic. If an input source is dynamic you can change the source url of the input dynamically using an input switch action. Currently, two input types support a dynamic url at this time, MP4_FILE and TS_FILE. By default all input sources are static.

```
STATIC
DYNAMIC
```

InputState

```
CREATING
DETACHED
ATTACHED
DELETING
DELETED
```

InputType

The different types of inputs that AWS Elemental MediaLive supports.

UDP_PUSH
RTP_PUSH
RTMP_PUSH
RTMP_PULL
URL_PULL
MP4_FILE
MEDIACONNECT
MULTICAST
INPUT_DEVICE
AWS_CDI
TS_FILE
SRT_CALLER
SDI

InputVpcRequest

Settings for a private VPC Input. When this property is specified, the input destination addresses will be created in a VPC rather than with public Internet addresses. This property requires setting the roleArn property on Input creation. Not compatible with the inputSecurityGroups property.

securityGroupIds

A list of up to 5 EC2 VPC security group IDs to attach to the Input VPC network interfaces. Requires subnetIds. If none are specified then the VPC default security group will be used.

Type: Array of type string
Required: False

subnetIds

A list of 2 VPC subnet IDs from the same VPC. Subnet IDs must be mapped to two unique availability zones (AZ).

Type: Array of type string

Required: True

InternalServiceError

message

Type: string
Required: False

InvalidRequest

message

Type: string
Required: False

LimitExceeded

message

Type: string
Required: False

ListInputsResultModel

inputs

Type: Array of type Input
Required: False

nextToken

Type: string
Required: False

MediaConnectFlow

The settings for a MediaConnect Flow.

flowArn

The unique ARN of the MediaConnect Flow being used as a source.

Type: string
Required: False

MediaConnectFlowRequest

The settings for a MediaConnect Flow.

flowArn

The ARN of the MediaConnect Flow that you want to use as a source.

Type: string
Required: False

Tags

key-value pairs

Type: string

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

ListInputs

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3

- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

CreateInput

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Inputs: input ID

URI

`/prod/inputs/`*`inputId`*

HTTP methods

DELETE

Operation ID: `DeleteInput`

Path parameters

Name	Type	Required	Description
`inputId`	String	True	

Responses

Status code	Response model	Description
200	Empty	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
409	ResourceConflict	409 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

GET

Operation ID: `DescribeInput`

Path parameters

Name	Type	Required	Description
inputId	String	True	

Responses

Status code	Response model	Description
200	Input	200 response
400	InvalidRequest	400 response

Status code	Response model	Description
403	AccessDenied	403 response
404	ResourceNotFound	404 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

PUT

Operation ID: `UpdateInput`

Path parameters

Name	Type	Required	Description
inputId	String	True	

Responses

Status code	Response model	Description
200	UpdateInputResultModel	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
409	ResourceConflict	409 response

Status code	Response model	Description
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

Schemas

Request bodies

PUT schema

```
{
  "destinations": [
    {
      "streamName": "string"
    }
  ],
  "inputDevices": [
    {
      "id": "string"
    }
  ],
  "inputSecurityGroups": [
    "string"
  ],
  "mediaConnectFlows": [
    {
      "flowArn": "string"
    }
  ],
  "name": "string",
  "roleArn": "string",
  "sources": [
    {
      "passwordParam": "string",
      "url": "string",
      "username": "string"
```

```
    }
  ]
}
```

Response bodies

Empty schema

```
{
}
```

Input schema

```
{
  "arn": "string",
  "attachedChannels": [
    "string"
  ],
  "destinations": [
    {
      "ip": "string",
      "port": "string",
      "url": "string",
      "vpc": {
        "availabilityZone": "string",
        "networkInterfaceId": "string"
      }
    }
  ],
  "id": "string",
  "inputClass": enum,
  "inputDevices": [
    {
      "id": "string"
    }
  ],
  "inputPartnerIds": [
    "string"
  ],
  "inputSourceType": enum,
  "mediaConnectFlows": [
    {
```

```
      "flowArn": "string"
    }
  ],
  "name": "string",
  "roleArn": "string",
  "securityGroups": [
    "string"
  ],
  "sources": [
    {
      "passwordParam": "string",
      "url": "string",
      "username": "string"
    }
  ],
  "state": enum,
  "tags": {
  },
  "type": enum
}
```

UpdateInputResultModel schema

```
{
  "input": {
    "arn": "string",
    "attachedChannels": [
      "string"
    ],
    "destinations": [
      {
        "ip": "string",
        "port": "string",
        "url": "string",
        "vpc": {
          "availabilityZone": "string",
          "networkInterfaceId": "string"
        }
      }
    ],
    "id": "string",
    "inputClass": enum,
    "inputDevices": [
```

```
      {
        "id": "string"
      }
    ],
    "inputPartnerIds": [
      "string"
    ],
    "inputSourceType": enum,
    "mediaConnectFlows": [
      {
        "flowArn": "string"
      }
    ],
    "name": "string",
    "roleArn": "string",
    "securityGroups": [
      "string"
    ],
    "sources": [
      {
        "passwordParam": "string",
        "url": "string",
        "username": "string"
      }
    ],
    "state": enum,
    "tags": {
    },
    "type": enum
  }
}
```

InvalidRequest schema

```
{
  "message": "string"
}
```

AccessDenied schema

```
{
  "message": "string"
```

```
}
```

ResourceNotFound schema

```
{
  "message": "string"
}
```

ResourceConflict schema

```
{
  "message": "string"
}
```

LimitExceeded schema

```
{
  "message": "string"
}
```

InternalServiceError schema

```
{
  "message": "string"
}
```

BadGatewayException schema

```
{
  "message": "string"
}
```

GatewayTimeoutException schema

```
{
  "message": "string"
}
```

Properties

AccessDenied

message

Type: string
Required: False

BadGatewayException

message

Type: string
Required: False

Empty

GatewayTimeoutException

message

Type: string
Required: False

Input

arn

The Unique ARN of the input (generated, immutable).

Type: string
Required: False

attachedChannels

A 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

destinations

A list of the destinations of the input (PUSH-type).

Type: Array of type InputDestination
Required: False

id

Read-only ID for the input. Unique in the AWS account.

Type: string
Required: False

inputClass

STANDARD - MediaLive expects two sources to be connected to this input. If the channel is also STANDARD, both sources will be ingested. If the channel is SINGLE_PIPELINE, only the first source will be ingested; the second source will always be ignored, even if the first source fails. SINGLE_PIPELINE - You can connect only one source to this input. If the ChannelClass is also SINGLE_PIPELINE, this value is valid. If the ChannelClass is STANDARD, this value is not valid because the channel requires two sources in the input.

Type: InputClass
Required: False

inputDevices

Settings for the input devices.

Type: Array of type InputDeviceSettings
Required: False

inputPartnerIds

A list of IDs for all Inputs which are partners of this one.

Type: Array of type string
Required: False

inputSourceType

Certain pull input sources can be dynamic, meaning that they can have their URL's dynamically changes during input switch actions. Presently, this functionality only works with MP4_FILE and TS_FILE inputs.

Type: InputSourceType
Required: False

mediaConnectFlows

A list of MediaConnect Flows for this input.

Type: Array of type MediaConnectFlow
Required: False

name

A modifiable ID for the input.

Type: string
Required: False

roleArn

The Amazon Resource Name (ARN) of the role this input assumes during and after creation.

Type: string
Required: False

securityGroups

A list of IDs for all the Input Security Groups attached to the input.

Type: Array of type string
Required: False

sources

A list of the sources of the input (PULL-type).

Type: Array of type InputSource
Required: False

state

Type: InputState
Required: False

tags

A collection of key-value pairs.

Type: Tags
Required: False

type

Type: InputType
Required: False

InputClass

A standard input has two sources and a single pipeline input only has one.

```
STANDARD
SINGLE_PIPELINE
```

InputDestination

The settings for a PUSH type input.

ip

The system-generated static IP address of endpoint. It remains fixed for the lifetime of the input.

Type: string
Required: False

port

The port number for the input.

Type: string
Required: False

url

This represents the endpoint that the customer stream will be pushed to.

Type: string
Required: False

vpc

Type: InputDestinationVpc
Required: False

InputDestinationRequest

Endpoint settings for a PUSH type input.

streamName

A unique name for the location the RTMP stream is being pushed to.

Type: string
Required: False

InputDestinationVpc

The properties for a VPC type input destination.

availabilityZone

The availability zone of the Input destination.

Type: string
Required: False

networkInterfaceId

The network interface ID of the Input destination in the VPC.

Type: string
Required: False

InputDeviceRequest

Settings for an input device.

id

The unique ID for the device.

Type: string
Required: False

InputDeviceSettings

Settings for an input device.

id

The unique ID for the device.

Type: string
Required: False

InputSource

The settings for a PULL type input.

passwordParam

The 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

The username for the input source.

Type: string
Required: False

InputSourceRequest

Settings for for a PULL type input.

passwordParam

The 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

The username for the input source.

Type: string
Required: False

InputSourceType

There are two types of input sources, static and dynamic. If an input source is dynamic you can change the source url of the input dynamically using an input switch action. Currently, two input types support a dynamic url at this time, MP4_FILE and TS_FILE. By default all input sources are static.

```
STATIC
DYNAMIC
```

InputState

```
CREATING
DETACHED
ATTACHED
DELETING
DELETED
```

InputType

The different types of inputs that AWS Elemental MediaLive supports.

```
UDP_PUSH
RTP_PUSH
RTMP_PUSH
RTMP_PULL
URL_PULL
MP4_FILE
```

MEDIACONNECT
MULTICAST
INPUT_DEVICE
AWS_CDI
TS_FILE
SRT_CALLER
SDI

InternalServiceError

message

Type: string
Required: False

InvalidRequest

message

Type: string
Required: False

LimitExceeded

message

Type: string
Required: False

MediaConnectFlow

The settings for a MediaConnect Flow.

flowArn

The unique ARN of the MediaConnect Flow being used as a source.

Type: string
Required: False

MediaConnectFlowRequest

The settings for a MediaConnect Flow.

flowArn

The ARN of the MediaConnect Flow that you want to use as a source.

Type: string
Required: False

ResourceConflict

message

Type: string
Required: False

ResourceNotFound

message

Type: string
Required: False

Tags

key-value pairs

Type: string

UpdateInput

destinations

Destination settings for PUSH type inputs.

Type: Array of type InputDestinationRequest

Required: False

inputDevices

Settings for the devices.

Type: Array of type InputDeviceRequest
Required: False

inputSecurityGroups

A list of security groups referenced by IDs to attach to the input.

Type: Array of type string
Required: False

mediaConnectFlows

A list of the MediaConnect Flow ARNs that you want to use as the source of the input. You can specify as few as one Flow and presently, as many as two. The only requirement is when you have more than one is that each Flow is in a separate Availability Zone as this ensures your EML input is redundant to AZ issues.

Type: Array of type MediaConnectFlowRequest
Required: False

name

Name of the input.

Type: string
Required: False

roleArn

The Amazon Resource Name (ARN) of the role this input assumes during and after creation.

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
Required: False

UpdateInputResultModel

input

Type: Input
Required: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

DeleteInput

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

DescribeInput

- AWS Command Line Interface

- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UpdateInput

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Inputs: partners

URI

`/prod/inputs/`*`inputId`*`/partners`

HTTP methods

POST

Operation ID: `CreatePartnerInput`

Path parameters

Name	Type	Required	Description
inputId	String	True	

Responses

Status code	Response model	Description
201	CreatePartnerInputResultModel	201 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

Schemas

Request bodies

POST schema

```
{
  "requestId": "string",
  "tags": {
  }
}
```

Response bodies

CreatePartnerInputResultModel schema

```
{
  "input": {
    "arn": "string",
    "attachedChannels": [
      "string"
    ],
    "destinations": [
      {
        "ip": "string",
        "port": "string",
        "url": "string",
        "vpc": {
          "availabilityZone": "string",
          "networkInterfaceId": "string"
        }
      }
    ],
    "id": "string",
    "inputClass": enum,
    "inputDevices": [
      {
        "id": "string"
      }
    ],
    "inputPartnerIds": [
      "string"
    ],
    "inputSourceType": enum,
    "mediaConnectFlows": [
      {
        "flowArn": "string"
      }
    ],
    "name": "string",
    "roleArn": "string",
    "securityGroups": [
      "string"
    ],
    "sources": [
      {
        "passwordParam": "string",
        "url": "string",
```

```
        "username": "string"
      }
    ],
    "state": enum,
    "tags": {
    },
    "type": enum
  }
}
```

InvalidRequest schema

```
{
  "message": "string"
}
```

AccessDenied schema

```
{
  "message": "string"
}
```

LimitExceeded schema

```
{
  "message": "string"
}
```

InternalServiceError schema

```
{
  "message": "string"
}
```

BadGatewayException schema

```
{
  "message": "string"
```

```
}
```

GatewayTimeoutException schema

```
{
  "message": "string"
}
```

Properties

AccessDenied

message

Type: string
Required: False

BadGatewayException

message

Type: string
Required: False

CreatePartnerInput

requestId

Unique identifier of the request to ensure the request is handled exactly once in case of retries.

Type: string
Required: False

tags

A collection of key-value pairs.

Type: Tags

Required: False

CreatePartnerInputResultModel

input

Type: Input
Required: False

GatewayTimeoutException

message

Type: string
Required: False

Input

arn

The Unique ARN of the input (generated, immutable).

Type: string
Required: False

attachedChannels

A 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

destinations

A list of the destinations of the input (PUSH-type).

Type: Array of type InputDestination
Required: False

id

Read-only ID for the input. Unique in the AWS account.

Type: string
Required: False

inputClass

STANDARD - MediaLive expects two sources to be connected to this input. If the channel is also STANDARD, both sources will be ingested. If the channel is SINGLE_PIPELINE, only the first source will be ingested; the second source will always be ignored, even if the first source fails. SINGLE_PIPELINE - You can connect only one source to this input. If the ChannelClass is also SINGLE_PIPELINE, this value is valid. If the ChannelClass is STANDARD, this value is not valid because the channel requires two sources in the input.

Type: InputClass
Required: False

inputDevices

Settings for the input devices.

Type: Array of type InputDeviceSettings
Required: False

inputPartnerIds

A list of IDs for all Inputs which are partners of this one.

Type: Array of type string
Required: False

inputSourceType

Certain pull input sources can be dynamic, meaning that they can have their URL's dynamically changes during input switch actions. Presently, this functionality only works with MP4_FILE and TS_FILE inputs.

Type: InputSourceType
Required: False

mediaConnectFlows

A list of MediaConnect Flows for this input.

Type: Array of type MediaConnectFlow
Required: False

name

A modifiable ID for the input.

Type: string
Required: False

roleArn

The Amazon Resource Name (ARN) of the role this input assumes during and after creation.

Type: string
Required: False

securityGroups

A list of IDs for all the Input Security Groups attached to the input.

Type: Array of type string
Required: False

sources

A list of the sources of the input (PULL-type).

Type: Array of type InputSource
Required: False

state

Type: InputState
Required: False

tags

A collection of key-value pairs.

Type: Tags
Required: False

type

Type: InputType
Required: False

InputClass

A standard input has two sources and a single pipeline input only has one.

```
STANDARD
SINGLE_PIPELINE
```

InputDestination

The settings for a PUSH type input.

ip

The system-generated static IP address of endpoint. It remains fixed for the lifetime of the input.

Type: string
Required: False

port

The port number for the input.

Type: string
Required: False

url

This represents the endpoint that the customer stream will be pushed to.

Type: string
Required: False

vpc

Type: InputDestinationVpc
Required: False

InputDestinationVpc

The properties for a VPC type input destination.

availabilityZone

The availability zone of the Input destination.

Type: string
Required: False

networkInterfaceId

The network interface ID of the Input destination in the VPC.

Type: string
Required: False

InputDeviceSettings

Settings for an input device.

id

The unique ID for the device.

Type: string
Required: False

InputSource

The settings for a PULL type input.

passwordParam

The 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

The username for the input source.

Type: string
Required: False

InputSourceType

There are two types of input sources, static and dynamic. If an input source is dynamic you can change the source url of the input dynamically using an input switch action. Currently, two input types support a dynamic url at this time, MP4_FILE and TS_FILE. By default all input sources are static.

`STATIC`
`DYNAMIC`

InputState

CREATING

DETACHED

ATTACHED

DELETING

DELETED

InputType

The different types of inputs that AWS Elemental MediaLive supports.

UDP_PUSH

RTP_PUSH

RTMP_PUSH

RTMP_PULL

URL_PULL

MP4_FILE

MEDIACONNECT

MULTICAST

INPUT_DEVICE

AWS_CDI

TS_FILE

SRT_CALLER

SDI

InternalServiceError

message

Type: string

Required: False

InvalidRequest

message

Type: string
Required: False

LimitExceeded

message

Type: string
Required: False

MediaConnectFlow

The settings for a MediaConnect Flow.

flowArn

The unique ARN of the MediaConnect Flow being used as a source.

Type: string
Required: False

Tags

key-value pairs

Type: string

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

CreatePartnerInput

- AWS Command Line Interface

- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Multiplex: describe program

URI

`/prod/multiplexes/multiplexId/programs/programName`

HTTP methods

DELETE

Operation ID: `DeleteMultiplexProgram`

Path parameters

Name	Type	Required	Description
multiplexId	String	True	
programName	String	True	

Responses

Status code	Response model	Description
200	MultiplexProgram	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response

Status code	Response model	Description
404	ResourceNotFound	404 response
409	ResourceConflict	409 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

GET

Operation ID: DescribeMultiplexProgram

Path parameters

Name	Type	Required	Description
multiplexId	String	True	
programName	String	True	

Responses

Status code	Response model	Description
200	MultiplexProgram	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
429	LimitExceeded	429 response

Status code	Response model	Description
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

PUT

Operation ID: UpdateMultiplexProgram

Path parameters

Name	Type	Required	Description
multiplexId	String	True	
programName	String	True	

Responses

Status code	Response model	Description
200	UpdateMultiplexProgramResultModel	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
409	ResourceConflict	409 response
422	MultiplexConfigurationValidationError	422 response
500	InternalServiceError	500 response

Status code	Response model	Description
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

Schemas

Request bodies

PUT schema

```
{
  "multiplexProgramSettings": {
    "preferredChannelPipeline": enum,
    "programNumber": integer,
    "serviceDescriptor": {
      "providerName": "string",
      "serviceName": "string"
    },
    "videoSettings": {
      "constantBitrate": integer,
      "statmuxSettings": {
        "maximumBitrate": integer,
        "minimumBitrate": integer,
        "priority": integer
      }
    }
  }
}
```

Response bodies

MultiplexProgram schema

```
{
  "channelId": "string",
  "multiplexProgramSettings": {
    "preferredChannelPipeline": enum,
    "programNumber": integer,
```

```
    "serviceDescriptor": {
      "providerName": "string",
      "serviceName": "string"
    },
    "videoSettings": {
      "constantBitrate": integer,
      "statmuxSettings": {
        "maximumBitrate": integer,
        "minimumBitrate": integer,
        "priority": integer
      }
    }
  },
  "packetIdentifiersMap": {
    "audioPids": [
      integer
    ],
    "dvbSubPids": [
      integer
    ],
    "dvbTeletextPid": integer,
    "etvPlatformPid": integer,
    "etvSignalPid": integer,
    "klvDataPids": [
      integer
    ],
    "pcrPid": integer,
    "pmtPid": integer,
    "privateMetadataPid": integer,
    "scte27Pids": [
      integer
    ],
    "scte35Pid": integer,
    "timedMetadataPid": integer,
    "videoPid": integer
  },
  "pipelineDetails": [
    {
      "activeChannelPipeline": "string",
      "pipelineId": "string"
    }
  ],
  "programName": "string"
```

```
}
```

UpdateMultiplexProgramResultModel schema

```
{
  "multiplexProgram": {
    "channelId": "string",
    "multiplexProgramSettings": {
      "preferredChannelPipeline": enum,
      "programNumber": integer,
      "serviceDescriptor": {
        "providerName": "string",
        "serviceName": "string"
      },
      "videoSettings": {
        "constantBitrate": integer,
        "statmuxSettings": {
          "maximumBitrate": integer,
          "minimumBitrate": integer,
          "priority": integer
        }
      }
    },
    "packetIdentifiersMap": {
      "audioPids": [
        integer
      ],
      "dvbSubPids": [
        integer
      ],
      "dvbTeletextPid": integer,
      "etvPlatformPid": integer,
      "etvSignalPid": integer,
      "klvDataPids": [
        integer
      ],
      "pcrPid": integer,
      "pmtPid": integer,
      "privateMetadataPid": integer,
      "scte27Pids": [
        integer
      ],
      "scte35Pid": integer,
```

```
      "timedMetadataPid": integer,
      "videoPid": integer
    },
    "pipelineDetails": [
      {
        "activeChannelPipeline": "string",
        "pipelineId": "string"
      }
    ],
    "programName": "string"
  }
}
```

InvalidRequest schema

```
{
  "message": "string"
}
```

AccessDenied schema

```
{
  "message": "string"
}
```

ResourceNotFound schema

```
{
  "message": "string"
}
```

ResourceConflict schema

```
{
  "message": "string"
}
```

MultiplexConfigurationValidationError schema

```
{
  "message": "string",
  "validationErrors": [
    {
      "elementPath": "string",
      "errorMessage": "string"
    }
  ]
}
```

LimitExceeded schema

```
{
  "message": "string"
}
```

InternalServiceError schema

```
{
  "message": "string"
}
```

BadGatewayException schema

```
{
  "message": "string"
}
```

GatewayTimeoutException schema

```
{
  "message": "string"
}
```

Properties

AccessDenied

message

Type: string
Required: False

BadGatewayException

message

Type: string
Required: False

GatewayTimeoutException

message

Type: string
Required: False

InternalServiceError

message

Type: string
Required: False

InvalidRequest

message

Type: string
Required: False

LimitExceeded

message

Type: string
Required: False

MultiplexConfigurationValidationError

message

The error message.

Type: string
Required: False

validationErrors

A collection of validation error responses.

Type: Array of type ValidationError
Required: False

MultiplexProgram

The multiplex program object.

channelId

The MediaLive channel associated with the program.

Type: string
Required: False

multiplexProgramSettings

The settings for this multiplex program.

Type: MultiplexProgramSettings

Required: False

packetIdentifiersMap

The packet identifier map for this multiplex program.

Type: MultiplexProgramPacketIdentifiersMap
Required: False

pipelineDetails

Contains information about the current sources for the specified program in the specified multiplex. Keep in mind that each multiplex pipeline connects to both pipelines in a given source channel (the channel identified by the program). But only one of those channel pipelines is ever active at one time.

Type: Array of type MultiplexProgramPipelineDetail
Required: False

programName

The name of the multiplex program.

Type: string
Required: False

MultiplexProgramPacketIdentifiersMap

Packet identifiers map for a given Multiplex program.

audioPids

Type: Array of type integer
Required: False

dvbSubPids

Type: Array of type integer

Required: False

dvbTeletextPid

Type: integer
Required: False

etvPlatformPid

Type: integer
Required: False

etvSignalPid

Type: integer
Required: False

klvDataPids

Type: Array of type integer
Required: False

pcrPid

Type: integer
Required: False

pmtPid

Type: integer
Required: False

privateMetadataPid

Type: integer
Required: False

scte27Pids

Type: Array of type integer
Required: False

scte35Pid

Type: integer
Required: False

timedMetadataPid

Type: integer
Required: False

videoPid

Type: integer
Required: False

MultiplexProgramPipelineDetail

The current source for one of the pipelines in the multiplex.

activeChannelPipeline

Identifies the channel pipeline that is currently active for the pipeline (identified by PipelineId) in the multiplex.

Type: string
Required: False

pipelineId

Identifies a specific pipeline in the multiplex.

Type: string
Required: False

MultiplexProgramServiceDescriptor

Transport stream service descriptor configuration for the Multiplex program.

providerName

Name of the provider.

Type: string
Required: True
MaxLength: 256

serviceName

Name of the service.

Type: string
Required: True
MaxLength: 256

MultiplexProgramSettings

Multiplex Program settings configuration.

preferredChannelPipeline

Indicates which pipeline is preferred by the multiplex for program ingest.

Type: PreferredChannelPipeline
Required: False

programNumber

Unique program number.

Type: integer
Required: True
Minimum: 0

Maximum: 65535

serviceDescriptor

Transport stream service descriptor configuration for the Multiplex program.

Type: MultiplexProgramServiceDescriptor
Required: False

videoSettings

Program video settings configuration.

Type: MultiplexVideoSettings
Required: False

MultiplexStatmuxVideoSettings

Statmux rate control settings

maximumBitrate

Maximum statmux bitrate.

Type: integer
Required: False
Minimum: 100000
Maximum: 100000000

minimumBitrate

Minimum statmux bitrate.

Type: integer
Required: False
Minimum: 100000
Maximum: 100000000

priority

The purpose of the priority is to use a combination of the\nmultiplex rate control algorithm and the QVBR capability of the\nencoder to prioritize the video quality of some channels in a \nmultiplex over others. Channels that have a higher priority will\nget higher video quality at the expense of the video quality of\nother channels in the multiplex with lower priority.

