# Table of Contents

- What Is AWS Elemental MediaLive? .............................................................................................................. 1
- How MediaLive Works ................................................................................................................................... 1
  - Pipelines .................................................................................................................................................. 2
  - MediaLive Inputs ................................................................................................................................... 2
  - MediaLive Channels ................................................................................................................................. 2
  - MediaLive Schedule ............................................................................................................................... 3
- Terminology .................................................................................................................................................. 4
- Related Services ........................................................................................................................................ 5
- Accessing AWS Elemental MediaLive ......................................................................................................... 6
- Pricing ........................................................................................................................................................ 6
- Setting Up .................................................................................................................................................. 8
  - Signing Up ........................................................................................................................................... 8
  - Creating an Administrator IAM User ................................................................................................... 9
  - Creating a Non-Administrator IAM User .......................................................................................... 10
    - Step 1: Create Customer Managed Policies .................................................................................. 11
    - Step 2: Create an IAM Group ........................................................................................................ 12
    - Step 3: Create or Add an IAM User to Your Group ................................................................... 13
  - Setting up as a Trusted Service ......................................................................................................... 14
- Data Retention ........................................................................................................................................ 15
-Deleting Data in AWS Elemental MediaLive ............................................................................................. 15

Setting Up for Production ........................................................................................................................... 16

- Setting Up Administrators and Users .................................................................................................. 17
  - Creating an Administrator User with Limited Access ................................................................... 17
  - Creating a Non-Administrator User ................................................................................................. 20
    - Step 1: Requirements for Permissions ....................................................................................... 21
    - Summary of Step 1 Access Requirements ................................................................................ 31
    - Step 2: Identify Categories of Users ........................................................................................... 33
    - Step 3: Create the Custom Policies ............................................................................................. 34
    - Step 4: Create the Groups ............................................................................................................. 34
    - Step 5: Create Users ....................................................................................................................... 35
    - Step 6: Setting Up Required Data ............................................................................................... 37
  - Setting Up AWS Elemental MediaLive as a Trusted Service ............................................................ 37
    - How the Trusted Entity Is Created and Attached ..................................................................... 38
    - Creating Trusted Entity Roles .................................................................................................... 38
    - Setting Up Permissions for Non-Administrator Users .................................................................. 44

Getting Started Tutorial ............................................................................................................................ 47

  - Prerequisites ....................................................................................................................................... 47
  - Step 1: Set Up the Upstream System ............................................................................................... 47
  - Step 2: Set Up the Downstream System ......................................................................................... 48
  - Step 3: Create an Input ...................................................................................................................... 48
  - Step 4: Set up Key Information ....................................................................................................... 49
  - Step 5: Attach the Input ..................................................................................................................... 49
  - Step 6: Set up Input Video, Audio, Captions ................................................................................ 49
  - Step 7: Create an HLS Output Group ............................................................................................ 50
  - Step 8: Set Up the Output and Encodes ....................................................................................... 51
  - Step 9: Create Your Channel ........................................................................................................ 51
  - Step 10: Start the Upstream System and the Channel ................................................................. 51
  - Step 11: Clean Up ............................................................................................................................. 52

Components of AWS Elemental MediaLive ................................................................................................. 53