Type: integer
Required: False
Minimum: -5
Maximum: 5

MultiplexVideoSettings

The video configuration for each program in a multiplex.

constantBitrate

The constant bitrate configuration for the video encode. When this field is defined, StatmuxSettings must be undefined.

Type: integer
Required: False
Minimum: 100000
Maximum: 100000000

statmuxSettings

Statmux rate control settings. When this field is defined, ConstantBitrate must be undefined.

Type: MultiplexStatmuxVideoSettings
Required: False

PreferredChannelPipeline

Indicates which pipeline is preferred by the multiplex for program ingest. If set to \"PIPELINE_0\" or \"PIPELINE_1\" and an unhealthy ingest causes the multiplex to switch to the non-preferred pipeline, it will switch back once that ingest is healthy again. If set to \"CURRENTLY_ACTIVE\",

it will not switch back to the other pipeline based on it recovering to a healthy state, it will only switch if the active pipeline becomes unhealthy.

CURRENTLY_ACTIVE
PIPELINE_0
PIPELINE_1

ResourceConflict

message

Type: string
Required: False

ResourceNotFound

message

Type: string
Required: False

UpdateMultiplexProgram

multiplexProgramSettings

The new settings for a multiplex program.

Type: MultiplexProgramSettings
Required: False

UpdateMultiplexProgramResultModel

multiplexProgram

The updated multiplex program.

Type: MultiplexProgram
Required: False

ValidationError

elementPath

Path to the source of the error.

Type: string
Required: False

errorMessage

The error message.

Type: string
Required: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

DeleteMultiplexProgram

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

DescribeMultiplexProgram

- AWS Command Line Interface

- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UpdateMultiplexProgram

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Multiplex: list programs

URI

`/prod/multiplexes/multiplexId/programs`

HTTP methods

GET

Operation ID: `ListMultiplexPrograms`

Path parameters

Name	Type	Required	Description
multiplexId	String	True	

Query parameters

Name	Type	Required	Description
nextToken	String	False	
maxResults	String	False	

Responses

Status code	Response model	Description
200	ListMultiplexProgramsResultModel	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

POST

Operation ID: CreateMultiplexProgram

Path parameters

Name	Type	Required	Description
multiplexId	String	True	

Responses

Status code	Response model	Description
201	CreateMultiplexProgramResultModel	201 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
409	ResourceConflict	409 response
422	MultiplexConfigurationValidationError	422 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

Schemas

Request bodies

POST schema

```
{
  "multiplexProgramSettings": {
    "preferredChannelPipeline": enum,
```

```
    "programNumber": integer,
    "serviceDescriptor": {
      "providerName": "string",
      "serviceName": "string"
    },
    "videoSettings": {
      "constantBitrate": integer,
      "statmuxSettings": {
        "maximumBitrate": integer,
        "minimumBitrate": integer,
        "priority": integer
      }
    }
  },
  "programName": "string",
  "requestId": "string"
}
```

Response bodies

ListMultiplexProgramsResultModel schema

```
{
  "multiplexPrograms": [
    {
      "channelId": "string",
      "programName": "string"
    }
  ],
  "nextToken": "string"
}
```

CreateMultiplexProgramResultModel schema

```
{
  "multiplexProgram": {
    "channelId": "string",
    "multiplexProgramSettings": {
      "preferredChannelPipeline": enum,
      "programNumber": integer,
      "serviceDescriptor": {
        "providerName": "string",
```

```
        "serviceName": "string"
      },
      "videoSettings": {
        "constantBitrate": integer,
        "statmuxSettings": {
          "maximumBitrate": integer,
          "minimumBitrate": integer,
          "priority": integer
        }
      }
    },
    "packetIdentifiersMap": {
      "audioPids": [
        integer
      ],
      "dvbSubPids": [
        integer
      ],
      "dvbTeletextPid": integer,
      "etvPlatformPid": integer,
      "etvSignalPid": integer,
      "klvDataPids": [
        integer
      ],
      "pcrPid": integer,
      "pmtPid": integer,
      "privateMetadataPid": integer,
      "scte27Pids": [
        integer
      ],
      "scte35Pid": integer,
      "timedMetadataPid": integer,
      "videoPid": integer
    },
    "pipelineDetails": [
      {
        "activeChannelPipeline": "string",
        "pipelineId": "string"
      }
    ],
    "programName": "string"
  }
}
```

InvalidRequest schema

```
{
  "message": "string"
}
```

AccessDenied schema

```
{
  "message": "string"
}
```

ResourceNotFound schema

```
{
  "message": "string"
}
```

ResourceConflict schema

```
{
  "message": "string"
}
```

MultiplexConfigurationValidationError schema

```
{
  "message": "string",
  "validationErrors": [
    {
      "elementPath": "string",
      "errorMessage": "string"
    }
  ]
}
```

LimitExceeded schema

```
{
```

```
  "message": "string"
}
```

InternalServiceError schema

```
{
  "message": "string"
}
```

BadGatewayException schema

```
{
  "message": "string"
}
```

GatewayTimeoutException schema

```
{
  "message": "string"
}
```

Properties

AccessDenied

message

Type: string

Required: False

BadGatewayException

message

Type: string

Required: False

CreateMultiplexProgram

multiplexProgramSettings

The settings for this multiplex program.

Type: MultiplexProgramSettings
Required: True

programName

Name of multiplex program.

Type: string
Required: True

requestId

Unique request ID. This prevents retries from creating multiple resources.

Type: string
Required: True

CreateMultiplexProgramResultModel

multiplexProgram

The newly created multiplex program.

Type: MultiplexProgram
Required: False

GatewayTimeoutException

message

Type: string
Required: False

InternalServiceError

message

Type: string
Required: False

InvalidRequest

message

Type: string
Required: False

LimitExceeded

message

Type: string
Required: False

ListMultiplexProgramsResultModel

multiplexPrograms

List of multiplex programs.

Type: Array of type MultiplexProgramSummary
Required: False

nextToken

Token for the next ListMultiplexProgram request.

Type: string
Required: False

MultiplexConfigurationValidationError

message

The error message.

Type: string
Required: False

validationErrors

A collection of validation error responses.

Type: Array of type ValidationError
Required: False

MultiplexProgram

The multiplex program object.

channelId

The MediaLive channel associated with the program.

Type: string
Required: False

multiplexProgramSettings

The settings for this multiplex program.

Type: MultiplexProgramSettings
Required: False

packetIdentifiersMap

The packet identifier map for this multiplex program.

Type: MultiplexProgramPacketIdentifiersMap
Required: False

pipelineDetails

Contains information about the current sources for the specified program in the specified multiplex. Keep in mind that each multiplex pipeline connects to both pipelines in a given source channel (the channel identified by the program). But only one of those channel pipelines is ever active at one time.

Type: Array of type MultiplexProgramPipelineDetail
Required: False

programName

The name of the multiplex program.

Type: string
Required: False

MultiplexProgramPacketIdentifiersMap

Packet identifiers map for a given Multiplex program.

audioPids

Type: Array of type integer
Required: False

dvbSubPids

Type: Array of type integer
Required: False

dvbTeletextPid

Type: integer
Required: False

etvPlatformPid

Type: integer

Required: False

etvSignalPid

Type: integer
Required: False

klvDataPids

Type: Array of type integer
Required: False

pcrPid

Type: integer
Required: False

pmtPid

Type: integer
Required: False

privateMetadataPid

Type: integer
Required: False

scte27Pids

Type: Array of type integer
Required: False

scte35Pid

Type: integer
Required: False

timedMetadataPid

Type: integer
Required: False

videoPid

Type: integer
Required: False

MultiplexProgramPipelineDetail

The current source for one of the pipelines in the multiplex.

activeChannelPipeline

Identifies the channel pipeline that is currently active for the pipeline (identified by PipelineId) in the multiplex.

Type: string
Required: False

pipelineId

Identifies a specific pipeline in the multiplex.

Type: string
Required: False

MultiplexProgramServiceDescriptor

Transport stream service descriptor configuration for the Multiplex program.

providerName

Name of the provider.

Type: string
Required: True
MaxLength: 256

serviceName

Name of the service.

Type: string
Required: True
MaxLength: 256

MultiplexProgramSettings

Multiplex Program settings configuration.

preferredChannelPipeline

Indicates which pipeline is preferred by the multiplex for program ingest.

Type: PreferredChannelPipeline
Required: False

programNumber

Unique program number.

Type: integer
Required: True
Minimum: 0
Maximum: 65535

serviceDescriptor

Transport stream service descriptor configuration for the Multiplex program.

Type: MultiplexProgramServiceDescriptor
Required: False

videoSettings

Program video settings configuration.

Type: MultiplexVideoSettings
Required: False

MultiplexProgramSummary

channelId

The MediaLive Channel associated with the program.

Type: string
Required: False

programName

The name of the multiplex program.

Type: string
Required: False

MultiplexStatmuxVideoSettings

Statmux rate control settings

maximumBitrate

Maximum statmux bitrate.

Type: integer
Required: False
Minimum: 100000
Maximum: 100000000

minimumBitrate

Minimum statmux bitrate.

Type: integer
Required: False
Minimum: 100000

Maximum: 100000000

priority

The purpose of the priority is to use a combination of the\nmultiplex rate control algorithm and the QVBR capability of the\nencoder to prioritize the video quality of some channels in a \nmultiplex over others. Channels that have a higher priority will\nget higher video quality at the expense of the video quality of\nother channels in the multiplex with lower priority.

Type: integer
Required: False
Minimum: -5
Maximum: 5

MultiplexVideoSettings

The video configuration for each program in a multiplex.

constantBitrate

The constant bitrate configuration for the video encode. When this field is defined, StatmuxSettings must be undefined.

Type: integer
Required: False
Minimum: 100000
Maximum: 100000000

statmuxSettings

Statmux rate control settings. When this field is defined, ConstantBitrate must be undefined.

Type: MultiplexStatmuxVideoSettings
Required: False

PreferredChannelPipeline

Indicates which pipeline is preferred by the multiplex for program ingest. If set to \"PIPELINE_0\" or \"PIPELINE_1\" and an unhealthy ingest causes the multiplex to switch to the non-preferred

pipeline, it will switch back once that ingest is healthy again. If set to \"CURRENTLY_ACTIVE\", it will not switch back to the other pipeline based on it recovering to a healthy state, it will only switch if the active pipeline becomes unhealthy.

CURRENTLY_ACTIVE

PIPELINE_0

PIPELINE_1

ResourceConflict

message

Type: string

Required: False

ResourceNotFound

message

Type: string

Required: False

ValidationError

elementPath

Path to the source of the error.

Type: string

Required: False

errorMessage

The error message.

Type: string

Required: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

ListMultiplexPrograms

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

CreateMultiplexProgram

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Multiplex: start

URI

`/prod/multiplexes/multiplexId/start`

HTTP methods

POST

Operation ID: `StartMultiplex`

Path parameters

Name	Type	Required	Description
multiplexId	String	True	

Responses

Status code	Response model	Description
202	Multiplex	202 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
409	ResourceConflict	409 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

Schemas

Response bodies

Multiplex schema

```
{
  "arn": "string",
  "availabilityZones": [
    "string"
  ],
  "destinations": [
    {
      "mediaConnectSettings": {
        "entitlementArn": "string"
      }
    }
  ],
  "id": "string",
  "multiplexSettings": {
    "maximumVideoBufferDelayMilliseconds": integer,
    "transportStreamBitrate": integer,
    "transportStreamId": integer,
    "transportStreamReservedBitrate": integer
  },
  "name": "string",
  "pipelinesRunningCount": integer,
  "programCount": integer,
  "state": enum,
  "tags": {
  }
}
```

InvalidRequest schema

```
{
  "message": "string"
}
```

AccessDenied schema

```
{
  "message": "string"
}
```

ResourceNotFound schema

```
{
  "message": "string"
}
```

ResourceConflict schema

```
{
  "message": "string"
}
```

LimitExceeded schema

```
{
  "message": "string"
}
```

InternalServiceError schema

```
{
  "message": "string"
}
```

BadGatewayException schema

```
{
  "message": "string"
}
```

GatewayTimeoutException schema

```
{
  "message": "string"
```

```
}
```

Properties

AccessDenied

message

Type: string
Required: False

BadGatewayException

message

Type: string
Required: False

GatewayTimeoutException

message

Type: string
Required: False

InternalServiceError

message

Type: string
Required: False

InvalidRequest

message

Type: string
Required: False

LimitExceeded

message

Type: string
Required: False

Multiplex

The multiplex object.

arn

The unique arn of the multiplex.

Type: string
Required: False

availabilityZones

A list of availability zones for the multiplex.

Type: Array of type string
Required: False

destinations

A list of the multiplex output destinations.

Type: Array of type MultiplexOutputDestination
Required: False

id

The unique id of the multiplex.

Type: string
Required: False

multiplexSettings

Configuration for a multiplex event.

Type: MultiplexSettings
Required: False

name

The name of the multiplex.

Type: string
Required: False

pipelinesRunningCount

The number of currently healthy pipelines.

Type: integer
Required: False

programCount

The number of programs in the multiplex.

Type: integer
Required: False

state

The current state of the multiplex.

Type: MultiplexState
Required: False

tags

A collection of key-value pairs.

Type: Tags
Required: False

MultiplexMediaConnectOutputDestinationSettings

Multiplex MediaConnect output destination settings.

entitlementArn

The MediaConnect entitlement ARN available as a Flow source.

Type: string
Required: False
MinLength: 1

MultiplexOutputDestination

Multiplex output destination settings

mediaConnectSettings

Multiplex MediaConnect output destination settings.

Type: MultiplexMediaConnectOutputDestinationSettings
Required: False

MultiplexSettings

Contains configuration for a Multiplex event

maximumVideoBufferDelayMilliseconds

Maximum video buffer delay in milliseconds.

Type: integer
Required: False
Minimum: 800
Maximum: 3000

transportStreamBitrate

Transport stream bit rate.

Type: integer
Required: True
Minimum: 1000000
Maximum: 100000000

transportStreamId

Transport stream ID.

Type: integer
Required: True
Minimum: 0
Maximum: 65535

transportStreamReservedBitrate

Transport stream reserved bit rate.

Type: integer
Required: False
Minimum: 0
Maximum: 100000000

MultiplexState

The current state of the multiplex.

```
CREATING
CREATE_FAILED
IDLE
STARTING
RUNNING
RECOVERING
```

STOPPING
DELETING
DELETED

ResourceConflict

message

Type: string
Required: False

ResourceNotFound

message

Type: string
Required: False

Tags

key-value pairs

Type: string

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

StartMultiplex

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2

- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Multiplex: stop

URI

`/prod/multiplexes/multiplexId/stop`

HTTP methods

POST

Operation ID: `StopMultiplex`

Path parameters

Name	Type	Required	Description
multiplexId	String	True	

Responses

Status code	Response model	Description
202	Multiplex	202 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
409	ResourceConflict	409 response
429	LimitExceeded	429 response

Status code	Response model	Description
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

Schemas

Response bodies

Multiplex schema

```
{
  "arn": "string",
  "availabilityZones": [
    "string"
  ],
  "destinations": [
    {
      "mediaConnectSettings": {
        "entitlementArn": "string"
      }
    }
  ],
  "id": "string",
  "multiplexSettings": {
    "maximumVideoBufferDelayMilliseconds": integer,
    "transportStreamBitrate": integer,
    "transportStreamId": integer,
    "transportStreamReservedBitrate": integer
  },
  "name": "string",
  "pipelinesRunningCount": integer,
  "programCount": integer,
  "state": enum,
  "tags": {
  }
}
```

InvalidRequest schema

```
{
  "message": "string"
}
```

AccessDenied schema

```
{
  "message": "string"
}
```

ResourceNotFound schema

```
{
  "message": "string"
}
```

ResourceConflict schema

```
{
  "message": "string"
}
```

LimitExceeded schema

```
{
  "message": "string"
}
```

InternalServiceError schema

```
{
  "message": "string"
```

```
}
```

BadGatewayException schema

```
{
  "message": "string"
}
```

GatewayTimeoutException schema

```
{
  "message": "string"
}
```

Properties

AccessDenied

message

Type: string

Required: False

BadGatewayException

message

Type: string

Required: False

GatewayTimeoutException

message

Type: string

Required: False

InternalServiceError

message

Type: string
Required: False

InvalidRequest

message

Type: string
Required: False

LimitExceeded

message

Type: string
Required: False

Multiplex

The multiplex object.

arn

The unique arn of the multiplex.

Type: string
Required: False

availabilityZones

A list of availability zones for the multiplex.

Type: Array of type string
Required: False

destinations

A list of the multiplex output destinations.

Type: Array of type MultiplexOutputDestination
Required: False

id

The unique id of the multiplex.

Type: string
Required: False

multiplexSettings

Configuration for a multiplex event.

Type: MultiplexSettings
Required: False

name

The name of the multiplex.

Type: string
Required: False

pipelinesRunningCount

The number of currently healthy pipelines.

Type: integer
Required: False

programCount

The number of programs in the multiplex.

Type: integer
Required: False

state

The current state of the multiplex.

Type: MultiplexState
Required: False

tags

A collection of key-value pairs.

Type: Tags
Required: False

MultiplexMediaConnectOutputDestinationSettings

Multiplex MediaConnect output destination settings.

entitlementArn

The MediaConnect entitlement ARN available as a Flow source.

Type: string
Required: False
MinLength: 1

MultiplexOutputDestination

Multiplex output destination settings

mediaConnectSettings

Multiplex MediaConnect output destination settings.

Type: MultiplexMediaConnectOutputDestinationSettings

Required: False

MultiplexSettings

Contains configuration for a Multiplex event

maximumVideoBufferDelayMilliseconds

Maximum video buffer delay in milliseconds.

Type: integer
Required: False
Minimum: 800
Maximum: 3000

transportStreamBitrate

Transport stream bit rate.

Type: integer
Required: True
Minimum: 1000000
Maximum: 100000000

transportStreamId

Transport stream ID.

Type: integer
Required: True
Minimum: 0
Maximum: 65535

transportStreamReservedBitrate

Transport stream reserved bit rate.

Type: integer

Required: False
Minimum: 0
Maximum: 100000000

MultiplexState

The current state of the multiplex.

CREATING
CREATE_FAILED
IDLE
STARTING
RUNNING
RECOVERING
STOPPING
DELETING
DELETED

ResourceConflict

message

Type: string
Required: False

ResourceNotFound

message

Type: string
Required: False

Tags

key-value pairs

Type: string

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

StopMultiplex

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Multiplexes

URI

`/prod/multiplexes`

HTTP methods

GET

Operation ID: `ListMultiplexes`

Query parameters

Name	Type	Required	Description
`nextToken`	String	False	
`maxResults`	String	False	

Responses

Status code	Response model	Description
200	ListMultiplexesResultModel	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

POST

Operation ID: `CreateMultiplex`

Responses

Status code	Response model	Description
201	CreateMultiplexResultModel	201 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
409	ResourceConflict	409 response
422	MultiplexConfigurationValidationError	422 response
429	LimitExceeded	429 response

Status code	Response model	Description
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

Schemas

Request bodies

POST schema

```
{
  "availabilityZones": [
    "string"
  ],
  "multiplexSettings": {
    "maximumVideoBufferDelayMilliseconds": integer,
    "transportStreamBitrate": integer,
    "transportStreamId": integer,
    "transportStreamReservedBitrate": integer
  },
  "name": "string",
  "requestId": "string",
  "tags": {
  }
}
```

Response bodies

ListMultiplexesResultModel schema

```
{
  "multiplexes": [
    {
      "arn": "string",
      "availabilityZones": [
```

```
        "string"
      ],
      "id": "string",
      "multiplexSettings": {
        "transportStreamBitrate": integer
      },
      "name": "string",
      "pipelinesRunningCount": integer,
      "programCount": integer,
      "state": enum,
      "tags": {
      }
    }
  ],
  "nextToken": "string"
}
```

CreateMultiplexResultModel schema

```
{
  "multiplex": {
    "arn": "string",
    "availabilityZones": [
      "string"
    ],
    "destinations": [
      {
        "mediaConnectSettings": {
          "entitlementArn": "string"
        }
      }
    ],
    "id": "string",
    "multiplexSettings": {
      "maximumVideoBufferDelayMilliseconds": integer,
      "transportStreamBitrate": integer,
      "transportStreamId": integer,
      "transportStreamReservedBitrate": integer
    },
    "name": "string",
    "pipelinesRunningCount": integer,
    "programCount": integer,
    "state": enum,
```

```
    "tags": {
    }
  }
}
```

InvalidRequest schema

```
{
  "message": "string"
}
```

AccessDenied schema

```
{
  "message": "string"
}
```

ResourceConflict schema

```
{
  "message": "string"
}
```

MultiplexConfigurationValidationError schema

```
{
  "message": "string",
  "validationErrors": [
    {
      "elementPath": "string",
      "errorMessage": "string"
    }
  ]
}
```

LimitExceeded schema

```
{
```

```
  "message": "string"
}
```

InternalServiceError schema

```
{
  "message": "string"
}
```

BadGatewayException schema

```
{
  "message": "string"
}
```

GatewayTimeoutException schema

```
{
  "message": "string"
}
```

Properties

AccessDenied

message

Type: string

Required: False

BadGatewayException

message

Type: string

Required: False

CreateMultiplex

availabilityZones

A list of availability zones for the multiplex. You must specify exactly two.

Type: Array of type string
Required: True

multiplexSettings

Configuration for a multiplex event.

Type: MultiplexSettings
Required: True

name

Name of multiplex.

Type: string
Required: True

requestId

Unique request ID. This prevents retries from creating multiple resources.

Type: string
Required: True

tags

A collection of key-value pairs.

Type: Tags
Required: False

CreateMultiplexResultModel

multiplex

The newly created multiplex.

Type: Multiplex
Required: False

GatewayTimeoutException

message

Type: string
Required: False

InternalServiceError

message

Type: string
Required: False

InvalidRequest

message

Type: string
Required: False

LimitExceeded

message

Type: string
Required: False

ListMultiplexesResultModel

multiplexes

List of multiplexes.

Type: Array of type MultiplexSummary
Required: False

nextToken

Token for the next ListMultiplexes request.

Type: string
Required: False

Multiplex

The multiplex object.

arn

The unique arn of the multiplex.

Type: string
Required: False

availabilityZones

A list of availability zones for the multiplex.

Type: Array of type string
Required: False

destinations

A list of the multiplex output destinations.

Type: Array of type MultiplexOutputDestination
Required: False

id

The unique id of the multiplex.

Type: string
Required: False

multiplexSettings

Configuration for a multiplex event.

Type: MultiplexSettings
Required: False

name

The name of the multiplex.

Type: string
Required: False

pipelinesRunningCount

The number of currently healthy pipelines.

Type: integer
Required: False

programCount

The number of programs in the multiplex.

Type: integer
Required: False

state

The current state of the multiplex.

Type: MultiplexState
Required: False

tags

A collection of key-value pairs.

Type: Tags
Required: False

MultiplexConfigurationValidationError

message

The error message.

Type: string
Required: False

validationErrors

A collection of validation error responses.

Type: Array of type ValidationError
Required: False

MultiplexMediaConnectOutputDestinationSettings

Multiplex MediaConnect output destination settings.

entitlementArn

The MediaConnect entitlement ARN available as a Flow source.

Type: string
Required: False
MinLength: 1

MultiplexOutputDestination

Multiplex output destination settings

mediaConnectSettings

Multiplex MediaConnect output destination settings.

Type: MultiplexMediaConnectOutputDestinationSettings
Required: False

MultiplexSettings

Contains configuration for a Multiplex event

maximumVideoBufferDelayMilliseconds

Maximum video buffer delay in milliseconds.

Type: integer
Required: False
Minimum: 800
Maximum: 3000

transportStreamBitrate

Transport stream bit rate.

Type: integer
Required: True
Minimum: 1000000
Maximum: 100000000

transportStreamId

Transport stream ID.

Type: integer

Required: True
Minimum: 0
Maximum: 65535

transportStreamReservedBitrate

Transport stream reserved bit rate.

Type: integer
Required: False
Minimum: 0
Maximum: 100000000

MultiplexSettingsSummary

Contains summary configuration for a Multiplex event.

transportStreamBitrate

Transport stream bit rate.

Type: integer
Required: False
Minimum: 1000000
Maximum: 100000000

MultiplexState

The current state of the multiplex.

```
CREATING
CREATE_FAILED
IDLE
STARTING
RUNNING
RECOVERING
```

STOPPING

DELETING

DELETED

MultiplexSummary

arn

The unique arn of the multiplex.

Type: string
Required: False

availabilityZones

A list of availability zones for the multiplex.

Type: Array of type string
Required: False

id

The unique id of the multiplex.

Type: string
Required: False

multiplexSettings

Configuration for a multiplex event.

Type: MultiplexSettingsSummary
Required: False

name

The name of the multiplex.

Type: string
Required: False

pipelinesRunningCount

The number of currently healthy pipelines.

Type: integer
Required: False

programCount

The number of programs in the multiplex.

Type: integer
Required: False

state

The current state of the multiplex.

Type: MultiplexState
Required: False

tags

A collection of key-value pairs.

Type: Tags
Required: False

ResourceConflict

message

Type: string
Required: False

Tags

key-value pairs

Type: string

ValidationError

elementPath

Path to the source of the error.

Type: string
Required: False

errorMessage

The error message.

Type: string
Required: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

ListMultiplexes

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python

- AWS SDK for Ruby V3

CreateMultiplex

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Multiplexes: multiplex ID

URI

`/prod/multiplexes/multiplexId`

HTTP methods

DELETE

Operation ID: `DeleteMultiplex`

Path parameters

Name	Type	Required	Description
multiplexId	String	True	

Responses

Status code	Response model	Description
202	Multiplex	202 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
409	ResourceConflict	409 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

GET

Operation ID: DescribeMultiplex

Path parameters

Name	Type	Required	Description
multiplexId	String	True	

Responses

Status code	Response model	Description
200	Multiplex	200 response
400	InvalidRequest	400 response

Status code	Response model	Description
403	AccessDenied	403 response
404	ResourceNotFound	404 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

PUT

Operation ID: UpdateMultiplex

Path parameters

Name	Type	Required	Description
multiplexId	String	True	

Responses

Status code	Response model	Description
200	UpdateMultiplexResultModel	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
409	ResourceConflict	409 response

Status code	Response model	Description
422	MultiplexConfigurationValidationError	422 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

Schemas

Request bodies

PUT schema

```
{
  "multiplexSettings": {
    "maximumVideoBufferDelayMilliseconds": integer,
    "transportStreamBitrate": integer,
    "transportStreamId": integer,
    "transportStreamReservedBitrate": integer
  },
  "name": "string"
}
```

Response bodies

UpdateMultiplexResultModel schema

```
{
  "multiplex": {
    "arn": "string",
    "availabilityZones": [
      "string"
    ],
    "destinations": [
      {
```

```
        "mediaConnectSettings": {
          "entitlementArn": "string"
        }
      }
    ],
    "id": "string",
    "multiplexSettings": {
      "maximumVideoBufferDelayMilliseconds": integer,
      "transportStreamBitrate": integer,
      "transportStreamId": integer,
      "transportStreamReservedBitrate": integer
    },
    "name": "string",
    "pipelinesRunningCount": integer,
    "programCount": integer,
    "state": enum,
    "tags": {
    }
  }
}
```

Multiplex schema

```
{
  "arn": "string",
  "availabilityZones": [
    "string"
  ],
  "destinations": [
    {
      "mediaConnectSettings": {
        "entitlementArn": "string"
      }
    }
  ],
  "id": "string",
  "multiplexSettings": {
    "maximumVideoBufferDelayMilliseconds": integer,
    "transportStreamBitrate": integer,
    "transportStreamId": integer,
    "transportStreamReservedBitrate": integer
  },
  "name": "string",
```

```
  "pipelinesRunningCount": integer,
  "programCount": integer,
  "state": enum,
  "tags": {
  }
}
```

InvalidRequest schema

```
{
  "message": "string"
}
```

AccessDenied schema

```
{
  "message": "string"
}
```

ResourceNotFound schema

```
{
  "message": "string"
}
```

ResourceConflict schema

```
{
  "message": "string"
}
```

MultiplexConfigurationValidationError schema

```
{
  "message": "string",
  "validationErrors": [
    {
```

```
      "elementPath": "string",
      "errorMessage": "string"
    }
  ]
}
```

LimitExceeded schema

```
{
  "message": "string"
}
```

InternalServiceError schema

```
{
  "message": "string"
}
```

BadGatewayException schema

```
{
  "message": "string"
}
```

GatewayTimeoutException schema

```
{
  "message": "string"
}
```

Properties

AccessDenied

message

Type: string
Required: False

BadGatewayException

message

Type: string
Required: False

GatewayTimeoutException

message

Type: string
Required: False

InternalServiceError

message

Type: string
Required: False

InvalidRequest

message

Type: string
Required: False

LimitExceeded

message

Type: string
Required: False

Multiplex

The multiplex object.

arn

The unique arn of the multiplex.

Type: string
Required: False

availabilityZones

A list of availability zones for the multiplex.

Type: Array of type string
Required: False

destinations

A list of the multiplex output destinations.

Type: Array of type MultiplexOutputDestination
Required: False

id

The unique id of the multiplex.

Type: string
Required: False

multiplexSettings

Configuration for a multiplex event.

Type: MultiplexSettings
Required: False

name

The name of the multiplex.

Type: string
Required: False

pipelinesRunningCount

The number of currently healthy pipelines.

Type: integer
Required: False

programCount

The number of programs in the multiplex.

Type: integer
Required: False

state

The current state of the multiplex.

Type: MultiplexState
Required: False

tags

A collection of key-value pairs.

Type: Tags
Required: False

MultiplexConfigurationValidationError

message

The error message.

Type: string

Required: False

validationErrors

A collection of validation error responses.

Type: Array of type ValidationError
Required: False

MultiplexMediaConnectOutputDestinationSettings

Multiplex MediaConnect output destination settings.

entitlementArn

The MediaConnect entitlement ARN available as a Flow source.

Type: string
Required: False
MinLength: 1

MultiplexOutputDestination

Multiplex output destination settings

mediaConnectSettings

Multiplex MediaConnect output destination settings.

Type: MultiplexMediaConnectOutputDestinationSettings
Required: False

MultiplexSettings

Contains configuration for a Multiplex event

maximumVideoBufferDelayMilliseconds

Maximum video buffer delay in milliseconds.

Type: integer
Required: False
Minimum: 800
Maximum: 3000

transportStreamBitrate

Transport stream bit rate.

Type: integer
Required: True
Minimum: 1000000
Maximum: 100000000

transportStreamId

Transport stream ID.

Type: integer
Required: True
Minimum: 0
Maximum: 65535

transportStreamReservedBitrate

Transport stream reserved bit rate.

Type: integer
Required: False
Minimum: 0
Maximum: 100000000

MultiplexState

The current state of the multiplex.

`CREATING`

CREATE_FAILED
IDLE
STARTING
RUNNING
RECOVERING
STOPPING
DELETING
DELETED

ResourceConflict

message

Type: string
Required: False

ResourceNotFound

message

Type: string
Required: False

Tags

key-value pairs

Type: string

UpdateMultiplex

multiplexSettings

The new settings for a multiplex.

Type: MultiplexSettings
Required: False

name

Name of the multiplex.

Type: string
Required: False

UpdateMultiplexResultModel

multiplex

The updated multiplex.

Type: Multiplex
Required: False

ValidationError

elementPath

Path to the source of the error.

Type: string
Required: False

errorMessage

The error message.

Type: string
Required: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

DeleteMultiplex

- AWS Command Line Interface

- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

DescribeMultiplex

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UpdateMultiplex

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python

- AWS SDK for Ruby V3

Offerings: describe offering

URI

`/prod/offerings/offeringId`

HTTP methods

GET

Operation ID: `DescribeOffering`

Path parameters

Name	Type	Required	Description
offeringId	String	True	

Responses

Status code	Response model	Description
200	Offering	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

Schemas

Response bodies

Offering schema

```
{
  "arn": "string",
  "currencyCode": "string",
  "duration": integer,
  "durationUnits": enum,
  "fixedPrice": number,
  "offeringDescription": "string",
  "offeringId": "string",
  "offeringType": enum,
  "region": "string",
  "resourceSpecification": {
    "channelClass": enum,
    "codec": enum,
    "maximumBitrate": enum,
    "maximumFramerate": enum,
    "resolution": enum,
    "resourceType": enum,
    "specialFeature": enum,
    "videoQuality": enum
  },
  "usagePrice": number
}
```

InvalidRequest schema

```
{
  "message": "string"
}
```

AccessDenied schema

```
{
  "message": "string"
}
```

ResourceNotFound schema

```
{
  "message": "string"
}
```

LimitExceeded schema

```
{
  "message": "string"
}
```

InternalServiceError schema

```
{
  "message": "string"
}
```

BadGatewayException schema

```
{
  "message": "string"
}
```

GatewayTimeoutException schema

```
{
  "message": "string"
}
```

Properties

AccessDenied

message

Type: string

Required: False

BadGatewayException

message

Type: string
Required: False

ChannelClass

A standard channel has two encoding pipelines and a single pipeline channel only has one.

STANDARD
SINGLE_PIPELINE

GatewayTimeoutException

message

Type: string
Required: False

InternalServiceError

message

Type: string
Required: False

InvalidRequest

message

Type: string
Required: False

LimitExceeded

message

Type: string
Required: False

Offering

Reserved resources available for purchase

arn

Unique offering ARN, e.g. 'arn:aws:medialive:us-west-2:123456789012:offering:87654321'

Type: string
Required: False

currencyCode

Currency code for usagePrice and fixedPrice in ISO-4217 format, e.g. 'USD'

Type: string
Required: False

duration

Lease duration, e.g. '12'

Type: integer
Required: False

durationUnits

Units for duration, e.g. 'MONTHS'

Type: OfferingDurationUnits
Required: False

fixedPrice

One-time charge for each reserved resource, e.g. '0.0' for a NO_UPFRONT offering

Type: number
Required: False

offeringDescription

Offering description, e.g. 'HD AVC output at 10-20 Mbps, 30 fps, and standard VQ in US West (Oregon)'

Type: string
Required: False

offeringId

Unique offering ID, e.g. '87654321'

Type: string
Required: False

offeringType

Offering type, e.g. 'NO_UPFRONT'

Type: OfferingType
Required: False

region

AWS Region, e.g. 'us-west-2'

Type: string
Required: False

resourceSpecification

Resource configuration details

Type: ReservationResourceSpecification
Required: False

usagePrice

Recurring usage charge for each reserved resource, e.g. '157.0'

Type: number
Required: False

OfferingDurationUnits

Units for duration, e.g. 'MONTHS'

MONTHS

OfferingType

Offering type, e.g. 'NO_UPFRONT'

NO_UPFRONT

ReservationCodec

Codec, 'MPEG2', 'AVC', 'HEVC', or 'AUDIO'

MPEG2
AVC
HEVC
AUDIO
LINK

ReservationMaximumBitrate

Maximum bitrate in megabits per second

MAX_10_MBPS

MAX_20_MBPS

MAX_50_MBPS

ReservationMaximumFramerate

Maximum framerate in frames per second (Outputs only)

MAX_30_FPS

MAX_60_FPS

ReservationResolution

Resolution based on lines of vertical resolution; SD is less than 720 lines, HD is 720 to 1080 lines, FHD is 1080 lines, UHD is greater than 1080 lines

SD

HD

FHD

UHD

ReservationResourceSpecification

Resource configuration (codec, resolution, bitrate, ...)

channelClass

Channel class, e.g. 'STANDARD'

Type: ChannelClass
Required: False

codec

Codec, e.g. 'AVC'

Type: ReservationCodec
Required: False

maximumBitrate

Maximum bitrate, e.g. 'MAX_20_MBPS'

Type: ReservationMaximumBitrate
Required: False

maximumFramerate

Maximum framerate, e.g. 'MAX_30_FPS' (Outputs only)

Type: ReservationMaximumFramerate
Required: False

resolution

Resolution, e.g. 'HD'

Type: ReservationResolution
Required: False

resourceType

Resource type, 'INPUT', 'OUTPUT', 'MULTIPLEX', or 'CHANNEL'

Type: ReservationResourceType
Required: False

specialFeature

Special feature, e.g. 'AUDIO_NORMALIZATION' (Channels only)

Type: ReservationSpecialFeature
Required: False

videoQuality

Video quality, e.g. 'STANDARD' (Outputs only)

Type: ReservationVideoQuality

Required: False

ReservationResourceType

Resource type, 'INPUT', 'OUTPUT', 'MULTIPLEX', or 'CHANNEL'

```
INPUT
OUTPUT
MULTIPLEX
CHANNEL
```

ReservationSpecialFeature

Special features, 'ADVANCED_AUDIO' 'AUDIO_NORMALIZATION' 'MGHD' or 'MGUHD'

```
ADVANCED_AUDIO
AUDIO_NORMALIZATION
MGHD
MGUHD
```

ReservationVideoQuality

Video quality, e.g. 'STANDARD' (Outputs only)

```
STANDARD
ENHANCED
PREMIUM
```

ResourceNotFound

message

Type: string

Required: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

DescribeOffering

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Offerings: list offerings

URI

`/prod/offerings`

HTTP methods

GET

Operation ID: `ListOfferings`

Query parameters

Name	Type	Required	Description
`resourceType`	String	False	
`nextToken`	String	False	

Name	Type	Required	Description
channelConfiguration	String	False	
duration	String	False	
codec	String	False	
videoQuality	String	False	
resolution	String	False	
maximumFramerate	String	False	
channelClass	String	False	
maxResults	String	False	
maximumBitrate	String	False	
specialFeature	String	False	

Responses

Status code	Response model	Description
200	ListOfferingsResultModel	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response

Status code	Response model	Description
504	GatewayTimeoutException	504 response

Schemas

Response bodies

ListOfferingsResultModel schema

```
{
  "nextToken": "string",
  "offerings": [
    {
      "arn": "string",
      "currencyCode": "string",
      "duration": integer,
      "durationUnits": enum,
      "fixedPrice": number,
      "offeringDescription": "string",
      "offeringId": "string",
      "offeringType": enum,
      "region": "string",
      "resourceSpecification": {
        "channelClass": enum,
        "codec": enum,
        "maximumBitrate": enum,
        "maximumFramerate": enum,
        "resolution": enum,
        "resourceType": enum,
        "specialFeature": enum,
        "videoQuality": enum
      },
      "usagePrice": number
    }
  ]
}
```

InvalidRequest schema

```
{
  "message": "string"
}
```

AccessDenied schema

```
{
  "message": "string"
}
```

LimitExceeded schema

```
{
  "message": "string"
}
```

InternalServiceError schema

```
{
  "message": "string"
}
```

BadGatewayException schema

```
{
  "message": "string"
}
```

GatewayTimeoutException schema

```
{
  "message": "string"
}
```

Properties

AccessDenied

message

Type: string
Required: False

BadGatewayException

message

Type: string
Required: False

ChannelClass

A standard channel has two encoding pipelines and a single pipeline channel only has one.

STANDARD
SINGLE_PIPELINE

GatewayTimeoutException

message

Type: string
Required: False

InternalServiceError

message

Type: string
Required: False

InvalidRequest

message

Type: string
Required: False

LimitExceeded

message

Type: string
Required: False

ListOfferingsResultModel

ListOfferings response

nextToken

Token to retrieve the next page of results

Type: string
Required: False

offerings

List of offerings

Type: Array of type Offering
Required: False

Offering

Reserved resources available for purchase

arn

Unique offering ARN, e.g. 'arn:aws:medialive:us-west-2:123456789012:offering:87654321'

Type: string
Required: False

currencyCode

Currency code for usagePrice and fixedPrice in ISO-4217 format, e.g. 'USD'

Type: string
Required: False

duration

Lease duration, e.g. '12'

Type: integer
Required: False

durationUnits

Units for duration, e.g. 'MONTHS'

Type: OfferingDurationUnits
Required: False

fixedPrice

One-time charge for each reserved resource, e.g. '0.0' for a NO_UPFRONT offering

Type: number
Required: False

offeringDescription

Offering description, e.g. 'HD AVC output at 10-20 Mbps, 30 fps, and standard VQ in US West (Oregon)'

Type: string
Required: False

offeringId

Unique offering ID, e.g. '87654321'

Type: string
Required: False

offeringType

Offering type, e.g. 'NO_UPFRONT'

Type: OfferingType
Required: False

region

AWS Region, e.g. 'us-west-2'

Type: string
Required: False

resourceSpecification

Resource configuration details

Type: ReservationResourceSpecification
Required: False

usagePrice

Recurring usage charge for each reserved resource, e.g. '157.0'

Type: number
Required: False

OfferingDurationUnits

Units for duration, e.g. 'MONTHS'

MONTHS

OfferingType

Offering type, e.g. 'NO_UPFRONT'

NO_UPFRONT

ReservationCodec

Codec, 'MPEG2', 'AVC', 'HEVC', or 'AUDIO'

MPEG2
AVC
HEVC
AUDIO
LINK

ReservationMaximumBitrate

Maximum bitrate in megabits per second

MAX_10_MBPS
MAX_20_MBPS
MAX_50_MBPS

ReservationMaximumFramerate

Maximum framerate in frames per second (Outputs only)

MAX_30_FPS
MAX_60_FPS

ReservationResolution

Resolution based on lines of vertical resolution; SD is less than 720 lines, HD is 720 to 1080 lines, FHD is 1080 lines, UHD is greater than 1080 lines

SD

HD

FHD

UHD

ReservationResourceSpecification

Resource configuration (codec, resolution, bitrate, ...)

channelClass

Channel class, e.g. 'STANDARD'

Type: ChannelClass
Required: False

codec

Codec, e.g. 'AVC'

Type: ReservationCodec
Required: False

maximumBitrate

Maximum bitrate, e.g. 'MAX_20_MBPS'

Type: ReservationMaximumBitrate
Required: False

maximumFramerate

Maximum framerate, e.g. 'MAX_30_FPS' (Outputs only)

Type: ReservationMaximumFramerate
Required: False

resolution

Resolution, e.g. 'HD'

Type: ReservationResolution
Required: False

resourceType

Resource type, 'INPUT', 'OUTPUT', 'MULTIPLEX', or 'CHANNEL'

Type: ReservationResourceType
Required: False

specialFeature

Special feature, e.g. 'AUDIO_NORMALIZATION' (Channels only)

Type: ReservationSpecialFeature
Required: False

videoQuality

Video quality, e.g. 'STANDARD' (Outputs only)

Type: ReservationVideoQuality
Required: False

ReservationResourceType

Resource type, 'INPUT', 'OUTPUT', 'MULTIPLEX', or 'CHANNEL'

```
INPUT
OUTPUT
MULTIPLEX
CHANNEL
```

ReservationSpecialFeature

Special features, 'ADVANCED_AUDIO' 'AUDIO_NORMALIZATION' 'MGHD' or 'MGUHD'

ADVANCED_AUDIO

AUDIO_NORMALIZATION

MGHD

MGUHD

ReservationVideoQuality

Video quality, e.g. 'STANDARD' (Outputs only)

STANDARD

ENHANCED

PREMIUM

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

ListOfferings

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Offerings: purchase offering

URI

/prod/offerings/*offeringId*/purchase

HTTP methods

POST

Operation ID: PurchaseOffering

Path parameters

Name	Type	Required	Description
offeringId	String	True	

Responses

Status code	Response model	Description
201	PurchaseOfferingResultModel	201 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
409	ResourceConflict	409 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

Schemas

Request bodies

POST schema

```
{
  "count": integer,
  "name": "string",
  "requestId": "string",
  "start": "string",
  "tags": {
  }
}
```

Response bodies

PurchaseOfferingResultModel schema

```
{
  "reservation": {
    "arn": "string",
    "count": integer,
    "currencyCode": "string",
    "duration": integer,
    "durationUnits": enum,
    "end": "string",
    "fixedPrice": number,
    "name": "string",
    "offeringDescription": "string",
    "offeringId": "string",
    "offeringType": enum,
    "region": "string",
    "reservationId": "string",
    "resourceSpecification": {
      "channelClass": enum,
      "codec": enum,
      "maximumBitrate": enum,
      "maximumFramerate": enum,
      "resolution": enum,
      "resourceType": enum,
      "specialFeature": enum,
```

```
      "videoQuality": enum
    },
    "start": "string",
    "state": enum,
    "tags": {
    },
    "usagePrice": number
  }
}
```

InvalidRequest schema

```
{
  "message": "string"
}
```

AccessDenied schema

```
{
  "message": "string"
}
```

ResourceNotFound schema

```
{
  "message": "string"
}
```

ResourceConflict schema

```
{
  "message": "string"
}
```

LimitExceeded schema

```
{
```

```
  "message": "string"
}
```

InternalServiceError schema

```
{
  "message": "string"
}
```

BadGatewayException schema

```
{
  "message": "string"
}
```

GatewayTimeoutException schema

```
{
  "message": "string"
}
```

Properties

AccessDenied

message

Type: string

Required: False

BadGatewayException

message

Type: string

Required: False

ChannelClass

A standard channel has two encoding pipelines and a single pipeline channel only has one.

```
STANDARD
SINGLE_PIPELINE
```

GatewayTimeoutException

message

Type: string
Required: False

InternalServiceError

message

Type: string
Required: False

InvalidRequest

message

Type: string
Required: False

LimitExceeded

message

Type: string
Required: False

OfferingDurationUnits

Units for duration, e.g. 'MONTHS'

MONTHS

OfferingType

Offering type, e.g. 'NO_UPFRONT'

NO_UPFRONT

PurchaseOffering

PurchaseOffering request

count

Number of resources

Type: integer
Required: True
Minimum: 1

name

Name for the new reservation

Type: string
Required: False

requestId

Unique request ID to be specified. This is needed to prevent retries from creating multiple resources.

Type: string
Required: False

start

Requested reservation start time (UTC) in ISO-8601 format. The specified time must be between the first day of the current month and one year from now. If no value is given, the default is now.

Type: string
Required: False

tags

A collection of key-value pairs

Type: Tags
Required: False

PurchaseOfferingResultModel

PurchaseOffering response

reservation

Type: Reservation
Required: False

Reservation

Reserved resources available to use

arn

Unique reservation ARN, e.g. 'arn:aws:medialive:us-west-2:123456789012:reservation:1234567'

Type: string
Required: False

count

Number of reserved resources

Type: integer
Required: False

currencyCode

Currency code for usagePrice and fixedPrice in ISO-4217 format, e.g. 'USD'

Type: string
Required: False

duration

Lease duration, e.g. '12'

Type: integer
Required: False

durationUnits

Units for duration, e.g. 'MONTHS'

Type: OfferingDurationUnits
Required: False

end

Reservation UTC end date and time in ISO-8601 format, e.g. '2019-03-01T00:00:00'

Type: string
Required: False

fixedPrice

One-time charge for each reserved resource, e.g. '0.0' for a NO_UPFRONT offering

Type: number
Required: False

name

User specified reservation name

Type: string
Required: False

offeringDescription

Offering description, e.g. 'HD AVC output at 10-20 Mbps, 30 fps, and standard VQ in US West (Oregon)'

Type: string
Required: False

offeringId

Unique offering ID, e.g. '87654321'

Type: string
Required: False

offeringType

Offering type, e.g. 'NO_UPFRONT'

Type: OfferingType
Required: False

region

AWS Region, e.g. 'us-west-2'

Type: string
Required: False

reservationId

Unique reservation ID, e.g. '1234567'

Type: string
Required: False

resourceSpecification

Resource configuration details

Type: ReservationResourceSpecification
Required: False

start

Reservation UTC start date and time in ISO-8601 format, e.g. '2018-03-01T00:00:00'

Type: string
Required: False

state

Current state of reservation, e.g. 'ACTIVE'

Type: ReservationState
Required: False

tags

A collection of key-value pairs

Type: Tags
Required: False

usagePrice

Recurring usage charge for each reserved resource, e.g. '157.0'

Type: number
Required: False

ReservationCodec

Codec, 'MPEG2', 'AVC', 'HEVC', or 'AUDIO'

MPEG2
AVC
HEVC

AUDIO
LINK

ReservationMaximumBitrate

Maximum bitrate in megabits per second

MAX_10_MBPS
MAX_20_MBPS
MAX_50_MBPS

ReservationMaximumFramerate

Maximum framerate in frames per second (Outputs only)

MAX_30_FPS
MAX_60_FPS

ReservationResolution

Resolution based on lines of vertical resolution; SD is less than 720 lines, HD is 720 to 1080 lines, FHD is 1080 lines, UHD is greater than 1080 lines

SD
HD
FHD
UHD

ReservationResourceSpecification

Resource configuration (codec, resolution, bitrate, ...)

channelClass

Channel class, e.g. 'STANDARD'

Type: ChannelClass
Required: False

codec

Codec, e.g. 'AVC'

Type: ReservationCodec
Required: False

maximumBitrate

Maximum bitrate, e.g. 'MAX_20_MBPS'

Type: ReservationMaximumBitrate
Required: False

maximumFramerate

Maximum framerate, e.g. 'MAX_30_FPS' (Outputs only)

Type: ReservationMaximumFramerate
Required: False

resolution

Resolution, e.g. 'HD'

Type: ReservationResolution
Required: False

resourceType

Resource type, 'INPUT', 'OUTPUT', 'MULTIPLEX', or 'CHANNEL'

Type: ReservationResourceType
Required: False

specialFeature

Special feature, e.g. 'AUDIO_NORMALIZATION' (Channels only)

Type: ReservationSpecialFeature

Required: False

videoQuality

Video quality, e.g. 'STANDARD' (Outputs only)

Type: ReservationVideoQuality

Required: False

ReservationResourceType

Resource type, 'INPUT', 'OUTPUT', 'MULTIPLEX', or 'CHANNEL'

```
INPUT
OUTPUT
MULTIPLEX
CHANNEL
```

ReservationSpecialFeature

Special features, 'ADVANCED_AUDIO' 'AUDIO_NORMALIZATION' 'MGHD' or 'MGUHD'

```
ADVANCED_AUDIO
AUDIO_NORMALIZATION
MGHD
MGUHD
```

ReservationState

Current reservation state

```
ACTIVE
EXPIRED
CANCELED
DELETED
```

ReservationVideoQuality

Video quality, e.g. 'STANDARD' (Outputs only)

STANDARD
ENHANCED
PREMIUM

ResourceConflict

message

Type: string
Required: False

ResourceNotFound

message

Type: string
Required: False

Tags

key-value pairs

Type: string

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

PurchaseOffering

- AWS Command Line Interface
- AWS SDK for .NET

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Reservations: describe reservation

URI

`/prod/reservations/reservationId`

HTTP methods

DELETE

Operation ID: `DeleteReservation`

Path parameters

Name	Type	Required	Description
reservationId	String	True	

Responses

Status code	Response model	Description
200	Reservation	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response

Status code	Response model	Description
409	ResourceConflict	409 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

GET

Operation ID: DescribeReservation

Path parameters

Name	Type	Required	Description
reservationId	String	True	

Responses

Status code	Response model	Description
200	Reservation	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response

Status code	Response model	Description
504	GatewayTimeoutException	504 response

PUT

Operation ID: UpdateReservation

Path parameters

Name	Type	Required	Description
reservationId	String	True	

Responses

Status code	Response model	Description
200	UpdateReservationResultModel	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
409	ResourceConflict	409 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

Schemas

Request bodies

PUT schema

```
{
  "name": "string"
}
```

Response bodies

Reservation schema

```
{
  "arn": "string",
  "count": integer,
  "currencyCode": "string",
  "duration": integer,
  "durationUnits": enum,
  "end": "string",
  "fixedPrice": number,
  "name": "string",
  "offeringDescription": "string",
  "offeringId": "string",
  "offeringType": enum,
  "region": "string",
  "reservationId": "string",
  "resourceSpecification": {
    "channelClass": enum,
    "codec": enum,
    "maximumBitrate": enum,
    "maximumFramerate": enum,
    "resolution": enum,
    "resourceType": enum,
    "specialFeature": enum,
    "videoQuality": enum
  },
  "start": "string",
  "state": enum,
  "tags": {
  },
```

```
  "usagePrice": number
}
```

UpdateReservationResultModel schema

```
{
  "reservation": {
    "arn": "string",
    "count": integer,
    "currencyCode": "string",
    "duration": integer,
    "durationUnits": enum,
    "end": "string",
    "fixedPrice": number,
    "name": "string",
    "offeringDescription": "string",
    "offeringId": "string",
    "offeringType": enum,
    "region": "string",
    "reservationId": "string",
    "resourceSpecification": {
      "channelClass": enum,
      "codec": enum,
      "maximumBitrate": enum,
      "maximumFramerate": enum,
      "resolution": enum,
      "resourceType": enum,
      "specialFeature": enum,
      "videoQuality": enum
    },
    "start": "string",
    "state": enum,
    "tags": {
    },
    "usagePrice": number
  }
}
```

InvalidRequest schema

```
{
```

```
  "message": "string"
}
```

AccessDenied schema

```
{
  "message": "string"
}
```

ResourceNotFound schema

```
{
  "message": "string"
}
```

ResourceConflict schema

```
{
  "message": "string"
}
```

LimitExceeded schema

```
{
  "message": "string"
}
```

InternalServiceError schema

```
{
  "message": "string"
}
```

BadGatewayException schema

```
{
```

```
  "message": "string"
}
```

GatewayTimeoutException schema

```
{
  "message": "string"
}
```

Properties

AccessDenied

message

Type: string
Required: False

BadGatewayException

message

Type: string
Required: False

ChannelClass

A standard channel has two encoding pipelines and a single pipeline channel only has one.