  - Inputs ................................................................................................................................................... 53
  - Channels ............................................................................................................................................. 53
  - Encodes ............................................................................................................................................. 53
  - Outputs ............................................................................................................................................. 54
  - Output Groups ................................................................................................................................. 54
Input Security Groups .......................................................................................................................... 54
How Components Are Associated .......................................................................................................... 54
Planning Your Workflow for AWS Elemental MediaLive ............................................................................. 55
Assessing the Video Source ...................................................................................................................... 55
Determining the Channel Class .................................................................................................................. 56
Setting Up the Upstream System ............................................................................................................. 56
MediaConnect Push .................................................................................................................................. 57
MP4 Pull .................................................................................................................................................. 57
RTP Push .................................................................................................................................................. 58
RTMP Push ................................................................................................................................................ 58
RTMP Pull ................................................................................................................................................ 59
HLS Pull ................................................................................................................................................... 59
Planning the Inputs .................................................................................................................................... 60
Planning the Channel ............................................................................................................................... 60
Rules for Encodes in an Output ................................................................................................................... 61
Examples of Channel Designs ..................................................................................................................... 61
Non-ABR Asset with Captions Embedded in the Video ............................................................................. 62
Non-ABR Segmented Asset with Captions as Separate Objects ................................................................. 62
Non-ABR Segmented Asset with Captions as Sidecars ........................................................................... 62
ABR Asset with Captions Embedded in the Video ..................................................................................... 63
ABR Asset with Captions in Sidecars ........................................................................................................ 63
Setting Up the Downstream System ........................................................................................................... 63
For an Archive Output ................................................................................................................................. 64
For an HLS Output ..................................................................................................................................... 64
For a MediaPackage Output ......................................................................................................................... 65
For a Frame Capture Output ....................................................................................................................... 65
For a Microsoft Smooth Output .................................................................................................................. 66
For an RTMP or RTMPS Output .................................................................................................................. 66
For a UDP Output ....................................................................................................................................... 66
Next Steps ................................................................................................................................................... 66
Working with Input Security Groups .......................................................................................................... 67
Purpose of an Input Security Group ............................................................................................................. 67
Creating an Input Security Group ............................................................................................................... 67
Editing an Input Security Group .................................................................................................................. 68
Deleting an Input Security Group ................................................................................................................ 68
Working with Inputs ................................................................................................................................. 70
Inputs, Input Security Groups, and Channels .............................................................................................. 70
Creating an Input ......................................................................................................................................... 70
Creating an HLS Pull Input .......................................................................................................................... 71
Creating a MediaConnect Push Input .......................................................................................................... 72
Creating an MP4 Pull Input .......................................................................................................................... 74
Creating an RTMP Push Input ..................................................................................................................... 75
Creating an RTMP Pull Input ....................................................................................................................... 76
Creating an RTMP Push Input in Amazon VPC ........................................................................................ 76
Creating an RTMP Pull Input ....................................................................................................................... 79
Creating an RTP Push Input ........................................................................................................................ 80
Creating an RTP Push Input in Amazon VPC ........................................................................................... 81
Editing an Input ........................................................................................................................................... 84
Deleting an Input ......................................................................................................................................... 85