STANDARD
SINGLE_PIPELINE

GatewayTimeoutException

message

Type: string

Required: False

InternalServiceError

message

Type: string
Required: False

InvalidRequest

message

Type: string
Required: False

LimitExceeded

message

Type: string
Required: False

OfferingDurationUnits

Units for duration, e.g. 'MONTHS'

`MONTHS`

OfferingType

Offering type, e.g. 'NO_UPFRONT'

`NO_UPFRONT`

Reservation

Reserved resources available to use

arn

Unique reservation ARN, e.g. 'arn:aws:medialive:us-west-2:123456789012:reservation:1234567'

Type: string
Required: False

count

Number of reserved resources

Type: integer
Required: False

currencyCode

Currency code for usagePrice and fixedPrice in ISO-4217 format, e.g. 'USD'

Type: string
Required: False

duration

Lease duration, e.g. '12'

Type: integer
Required: False

durationUnits

Units for duration, e.g. 'MONTHS'

Type: OfferingDurationUnits
Required: False

end

Reservation UTC end date and time in ISO-8601 format, e.g. '2019-03-01T00:00:00'

Type: string
Required: False

fixedPrice

One-time charge for each reserved resource, e.g. '0.0' for a NO_UPFRONT offering

Type: number
Required: False

name

User specified reservation name

Type: string
Required: False

offeringDescription

Offering description, e.g. 'HD AVC output at 10-20 Mbps, 30 fps, and standard VQ in US West (Oregon)'

Type: string
Required: False

offeringId

Unique offering ID, e.g. '87654321'

Type: string
Required: False

offeringType

Offering type, e.g. 'NO_UPFRONT'

Type: OfferingType
Required: False

region

AWS Region, e.g. 'us-west-2'

Type: string
Required: False

reservationId

Unique reservation ID, e.g. '1234567'

Type: string
Required: False

resourceSpecification

Resource configuration details

Type: ReservationResourceSpecification
Required: False

start

Reservation UTC start date and time in ISO-8601 format, e.g. '2018-03-01T00:00:00'

Type: string
Required: False

state

Current state of reservation, e.g. 'ACTIVE'

Type: ReservationState
Required: False

tags

A collection of key-value pairs

Type: Tags

Required: False

usagePrice

Recurring usage charge for each reserved resource, e.g. '157.0'

Type: number
Required: False

ReservationCodec

Codec, 'MPEG2', 'AVC', 'HEVC', or 'AUDIO'

```
MPEG2
AVC
HEVC
AUDIO
LINK
```

ReservationMaximumBitrate

Maximum bitrate in megabits per second

```
MAX_10_MBPS
MAX_20_MBPS
MAX_50_MBPS
```

ReservationMaximumFramerate

Maximum framerate in frames per second (Outputs only)

```
MAX_30_FPS
MAX_60_FPS
```

ReservationResolution

Resolution based on lines of vertical resolution; SD is less than 720 lines, HD is 720 to 1080 lines, FHD is 1080 lines, UHD is greater than 1080 lines

SD

HD

FHD

UHD

ReservationResourceSpecification

Resource configuration (codec, resolution, bitrate, ...)

channelClass

Channel class, e.g. 'STANDARD'

Type: ChannelClass
Required: False

codec

Codec, e.g. 'AVC'

Type: ReservationCodec
Required: False

maximumBitrate

Maximum bitrate, e.g. 'MAX_20_MBPS'

Type: ReservationMaximumBitrate
Required: False

maximumFramerate

Maximum framerate, e.g. 'MAX_30_FPS' (Outputs only)

Type: ReservationMaximumFramerate
Required: False

resolution

Resolution, e.g. 'HD'

Type: ReservationResolution
Required: False

resourceType

Resource type, 'INPUT', 'OUTPUT', 'MULTIPLEX', or 'CHANNEL'

Type: ReservationResourceType
Required: False

specialFeature

Special feature, e.g. 'AUDIO_NORMALIZATION' (Channels only)

Type: ReservationSpecialFeature
Required: False

videoQuality

Video quality, e.g. 'STANDARD' (Outputs only)

Type: ReservationVideoQuality
Required: False

ReservationResourceType

Resource type, 'INPUT', 'OUTPUT', 'MULTIPLEX', or 'CHANNEL'

`INPUT`
`OUTPUT`
`MULTIPLEX`
`CHANNEL`

ReservationSpecialFeature

Special features, 'ADVANCED_AUDIO' 'AUDIO_NORMALIZATION' 'MGHD' or 'MGUHD'

ADVANCED_AUDIO
AUDIO_NORMALIZATION
MGHD
MGUHD

ReservationState

Current reservation state

ACTIVE
EXPIRED
CANCELED
DELETED

ReservationVideoQuality

Video quality, e.g. 'STANDARD' (Outputs only)

STANDARD
ENHANCED
PREMIUM

ResourceConflict

message

Type: string
Required: False

ResourceNotFound

message

Type: string

Required: False

Tags

key-value pairs

Type: string

UpdateReservation

UpdateReservation request

name

Name of the reservation

Type: string
Required: False

UpdateReservationResultModel

UpdateReservation response

reservation

Type: Reservation
Required: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

DeleteReservation

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++

- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

DescribeReservation

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UpdateReservation

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Reservations: list reservations

URI

`/prod/reservations`

HTTP methods

GET

Operation ID: `ListReservations`

Query parameters

Name	Type	Required	Description
`resourceType`	String	False	
`nextToken`	String	False	
`codec`	String	False	
`videoQuality`	String	False	
`resolution`	String	False	
`maximumFramerate`	String	False	
`channelClass`	String	False	
`maxResults`	String	False	
`maximumBitrate`	String	False	
`specialFeature`	String	False	

Responses

Status code	Response model	Description
200	ListReservationsResultModel	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
429	LimitExceeded	429 response
500	InternalServiceError	500 response
502	BadGatewayException	502 response
504	GatewayTimeoutException	504 response

Schemas

Response bodies

ListReservationsResultModel schema

```
{
  "nextToken": "string",
  "reservations": [
    {
      "arn": "string",
      "count": integer,
      "currencyCode": "string",
      "duration": integer,
      "durationUnits": enum,
      "end": "string",
      "fixedPrice": number,
      "name": "string",
      "offeringDescription": "string",
      "offeringId": "string",
      "offeringType": enum,
```

```
      "region": "string",
      "reservationId": "string",
      "resourceSpecification": {
        "channelClass": enum,
        "codec": enum,
        "maximumBitrate": enum,
        "maximumFramerate": enum,
        "resolution": enum,
        "resourceType": enum,
        "specialFeature": enum,
        "videoQuality": enum
      },
      "start": "string",
      "state": enum,
      "tags": {
      },
      "usagePrice": number
    }
  ]
}
```

InvalidRequest schema

```
{
  "message": "string"
}
```

AccessDenied schema

```
{
  "message": "string"
}
```

LimitExceeded schema

```
{
  "message": "string"
}
```

InternalServiceError schema

```
{
  "message": "string"
}
```

BadGatewayException schema

```
{
  "message": "string"
}
```

GatewayTimeoutException schema

```
{
  "message": "string"
}
```

Properties

AccessDenied

message

Type: string
Required: False

BadGatewayException

message

Type: string
Required: False

ChannelClass

A standard channel has two encoding pipelines and a single pipeline channel only has one.

STANDARD
SINGLE_PIPELINE

GatewayTimeoutException

message

Type: string
Required: False

InternalServiceError

message

Type: string
Required: False

InvalidRequest

message

Type: string
Required: False

LimitExceeded

message

Type: string
Required: False

ListReservationsResultModel

ListReservations response

nextToken

Token to retrieve the next page of results

Type: string
Required: False

reservations

List of reservations

Type: Array of type Reservation
Required: False

OfferingDurationUnits

Units for duration, e.g. 'MONTHS'

```
MONTHS
```

OfferingType

Offering type, e.g. 'NO_UPFRONT'

```
NO_UPFRONT
```

Reservation

Reserved resources available to use

arn

Unique reservation ARN, e.g. 'arn:aws:medialive:us-west-2:123456789012:reservation:1234567'

Type: string
Required: False

count

Number of reserved resources

Type: integer
Required: False

currencyCode

Currency code for usagePrice and fixedPrice in ISO-4217 format, e.g. 'USD'

Type: string
Required: False

duration

Lease duration, e.g. '12'

Type: integer
Required: False

durationUnits

Units for duration, e.g. 'MONTHS'

Type: OfferingDurationUnits
Required: False

end

Reservation UTC end date and time in ISO-8601 format, e.g. '2019-03-01T00:00:00'

Type: string
Required: False

fixedPrice

One-time charge for each reserved resource, e.g. '0.0' for a NO_UPFRONT offering

Type: number
Required: False

name

User specified reservation name

Type: string
Required: False

offeringDescription

Offering description, e.g. 'HD AVC output at 10-20 Mbps, 30 fps, and standard VQ in US West (Oregon)'

Type: string
Required: False

offeringId

Unique offering ID, e.g. '87654321'

Type: string
Required: False

offeringType

Offering type, e.g. 'NO_UPFRONT'

Type: OfferingType
Required: False

region

AWS Region, e.g. 'us-west-2'

Type: string
Required: False

reservationId

Unique reservation ID, e.g. '1234567'

Type: string
Required: False

resourceSpecification

Resource configuration details

Type: ReservationResourceSpecification
Required: False

start

Reservation UTC start date and time in ISO-8601 format, e.g. '2018-03-01T00:00:00'

Type: string
Required: False

state

Current state of reservation, e.g. 'ACTIVE'

Type: ReservationState
Required: False

tags

A collection of key-value pairs

Type: Tags
Required: False

usagePrice

Recurring usage charge for each reserved resource, e.g. '157.0'

Type: number
Required: False

ReservationCodec

Codec, 'MPEG2', 'AVC', 'HEVC', or 'AUDIO'

MPEG2
AVC
HEVC
AUDIO
LINK

ReservationMaximumBitrate

Maximum bitrate in megabits per second

MAX_10_MBPS
MAX_20_MBPS
MAX_50_MBPS

ReservationMaximumFramerate

Maximum framerate in frames per second (Outputs only)

MAX_30_FPS
MAX_60_FPS

ReservationResolution

Resolution based on lines of vertical resolution; SD is less than 720 lines, HD is 720 to 1080 lines, FHD is 1080 lines, UHD is greater than 1080 lines

SD
HD
FHD
UHD

ReservationResourceSpecification

Resource configuration (codec, resolution, bitrate, ...)

channelClass

Channel class, e.g. 'STANDARD'

Type: ChannelClass
Required: False

codec

Codec, e.g. 'AVC'

Type: ReservationCodec
Required: False

maximumBitrate

Maximum bitrate, e.g. 'MAX_20_MBPS'

Type: ReservationMaximumBitrate
Required: False

maximumFramerate

Maximum framerate, e.g. 'MAX_30_FPS' (Outputs only)

Type: ReservationMaximumFramerate
Required: False

resolution

Resolution, e.g. 'HD'

Type: ReservationResolution
Required: False

resourceType

Resource type, 'INPUT', 'OUTPUT', 'MULTIPLEX', or 'CHANNEL'

Type: ReservationResourceType
Required: False

specialFeature

Special feature, e.g. 'AUDIO_NORMALIZATION' (Channels only)

Type: ReservationSpecialFeature
Required: False

videoQuality

Video quality, e.g. 'STANDARD' (Outputs only)

Type: ReservationVideoQuality
Required: False

ReservationResourceType

Resource type, 'INPUT', 'OUTPUT', 'MULTIPLEX', or 'CHANNEL'

```
INPUT
OUTPUT
MULTIPLEX
CHANNEL
```

ReservationSpecialFeature

Special features, 'ADVANCED_AUDIO' 'AUDIO_NORMALIZATION' 'MGHD' or 'MGUHD'

```
ADVANCED_AUDIO
AUDIO_NORMALIZATION
MGHD
MGUHD
```

ReservationState

Current reservation state

```
ACTIVE
```

EXPIRED

CANCELED

DELETED

ReservationVideoQuality

Video quality, e.g. 'STANDARD' (Outputs only)

STANDARD

ENHANCED

PREMIUM

Tags

key-value pairs

Type: string

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

ListReservations

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Tags: list tags

URI

`/prod/tags/resource-arn`

HTTP methods

DELETE

Operation ID: `DeleteTags`

Path parameters

Name	Type	Required	Description
resource-arn	String	True	

Query parameters

Name	Type	Required	Description
tagKeys	String	True	

Responses

Status code	Response model	Description
204	None	204 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
500	InternalServiceError	500 response

GET

Operation ID: `ListTagsForResource`

Path parameters

Name	Type	Required	Description
resource-arn	String	True	

Responses

Status code	Response model	Description
200	TagsModel	200 response
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
500	InternalServiceError	500 response

POST

Operation ID: `CreateTags`

Path parameters

Name	Type	Required	Description
resource-arn	String	True	

Responses

Status code	Response model	Description
204	None	204 response

Status code	Response model	Description
400	InvalidRequest	400 response
403	AccessDenied	403 response
404	ResourceNotFound	404 response
500	InternalServiceError	500 response

Schemas

Request bodies

POST schema

```
{
  "tags": {
  }
}
```

Response bodies

TagsModel schema

```
{
  "tags": {
  }
}
```

InvalidRequest schema

```
{
  "message": "string"
}
```

AccessDenied schema

```
{
```

```
  "message": "string"
}
```

ResourceNotFound schema

```
{
  "message": "string"
}
```

InternalServiceError schema

```
{
  "message": "string"
}
```

Properties

AccessDenied

message

Type: string
Required: False

InternalServiceError

message

Type: string
Required: False

InvalidRequest

message

Type: string
Required: False

ResourceNotFound

message

Type: string
Required: False

Tags

key-value pairs

Type: string

TagsModel

tags

Type: Tags
Required: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

DeleteTags

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python

- AWS SDK for Ruby V3

ListTagsForResource

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

CreateTags

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Workflow monitor: CloudWatch alarm template groups

URI

```
/prod/cloudwatch-alarm-template-groups
```

HTTP methods

GET

Operation ID: `ListCloudWatchAlarmTemplateGroups`

Query parameters

Name	Type	Required	Description
`signalMapIdentifier`	String	False	
`scope`	String	False	
`nextToken`	String	False	
`maxResults`	String	False	

Responses

Status code	Response model	Description
200	ListCloudWatchAlarmTemplateGroupsResponseContent	200 response
400	BadRequestExceptionResponseContent	400 response
403	ForbiddenExceptionResponseContent	403 response
404	NotFoundExceptionResponseContent	404 response
429	TooManyRequestsExceptionResponseContent	429 response

Status code	Response model	Description
500	InternalServerErrorExceptionResponseContent	500 response

OPTIONS

Operation ID: CorsCloudwatch_alarm_template_groups

Responses

Status code	Response model	Description
200	None	200 response

POST

Operation ID: CreateCloudWatchAlarmTemplateGroup

Responses

Status code	Response model	Description
201	CreateCloudWatchAlarmTemplateGroupResponseContent	201 response
400	BadRequestExceptionResponseContent	400 response
403	ForbiddenExceptionResponseContent	403 response
404	NotFoundExceptionResponseContent	404 response
409	ConflictExceptionResponseContent	409 response

Status code	Response model	Description
429	TooManyRequestsExceptionResponseContent	429 response
500	InternalServerErrorExceptionResponseContent	500 response

Schemas

Request bodies

POST schema

```
{
  "description": "string",
  "name": "string"
}
```

Response bodies

ListCloudWatchAlarmTemplateGroupsResponseContent schema

```
{
  "cloudWatchAlarmTemplateGroups": [
    {
      "arn": "string",
      "createdAt": "string",
      "description": "string",
      "id": "string",
      "modifiedAt": "string",
      "name": "string",
      "templateCount": number
    }
  ],
  "nextToken": "string"
}
```

CreateCloudWatchAlarmTemplateGroupResponseContent schema

```
{
  "arn": "string",
  "createdAt": "string",
  "description": "string",
  "id": "string",
  "modifiedAt": "string",
  "name": "string"
}
```

BadRequestExceptionResponseContent schema

```
{
  "message": "string"
}
```

ForbiddenExceptionResponseContent schema

```
{
  "message": "string"
}
```

NotFoundExceptionResponseContent schema

```
{
  "message": "string"
}
```

ConflictExceptionResponseContent schema

```
{
  "message": "string"
}
```

TooManyRequestsExceptionResponseContent schema

```
{
```

```
  "message": "string"
}
```

InternalServerErrorExceptionResponseContent schema

```
{
  "message": "string"
}
```

Properties

BadRequestExceptionResponseContent

The input fails to satisfy the constraints specified by an Amazon Web Services service.

message

Exception error message.

Type: string
Required: False

CloudWatchAlarmTemplateGroupSummary

A summary of the alarm template groups.

arn

A cloudwatch alarm template group's ARN (Amazon Resource Name)

Type: string
Required: True
Pattern: `^arn:.+:medialive:.+:cloudwatch-alarm-template-group:.+$`

createdAt

The date and time of resource creation.

Type: string
Required: True

Format: date-time

description

A resource's optional description.

Type: string
Required: False
MinLength: 0
MaxLength: 1024

id

A CloudWatch alarm template group's id. Amazon Web Services provided template groups have ids that start with <code>`aws-`</code>

Type: string
Required: True
Pattern: `^(aws-)?[0-9]{7}$`
MinLength: 7
MaxLength: 11

modifiedAt

The date and time of latest resource modification.

Type: string
Required: False
Format: date-time

name

A resource's name. Names must be unique within the scope of a resource type in a specific region.

Type: string
Required: True
Pattern: `^[^\s]+$`
MinLength: 1

MaxLength: 255

templateCount

The number of templates in the group.

Type: number
Required: True

ConflictExceptionResponseContent

Updating or deleting a resource can cause an inconsistent state.

message

Exception error message.

Type: string
Required: False

CreateCloudWatchAlarmTemplateGroupRequestContent

description

A resource's optional description.

Type: string
Required: False
MinLength: 0
MaxLength: 1024

name

A resource's name. Names must be unique within the scope of a resource type in a specific region.

Type: string
Required: True
Pattern: ^[^\s]+$
MinLength: 1

MaxLength: 255

CreateCloudWatchAlarmTemplateGroupResponseContent

arn

A cloudwatch alarm template group's ARN (Amazon Resource Name)

Type: string
Required: True
Pattern: `^arn:.+:medialive:.+:cloudwatch-alarm-template-group:.+$`

createdAt

The date and time of resource creation.

Type: string
Required: True
Format: date-time

description

A resource's optional description.

Type: string
Required: False
MinLength: 0
MaxLength: 1024

id

A CloudWatch alarm template group's id. Amazon Web Services provided template groups have ids that start with <code>`aws-`</code>

Type: string
Required: True
Pattern: `^(aws-)?[0-9]{7}$`
MinLength: 7
MaxLength: 11

modifiedAt

The date and time of latest resource modification.

Type: string
Required: False
Format: date-time

name

A resource's name. Names must be unique within the scope of a resource type in a specific region.

Type: string
Required: True
Pattern: ^[^\s]+$
MinLength: 1
MaxLength: 255

ForbiddenExceptionResponseContent

User does not have sufficient access to perform this action.

message

Exception error message.

Type: string
Required: False

InternalServerErrorExceptionResponseContent

Unexpected error during processing of request.

message

Exception error message.

Type: string
Required: False

ListCloudWatchAlarmTemplateGroupsResponseContent

cloudWatchAlarmTemplateGroups

A summary of the alarm template groups.

Type: Array of type CloudWatchAlarmTemplateGroupSummary
Required: True

nextToken

A token used to retrieve the next set of results in paginated list responses.

Type: string
Required: False
MinLength: 1
MaxLength: 2048

NotFoundExceptionResponseContent

Request references a resource which does not exist.

message

Exception error message.

Type: string
Required: False

TooManyRequestsExceptionResponseContent

Request was denied due to request throttling.

message

Exception error message.

Type: string
Required: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

ListCloudWatchAlarmTemplateGroups

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

CorsCloudwatch_alarm_template_groups

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

CreateCloudWatchAlarmTemplateGroup

- AWS Command Line Interface
- AWS SDK for .NET

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Workflow monitor: CloudWatch alarm template groups ID

URI

`/prod/cloudwatch-alarm-template-groups/identifier`

HTTP methods

DELETE

Operation ID: `DeleteCloudWatchAlarmTemplateGroup`

Path parameters

Name	Type	Required	Description
identifier	String	True	

Responses

Status code	Response model	Description
204	None	204 response
400	BadRequestExceptionResponseContent	400 response
403	ForbiddenExceptionResponseContent	403 response

Status code	Response model	Description
404	NotFoundExceptionResponseContent	404 response
409	ConflictExceptionResponseContent	409 response
429	TooManyRequestsExceptionResponseContent	429 response
500	InternalServerErrorExceptionResponseContent	500 response

GET

Operation ID: GetCloudWatchAlarmTemplateGroup

Path parameters

Name	Type	Required	Description
identifier	String	True	

Responses

Status code	Response model	Description
200	GetCloudWatchAlarmTemplateGroupResponseContent	200 response
400	BadRequestExceptionResponseContent	400 response

Status code	Response model	Description
403	ForbiddenExceptionResponseContent	403 response
404	NotFoundExceptionResponseContent	404 response
429	TooManyRequestsExceptionResponseContent	429 response
500	InternalServerErrorExceptionResponseContent	500 response

OPTIONS

Operation ID: `CorsCloudwatch_alarm_template_groupsIdentifier`

Path parameters

Name	Type	Required	Description
identifier	String	True	

Responses

Status code	Response model	Description
200	None	200 response

PATCH

Operation ID: `UpdateCloudWatchAlarmTemplateGroup`

Path parameters

Name	Type	Required	Description
identifier	String	True	

Responses

Status code	Response model	Description
200	UpdateCloudWatchAlarmTemplateGroupResponseContent	200 response
400	BadRequestExceptionResponseContent	400 response
403	ForbiddenExceptionResponseContent	403 response
404	NotFoundExceptionResponseContent	404 response
409	ConflictExceptionResponseContent	409 response
429	TooManyRequestsExceptionResponseContent	429 response
500	InternalServerErrorExceptionResponseContent	500 response

Schemas

Request bodies

PATCH schema

```
{
  "description": "string"
}
```

Response bodies

GetCloudWatchAlarmTemplateGroupResponseContent schema

```
{
  "arn": "string",
  "createdAt": "string",
  "description": "string",
  "id": "string",
  "modifiedAt": "string",
  "name": "string"
}
```

UpdateCloudWatchAlarmTemplateGroupResponseContent schema

```
{
  "arn": "string",
  "createdAt": "string",
  "description": "string",
  "id": "string",
  "modifiedAt": "string",
  "name": "string"
}
```

BadRequestExceptionResponseContent schema

```
{
  "message": "string"
}
```

ForbiddenExceptionResponseContent schema

```
{
```

```
  "message": "string"
}
```

NotFoundExceptionResponseContent schema

```
{
  "message": "string"
}
```

ConflictExceptionResponseContent schema

```
{
  "message": "string"
}
```

TooManyRequestsExceptionResponseContent schema

```
{
  "message": "string"
}
```

InternalServerErrorExceptionResponseContent schema

```
{
  "message": "string"
}
```

Properties

BadRequestExceptionResponseContent

The input fails to satisfy the constraints specified by an Amazon Web Services service.

message

Exception error message.

Type: string
Required: False

ConflictExceptionResponseContent

Updating or deleting a resource can cause an inconsistent state.

message

Exception error message.

Type: string
Required: False

ForbiddenExceptionResponseContent

User does not have sufficient access to perform this action.

message

Exception error message.

Type: string
Required: False

GetCloudWatchAlarmTemplateGroupResponseContent

arn

A cloudwatch alarm template group's ARN (Amazon Resource Name)

Type: string
Required: True
Pattern: `^arn:.+:medialive:.+:cloudwatch-alarm-template-group:.+$`

createdAt

The date and time of resource creation.

Type: string
Required: True
Format: date-time

description

A resource's optional description.

Type: string
Required: False
MinLength: 0
MaxLength: 1024

id

A CloudWatch alarm template group's id. Amazon Web Services provided template groups have ids that start with <code>`aws-`</code>

Type: string
Required: True
Pattern: `^(aws-)?[0-9]{7}$`
MinLength: 7
MaxLength: 11

modifiedAt

The date and time of latest resource modification.

Type: string
Required: False
Format: date-time

name

A resource's name. Names must be unique within the scope of a resource type in a specific region.

Type: string
Required: True
Pattern: `^[^\s]+$`
MinLength: 1
MaxLength: 255

InternalServerErrorExceptionResponseContent

Unexpected error during processing of request.

message

Exception error message.

Type: string
Required: False

NotFoundExceptionResponseContent

Request references a resource which does not exist.

message

Exception error message.

Type: string
Required: False

TooManyRequestsExceptionResponseContent

Request was denied due to request throttling.

message

Exception error message.

Type: string
Required: False

UpdateCloudWatchAlarmTemplateGroupRequestContent

description

A resource's optional description.

Type: string
Required: False

MinLength: 0
MaxLength: 1024

UpdateCloudWatchAlarmTemplateGroupResponseContent

arn

A cloudwatch alarm template group's ARN (Amazon Resource Name)

Type: string
Required: True
Pattern: `^arn:.+:medialive:.+:cloudwatch-alarm-template-group:.+$`

createdAt

The date and time of resource creation.

Type: string
Required: True
Format: date-time

description

A resource's optional description.

Type: string
Required: False
MinLength: 0
MaxLength: 1024

id

A CloudWatch alarm template group's id. Amazon Web Services provided template groups have ids that start with <code>`aws-`</code>

Type: string
Required: True
Pattern: `^(aws-)?[0-9]{7}$`
MinLength: 7

MaxLength: 11

modifiedAt

The date and time of latest resource modification.

Type: string
Required: False
Format: date-time

name

A resource's name. Names must be unique within the scope of a resource type in a specific region.

Type: string
Required: True
Pattern: ^[^\s]+$
MinLength: 1
MaxLength: 255

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

DeleteCloudWatchAlarmTemplateGroup

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

GetCloudWatchAlarmTemplateGroup

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

CorsCloudwatch_alarm_template_groupsIdentifier

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UpdateCloudWatchAlarmTemplateGroup

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3

- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Workflow monitor: CloudWatch alarm templates

URI

`/prod/cloudwatch-alarm-templates`

HTTP methods

GET

Operation ID: `ListCloudWatchAlarmTemplates`

Query parameters

Name	Type	Required	Description
`groupIdentifier`	String	False	
`signalMapIdentifier`	String	False	
`scope`	String	False	
`nextToken`	String	False	
`maxResults`	String	False	

Responses

Status code	Response model	Description
`200`	`ListCloudWatchAlarmTemplatesResponseContent`	200 response

Status code	Response model	Description
400	BadRequestExceptionResponseContent	400 response
403	ForbiddenExceptionResponseContent	403 response
404	NotFoundExceptionResponseContent	404 response
429	TooManyRequestsExceptionResponseContent	429 response
500	InternalServerErrorExceptionResponseContent	500 response

OPTIONS

Operation ID: `CorsCloudwatch_alarm_templates`

Responses

Status code	Response model	Description
200	None	200 response

POST

Operation ID: `CreateCloudWatchAlarmTemplate`

Responses

Status code	Response model	Description
201	CreateCloudWatchAlarmTemplateResponseContent	201 response
400	BadRequestExceptionResponseContent	400 response
403	ForbiddenExceptionResponseContent	403 response
404	NotFoundExceptionResponseContent	404 response
409	ConflictExceptionResponseContent	409 response
429	TooManyRequestsExceptionResponseContent	429 response
500	InternalServerErrorExceptionResponseContent	500 response

Schemas

Request bodies

POST schema

```
{
  "comparisonOperator": enum,
  "datapointsToAlarm": number,
  "description": "string",
  "evaluationPeriods": number,
  "groupIdentifier": "string",
```

```
  "metricName": "string",
  "name": "string",
  "period": number,
  "standardDeviationThreshold": number,
  "statistic": enum,
  "targetResourceType": enum,
  "threshold": number,
  "treatMissingData": enum
}
```

Response bodies

ListCloudWatchAlarmTemplatesResponseContent schema

```
{
  "cloudWatchAlarmTemplates": [
    {
      "arn": "string",
      "comparisonOperator": enum,
      "createdAt": "string",
      "datapointsToAlarm": number,
      "description": "string",
      "evaluationPeriods": number,
      "groupId": "string",
      "id": "string",
      "metricName": "string",
      "modifiedAt": "string",
      "name": "string",
      "period": number,
      "standardDeviationThreshold": number,
      "statistic": enum,
      "targetResourceType": enum,
      "threshold": number,
      "treatMissingData": enum
    }
  ],
  "nextToken": "string"
}
```

CreateCloudWatchAlarmTemplateResponseContent schema

```
{
```

```
  "arn": "string",
  "comparisonOperator": enum,
  "createdAt": "string",
  "datapointsToAlarm": number,
  "description": "string",
  "evaluationPeriods": number,
  "groupId": "string",
  "id": "string",
  "metricName": "string",
  "modifiedAt": "string",
  "name": "string",
  "period": number,
  "standardDeviationThreshold": number,
  "statistic": enum,
  "targetResourceType": enum,
  "threshold": number,
  "treatMissingData": enum
}
```

BadRequestExceptionResponseContent schema

```
{
  "message": "string"
}
```

ForbiddenExceptionResponseContent schema

```
{
  "message": "string"
}
```

NotFoundExceptionResponseContent schema

```
{
  "message": "string"
}
```

ConflictExceptionResponseContent schema

```
{
```

```
  "message": "string"
}
```

TooManyRequestsExceptionResponseContent schema

```
{
  "message": "string"
}
```

InternalServerErrorExceptionResponseContent schema

```
{
  "message": "string"
}
```

Properties

BadRequestExceptionResponseContent

The input fails to satisfy the constraints specified by an Amazon Web Services service.

message

Exception error message.

Type: string
Required: False

CloudWatchAlarmTemplateComparisonOperator

The comparison operator used to compare the specified statistic and the threshold.

`GreaterThanOrEqualToThreshold`
`GreaterThanThreshold`
`LessThanThreshold`
`LessThanOrEqualToThreshold`
`LessThanLowerOrGreaterThanUpperThreshold`

`LessThanLowerThreshold`
`GreaterThanUpperThreshold`

CloudWatchAlarmTemplateStatistic

The statistic to apply to the alarm's metric data.

`SampleCount`
`Average`
`Sum`
`Minimum`
`Maximum`

CloudWatchAlarmTemplateSummary

A summary of the alarm templates.

arn

A cloudwatch alarm template's ARN (Amazon Resource Name)

Type: string
Required: True
Pattern: `^arn:.+:medialive:.+:cloudwatch-alarm-template:.+$`

comparisonOperator

The comparison operator used to compare the specified statistic and the threshold.

Type: CloudWatchAlarmTemplateComparisonOperator
Required: True

createdAt

The date and time of resource creation.

Type: string
Required: True

Format: date-time

datapointsToAlarm

The number of datapoints within the evaluation period that must be breaching to trigger the alarm.

Type: number
Required: False
Minimum: 1

description

A resource's optional description.

Type: string
Required: False
MinLength: 0
MaxLength: 1024

evaluationPeriods

The number of periods over which data is compared to the specified threshold.

Type: number
Required: True
Minimum: 1

groupId

A CloudWatch alarm template group's id. Amazon Web Services provided template groups have ids that start with <code>`aws-`</code>

Type: string
Required: True
Pattern: `^(aws-)?[0-9]{7}$`
MinLength: 7

MaxLength: 11

id

A cloudwatch alarm template's id. Amazon Web Services provided templates have ids that start with <code>`aws-`</code>.

Type: string
Required: True
Pattern: `^(aws-)?[0-9]{7}$`
MinLength: 7
MaxLength: 11

metricName

The name of the metric associated with the alarm. Must be compatible with targetResourceType.

Type: string
Required: True
MaxLength: 64

modifiedAt

The date and time of latest resource modification.

Type: string
Required: False
Format: date-time

name

A resource's name. Names must be unique within the scope of a resource type in a specific region.

Type: string
Required: True
Pattern: `^[^\s]+$`
MinLength: 1
MaxLength: 255

period

The period, in seconds, over which the specified statistic is applied.

Type: number
Required: True
Minimum: 10
Maximum: 86400

standardDeviationThreshold

The number of standard deviations by which the metric value can deviate from the expected value before the alarm is triggered.

Type: number
Required: False
Minimum: 1

statistic

The statistic to apply to the alarm's metric data.

Type: CloudWatchAlarmTemplateStatistic
Required: True

targetResourceType

The resource type this template should dynamically generate CloudWatch metric alarms for.

Type: CloudWatchAlarmTemplateTargetResourceType
Required: True

threshold

The threshold value to compare with the specified statistic.

Type: number
Required: False
Format: double

treatMissingData

Specifies how missing data points are treated when evaluating the alarm's condition.

Type: CloudWatchAlarmTemplateTreatMissingData
Required: True

CloudWatchAlarmTemplateTargetResourceType

The resource type this template should dynamically generate cloudwatch metric alarms for.

```
CLOUDFRONT_DISTRIBUTION
MEDIALIVE_MULTIPLEX
MEDIALIVE_CHANNEL
MEDIALIVE_INPUT_DEVICE
MEDIAPACKAGE_CHANNEL
MEDIAPACKAGE_ORIGIN_ENDPOINT
MEDIACONNECT_FLOW
S3_BUCKET
```

CloudWatchAlarmTemplateTreatMissingData

Specifies how missing data points are treated when evaluating the alarm's condition.

```
notBreaching
breaching
ignore
missing
```

ConflictExceptionResponseContent

Updating or deleting a resource can cause an inconsistent state.

message

Exception error message.

Type: string
Required: False

CreateCloudWatchAlarmTemplateRequestContent

comparisonOperator

The comparison operator used to compare the specified statistic and the threshold.

Type: CloudWatchAlarmTemplateComparisonOperator
Required: True

datapointsToAlarm

The number of datapoints within the evaluation period that must be breaching to trigger the alarm.

Type: number
Required: False
Minimum: 1

description

A resource's optional description.

Type: string
Required: False
MinLength: 0
MaxLength: 1024

evaluationPeriods

The number of periods over which data is compared to the specified threshold.

Type: number
Required: True
Minimum: 1

groupIdentifier

A cloudwatch alarm template group's identifier. Can be either be its id or current name.

Type: string
Required: True
Pattern: ^[^\s]+$

metricName

The name of the metric associated with the alarm. Must be compatible with targetResourceType.

Type: string
Required: True
MaxLength: 64

name

A resource's name. Names must be unique within the scope of a resource type in a specific region.

Type: string
Required: True
Pattern: ^[^\s]+$
MinLength: 1
MaxLength: 255

period

The period, in seconds, over which the specified statistic is applied.

Type: number
Required: True
Minimum: 10
Maximum: 86400

standardDeviationThreshold

The number of standard deviations by which the metric value can deviate from the expected value before the alarm is triggered.

Type: number
Required: False

Minimum: 1

statistic

The statistic to apply to the alarm's metric data.

Type: CloudWatchAlarmTemplateStatistic
Required: True

targetResourceType

The resource type this template should dynamically generate CloudWatch metric alarms for.

Type: CloudWatchAlarmTemplateTargetResourceType
Required: True

threshold

The threshold value to compare with the specified statistic.

Type: number
Required: False
Format: double

treatMissingData

Specifies how missing data points are treated when evaluating the alarm's condition.

Type: CloudWatchAlarmTemplateTreatMissingData
Required: True

CreateCloudWatchAlarmTemplateResponseContent

arn

A cloudwatch alarm template's ARN (Amazon Resource Name)

Type: string
Required: True

Pattern: `^arn:.+:medialive:.+:cloudwatch-alarm-template:.+$`

comparisonOperator

The comparison operator used to compare the specified statistic and the threshold.

Type: CloudWatchAlarmTemplateComparisonOperator
Required: True

createdAt

The date and time of resource creation.

Type: string
Required: True
Format: date-time

datapointsToAlarm

The number of datapoints within the evaluation period that must be breaching to trigger the alarm.

Type: number
Required: False
Minimum: 1

description

A resource's optional description.

Type: string
Required: False
MinLength: 0
MaxLength: 1024

evaluationPeriods

The number of periods over which data is compared to the specified threshold.

Type: number
Required: True
Minimum: 1

groupId

A CloudWatch alarm template group's id. Amazon Web Services provided template groups have ids that start with <code>`aws-`</code>

Type: string
Required: True
Pattern: `^(aws-)?[0-9]{7}$`
MinLength: 7
MaxLength: 11

id

A cloudwatch alarm template's id. Amazon Web Services provided templates have ids that start with <code>`aws-`</code>.

Type: string
Required: True
Pattern: `^(aws-)?[0-9]{7}$`
MinLength: 7
MaxLength: 11

metricName

The name of the metric associated with the alarm. Must be compatible with targetResourceType.

Type: string
Required: True
MaxLength: 64

modifiedAt

The date and time of latest resource modification.

Type: string
Required: False
Format: date-time

name

A resource's name. Names must be unique within the scope of a resource type in a specific region.

Type: string
Required: True
Pattern: ^[^\s]+$
MinLength: 1
MaxLength: 255

period

The period, in seconds, over which the specified statistic is applied.

Type: number
Required: True
Minimum: 10
Maximum: 86400

standardDeviationThreshold

The number of standard deviations by which the metric value can deviate from the expected value before the alarm is triggered.

Type: number
Required: False
Minimum: 1

statistic

The statistic to apply to the alarm's metric data.

Type: CloudWatchAlarmTemplateStatistic

Required: True

targetResourceType

The resource type this template should dynamically generate CloudWatch metric alarms for.

Type: CloudWatchAlarmTemplateTargetResourceType
Required: True

threshold

The threshold value to compare with the specified statistic.

Type: number
Required: False
Format: double

treatMissingData

Specifies how missing data points are treated when evaluating the alarm's condition.

Type: CloudWatchAlarmTemplateTreatMissingData
Required: True

ForbiddenExceptionResponseContent

User does not have sufficient access to perform this action.

message

Exception error message.

Type: string
Required: False

InternalServerErrorExceptionResponseContent

Unexpected error during processing of request.

message

Exception error message.

Type: string
Required: False

ListCloudWatchAlarmTemplatesResponseContent

cloudWatchAlarmTemplates

A summary of the alarm templates.

Type: Array of type CloudWatchAlarmTemplateSummary
Required: True

nextToken

A token used to retrieve the next set of results in paginated list responses.

Type: string
Required: False
MinLength: 1
MaxLength: 2048

NotFoundExceptionResponseContent

Request references a resource which does not exist.

message

Exception error message.

Type: string
Required: False

TooManyRequestsExceptionResponseContent

Request was denied due to request throttling.

message

Exception error message.

Type: string
Required: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

ListCloudWatchAlarmTemplates

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

CorsCloudwatch_alarm_templates

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python

- AWS SDK for Ruby V3

CreateCloudWatchAlarmTemplate

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Workflow monitor: CloudWatch alarm templates ID

URI

`/prod/cloudwatch-alarm-templates/identifier`

HTTP methods

DELETE

Operation ID: `DeleteCloudWatchAlarmTemplate`

Path parameters

Name	Type	Required	Description
identifier	String	True	

Responses

Status code	Response model	Description
204	None	204 response
400	BadRequestExceptionResponseContent	400 response
403	ForbiddenExceptionResponseContent	403 response
404	NotFoundExceptionResponseContent	404 response
409	ConflictExceptionResponseContent	409 response
429	TooManyRequestsExceptionResponseContent	429 response
500	InternalServerErrorExceptionResponseContent	500 response

GET

Operation ID: `GetCloudWatchAlarmTemplate`

Path parameters

Name	Type	Required	Description
identifier	String	True	

Responses

Status code	Response model	Description
200	GetCloudWatchAlarmTemplateResponseContent	200 response
400	BadRequestExceptionResponseContent	400 response
403	ForbiddenExceptionResponseContent	403 response
404	NotFoundExceptionResponseContent	404 response
429	TooManyRequestsExceptionResponseContent	429 response
500	InternalServerErrorExceptionResponseContent	500 response

OPTIONS

Operation ID: `CorsCloudwatch_alarm_templatesIdentifier`

Path parameters

Name	Type	Required	Description
identifier	String	True	

Responses

Status code	Response model	Description
200	None	200 response

PATCH

Operation ID: `UpdateCloudWatchAlarmTemplate`

Path parameters

Name	Type	Required	Description
identifier	String	True	

Responses

Status code	Response model	Description
200	UpdateCloudWatchAlarmTemplateResponseContent	200 response
400	BadRequestExceptionResponseContent	400 response
403	ForbiddenExceptionResponseContent	403 response
404	NotFoundExceptionResponseContent	404 response
409	ConflictExceptionResponseContent	409 response
429	TooManyRequestsExceptionResponseContent	429 response

Status code	Response model	Description
500	InternalServerErrorExceptionResponseContent	500 response

Schemas

Request bodies

PATCH schema

```
{
  "comparisonOperator": enum,
  "datapointsToAlarm": number,
  "description": "string",
  "evaluationPeriods": number,
  "groupIdentifier": "string",
  "metricName": "string",
  "name": "string",
  "period": number,
  "standardDeviationThreshold": number,
  "statistic": enum,
  "targetResourceType": enum,
  "threshold": number,
  "treatMissingData": enum
}
```

Response bodies

GetCloudWatchAlarmTemplateResponseContent schema

```
{
  "arn": "string",
  "comparisonOperator": enum,
  "createdAt": "string",
  "datapointsToAlarm": number,
  "description": "string",
  "evaluationPeriods": number,
  "groupId": "string",
```

```
  "id": "string",
  "metricName": "string",
  "modifiedAt": "string",
  "name": "string",
  "period": number,
  "standardDeviationThreshold": number,
  "statistic": enum,
  "targetResourceType": enum,
  "threshold": number,
  "treatMissingData": enum
}
```

UpdateCloudWatchAlarmTemplateResponseContent schema

```
{
  "arn": "string",
  "comparisonOperator": enum,
  "createdAt": "string",
  "datapointsToAlarm": number,
  "description": "string",
  "evaluationPeriods": number,
  "groupId": "string",
  "id": "string",
  "metricName": "string",
  "modifiedAt": "string",
  "name": "string",
  "period": number,
  "standardDeviationThreshold": number,
  "statistic": enum,
  "targetResourceType": enum,
  "threshold": number,
  "treatMissingData": enum
}
```

BadRequestExceptionResponseContent schema

```
{
  "message": "string"
}
```

ForbiddenExceptionResponseContent schema

```
{
  "message": "string"
}
```

NotFoundExceptionResponseContent schema

```
{
  "message": "string"
}
```

ConflictExceptionResponseContent schema

```
{
  "message": "string"
}
```

TooManyRequestsExceptionResponseContent schema

```
{
  "message": "string"
}
```

InternalServerErrorExceptionResponseContent schema

```
{
  "message": "string"
}
```

Properties

BadRequestExceptionResponseContent

The input fails to satisfy the constraints specified by an Amazon Web Services service.

message

Exception error message.

Type: string
Required: False

CloudWatchAlarmTemplateComparisonOperator

The comparison operator used to compare the specified statistic and the threshold.

```
GreaterThanOrEqualToThreshold
GreaterThanThreshold
LessThanThreshold
LessThanOrEqualToThreshold
LessThanLowerOrGreaterThanUpperThreshold
LessThanLowerThreshold
GreaterThanUpperThreshold
```

CloudWatchAlarmTemplateStatistic

The statistic to apply to the alarm's metric data.

```
SampleCount
Average
Sum
Minimum
Maximum
```

CloudWatchAlarmTemplateTargetResourceType

The resource type this template should dynamically generate cloudwatch metric alarms for.

```
CLOUDFRONT_DISTRIBUTION
MEDIALIVE_MULTIPLEX
MEDIALIVE_CHANNEL
MEDIALIVE_INPUT_DEVICE
MEDIAPACKAGE_CHANNEL
MEDIAPACKAGE_ORIGIN_ENDPOINT
MEDIACONNECT_FLOW
S3_BUCKET
```

CloudWatchAlarmTemplateTreatMissingData

Specifies how missing data points are treated when evaluating the alarm's condition.

```
notBreaching
breaching
ignore
missing
```

ConflictExceptionResponseContent

Updating or deleting a resource can cause an inconsistent state.

message

Exception error message.

Type: string
Required: False

ForbiddenExceptionResponseContent

User does not have sufficient access to perform this action.

message

Exception error message.

Type: string
Required: False

GetCloudWatchAlarmTemplateResponseContent

arn

A cloudwatch alarm template's ARN (Amazon Resource Name)

Type: string
Required: True

Pattern: `^arn:.+:medialive:.+:cloudwatch-alarm-template:.+$`

comparisonOperator

The comparison operator used to compare the specified statistic and the threshold.

Type: CloudWatchAlarmTemplateComparisonOperator
Required: True

createdAt

The date and time of resource creation.

Type: string
Required: True
Format: date-time

datapointsToAlarm

The number of datapoints within the evaluation period that must be breaching to trigger the alarm.

Type: number
Required: False
Minimum: 1

description

A resource's optional description.

Type: string
Required: False
MinLength: 0
MaxLength: 1024

evaluationPeriods

The number of periods over which data is compared to the specified threshold.

Type: number
Required: True
Minimum: 1

groupId

A CloudWatch alarm template group's id. Amazon Web Services provided template groups have ids that start with <code>`aws-`</code>

Type: string
Required: True
Pattern: `^(aws-)?[0-9]{7}$`
MinLength: 7
MaxLength: 11

id

A cloudwatch alarm template's id. Amazon Web Services provided templates have ids that start with <code>`aws-`</code>.

Type: string
Required: True
Pattern: `^(aws-)?[0-9]{7}$`
MinLength: 7
MaxLength: 11

metricName

The name of the metric associated with the alarm. Must be compatible with targetResourceType.

Type: string
Required: True
MaxLength: 64

modifiedAt

The date and time of latest resource modification.

Type: string
Required: False
Format: date-time

name

A resource's name. Names must be unique within the scope of a resource type in a specific region.

Type: string
Required: True
Pattern: ^[^\s]+$
MinLength: 1
MaxLength: 255

period

The period, in seconds, over which the specified statistic is applied.

Type: number
Required: True
Minimum: 10
Maximum: 86400

standardDeviationThreshold

The number of standard deviations by which the metric value can deviate from the expected value before the alarm is triggered.

Type: number
Required: False
Minimum: 1

statistic

The statistic to apply to the alarm's metric data.

Type: CloudWatchAlarmTemplateStatistic

Required: True

targetResourceType

The resource type this template should dynamically generate CloudWatch metric alarms for.

Type: CloudWatchAlarmTemplateTargetResourceType
Required: True

threshold

The threshold value to compare with the specified statistic.

Type: number
Required: False
Format: double

treatMissingData

Specifies how missing data points are treated when evaluating the alarm's condition.

Type: CloudWatchAlarmTemplateTreatMissingData
Required: True

InternalServerErrorExceptionResponseContent

Unexpected error during processing of request.

message

Exception error message.

Type: string
Required: False

NotFoundExceptionResponseContent

Request references a resource which does not exist.

message

Exception error message.