Creating a Channel from Scratch .............................................................................................................. 86
Step 1: Complete the Channel and Input Details ....................................................................................... 86
IAM Role and ARN ....................................................................................................................................... 87
Channel Class ............................................................................................................................................ 88
Input Specifications Settings ....................................................................................................................... 88
Step 2: Attach Inputs to the Channel ......................................................................................................... 89
Channel Input—HLS Pull Input .................................................................................................................. 89
Channel Input—MediaConnect Push Input .................................................................................................. 90
Channel Input—MP4 Pull Input ................................................................................................................... 90
Enabling Channel Logs ................................................................. 129
Working with Logs ................................................................. 129
Logging Using CloudTrail ............................................................. 130
MediaLive Information in CloudTrail ................................................ 131
Understanding MediaLive Log File Entries ...................................... 131
Editing and Deleting a Channel ..................................................... 133
Editing a Channel ........................................................................ 133
Changing the Channel Class ......................................................... 133
Editing the Tags Associated with a Channel .................................... 134
Deleting a Channel ................................................................. 135
Working with the Schedule ........................................................... 136
Actions in the Schedule .............................................................. 136
Types of Schedule Actions for Input Switching ................................. 136
Types of Schedule Actions for Image Overlay ................................. 137
Types of Schedule Actions for SCTE-35 .......................................... 137
Types of Schedule Actions for ID3 Metadata .................................... 137
Types of Schedule Actions for Pause ............................................. 138
Working with the Schedule (Console) ............................................... 138
Creating Actions ......................................................................... 138
Deleting Actions .......................................................................... 144
Modifying Actions ....................................................................... 145
Viewing the Schedule ................................................................... 147
Working with the Schedule (AWS CLI) ............................................. 148
Creating and Deleting Using a Batch Command .............................. 148
Submitting a Batch Update Schedule Command .............................. 150
JSON for Creates ......................................................................... 151
JSON for Deletes ......................................................................... 162
JSON for Combinations ............................................................... 163
Viewing the Schedule ................................................................... 164
Working with Captions ................................................................. 167
Supported Features ....................................................................... 167
Supported Formats ........................................................................ 167
Format Support by Output Container ............................................. 167
Support for Languages ................................................................... 168
Support for Font Styles in Output Captions ..................................... 168
Captions in Channels with Multiple Inputs ...................................... 169
Typical Scenarios .......................................................................... 170
Use Case A ................................................................................... 170
Use Case B ................................................................................... 171
Use Case C ................................................................................... 171
Use Case D ................................................................................... 172
Setting Up for Captions ............................................................... 172
Step 1: Create Captions Selectors in the Input ................................. 172
Step 2: Plan Captions for the Outputs ............................................ 176
Step 3: Match Formats to Categories ........................................... 177
Step 4: Create Captions Encodes .................................................. 179
Examples ...................................................................................... 182
Use Case A ................................................................................... 182
Use Case B ................................................................................... 183
Use Case C ................................................................................... 184
Use Case D ................................................................................... 185
Input Switching ............................................................................ 189
Typical Use Cases ......................................................................... 189
Use Case 1: One Live Feed and One File Input Alternating ............. 189
Use Case 2: One Live Feed and File Inputs, and the Channel Starts with a File Input 190
Use Case 3: Two Live Feeds .......................................................... 190
Recommended Procedure ............................................................. 190
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rules and Limits for Input Switches</td>
<td>191</td>
</tr>
<tr>
<td>Rules for Types of Inputs</td>
<td>191</td>
</tr>
<tr>
<td>Limit to Inputs in a Channel</td>
<td>191</td>
</tr>
<tr>
<td>No Limits to the Number of Input Switches</td>
<td>191</td>
</tr>
<tr>
<td>Reusing a File Input</td>
<td>192</td>
</tr>
<tr>
<td>Types of Switches—Fixed, Follow, and Follow Chains</td>
<td>192</td>
</tr>
<tr>
<td>Planning the Schedule of Input Switches</td>
<td>193</td>
</tr>
<tr>
<td>Planning the Names</td>
<td>193</td>
</tr>
<tr>
<td>Handling the Transition When the Next Input is Fixed</td>
<td>193</td>
</tr>
<tr>
<td>Handling the Transition When the Next Input is Follow</td>
<td>194</td>
</tr>
<tr>
<td>Creating Inputs for Input Switching</td>
<td>194</td>
</tr>
<tr>
<td>Rules for Setup</td>
<td>194</td>
</tr>
<tr>
<td>Creating a Channel with Multiple Inputs</td>
<td>195</td>
</tr>
<tr>
<td>Setting Up the Schedule with Input Switches</td>
<td>196</td>
</tr>
<tr>
<td>Starting a Channel That Has Multiple Inputs</td>
<td>196</td>
</tr>
<tr>
<td>What Happens at Runtime</td>
<td>196</td>
</tr>
<tr>
<td>Restarting a Channel</td>
<td>196</td>
</tr>
<tr>
<td>Working with ID3 Metadata</td>
<td>198</td>
</tr>
<tr>
<td>Enabling ID3 Metadata</td>
<td>198</td>
</tr>
<tr>
<td>Enabling in Archive Outputs</td>
<td>198</td>
</tr>
<tr>
<td>Enabling in HLS Outputs</td>
<td>198</td>
</tr>
<tr>
<td>Enabling in MediaPackage Outputs</td>
<td>199</td>
</tr>
<tr>
<td>Enabling in UDP Outputs</td>
<td>199</td>
</tr>
<tr>
<td>Results of Enabling ID3 Metadata</td>
<td>199</td>
</tr>
<tr>
<td>Passing Through ID3 Metadata</td>
<td>200</td>