Type: string
Required: False

TooManyRequestsExceptionResponseContent

Request was denied due to request throttling.

message

Exception error message.

Type: string
Required: False

UpdateCloudWatchAlarmTemplateRequestContent

comparisonOperator

The comparison operator used to compare the specified statistic and the threshold.

Type: CloudWatchAlarmTemplateComparisonOperator
Required: False

datapointsToAlarm

The number of datapoints within the evaluation period that must be breaching to trigger the alarm.

Type: number
Required: False
Minimum: 1

description

A resource's optional description.

Type: string
Required: False
MinLength: 0
MaxLength: 1024

evaluationPeriods

The number of periods over which data is compared to the specified threshold.

Type: number
Required: False
Minimum: 1

groupIdentifier

A cloudwatch alarm template group's identifier. Can be either be its id or current name.

Type: string
Required: False
Pattern: ^[^\s]+$

metricName

The name of the metric associated with the alarm. Must be compatible with targetResourceType.

Type: string
Required: False
MaxLength: 64

name

A resource's name. Names must be unique within the scope of a resource type in a specific region.

Type: string
Required: False
Pattern: ^[^\s]+$
MinLength: 1
MaxLength: 255

period

The period, in seconds, over which the specified statistic is applied.

Type: number
Required: False
Minimum: 10
Maximum: 86400

standardDeviationThreshold

The number of standard deviations by which the metric value can deviate from the expected value before the alarm is triggered.

Type: number
Required: False
Minimum: 1

statistic

The statistic to apply to the alarm's metric data.

Type: CloudWatchAlarmTemplateStatistic
Required: False

targetResourceType

The resource type this template should dynamically generate CloudWatch metric alarms for.

Type: CloudWatchAlarmTemplateTargetResourceType
Required: False

threshold

The threshold value to compare with the specified statistic.

Type: number
Required: False

Format: double

treatMissingData

Specifies how missing data points are treated when evaluating the alarm's condition.

Type: CloudWatchAlarmTemplateTreatMissingData
Required: False

UpdateCloudWatchAlarmTemplateResponseContent

arn

A cloudwatch alarm template's ARN (Amazon Resource Name)

Type: string
Required: True
Pattern: `^arn:.+:medialive:.+:cloudwatch-alarm-template:.+$`

comparisonOperator

The comparison operator used to compare the specified statistic and the threshold.

Type: CloudWatchAlarmTemplateComparisonOperator
Required: True

createdAt

The date and time of resource creation.

Type: string
Required: True
Format: date-time

datapointsToAlarm

The number of datapoints within the evaluation period that must be breaching to trigger the alarm.

Type: number
Required: False
Minimum: 1

description

A resource's optional description.

Type: string
Required: False
MinLength: 0
MaxLength: 1024

evaluationPeriods

The number of periods over which data is compared to the specified threshold.

Type: number
Required: True
Minimum: 1

groupId

A CloudWatch alarm template group's id. Amazon Web Services provided template groups have ids that start with <code>`aws-`</code>

Type: string
Required: True
Pattern: `^(aws-)?[0-9]{7}$`
MinLength: 7
MaxLength: 11

id

A cloudwatch alarm template's id. Amazon Web Services provided templates have ids that start with <code>`aws-`</code>.

Type: string

Required: True
Pattern: ^(aws-)?[0-9]{7}$
MinLength: 7
MaxLength: 11

metricName

The name of the metric associated with the alarm. Must be compatible with targetResourceType.

Type: string
Required: True
MaxLength: 64

modifiedAt

The date and time of latest resource modification.

Type: string
Required: False
Format: date-time

name

A resource's name. Names must be unique within the scope of a resource type in a specific region.

Type: string
Required: True
Pattern: ^[^\s]+$
MinLength: 1
MaxLength: 255

period

The period, in seconds, over which the specified statistic is applied.

Type: number
Required: True
Minimum: 10

Maximum: 86400

standardDeviationThreshold

The number of standard deviations by which the metric value can deviate from the expected value before the alarm is triggered.

Type: number
Required: False
Minimum: 1

statistic

The statistic to apply to the alarm's metric data.

Type: CloudWatchAlarmTemplateStatistic
Required: True

targetResourceType

The resource type this template should dynamically generate CloudWatch metric alarms for.

Type: CloudWatchAlarmTemplateTargetResourceType
Required: True

threshold

The threshold value to compare with the specified statistic.

Type: number
Required: False
Format: double

treatMissingData

Specifies how missing data points are treated when evaluating the alarm's condition.

Type: CloudWatchAlarmTemplateTreatMissingData
Required: True

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

DeleteCloudWatchAlarmTemplate

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

GetCloudWatchAlarmTemplate

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

CorsCloudwatch_alarm_templatesIdentifier

- AWS Command Line Interface

- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UpdateCloudWatchAlarmTemplate

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Workflow monitor: EventBridge rule template groups

URI

`/prod/eventbridge-rule-template-groups`

HTTP methods

GET

Operation ID: `ListEventBridgeRuleTemplateGroups`

Query parameters

Name	Type	Required	Description
signalMapIdentifier	String	False	
nextToken	String	False	
maxResults	String	False	

Responses

Status code	Response model	Description
200	ListEventBridgeRuleTemplateGroupsResponseContent	200 response
400	BadRequestExceptionResponseContent	400 response
403	ForbiddenExceptionResponseContent	403 response
404	NotFoundExceptionResponseContent	404 response
429	TooManyRequestsExceptionResponseContent	429 response
500	InternalServerErrorExceptionResponseContent	500 response

OPTIONS

Operation ID: CorsEventbridge_rule_template_groups

Responses

Status code	Response model	Description
200	None	200 response

POST

Operation ID: `CreateEventBridgeRuleTemplateGroup`

Responses

Status code	Response model	Description
201	CreateEventBridgeRuleTemplateGroupResponseContent	201 response
400	BadRequestExceptionResponseContent	400 response
403	ForbiddenExceptionResponseContent	403 response
404	NotFoundExceptionResponseContent	404 response
409	ConflictExceptionResponseContent	409 response
429	TooManyRequestsExceptionResponseContent	429 response
500	InternalServerErrorExceptionResponseContent	500 response

Schemas

Request bodies

POST schema

```
{
  "description": "string",
  "name": "string"
}
```

Response bodies

ListEventBridgeRuleTemplateGroupsResponseContent schema

```
{
  "eventBridgeRuleTemplateGroups": [
    {
      "arn": "string",
      "createdAt": "string",
      "description": "string",
      "id": "string",
      "modifiedAt": "string",
      "name": "string",
      "templateCount": number
    }
  ],
  "nextToken": "string"
}
```

CreateEventBridgeRuleTemplateGroupResponseContent schema

```
{
  "arn": "string",
  "createdAt": "string",
  "description": "string",
  "id": "string",
  "modifiedAt": "string",
  "name": "string"
}
```

BadRequestExceptionResponseContent schema

```
{
  "message": "string"
}
```

ForbiddenExceptionResponseContent schema

```
{
  "message": "string"
}
```

NotFoundExceptionResponseContent schema

```
{
  "message": "string"
}
```

ConflictExceptionResponseContent schema

```
{
  "message": "string"
}
```

TooManyRequestsExceptionResponseContent schema

```
{
  "message": "string"
}
```

InternalServerErrorExceptionResponseContent schema

```
{
  "message": "string"
}
```

Properties

BadRequestExceptionResponseContent

The input fails to satisfy the constraints specified by an Amazon Web Services service.

message

Exception error message.

Type: string
Required: False

ConflictExceptionResponseContent

Updating or deleting a resource can cause an inconsistent state.

message

Exception error message.

Type: string
Required: False

CreateEventBridgeRuleTemplateGroupRequestContent

description

A resource's optional description.

Type: string
Required: False
MinLength: 0
MaxLength: 1024

name

A resource's name. Names must be unique within the scope of a resource type in a specific region.

Type: string

Required: True
Pattern: ^[^\s]+$
MinLength: 1
MaxLength: 255

CreateEventBridgeRuleTemplateGroupResponseContent

arn

An eventbridge rule template group's ARN (Amazon Resource Name)

Type: string
Required: True
Pattern: ^arn:.+:medialive:.+:eventbridge-rule-template-group:.+$

createdAt

The date and time of resource creation.

Type: string
Required: True
Format: date-time

description

A resource's optional description.

Type: string
Required: False
MinLength: 0
MaxLength: 1024

id

An eventbridge rule template group's id. Amazon Web Services provided template groups have ids that start with <code>`aws-`</code>.

Type: string

Required: True
Pattern: ^(aws-)?[0-9]{7}$
MinLength: 7
MaxLength: 11

modifiedAt

The date and time of latest resource modification.

Type: string
Required: False
Format: date-time

name

A resource's name. Names must be unique within the scope of a resource type in a specific region.

Type: string
Required: True
Pattern: ^[^\s]+$
MinLength: 1
MaxLength: 255

EventBridgeRuleTemplateGroupSummary

arn

An eventbridge rule template group's ARN (Amazon Resource Name)

Type: string
Required: True
Pattern: ^arn:.+:medialive:.+:eventbridge-rule-template-group:.+$

createdAt

The date and time of resource creation.

Type: string

Required: True
Format: date-time

description

A resource's optional description.

Type: string
Required: False
MinLength: 0
MaxLength: 1024

id

An eventbridge rule template group's id. Amazon Web Services provided template groups have ids that start with <code>`aws-`</code>.

Type: string
Required: True
Pattern: ^(aws-)?[0-9]{7}$
MinLength: 7
MaxLength: 11

modifiedAt

The date and time of latest resource modification.

Type: string
Required: False
Format: date-time

name

A resource's name. Names must be unique within the scope of a resource type in a specific region.

Type: string
Required: True
Pattern: ^[^\s]+$

MinLength: 1
MaxLength: 255

templateCount

The number of templates in the group.

Type: number
Required: True

ForbiddenExceptionResponseContent

User does not have sufficient access to perform this action.

message

Exception error message.

Type: string
Required: False

InternalServerErrorExceptionResponseContent

Unexpected error during processing of request.

message

Exception error message.

Type: string
Required: False

ListEventBridgeRuleTemplateGroupsResponseContent

eventBridgeRuleTemplateGroups

A list of EventBridge rule template groups.

Type: Array of type EventBridgeRuleTemplateGroupSummary
Required: True

nextToken

A token used to retrieve the next set of results in paginated list responses.

Type: string
Required: False
MinLength: 1
MaxLength: 2048

NotFoundExceptionResponseContent

Request references a resource which does not exist.

message

Exception error message.

Type: string
Required: False

TooManyRequestsExceptionResponseContent

Request was denied due to request throttling.

message

Exception error message.

Type: string
Required: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

ListEventBridgeRuleTemplateGroups

- AWS Command Line Interface

- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

CorsEventbridge_rule_template_groups

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

CreateEventBridgeRuleTemplateGroup

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Workflow monitor: EventBridge rule template groups ID

URI

`/prod/eventbridge-rule-template-groups/identifier`

HTTP methods

DELETE

Operation ID: `DeleteEventBridgeRuleTemplateGroup`

Path parameters

Name	Type	Required	Description
identifier	String	True	

Responses

Status code	Response model	Description
204	None	204 response
400	BadRequestExceptionResponseContent	400 response
403	ForbiddenExceptionResponseContent	403 response
404	NotFoundExceptionResponseContent	404 response
409	ConflictExceptionResponseContent	409 response
429	TooManyRequestsExceptionResponseContent	429 response

Status code	Response model	Description
500	InternalServerErrorExceptionResponseContent	500 response

GET

Operation ID: GetEventBridgeRuleTemplateGroup

Path parameters

Name	Type	Required	Description
identifier	String	True	

Responses

Status code	Response model	Description
200	GetEventBridgeRuleTemplateGroupResponseContent	200 response
400	BadRequestExceptionResponseContent	400 response
403	ForbiddenExceptionResponseContent	403 response
404	NotFoundExceptionResponseContent	404 response
429	TooManyRequestsExceptionResponseContent	429 response

Status code	Response model	Description
500	InternalServerErrorExceptionResponseContent	500 response

OPTIONS

Operation ID: CorsEventbridge_rule_template_groupsIdentifier

Path parameters

Name	Type	Required	Description
identifier	String	True	

Responses

Status code	Response model	Description
200	None	200 response

PATCH

Operation ID: UpdateEventBridgeRuleTemplateGroup

Path parameters

Name	Type	Required	Description
identifier	String	True	

Responses

Status code	Response model	Description
200	UpdateEventBridgeRuleTemplateGroupResponseContent	200 response
400	BadRequestExceptionResponseContent	400 response
403	ForbiddenExceptionResponseContent	403 response
404	NotFoundExceptionResponseContent	404 response
409	ConflictExceptionResponseContent	409 response
429	TooManyRequestsExceptionResponseContent	429 response
500	InternalServerErrorExceptionResponseContent	500 response

Schemas

Request bodies

PATCH schema

```
{
  "description": "string"
}
```

Response bodies

GetEventBridgeRuleTemplateGroupResponseContent schema

```
{
  "arn": "string",
  "createdAt": "string",
  "description": "string",
  "id": "string",
  "modifiedAt": "string",
  "name": "string"
}
```

UpdateEventBridgeRuleTemplateGroupResponseContent schema

```
{
  "arn": "string",
  "createdAt": "string",
  "description": "string",
  "id": "string",
  "modifiedAt": "string",
  "name": "string"
}
```

BadRequestExceptionResponseContent schema

```
{
  "message": "string"
}
```

ForbiddenExceptionResponseContent schema

```
{
  "message": "string"
}
```

NotFoundExceptionResponseContent schema

```
{
  "message": "string"
}
```

ConflictExceptionResponseContent schema

```
{
  "message": "string"
}
```

TooManyRequestsExceptionResponseContent schema

```
{
  "message": "string"
}
```

InternalServerErrorExceptionResponseContent schema

```
{
  "message": "string"
}
```

Properties

BadRequestExceptionResponseContent

The input fails to satisfy the constraints specified by an Amazon Web Services service.

message

Exception error message.

Type: string
Required: False

ConflictExceptionResponseContent

Updating or deleting a resource can cause an inconsistent state.

message

Exception error message.

Type: string

Required: False

ForbiddenExceptionResponseContent

User does not have sufficient access to perform this action.

message

Exception error message.

Type: string
Required: False

GetEventBridgeRuleTemplateGroupResponseContent

arn

An eventbridge rule template group's ARN (Amazon Resource Name)

Type: string
Required: True
Pattern: `^arn:.+:medialive:.+:eventbridge-rule-template-group:.+$`

createdAt

The date and time of resource creation.

Type: string
Required: True
Format: date-time

description

A resource's optional description.

Type: string
Required: False
MinLength: 0
MaxLength: 1024

id

An eventbridge rule template group's id. Amazon Web Services provided template groups have ids that start with <code>`aws-`</code>.

Type: string
Required: True
Pattern: `^(aws-)?[0-9]{7}$`
MinLength: 7
MaxLength: 11

modifiedAt

The date and time of latest resource modification.

Type: string
Required: False
Format: date-time

name

A resource's name. Names must be unique within the scope of a resource type in a specific region.

Type: string
Required: True
Pattern: `^[^\s]+$`
MinLength: 1
MaxLength: 255

InternalServerErrorExceptionResponseContent

Unexpected error during processing of request.

message

Exception error message.

Type: string

Required: False

NotFoundExceptionResponseContent

Request references a resource which does not exist.

message

Exception error message.

Type: string
Required: False

TooManyRequestsExceptionResponseContent

Request was denied due to request throttling.

message

Exception error message.

Type: string
Required: False

UpdateEventBridgeRuleTemplateGroupRequestContent

description

A resource's optional description.

Type: string
Required: False
MinLength: 0
MaxLength: 1024

UpdateEventBridgeRuleTemplateGroupResponseContent

arn

An eventbridge rule template group's ARN (Amazon Resource Name)

Type: string
Required: True
Pattern: `^arn:.+:medialive:.+:eventbridge-rule-template-group:.+$`

createdAt

The date and time of resource creation.

Type: string
Required: True
Format: date-time

description

A resource's optional description.

Type: string
Required: False
MinLength: 0
MaxLength: 1024

id

An eventbridge rule template group's id. Amazon Web Services provided template groups have ids that start with <code>`aws-`</code>.

Type: string
Required: True
Pattern: `^(aws-)?[0-9]{7}$`
MinLength: 7
MaxLength: 11

modifiedAt

The date and time of latest resource modification.

Type: string

Required: False
Format: date-time

name

A resource's name. Names must be unique within the scope of a resource type in a specific region.

Type: string
Required: True
Pattern: ^[^\s]+$
MinLength: 1
MaxLength: 255

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

DeleteEventBridgeRuleTemplateGroup

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

GetEventBridgeRuleTemplateGroup

- AWS Command Line Interface
- AWS SDK for .NET

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

CorsEventbridge_rule_template_groupsIdentifier

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UpdateEventBridgeRuleTemplateGroup

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Workflow monitor: EventBridge rule templates

URI

`/prod/eventbridge-rule-templates`

HTTP methods

GET

Operation ID: `ListEventBridgeRuleTemplates`

Query parameters

Name	Type	Required	Description
`groupIdentifier`	String	False	
`signalMapIdentifier`	String	False	
`nextToken`	String	False	
`maxResults`	String	False	

Responses

Status code	Response model	Description
200	ListEventBridgeRuleTemplatesResponseContent	200 response
400	BadRequestExceptionResponseContent	400 response
403	ForbiddenExceptionResponseContent	403 response

Status code	Response model	Description
404	NotFoundExceptionResponseContent	404 response
429	TooManyRequestsExceptionResponseContent	429 response
500	InternalServerErrorExceptionResponseContent	500 response

OPTIONS

Operation ID: CorsEventbridge_rule_templates

Responses

Status code	Response model	Description
200	None	200 response

POST

Operation ID: CreateEventBridgeRuleTemplate

Responses

Status code	Response model	Description
201	CreateEventBridgeRuleTemplateResponseContent	201 response
400	BadRequestExceptionResponseContent	400 response

Status code	Response model	Description
403	ForbiddenExceptionResponseContent	403 response
404	NotFoundExceptionResponseContent	404 response
409	ConflictExceptionResponseContent	409 response
429	TooManyRequestsExceptionResponseContent	429 response
500	InternalServerErrorExceptionResponseContent	500 response

Schemas

Request bodies

POST schema

```
{
  "description": "string",
  "eventTargets": [
    {
      "arn": "string"
    }
  ],
  "eventType": enum,
  "groupIdentifier": "string",
  "name": "string"
}
```

Response bodies

ListEventBridgeRuleTemplatesResponseContent schema

```
{
  "eventBridgeRuleTemplates": [
    {
      "arn": "string",
      "createdAt": "string",
      "description": "string",
      "eventTargetCount": number,
      "eventType": enum,
      "groupId": "string",
      "id": "string",
      "modifiedAt": "string",
      "name": "string"
    }
  ],
  "nextToken": "string"
}
```

CreateEventBridgeRuleTemplateResponseContent schema

```
{
  "arn": "string",
  "createdAt": "string",
  "description": "string",
  "eventTargets": [
    {
      "arn": "string"
    }
  ],
  "eventType": enum,
  "groupId": "string",
  "id": "string",
  "modifiedAt": "string",
  "name": "string"
}
```

BadRequestExceptionResponseContent schema

```
{
  "message": "string"
```

```
}
```

ForbiddenExceptionResponseContent schema

```
{
  "message": "string"
}
```

NotFoundExceptionResponseContent schema

```
{
  "message": "string"
}
```

ConflictExceptionResponseContent schema

```
{
  "message": "string"
}
```

TooManyRequestsExceptionResponseContent schema

```
{
  "message": "string"
}
```

InternalServerErrorExceptionResponseContent schema

```
{
  "message": "string"
}
```

Properties

BadRequestExceptionResponseContent

The input fails to satisfy the constraints specified by an Amazon Web Services service.

message

Exception error message.

Type: string
Required: False

ConflictExceptionResponseContent

Updating or deleting a resource can cause an inconsistent state.

message

Exception error message.

Type: string
Required: False

CreateEventBridgeRuleTemplateRequestContent

description

A resource's optional description.

Type: string
Required: False
MinLength: 0
MaxLength: 1024

eventTargets

The destinations that will receive the event notifications.

Type: Array of type EventBridgeRuleTemplateTarget
Required: True

eventType

The type of event to match with the rule.

Type: EventBridgeRuleTemplateEventType
Required: True

groupIdentifier

An eventbridge rule template group's identifier. Can be either be its id or current name.

Type: string
Required: True
Pattern: ^[^\s]+$

name

A resource's name. Names must be unique within the scope of a resource type in a specific region.

Type: string
Required: True
Pattern: ^[^\s]+$
MinLength: 1
MaxLength: 255

CreateEventBridgeRuleTemplateResponseContent

arn

An eventbridge rule template's ARN (Amazon Resource Name)

Type: string
Required: True
Pattern: ^arn:.+:medialive:.+:eventbridge-rule-template:.+$

createdAt

The date and time of resource creation.

Type: string
Required: True
Format: date-time

description

A resource's optional description.

Type: string
Required: False
MinLength: 0
MaxLength: 1024

eventTargets

The destinations that will receive the event notifications.

Type: Array of type EventBridgeRuleTemplateTarget
Required: True

eventType

The type of event to match with the rule.

Type: EventBridgeRuleTemplateEventType
Required: True

groupId

An eventbridge rule template group's id. Amazon Web Services provided template groups have ids that start with <code>`aws-`</code>.

Type: string
Required: True
Pattern: `^(aws-)?[0-9]{7}$`
MinLength: 7
MaxLength: 11

id

An eventbridge rule template's id. Amazon Web Services provided templates have ids that start with <code>`aws-`</code>

Type: string
Required: True
Pattern: ^(aws-)?[0-9]{7}$
MinLength: 7
MaxLength: 11

modifiedAt

The date and time of latest resource modification.

Type: string
Required: False
Format: date-time

name

A resource's name. Names must be unique within the scope of a resource type in a specific region.

Type: string
Required: True
Pattern: ^[^\s]+$
MinLength: 1
MaxLength: 255

EventBridgeRuleTemplateEventType

The type of event to match with the rule.

MEDIALIVE_MULTIPLEX_ALERT
MEDIALIVE_MULTIPLEX_STATE_CHANGE
MEDIALIVE_CHANNEL_ALERT
MEDIALIVE_CHANNEL_INPUT_CHANGE
MEDIALIVE_CHANNEL_STATE_CHANGE
MEDIAPACKAGE_INPUT_NOTIFICATION
MEDIAPACKAGE_KEY_PROVIDER_NOTIFICATION
MEDIAPACKAGE_HARVEST_JOB_NOTIFICATION

SIGNAL_MAP_ACTIVE_ALARM

MEDIACONNECT_ALERT

MEDIACONNECT_SOURCE_HEALTH

MEDIACONNECT_OUTPUT_HEALTH

MEDIACONNECT_FLOW_STATUS_CHANGE

EventBridgeRuleTemplateSummary

arn

An eventbridge rule template's ARN (Amazon Resource Name)

Type: string

Required: True

Pattern: ^arn:.+:medialive:.+:eventbridge-rule-template:.+$

createdAt

The date and time of resource creation.

Type: string

Required: True

Format: date-time

description

A resource's optional description.

Type: string

Required: False

MinLength: 0

MaxLength: 1024

eventTargetCount

The number of event targets.

Type: number
Required: True

eventType

The type of event to match with the rule.

Type: EventBridgeRuleTemplateEventType
Required: True

groupId

An eventbridge rule template group's id. Amazon Web Services provided template groups have ids that start with <code>`aws-`</code>.

Type: string
Required: True
Pattern: `^(aws-)?[0-9]{7}$`
MinLength: 7
MaxLength: 11

id

An eventbridge rule template's id. Amazon Web Services provided templates have ids that start with <code>`aws-`</code>

Type: string
Required: True
Pattern: `^(aws-)?[0-9]{7}$`
MinLength: 7
MaxLength: 11

modifiedAt

The date and time of latest resource modification.

Type: string

Required: False
Format: date-time

name

A resource's name. Names must be unique within the scope of a resource type in a specific region.

Type: string
Required: True
Pattern: `^[^\s]+$`
MinLength: 1
MaxLength: 255

EventBridgeRuleTemplateTarget

The target to which to send matching events.

arn

Target ARNs must be either an SNS topic or CloudWatch log group.

Type: string
Required: True
Pattern: `^arn.+$`
MinLength: 1
MaxLength: 2048

ForbiddenExceptionResponseContent

User does not have sufficient access to perform this action.

message

Exception error message.

Type: string
Required: False

InternalServerErrorExceptionResponseContent

Unexpected error during processing of request.

message

Exception error message.

Type: string
Required: False

ListEventBridgeRuleTemplatesResponseContent

eventBridgeRuleTemplates

A list of the EventBridge rule templates.

Type: Array of type EventBridgeRuleTemplateSummary
Required: True

nextToken

A token used to retrieve the next set of results in paginated list responses.

Type: string
Required: False
MinLength: 1
MaxLength: 2048

NotFoundExceptionResponseContent

Request references a resource which does not exist.

message

Exception error message.

Type: string
Required: False

TooManyRequestsExceptionResponseContent

Request was denied due to request throttling.

message

Exception error message.