</tr>
<tr>
<td>Inserting ID3 Metadata When Creating the Channel</td>
<td>200</td>
</tr>
<tr>
<td>Inserting ID3 Metadata Using the Schedule</td>
<td>201</td>
</tr>
<tr>
<td>SCTE-35 Message Processing</td>
<td>202</td>
</tr>
<tr>
<td>About Message Processing</td>
<td>202</td>
</tr>
<tr>
<td>Supported Features by Input Type</td>
<td>203</td>
</tr>
<tr>
<td>Supported Output Features</td>
<td>203</td>
</tr>
<tr>
<td>Processing Features – Default Behavior</td>
<td>204</td>
</tr>
<tr>
<td>Scope of Processing by Feature</td>
<td>204</td>
</tr>
<tr>
<td>Supported Features by Output Type</td>
<td>205</td>
</tr>
<tr>
<td>Getting Ready: Set the Ad Avail Mode</td>
<td>208</td>
</tr>
<tr>
<td>Enabling Manifest Decoration</td>
<td>209</td>
</tr>
<tr>
<td>Enabling Decoration – HLS</td>
<td>209</td>
</tr>
<tr>
<td>Enabling Decoration – Microsoft Smooth</td>
<td>210</td>
</tr>
<tr>
<td>How SCTE-35 Events Are Handled in Manifests and Sparse Tracks</td>
<td>210</td>
</tr>
<tr>
<td>Enabling Ad Avail Blanking</td>
<td>212</td>
</tr>
<tr>
<td>Enabling Blanking</td>
<td>212</td>
</tr>
<tr>
<td>Triggers for Ad Avail Blanking</td>
<td>213</td>
</tr>
<tr>
<td>Ad Avail Blanking Restriction Flags</td>
<td>215</td>
</tr>
<tr>
<td>Enabling Blackout</td>
<td>216</td>
</tr>
<tr>
<td>Enabling Blackout</td>
<td>217</td>
</tr>
<tr>
<td>Triggers for Blackout</td>
<td>218</td>
</tr>
<tr>
<td>Blackout Restriction Flags</td>
<td>219</td>
</tr>
<tr>
<td>Enabling SCTE-35 Passthrough or Removal</td>
<td>220</td>
</tr>
<tr>
<td>Enabling Passthrough for Archive Outputs</td>
<td>220</td>
</tr>
<tr>
<td>Enabling Passthrough for HLS Outputs</td>
<td>221</td>
</tr>
<tr>
<td>Enabling Passthrough for UDP Outputs</td>
<td>221</td>
</tr>
<tr>
<td>Sample Manifests - HLS</td>
<td>222</td>
</tr>
<tr>
<td>Ad Marker: Adobe</td>
<td>222</td>
</tr>
<tr>
<td>Ad Marker: Elemental</td>
<td>222</td>
</tr>
<tr>
<td>Ad Marker: SCTE-35 Enhanced</td>
<td>223</td>
</tr>
<tr>
<td>Working with Image Overlays</td>
<td>225</td>
</tr>
</tbody>
</table>
Examples ........................................................................................................................................ 225
Features of the Static Image Overlay ................................................................................................... 225
Step 1: Prepare the Static Image Overlay File .................................................................................. 226
Step 2: Insert the Overlay ............................................................................................................. 226
Tagging Resources .......................................................................................................................... 227
Supported Resources in AWS Elemental MediaLive .......................................................................... 227
Tag Restrictions ............................................................................................................................ 227
Managing Tags .................................................................................................................................. 228
Tagging Using the AWS Management Console .................................................................................. 228
Tagging Using MediaLive ................................................................................................................. 228
Reference: Supported Captions ......................................................................................................... 229
How to Read the Supported Captions Information ............................................................................ 229
General Information About Supported Formats ............................................................................... 229
Formats Supported in an HLS Output or a MediaPackage Output ........................................................ 231
Formats Supported in an RTMP Output ............................................................................................. 232
Formats Supported in an MPEG2-TS File Output or MPEG2-UDP Streaming Output .................. 233
Formats Supported in a Microsoft Smooth Output ............................................................................ 234
Reference: Supported Containers and Codecs ............................................................................... 236
Supported Input Types and Upstream Systems ................................................................................... 236
Ingesting with Push .......................................................................................................................... 237
Ingesting with Pull ............................................................................................................................. 238
Supported Codecs for Inputs ............................................................................................................. 238
Supported Containers and Downstream Systems ............................................................................. 238
Supported Codecs for Outputs ......................................................................................................... 240
Supported Codecs for File Outputs .................................................................................................. 240
Supported Codecs for Streaming Outputs ....................................................................................... 240
Reference: Identifiers for Variable Data ............................................................................................. 241
Working with Reservations .............................................................................................................. 243
Input and Output Reservations ......................................................................................................... 243
Input Reservation Attributes and Matching ....................................................................................... 243
Output Reservation Attributes and Matching .................................................................................... 243
How an Input or Output Reservation Is Applied ................................................................................ 244
Add-on Reservations ....................................................................................................................... 245
Reservation Attributes ..................................................................................................................... 245
How an Add-on Reservation Is Applied ............................................................................................ 245
Purchasing a Reservation ................................................................................................................. 247
Filtering on the Offerings Page ......................................................................................................... 247
Viewing Purchased Reservations ...................................................................................................... 247
Deleting a Reservation ...................................................................................................................... 248
Limits ................................................................................................................................................ 249
Soft Limits ...................................................................................................................................... 249
Hard Limits .................................................................................................................................... 249
Document History ............................................................................................................................ 250
AWS Glossary ................................................................................................................................. 255
What Is AWS Elemental MediaLive?