Type: string
Required: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

ListEventBridgeRuleTemplates

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

CorsEventbridge_rule_templates

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2

- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

CreateEventBridgeRuleTemplate

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Workflow monitor: EventBridge rule templates ID

URI

`/prod/eventbridge-rule-templates/identifier`

HTTP methods

DELETE

Operation ID: `DeleteEventBridgeRuleTemplate`

Path parameters

Name	Type	Required	Description
identifier	String	True	

Responses

Status code	Response model	Description
204	None	204 response
400	BadRequestExceptionResponseContent	400 response
403	ForbiddenExceptionResponseContent	403 response
404	NotFoundExceptionResponseContent	404 response
409	ConflictExceptionResponseContent	409 response
429	TooManyRequestsExceptionResponseContent	429 response
500	InternalServerErrorExceptionResponseContent	500 response

GET

Operation ID: `GetEventBridgeRuleTemplate`

Path parameters

Name	Type	Required	Description
identifier	String	True	

Responses

Status code	Response model	Description
200	GetEventBridgeRuleTemplateResponseContent	200 response
400	BadRequestExceptionResponseContent	400 response
403	ForbiddenExceptionResponseContent	403 response
404	NotFoundExceptionResponseContent	404 response
429	TooManyRequestsExceptionResponseContent	429 response
500	InternalServerErrorExceptionResponseContent	500 response

OPTIONS

Operation ID: `CorsEventbridge_rule_templatesIdentifier`

Path parameters

Name	Type	Required	Description
identifier	String	True	

Responses

Status code	Response model	Description
200	None	200 response

PATCH

Operation ID: `UpdateEventBridgeRuleTemplate`

Path parameters

Name	Type	Required	Description
identifier	String	True	

Responses

Status code	Response model	Description
200	UpdateEventBridgeRuleTemplateResponseContent	200 response
400	BadRequestExceptionResponseContent	400 response
403	ForbiddenExceptionResponseContent	403 response
404	NotFoundExceptionResponseContent	404 response
409	ConflictExceptionResponseContent	409 response
429	TooManyRequestsExceptionResponseContent	429 response

Status code	Response model	Description
500	InternalServerErrorExceptionResponseContent	500 response

Schemas

Request bodies

PATCH schema

```
{
  "description": "string",
  "eventTargets": [
    {
      "arn": "string"
    }
  ],
  "eventType": enum,
  "groupIdentifier": "string",
  "name": "string"
}
```

Response bodies

GetEventBridgeRuleTemplateResponseContent schema

```
{
  "arn": "string",
  "createdAt": "string",
  "description": "string",
  "eventTargets": [
    {
      "arn": "string"
    }
  ],
  "eventType": enum,
  "groupId": "string",
  "id": "string",
```

```
  "modifiedAt": "string",
  "name": "string"
}
```

UpdateEventBridgeRuleTemplateResponseContent schema

```
{
  "arn": "string",
  "createdAt": "string",
  "description": "string",
  "eventTargets": [
    {
      "arn": "string"
    }
  ],
  "eventType": enum,
  "groupId": "string",
  "id": "string",
  "modifiedAt": "string",
  "name": "string"
}
```

BadRequestExceptionResponseContent schema

```
{
  "message": "string"
}
```

ForbiddenExceptionResponseContent schema

```
{
  "message": "string"
}
```

NotFoundExceptionResponseContent schema

```
{
  "message": "string"
```

```
}
```

ConflictExceptionResponseContent schema

```
{
  "message": "string"
}
```

TooManyRequestsExceptionResponseContent schema

```
{
  "message": "string"
}
```

InternalServerErrorExceptionResponseContent schema

```
{
  "message": "string"
}
```

Properties

BadRequestExceptionResponseContent

The input fails to satisfy the constraints specified by an Amazon Web Services service.

message

Exception error message.

Type: string
Required: False

ConflictExceptionResponseContent

Updating or deleting a resource can cause an inconsistent state.

message

Exception error message.

Type: string
Required: False

EventBridgeRuleTemplateEventType

The type of event to match with the rule.

MEDIALIVE_MULTIPLEX_ALERT
MEDIALIVE_MULTIPLEX_STATE_CHANGE
MEDIALIVE_CHANNEL_ALERT
MEDIALIVE_CHANNEL_INPUT_CHANGE
MEDIALIVE_CHANNEL_STATE_CHANGE
MEDIAPACKAGE_INPUT_NOTIFICATION
MEDIAPACKAGE_KEY_PROVIDER_NOTIFICATION
MEDIAPACKAGE_HARVEST_JOB_NOTIFICATION
SIGNAL_MAP_ACTIVE_ALARM
MEDIACONNECT_ALERT
MEDIACONNECT_SOURCE_HEALTH
MEDIACONNECT_OUTPUT_HEALTH
MEDIACONNECT_FLOW_STATUS_CHANGE

EventBridgeRuleTemplateTarget

The target to which to send matching events.

arn

Target ARNs must be either an SNS topic or CloudWatch log group.

Type: string
Required: True
Pattern: `^arn.+$`
MinLength: 1

MaxLength: 2048

ForbiddenExceptionResponseContent

User does not have sufficient access to perform this action.

message

Exception error message.

Type: string
Required: False

GetEventBridgeRuleTemplateResponseContent

arn

An eventbridge rule template's ARN (Amazon Resource Name)

Type: string
Required: True
Pattern: `^arn:.+:medialive:.+:eventbridge-rule-template:.+$`

createdAt

The date and time of resource creation.

Type: string
Required: True
Format: date-time

description

A resource's optional description.

Type: string
Required: False
MinLength: 0

MaxLength: 1024

eventTargets

The destinations that will receive the event notifications.

Type: Array of type EventBridgeRuleTemplateTarget
Required: True

eventType

The type of event to match with the rule.

Type: EventBridgeRuleTemplateEventType
Required: True

groupId

An eventbridge rule template group's id. Amazon Web Services provided template groups have ids that start with <code>`aws-`</code>.

Type: string
Required: True
Pattern: ^(aws-)?[0-9]{7}$
MinLength: 7
MaxLength: 11

id

An eventbridge rule template's id. Amazon Web Services provided templates have ids that start with <code>`aws-`</code>

Type: string
Required: True
Pattern: ^(aws-)?[0-9]{7}$
MinLength: 7
MaxLength: 11

modifiedAt

The date and time of latest resource modification.

Type: string
Required: False
Format: date-time

name

A resource's name. Names must be unique within the scope of a resource type in a specific region.

Type: string
Required: True
Pattern: ^[^\s]+$
MinLength: 1
MaxLength: 255

InternalServerErrorExceptionResponseContent

Unexpected error during processing of request.

message

Exception error message.

Type: string
Required: False

NotFoundExceptionResponseContent

Request references a resource which does not exist.

message

Exception error message.

Type: string
Required: False

TooManyRequestsExceptionResponseContent

Request was denied due to request throttling.

message

Exception error message.

Type: string
Required: False

UpdateEventBridgeRuleTemplateRequestContent

description

A resource's optional description.

Type: string
Required: False
MinLength: 0
MaxLength: 1024

eventTargets

The destinations that will receive the event notifications.

Type: Array of type EventBridgeRuleTemplateTarget
Required: False

eventType

The type of event to match with the rule.

Type: EventBridgeRuleTemplateEventType
Required: False

groupIdentifier

An eventbridge rule template group's identifier. Can be either be its id or current name.

Type: string
Required: False
Pattern: ^[^\s]+$

name

A resource's name. Names must be unique within the scope of a resource type in a specific region.

Type: string
Required: False
Pattern: ^[^\s]+$
MinLength: 1
MaxLength: 255

UpdateEventBridgeRuleTemplateResponseContent

arn

An eventbridge rule template's ARN (Amazon Resource Name)

Type: string
Required: True
Pattern: ^arn:.+:medialive:.+:eventbridge-rule-template:.+$

createdAt

The date and time of resource creation.

Type: string
Required: True
Format: date-time

description

A resource's optional description.

Type: string
Required: False

MinLength: 0
MaxLength: 1024

eventTargets

The destinations that will receive the event notifications.

Type: Array of type EventBridgeRuleTemplateTarget
Required: True

eventType

The type of event to match with the rule.

Type: EventBridgeRuleTemplateEventType
Required: True

groupId

An eventbridge rule template group's id. Amazon Web Services provided template groups have ids that start with <code>`aws-`</code>.

Type: string
Required: True
Pattern: `^(aws-)?[0-9]{7}$`
MinLength: 7
MaxLength: 11

id

An eventbridge rule template's id. Amazon Web Services provided templates have ids that start with <code>`aws-`</code>

Type: string
Required: True
Pattern: `^(aws-)?[0-9]{7}$`
MinLength: 7

MaxLength: 11

modifiedAt

The date and time of latest resource modification.

Type: string
Required: False
Format: date-time

name

A resource's name. Names must be unique within the scope of a resource type in a specific region.

Type: string
Required: True
Pattern: ^[^\s]+$
MinLength: 1
MaxLength: 255

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

DeleteEventBridgeRuleTemplate

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

GetEventBridgeRuleTemplate

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

CorsEventbridge_rule_templatesIdentifier

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UpdateEventBridgeRuleTemplate

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3

- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Workflow monitor: Signal map monitor deployment

URI

`/prod/signal-maps/identifier/monitor-deployment`

HTTP methods

DELETE

Operation ID: `StartDeleteMonitorDeployment`

Path parameters

Name	Type	Required	Description
identifier	String	True	

Responses

Status code	Response model	Description
202	StartDeleteMonitorDeploymentResponseContent	202 response
400	BadRequestExceptionResponseContent	400 response
403	ForbiddenExceptionResponseContent	403 response
404	NotFoundExceptionResponseContent	404 response

Status code	Response model	Description
409	ConflictExceptionResponseContent	409 response
429	TooManyRequestsExceptionResponseContent	429 response
500	InternalServerErrorExceptionResponseContent	500 response

OPTIONS

Operation ID: `CorsSignal_mapsIdentifierMonitor_deployment`

Path parameters

Name	Type	Required	Description
identifier	String	True	

Responses

Status code	Response model	Description
200	None	200 response

POST

Operation ID: `StartMonitorDeployment`

Path parameters

Name	Type	Required	Description
identifier	String	True	

Responses

Status code	Response model	Description
202	StartMonitorDeploymentResponseContent	202 response
400	BadRequestExceptionResponseContent	400 response
403	ForbiddenExceptionResponseContent	403 response
404	NotFoundExceptionResponseContent	404 response
409	ConflictExceptionResponseContent	409 response
429	TooManyRequestsExceptionResponseContent	429 response
500	InternalServerErrorExceptionResponseContent	500 response

Schemas

Request bodies

POST schema

```
{
  "dryRun": boolean
}
```

Response bodies

StartDeleteMonitorDeploymentResponseContent schema

```
{
  "arn": "string",
  "cloudWatchAlarmTemplateGroupIds": [
    "string"
  ],
  "createdAt": "string",
  "description": "string",
  "discoveryEntryPointArn": "string",
  "errorMessage": "string",
  "eventBridgeRuleTemplateGroupIds": [
    "string"
  ],
  "failedMediaResourceMap": {
  },
  "id": "string",
  "lastDiscoveredAt": "string",
  "lastSuccessfulMonitorDeployment": {
    "detailsUri": "string",
    "status": enum
  },
  "mediaResourceMap": {
  },
  "modifiedAt": "string",
  "monitorChangesPendingDeployment": boolean,
  "monitorDeployment": {
    "detailsUri": "string",
    "errorMessage": "string",
    "status": enum
  },
  "name": "string",
  "status": enum
}
```

StartMonitorDeploymentResponseContent schema

```
{
  "arn": "string",
  "cloudWatchAlarmTemplateGroupIds": [
    "string"
  ],
```

```
  "createdAt": "string",
  "description": "string",
  "discoveryEntryPointArn": "string",
  "errorMessage": "string",
  "eventBridgeRuleTemplateGroupIds": [
    "string"
  ],
  "failedMediaResourceMap": {
  },
  "id": "string",
  "lastDiscoveredAt": "string",
  "lastSuccessfulMonitorDeployment": {
    "detailsUri": "string",
    "status": enum
  },
  "mediaResourceMap": {
  },
  "modifiedAt": "string",
  "monitorChangesPendingDeployment": boolean,
  "monitorDeployment": {
    "detailsUri": "string",
    "errorMessage": "string",
    "status": enum
  },
  "name": "string",
  "status": enum
}
```


BadRequestExceptionResponseContent schema


```
{
  "message": "string"
}
```


ForbiddenExceptionResponseContent schema


```
{
  "message": "string"
}
```

NotFoundExceptionResponseContent schema

```
{
  "message": "string"
}
```

ConflictExceptionResponseContent schema

```
{
  "message": "string"
}
```

TooManyRequestsExceptionResponseContent schema

```
{
  "message": "string"
}
```

InternalServerErrorExceptionResponseContent schema

```
{
  "message": "string"
}
```

Properties

BadRequestExceptionResponseContent

The input fails to satisfy the constraints specified by an Amazon Web Services service.

message

Exception error message.

Type: string

Required: False

ConflictExceptionResponseContent

Updating or deleting a resource can cause an inconsistent state.

message

Exception error message.

Type: string
Required: False

FailedMediaResourceMap

A map representing an incomplete Amazon Web Services media workflow as a graph.

key-value pairs

Type: object

ForbiddenExceptionResponseContent

User does not have sufficient access to perform this action.

message

Exception error message.

Type: string
Required: False

InternalServerErrorExceptionResponseContent

Unexpected error during processing of request.

message

Exception error message.

Type: string
Required: False

MediaResource

An Amazon Web Services resource used in media workflows.

destinations

A direct destination neighbor to an Amazon Web Services media resource.

Type: Array of type MediaResourceNeighbor
Required: False

name

The logical name of an Amazon Web Services media resource.

Type: string
Required: False
MinLength: 1
MaxLength: 256

sources

A direct source neighbor to an Amazon Web Services media resource.

Type: Array of type MediaResourceNeighbor
Required: False

MediaResourceMap

A map representing an Amazon Web Services media workflow as a graph.

key-value pairs

Type: object

MediaResourceNeighbor

A direct source or destination neighbor to an Amazon Web Services media resource.

arn

The ARN of a resource used in Amazon Web Services media workflows.

Type: string
Required: True
Pattern: `^arn.+$`
MinLength: 1
MaxLength: 2048

name

The logical name of an Amazon Web Services media resource.

Type: string
Required: False
MinLength: 1
MaxLength: 256

MonitorDeployment

Represents the latest monitor deployment of a signal map.

detailsUri

URI associated with a signal map's monitor deployment.

Type: string
Required: False
MinLength: 1
MaxLength: 2048

errorMessage

Error message associated with a failed monitor deployment of a signal map.

Type: string
Required: False
MinLength: 1

MaxLength: 2048

status

The signal map monitor deployment status.

Type: SignalMapMonitorDeploymentStatus
Required: True

NotFoundExceptionResponseContent

Request references a resource which does not exist.

message

Exception error message.

Type: string
Required: False

SignalMapMonitorDeploymentStatus

A signal map's monitor deployment status.

```
NOT_DEPLOYED
DRY_RUN_DEPLOYMENT_COMPLETE
DRY_RUN_DEPLOYMENT_FAILED
DRY_RUN_DEPLOYMENT_IN_PROGRESS
DEPLOYMENT_COMPLETE
DEPLOYMENT_FAILED
DEPLOYMENT_IN_PROGRESS
DELETE_COMPLETE
DELETE_FAILED
DELETE_IN_PROGRESS
```

SignalMapStatus

A signal map's current status which is dependent on its lifecycle actions or associated jobs.

CREATE_IN_PROGRESS
CREATE_COMPLETE
CREATE_FAILED
UPDATE_IN_PROGRESS
UPDATE_COMPLETE
UPDATE_REVERTED
UPDATE_FAILED
READY
NOT_READY

StartDeleteMonitorDeploymentResponseContent

arn

A signal map's ARN (Amazon Resource Name)

Type: string
Required: True
Pattern: `^arn:.+:medialive:.+:signal-map:.+$`

cloudWatchAlarmTemplateGroupIds

An alarm template group's id.

Type: Array of type string
Required: False
Pattern: `^(aws-)?[0-9]{7}$`
MinLength: 7
MaxLength: 11

createdAt

The date and time of resource creation.

Type: string
Required: True

Format: date-time

description

A resource's optional description.

Type: string
Required: False
MinLength: 0
MaxLength: 1024

discoveryEntryPointArn

A top-level supported Amazon Web Services resource ARN to discover a signal map from.

Type: string
Required: True
MinLength: 1
MaxLength: 2048

errorMessage

Error message associated with a failed creation or failed update attempt of a signal map.

Type: string
Required: False
MinLength: 1
MaxLength: 2048

eventBridgeRuleTemplateGroupIds

An eventbridge rule template group's id.

Type: Array of type string
Required: False
Pattern: `^(aws-)?[0-9]{7}$`
MinLength: 7
MaxLength: 11

failedMediaResourceMap

A map representing an incomplete Amazon Web Services media workflow as a graph.

Type: FailedMediaResourceMap
Required: False

id

A signal map's id.

Type: string
Required: True
Pattern: `^(aws-)?[0-9]{7}$`
MinLength: 7
MaxLength: 11

lastDiscoveredAt

The date and time of latest discovery.

Type: string
Required: False
Format: date-time

lastSuccessfulMonitorDeployment

The date and time of latest successful deployment.

Type: SuccessfulMonitorDeployment
Required: False

mediaResourceMap

A map representing an Amazon Web Services media workflow as a graph.

Type: MediaResourceMap
Required: False

modifiedAt

The date and time of latest resource modification.

Type: string
Required: False
Format: date-time

monitorChangesPendingDeployment

If true, there are pending monitor changes for this signal map that can be deployed.

Type: boolean
Required: True

monitorDeployment

Represents the latest monitor deployment of a signal map.

Type: MonitorDeployment
Required: False

name

A resource's name. Names must be unique within the scope of a resource type in a specific region.

Type: string
Required: True
Pattern: `^[^\s]+$`
MinLength: 1
MaxLength: 255

status

A signal map's current status, which is dependent on its lifecycle actions or associated jobs.

Type: SignalMapStatus
Required: True

StartMonitorDeploymentRequestContent

dryRun

Type: boolean
Required: False

StartMonitorDeploymentResponseContent

arn

A signal map's ARN (Amazon Resource Name)

Type: string
Required: True
Pattern: `^arn:.+:medialive:.+:signal-map:.+$`

cloudWatchAlarmTemplateGroupIds

An alarm template group's id.

Type: Array of type string
Required: False
Pattern: `^(aws-)?[0-9]{7}$`
MinLength: 7
MaxLength: 11

createdAt

The date and time of resource creation.

Type: string
Required: True
Format: date-time

description

A resource's optional description.

Type: string
Required: False
MinLength: 0
MaxLength: 1024

discoveryEntryPointArn

A top-level supported Amazon Web Services resource ARN to discover a signal map from.

Type: string
Required: True
MinLength: 1
MaxLength: 2048

errorMessage

Error message associated with a failed creation or failed update attempt of a signal map.

Type: string
Required: False
MinLength: 1
MaxLength: 2048

eventBridgeRuleTemplateGroupIds

An eventbridge rule template group's id.

Type: Array of type string
Required: False
Pattern: `^(aws-)?[0-9]{7}$`
MinLength: 7
MaxLength: 11

failedMediaResourceMap

A map representing an incomplete Amazon Web Services media workflow as a graph.

Type: FailedMediaResourceMap

Required: False

id

A signal map's id.

Type: string
Required: True
Pattern: `^(aws-)?[0-9]{7}$`
MinLength: 7
MaxLength: 11

lastDiscoveredAt

The date and time of latest discovery.

Type: string
Required: False
Format: date-time

lastSuccessfulMonitorDeployment

The date and time of latest successful deployment.

Type: SuccessfulMonitorDeployment
Required: False

mediaResourceMap

A map representing an Amazon Web Services media workflow as a graph.

Type: MediaResourceMap
Required: False

modifiedAt

The date and time of latest resource modification.

Type: string

Required: False
Format: date-time

monitorChangesPendingDeployment

If true, there are pending monitor changes for this signal map that can be deployed.

Type: boolean
Required: True

monitorDeployment

Represents the latest monitor deployment of a signal map.

Type: MonitorDeployment
Required: False

name

A resource's name. Names must be unique within the scope of a resource type in a specific region.

Type: string
Required: True
Pattern: ^[^\s]+$
MinLength: 1
MaxLength: 255

status

A signal map's current status, which is dependent on its lifecycle actions or associated jobs.

Type: SignalMapStatus
Required: True

SuccessfulMonitorDeployment

Represents the latest successful monitor deployment of a signal map.

detailsUri

URI associated with a signal map's monitor deployment.

Type: string
Required: True
MinLength: 1
MaxLength: 2048

status

A signal map's monitor deployment status.

Type: SignalMapMonitorDeploymentStatus
Required: True

TooManyRequestsExceptionResponseContent

Request was denied due to request throttling.

message

Exception error message.

Type: string
Required: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

StartDeleteMonitorDeployment

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++

- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

CorsSignal_mapsIdentifierMonitor_deployment

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

StartMonitorDeployment

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Workflow monitor: Signal maps

URI

`/prod/signal-maps`

HTTP methods

GET

Operation ID: `ListSignalMaps`

Query parameters

Name	Type	Required	Description
`eventBridgeRuleTemplateGroupIdentifier`	String	False	
`nextToken`	String	False	
`cloudWatchAlarmTemplateGroupIdentifier`	String	False	
`maxResults`	String	False	

Responses

Status code	Response model	Description
200	ListSignalMapsResponseContent	200 response
400	BadRequestExceptionResponseContent	400 response

Status code	Response model	Description
403	ForbiddenExceptionResponseContent	403 response
404	NotFoundExceptionResponseContent	404 response
429	TooManyRequestsExceptionResponseContent	429 response
500	InternalServerErrorExceptionResponseContent	500 response

OPTIONS

Operation ID: CorsSignal_maps

Responses

Status code	Response model	Description
200	None	200 response

POST

Operation ID: CreateSignalMap

Responses

Status code	Response model	Description
201	CreateSignalMapResponseContent	201 response
400	BadRequestExceptionResponseContent	400 response

Status code	Response model	Description
403	ForbiddenExceptionResponseContent	403 response
404	NotFoundExceptionResponseContent	404 response
409	ConflictExceptionResponseContent	409 response
429	TooManyRequestsExceptionResponseContent	429 response
500	InternalServerErrorExceptionResponseContent	500 response

Schemas

Request bodies

POST schema

```
{
  "cloudWatchAlarmTemplateGroupIdentifiers": [
    "string"
  ],
  "description": "string",
  "discoveryEntryPointArn": "string",
  "eventBridgeRuleTemplateGroupIdentifiers": [
    "string"
  ],
  "name": "string"
}
```

Response bodies

ListSignalMapsResponseContent schema

```
{
  "nextToken": "string",
  "signalMaps": [
    {
      "arn": "string",
      "createdAt": "string",
      "description": "string",
      "id": "string",
      "modifiedAt": "string",
      "monitorDeploymentStatus": enum,
      "name": "string",
      "status": enum
    }
  ]
}
```

CreateSignalMapResponseContent schema

```
{
  "arn": "string",
  "cloudWatchAlarmTemplateGroupIds": [
    "string"
  ],
  "createdAt": "string",
  "description": "string",
  "discoveryEntryPointArn": "string",
  "errorMessage": "string",
  "eventBridgeRuleTemplateGroupIds": [
    "string"
  ],
  "failedMediaResourceMap": {
  },
  "id": "string",
  "lastDiscoveredAt": "string",
  "lastSuccessfulMonitorDeployment": {
    "detailsUri": "string",
    "status": enum
  },
  "mediaResourceMap": {
  },
  "modifiedAt": "string",
```

```
  "monitorChangesPendingDeployment": boolean,
  "monitorDeployment": {
    "detailsUri": "string",
    "errorMessage": "string",
    "status": enum
  },
  "name": "string",
  "status": enum
}
```

BadRequestExceptionResponseContent schema

```
{
  "message": "string"
}
```

ForbiddenExceptionResponseContent schema

```
{
  "message": "string"
}
```

NotFoundExceptionResponseContent schema

```
{
  "message": "string"
}
```

ConflictExceptionResponseContent schema

```
{
  "message": "string"
}
```

TooManyRequestsExceptionResponseContent schema

```
{
```

```
  "message": "string"
}
```

InternalServerErrorExceptionResponseContent schema

```
{
  "message": "string"
}
```

Properties

BadRequestExceptionResponseContent

The input fails to satisfy the constraints specified by an Amazon Web Services service.

message

Exception error message.

Type: string
Required: False

ConflictExceptionResponseContent

Updating or deleting a resource can cause an inconsistent state.

message

Exception error message.

Type: string
Required: False

CreateSignalMapRequestContent

cloudWatchAlarmTemplateGroupIdentifiers

A cloudwatch alarm template group's identifier. Can be either be its id or current name.

Type: Array of type string
Required: False
Pattern: ^[^\s]+$

description

A resource's optional description.

Type: string
Required: False
MinLength: 0
MaxLength: 1024

discoveryEntryPointArn

A top-level supported Amazon Web Services resource ARN to discover a signal map from.

Type: string
Required: True
MinLength: 1
MaxLength: 2048

eventBridgeRuleTemplateGroupIdentifiers

An eventbridge rule template group's identifier. Can be either be its id or current name.

Type: Array of type string
Required: False
Pattern: ^[^\s]+$

name

A resource's name. Names must be unique within the scope of a resource type in a specific region.

Type: string
Required: True
Pattern: ^[^\s]+$

MinLength: 1
MaxLength: 255

CreateSignalMapResponseContent

arn

A signal map's ARN (Amazon Resource Name)

Type: string
Required: True
Pattern: `^arn:.+:medialive:.+:signal-map:.+$`

cloudWatchAlarmTemplateGroupIds

An alarm template group's id.

Type: Array of type string
Required: False
Pattern: `^(aws-)?[0-9]{7}$`
MinLength: 7
MaxLength: 11

createdAt

The date and time of resource creation.

Type: string
Required: True
Format: date-time

description

A resource's optional description.

Type: string
Required: False
MinLength: 0

MaxLength: 1024

discoveryEntryPointArn

A top-level supported Amazon Web Services resource ARN to discover a signal map from.

Type: string
Required: True
MinLength: 1
MaxLength: 2048

errorMessage

Error message associated with a failed creation or failed update attempt of a signal map.

Type: string
Required: False
MinLength: 1
MaxLength: 2048

eventBridgeRuleTemplateGroupIds

An eventbridge rule template group's id. Amazon Web Services provided template groups have ids that start with `aws-`.

Type: Array of type string
Required: False
Pattern: `^(aws-)?[0-9]{7}$`
MinLength: 7
MaxLength: 11

failedMediaResourceMap

A map representing an incomplete Amazon Web Services media workflow as a graph.

Type: FailedMediaResourceMap
Required: False

id

A signal map's id.

Type: string
Required: True
Pattern: `^(aws-)?[0-9]{7}$`
MinLength: 7
MaxLength: 11

lastDiscoveredAt

The date and time of latest discovery.

Type: string
Required: False
Format: date-time

lastSuccessfulMonitorDeployment

The date and time of latest successful deployment.

Type: SuccessfulMonitorDeployment
Required: False

mediaResourceMap

A map representing an Amazon Web Services media workflow as a graph.

Type: MediaResourceMap
Required: False

modifiedAt

The date and time of latest resource modification.

Type: string
Required: False
Format: date-time

monitorChangesPendingDeployment

If true, there are pending monitor changes for this signal map that can be deployed.

Type: boolean
Required: True

monitorDeployment

Represents the latest monitor deployment of a signal map.

Type: MonitorDeployment
Required: False

name

A resource's name. Names must be unique within the scope of a resource type in a specific region.

Type: string
Required: True
Pattern: `^[^\s]+$`
MinLength: 1
MaxLength: 255

status

A signal map's current status, which is dependent on its lifecycle actions or associated jobs.

Type: SignalMapStatus
Required: True

FailedMediaResourceMap

A map representing an incomplete Amazon Web Services media workflow as a graph.

key-value pairs

Type: object

ForbiddenExceptionResponseContent

User does not have sufficient access to perform this action.

message

Exception error message.

Type: string
Required: False

InternalServerErrorExceptionResponseContent

Unexpected error during processing of request.

message

Exception error message.

Type: string
Required: False

ListSignalMapsResponseContent

nextToken

A token used to retrieve the next set of results in paginated list responses.

Type: string
Required: False
MinLength: 1
MaxLength: 2048

signalMaps

Details of the signal map

Type: Array of type SignalMapSummary
Required: True

MediaResource

An Amazon Web Services resource used in media workflows.

destinations

A direct destination neighbor to an Amazon Web Services media resource.

Type: Array of type MediaResourceNeighbor
Required: False

name

The logical name of an Amazon Web Services media resource.

Type: string
Required: False
MinLength: 1
MaxLength: 256

sources

A direct source neighbor to an Amazon Web Services media resource.

Type: Array of type MediaResourceNeighbor
Required: False

MediaResourceMap

A map representing an Amazon Web Services media workflow as a graph.

key-value pairs

Type: object

MediaResourceNeighbor

A direct source or destination neighbor to an Amazon Web Services media resource.

arn

The ARN of a resource used in Amazon Web Services media workflows.

Type: string
Required: True
Pattern: `^arn.+$`
MinLength: 1
MaxLength: 2048

name

The logical name of an Amazon Web Services media resource.

Type: string
Required: False
MinLength: 1
MaxLength: 256

MonitorDeployment

Represents the latest monitor deployment of a signal map.

detailsUri

URI associated with a signal map's monitor deployment.

Type: string
Required: False
MinLength: 1
MaxLength: 2048

errorMessage

Error message associated with a failed monitor deployment of a signal map.

Type: string

Required: False

MinLength: 1

MaxLength: 2048

status

The signal map monitor deployment status.

Type: SignalMapMonitorDeploymentStatus

Required: True

NotFoundExceptionResponseContent

Request references a resource which does not exist.

message

Exception error message.

Type: string

Required: False

SignalMapMonitorDeploymentStatus

A signal map's monitor deployment status.

NOT_DEPLOYED

DRY_RUN_DEPLOYMENT_COMPLETE

DRY_RUN_DEPLOYMENT_FAILED

DRY_RUN_DEPLOYMENT_IN_PROGRESS

DEPLOYMENT_COMPLETE

DEPLOYMENT_FAILED

DEPLOYMENT_IN_PROGRESS

DELETE_COMPLETE

DELETE_FAILED

DELETE_IN_PROGRESS

SignalMapStatus

A signal map's current status which is dependent on its lifecycle actions or associated jobs.

CREATE_IN_PROGRESS
CREATE_COMPLETE
CREATE_FAILED
UPDATE_IN_PROGRESS
UPDATE_COMPLETE
UPDATE_REVERTED
UPDATE_FAILED
READY
NOT_READY

SignalMapSummary

Details of the signal map

arn

A signal map's ARN (Amazon Resource Name)

Type: string
Required: True
Pattern: ^arn:.+:medialive:.+:signal-map:.+$

createdAt

The date and time of resource creation.

Type: string
Required: True
Format: date-time

description

A resource's optional description.

Type: string

Required: False
MinLength: 0
MaxLength: 1024

id

A signal map's id.

Type: string
Required: True
Pattern: `^(aws-)?[0-9]{7}$`
MinLength: 7
MaxLength: 11

modifiedAt

The date and time of latest resource modification.

Type: string
Required: False
Format: date-time

monitorDeploymentStatus

A signal map's monitor deployment status.

Type: SignalMapMonitorDeploymentStatus
Required: True

name

A resource's name. Names must be unique within the scope of a resource type in a specific region.

Type: string
Required: True
Pattern: `^[^\s]+$`
MinLength: 1

MaxLength: 255

status

A signal map's current status, which is dependent on its lifecycle actions or associated jobs.

Type: SignalMapStatus
Required: True

SuccessfulMonitorDeployment

Represents the latest successful monitor deployment of a signal map.

detailsUri

URI associated with a signal map's monitor deployment.

Type: string
Required: True
MinLength: 1
MaxLength: 2048

status

A signal map's monitor deployment status.

Type: SignalMapMonitorDeploymentStatus
Required: True

TooManyRequestsExceptionResponseContent

Request was denied due to request throttling.

message

Exception error message.

Type: string
Required: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

ListSignalMaps

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

CorsSignal_maps

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

CreateSignalMap

- AWS Command Line Interface
- AWS SDK for .NET

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Workflow monitor: Signal maps ID

URI

`/prod/signal-maps/identifier`

HTTP methods

DELETE

Operation ID: `DeleteSignalMap`

Path parameters

Name	Type	Required	Description
identifier	String	True	

Responses

Status code	Response model	Description
204	None	204 response
400	BadRequestExceptionResponseContent	400 response
403	ForbiddenExceptionResponseContent	403 response

Status code	Response model	Description
404	NotFoundExceptionResponseContent	404 response
409	ConflictExceptionResponseContent	409 response
429	TooManyRequestsExceptionResponseContent	429 response
500	InternalServerErrorExceptionResponseContent	500 response

GET

Operation ID: GetSignalMap

Path parameters

Name	Type	Required	Description
identifier	String	True	

Responses

Status code	Response model	Description
200	GetSignalMapResponseContent	200 response
400	BadRequestExceptionResponseContent	400 response
403	ForbiddenExceptionResponseContent	403 response

Status code	Response model	Description
404	NotFoundExceptionResponseContent	404 response
429	TooManyRequestsExceptionResponseContent	429 response
500	InternalServerErrorExceptionResponseContent	500 response

OPTIONS

Operation ID: `CorsSignal_mapsIdentifier`

Path parameters

Name	Type	Required	Description
identifier	String	True	

Responses

Status code	Response model	Description
200	None	200 response

PATCH

Operation ID: `StartUpdateSignalMap`

Path parameters

Name	Type	Required	Description
identifier	String	True	

Responses

Status code	Response model	Description
202	StartUpdateSignalMapResponseContent	202 response
400	BadRequestExceptionResponseContent	400 response
403	ForbiddenExceptionResponseContent	403 response
404	NotFoundExceptionResponseContent	404 response
409	ConflictExceptionResponseContent	409 response
429	TooManyRequestsExceptionResponseContent	429 response
500	InternalServerErrorExceptionResponseContent	500 response

Schemas

Request bodies

PATCH schema

```
{
  "cloudWatchAlarmTemplateGroupIdentifiers": [
    "string"
  ],
  "description": "string",
  "discoveryEntryPointArn": "string",
  "eventBridgeRuleTemplateGroupIdentifiers": [
```

```
    "string"
  ],
  "forceRediscovery": boolean,
  "name": "string"
}
```

Response bodies

GetSignalMapResponseContent schema

```
{
  "arn": "string",
  "cloudWatchAlarmTemplateGroupIds": [
    "string"
  ],
  "createdAt": "string",
  "description": "string",
  "discoveryEntryPointArn": "string",
  "errorMessage": "string",
  "eventBridgeRuleTemplateGroupIds": [
    "string"
  ],
  "failedMediaResourceMap": {
  },
  "id": "string",
  "lastDiscoveredAt": "string",
  "lastSuccessfulMonitorDeployment": {
    "detailsUri": "string",
    "status": enum
  },
  "mediaResourceMap": {
  },
  "modifiedAt": "string",
  "monitorChangesPendingDeployment": boolean,
  "monitorDeployment": {
    "detailsUri": "string",
    "errorMessage": "string",
    "status": enum
  },
  "name": "string",
  "status": enum
}
```

StartUpdateSignalMapResponseContent schema

```
{
  "arn": "string",
  "cloudWatchAlarmTemplateGroupIds": [
    "string"
  ],
  "createdAt": "string",
  "description": "string",
  "discoveryEntryPointArn": "string",
  "errorMessage": "string",
  "eventBridgeRuleTemplateGroupIds": [
    "string"
  ],
  "failedMediaResourceMap": {
  },
  "id": "string",
  "lastDiscoveredAt": "string",
  "lastSuccessfulMonitorDeployment": {
    "detailsUri": "string",
    "status": enum
  },
  "mediaResourceMap": {
  },
  "modifiedAt": "string",
  "monitorChangesPendingDeployment": boolean,
  "monitorDeployment": {
    "detailsUri": "string",
    "errorMessage": "string",
    "status": enum
  },
  "name": "string",
  "status": enum
}
```

BadRequestExceptionResponseContent schema

```
{
  "message": "string"
}
```

ForbiddenExceptionResponseContent schema

```
{
  "message": "string"
}
```

NotFoundExceptionResponseContent schema

```
{
  "message": "string"
}
```

ConflictExceptionResponseContent schema

```
{
  "message": "string"
}
```

TooManyRequestsExceptionResponseContent schema

```
{
  "message": "string"
}
```

InternalServerErrorExceptionResponseContent schema

```
{
  "message": "string"
}
```

Properties

BadRequestExceptionResponseContent

The input fails to satisfy the constraints specified by an Amazon Web Services service.

message

Exception error message.

Type: string
Required: False

ConflictExceptionResponseContent

Updating or deleting a resource can cause an inconsistent state.

message

Exception error message.

Type: string
Required: False

FailedMediaResourceMap

A map representing an incomplete Amazon Web Services media workflow as a graph.

key-value pairs

Type: object

ForbiddenExceptionResponseContent

User does not have sufficient access to perform this action.

message

Exception error message.

Type: string
Required: False

GetSignalMapResponseContent

arn

A signal map's ARN (Amazon Resource Name)

Type: string

Required: True
Pattern: `^arn:.+:medialive:.+:signal-map:.+$`

cloudWatchAlarmTemplateGroupIds

An alarm template group's id.

Type: Array of type string
Required: False
Pattern: `^(aws-)?[0-9]{7}$`
MinLength: 7
MaxLength: 11

createdAt

The date and time of resource creation.

Type: string
Required: True
Format: date-time

description

A resource's optional description.

Type: string
Required: False
MinLength: 0
MaxLength: 1024

discoveryEntryPointArn

A top-level supported Amazon Web Services resource ARN to discover a signal map from.

Type: string
Required: True
MinLength: 1
MaxLength: 2048

errorMessage

Error message associated with a failed creation or failed update attempt of a signal map.

Type: string
Required: False
MinLength: 1
MaxLength: 2048

eventBridgeRuleTemplateGroupIds

An eventbridge rule template group's id.

Type: Array of type string
Required: False
Pattern: `^(aws-)?[0-9]{7}$`
MinLength: 7
MaxLength: 11

failedMediaResourceMap

A map representing an incomplete Amazon Web Services media workflow as a graph.

Type: FailedMediaResourceMap
Required: False

id

A signal map's id.

Type: string
Required: True
Pattern: `^(aws-)?[0-9]{7}$`
MinLength: 7
MaxLength: 11

lastDiscoveredAt

The date and time of latest discovery.

Type: string
Required: False
Format: date-time

lastSuccessfulMonitorDeployment

The date and time of latest successful deployment.

Type: SuccessfulMonitorDeployment
Required: False

mediaResourceMap

A map representing an Amazon Web Services media workflow as a graph.

Type: MediaResourceMap
Required: False

modifiedAt

The date and time of latest resource modification.

Type: string
Required: False
Format: date-time

monitorChangesPendingDeployment

If true, there are pending monitor changes for this signal map that can be deployed.

Type: boolean
Required: True

monitorDeployment

Represents the latest monitor deployment of a signal map.

Type: MonitorDeployment

Required: False

name

A resource's name. Names must be unique within the scope of a resource type in a specific region.

Type: string
Required: True
Pattern: ^[^\s]+$
MinLength: 1
MaxLength: 255

status

A signal map's current status, which is dependent on its lifecycle actions or associated jobs.

Type: SignalMapStatus
Required: True

InternalServerErrorExceptionResponseContent

Unexpected error during processing of request.

message

Exception error message.

Type: string
Required: False

MediaResource

An Amazon Web Services resource used in media workflows.

destinations

A direct destination neighbor to an Amazon Web Services media resource.

Type: Array of type MediaResourceNeighbor

Required: False

name

The logical name of an Amazon Web Services media resource.

Type: string
Required: False
MinLength: 1
MaxLength: 256

sources

A direct source neighbor to an Amazon Web Services media resource.

Type: Array of type MediaResourceNeighbor
Required: False

MediaResourceMap

A map representing an Amazon Web Services media workflow as a graph.

key-value pairs

Type: object

MediaResourceNeighbor

A direct source or destination neighbor to an Amazon Web Services media resource.

arn

The ARN of a resource used in Amazon Web Services media workflows.

Type: string
Required: True
Pattern: `^arn.+$`
MinLength: 1

MaxLength: 2048

name

The logical name of an Amazon Web Services media resource.

Type: string
Required: False
MinLength: 1
MaxLength: 256

MonitorDeployment

Represents the latest monitor deployment of a signal map.

detailsUri

URI associated with a signal map's monitor deployment.

Type: string
Required: False
MinLength: 1
MaxLength: 2048

errorMessage

Error message associated with a failed monitor deployment of a signal map.

Type: string
Required: False
MinLength: 1
MaxLength: 2048

status

The signal map monitor deployment status.

Type: SignalMapMonitorDeploymentStatus

Required: True

NotFoundExceptionResponseContent

Request references a resource which does not exist.

message

Exception error message.

Type: string
Required: False

SignalMapMonitorDeploymentStatus

A signal map's monitor deployment status.

NOT_DEPLOYED
DRY_RUN_DEPLOYMENT_COMPLETE
DRY_RUN_DEPLOYMENT_FAILED
DRY_RUN_DEPLOYMENT_IN_PROGRESS
DEPLOYMENT_COMPLETE
DEPLOYMENT_FAILED
DEPLOYMENT_IN_PROGRESS
DELETE_COMPLETE
DELETE_FAILED
DELETE_IN_PROGRESS

SignalMapStatus

A signal map's current status which is dependent on its lifecycle actions or associated jobs.

CREATE_IN_PROGRESS
CREATE_COMPLETE
CREATE_FAILED
UPDATE_IN_PROGRESS
UPDATE_COMPLETE

UPDATE_REVERTED
UPDATE_FAILED
READY
NOT_READY

StartUpdateSignalMapRequestContent

cloudWatchAlarmTemplateGroupIdentifiers

A cloudwatch alarm template group's identifier. Can be either be its id or current name.

Type: Array of type string
Required: False
Pattern: ^[^\s]+$

description

A resource's optional description.

Type: string
Required: False
MinLength: 0
MaxLength: 1024

discoveryEntryPointArn

A top-level supported Amazon Web Services resource ARN to discover a signal map from.

Type: string
Required: False
MinLength: 1
MaxLength: 2048

eventBridgeRuleTemplateGroupIdentifiers

An eventbridge rule template group's identifier. Can be either be its id or current name.

Type: Array of type string

Required: False
Pattern: ^[^\s]+$

forceRediscovery

If true, will force a rediscovery of a signal map if an unchanged discoveryEntryPointArn is provided.

Type: boolean
Required: False

name

A resource's name. Names must be unique within the scope of a resource type in a specific region.

Type: string
Required: False
Pattern: ^[^\s]+$
MinLength: 1
MaxLength: 255

StartUpdateSignalMapResponseContent

arn

A signal map's ARN (Amazon Resource Name)

Type: string
Required: True
Pattern: ^arn:.+:medialive:.+:signal-map:.+$

cloudWatchAlarmTemplateGroupIds

An alarm template group's id.

Type: Array of type string
Required: False
Pattern: ^(aws-)?[0-9]{7}$

MinLength: 7
MaxLength: 11

createdAt

The date and time of resource creation.

Type: string
Required: True
Format: date-time

description

A resource's optional description.

Type: string
Required: False
MinLength: 0
MaxLength: 1024

discoveryEntryPointArn

A top-level supported Amazon Web Services resource ARN to discover a signal map from.

Type: string
Required: True
MinLength: 1
MaxLength: 2048

errorMessage

Error message associated with a failed creation or failed update attempt of a signal map.

Type: string
Required: False
MinLength: 1
MaxLength: 2048

eventBridgeRuleTemplateGroupIds

An eventbridge rule template group's id. Amazon Web Services provided template groups have ids that start with <code>`aws-`</code>.

Type: Array of type string
Required: False
Pattern: `^(aws-)?[0-9]{7}$`
MinLength: 7
MaxLength: 11

failedMediaResourceMap

A map representing an incomplete Amazon Web Services media workflow as a graph.

Type: FailedMediaResourceMap
Required: False

id

A signal map's id.

Type: string
Required: True
Pattern: `^(aws-)?[0-9]{7}$`
MinLength: 7
MaxLength: 11

lastDiscoveredAt

The date and time of latest discovery.

Type: string
Required: False
Format: date-time

lastSuccessfulMonitorDeployment

The date and time of latest successful deployment.

Type: SuccessfulMonitorDeployment
Required: False

mediaResourceMap

A map representing an Amazon Web Services media workflow as a graph.

Type: MediaResourceMap
Required: False

modifiedAt

The date and time of latest resource modification.

Type: string
Required: False
Format: date-time

monitorChangesPendingDeployment

If true, there are pending monitor changes for this signal map that can be deployed.

Type: boolean
Required: True

monitorDeployment

Represents the latest monitor deployment of a signal map.

Type: MonitorDeployment
Required: False

name

A resource's name. Names must be unique within the scope of a resource type in a specific region.

Type: string
Required: True
Pattern: `^[^\s]+$`

MinLength: 1
MaxLength: 255

status

A signal map's current status, which is dependent on its lifecycle actions or associated jobs.

Type: SignalMapStatus
Required: True

SuccessfulMonitorDeployment

Represents the latest successful monitor deployment of a signal map.

detailsUri

URI associated with a signal map's monitor deployment.

Type: string
Required: True
MinLength: 1
MaxLength: 2048

status

A signal map's monitor deployment status.

Type: SignalMapMonitorDeploymentStatus
Required: True

TooManyRequestsExceptionResponseContent

Request was denied due to request throttling.

message

Exception error message.

Type: string
Required: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

DeleteSignalMap

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

GetSignalMap

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

CorsSignal_mapsIdentifier

- AWS Command Line Interface
- AWS SDK for .NET

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

StartUpdateSignalMap

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

Document History

The following table describes the main changes to this documentation.

- **API version: latest**

Change	Description	Date
Workflow monitor	Workflow monitor features have been added to the API. Workflow monitor is a tool to analyze AWS media services and create signal maps, visualizations of the media workflow, between those services. Use the signal maps to generate monitoring alarms and notifications using CloudWatch, EventBridge, and AWS CloudFormation.	April 11, 2024
Features added since November, 2021	Documentation for features added from November 2021 to July 2022, including: • AWS Elemental Link remote reboots and remote software updates • Reservation automatic renewal • SCTE-35 input PID selection • PDT clock source selector for HLS outputs	July 19, 2022

Change	Description	Date
	• Accessibility captions for HLS and MediaPackage outputs	
Features added since April, 2021	Documentation for features added from April 2021 to November 2021, including: • Transport Stream (TS) file inputs • Nielsen watermarks • Support for font styles with WebVTT captions • WebVTT output captions • Amazon CloudWatch metrics • Pipeline locking with UDP outputs • Trick-play in MediaPackage output group	November 5, 2021

Change	Description	Date
Features added since December, 2020	Documentation for features added from December 2020 to April 2021, including: • Delivery via your VPC • Captions in CDI inputs • Clipping a file input that is stored on an HTTP server • Trick-play in HLS outputs (according to the Image media playlist specification) • Automatic input failover with CDI inputs • Support for transferring AWS Elemental Link devices between Regions • Support for ACLs when delivering outputs to Amazon S3 • Support for motion graphics overlays via the channel schedule • Font styling on TTML output captions	May 3, 2020

Change	Description	Date
Features added since August, 2020	Documentation for features added since August, 2020, including: • Batch operations for the channel • Operations to handle transfers of AWS Elemental Link devices • Operations for input thumbnails.	December 28, 2020
Rewrite of resources.	The descriptions of many of the elements in the resources other than Channel have been revised.	August 26, 2020
Rewrite of Channels and Channels (ID) resources.	The descriptions of many of the elements in these two resources have been revised.	August 20, 2020

Change	Description	Date
Features added since September, 2019	Documentation for features added since September, 2019, including: • EBU-TT-D captions in a captions output • AWS Elemental Devices and the AWS Elemental Link hardware device • Multiplex • Automatic input failover • Input prepare action in the schedule • Immediate mode for all actions in the schedule • Follow mode for SCTE-35 actions in the schedule • Enhanced VQ mode • ID3 segment tagging • Nielsen watermarks	August 10, 2020
Changes to support the H.265 (HEVC) codec	Documentation for configuri ng H.265 in Channels	September 12, 2019

Change	Description	Date
Changes to support SDR and HDR color space handling	Documentation for handling color space in the input, in VideoSelector in Channels Documentation for handling color space in the output, in H264ColorMetadata, H264ColorSpaceSettings, H265ColorMetadata and H265ColorSpaceSettings in Channels	September 12, 2019
Changes to support input clipping when switching inputs, in the channel schedule	Documentation for InputClippingSettings in Channels channelId Schedule	July 25, 2019
Changes to support switching inputs immediately, in the channel schedule	Documentation for urlPath in InputSwitchScheduleActionSettings in Channels channelId Schedule	July 25, 2019
Changes to support dynamic inputs in the channel schedule	Documentation for urlPath in InputSwitchScheduleActionSettings in Channels channelId Schedule	July 25, 2019

Change	Description	Date
UpdateChannelClass operation in the channel, and more	Documentation for the UpdateChannelClass operation in the channel Documentation for the MediaPackage output group type Documentation for pausing and unpausing a channel using the channel schedule Documentation for the RTP push input and RTMP push input connected to an upstream system that is in your Amazon VPC Documentation for tagging using MediaLive Documentation for the frame capture output group	May 2, 2019

Change	Description	Date
Integration with AWS Elemental MediaConnect, and more	Documentation for the MediaConnect input type in the Channel resource Documentation for Input switching using the channel schedule Documentation for Schedule feature in the Channel resource Documentation for Reservations resource Documentation for the RTMP output type in the Channel resource	December 7, 2018
New AWS Elemental MediaLive service release	Initial documentation for the MediaLive service.	November 27, 2017

AWS Glossary

For the latest AWS terminology, see the AWS glossary in the *AWS Glossary Reference*.