AWS Elemental MediaLive is a real-time video service that lets you create live outputs for broadcast and streaming delivery.

You use MediaLive to transform live video content from one format and package into other formats and packages. You typically need to transform the content in order to provide a format and package that a playback device can handle. Playback devices include smartphones and set-top boxes attached to televisions.

**Topics**

- How AWS Elemental MediaLive Works (p. 1)
- AWS Elemental MediaLive Terminology (p. 4)
- Related Services (p. 5)
- Accessing AWS Elemental MediaLive (p. 6)
- Pricing (p. 6)

How AWS Elemental MediaLive Works

From the point of view of AWS Elemental MediaLive, a live streaming workflow that includes MediaLive involves three systems:

- A MediaLive **channel**, which ingests and transcodes source content.
- One or more **upstream systems** that provide the **source content** (the video) to MediaLive.

  Examples of an upstream system are a streaming camera or appliance that is directly connected to the internet, or a contribution encoder that is located in a sports stadium where a sports event is being held.

  The source content is in a specific package format and protocol. For example, the source content might be available as streaming HLS or streaming TS (transport stream). The source content contains video, audio, and optional captions streams that are in specific codecs or formats.

- One or more **downstream systems** that are the destinations for the output that MediaLive produces.

  A typical downstream system consists of an origin service or a packager that is connected to MediaLive, a content distribution network (CDN) that is downstream of the origin service or the packager, and a playback device or website where the users view the content. AWS Elemental MediaPackage is an example of an origin service and packager. Amazon CloudFront is an example of a CDN.

  To create a MediaLive workflow, you create one or more MediaLive inputs. The inputs contain information about how MediaLive and the upstream system are connected. You also create a MediaLive channel and attach the inputs to the channel. The channel configuration data includes information about how MediaLive connects to the downstream systems.

  This setup connects the components as illustrated in this diagram.
To start processing the content, you start the channel. When the channel is running, it ingests the source content from the upstream system that is identified by the input. The channel then transcodes that video (and the related audio, captions, and metadata) and creates outputs. MediaLive sends the outputs to the specified downstream systems.

**Pipelines**

The processing within MediaLive occurs within one or two pipelines.

If you set up the workflow so that the channel and inputs have two pipelines (recommended), both pipelines work independently of each other but perform identical processing. Setting up with two pipelines provides resiliency within MediaLive.

With two pipelines, the upstream system must be set up to provide two sources, and the downstream system must be set up to receive two outputs.

**AWS Elemental MediaLive Inputs**

An input contains information about how the upstream system and the channel connect to each other. The connection between the input and the upstream system might be a push (the upstream system pushes the content) or a pull (MediaLive pulls the content from the upstream system).

A push input has a MediaLive input security group associated with it. The input security group identifies a range of IP addresses that includes the source addresses on the upstream system. IP addresses within this range are allowed to push content to the input.

**AWS Elemental MediaLive Channels**

A channel can have several inputs attached to it, but it only ingests source content from one input at a time. (You use the channel schedule (p. 3) to set up the channel to switch from one input to another.)

The channel ingests the source content, transcodes it (decodes and encodes it), and packages it into output groups.
The channel contains one or more output groups. There are different types of output groups to handle the requirements of different downstream systems.

The output group consists of one or more outputs. Each output contains a specific combination of encodes. An encode is one video stream, one audio stream, or one captions track. Different encodes have different characteristics. The rules for combining encodes into outputs and for combining outputs into output groups depend on the type of the output group.

The following diagram is a detailed illustration of the workflow.

The illustration shows a channel with only one output group.

As another example, the channel might contain one HLS output group and one RTMP output group. The HLS output group might contain two outputs. One HLS output contains one high-resolution video, one audio, and one captions encode. The other HLS output contains one low-resolution video, one audio, and no captions. The RTMP output group contains one output that contains one video and one audio.

For information about designing this workflow and creating a channel, see Planning Your Workflow for AWS Elemental MediaLive (p. 55).

**AWS Elemental MediaLive Schedule**

Each MediaLive channel has one schedule associated with it. You add actions to the schedule to suit your requirements. There are different types of actions, including “switch input” (to switch to ingesting a different input) and “insert image overlay” (to overlay an image that you specify onto the video).
You can add these actions when the channel isn't running or when it is running. MediaLive sends the actions to the channel at the time identified in the schedule, and the channel performs the action.

For more information about schedules, see *Working with the Schedule* (p. 136)

## AWS Elemental MediaLive Terminology

### CDN

A content distribution network (CDN) is a network of servers that is downstream of the origin server or packager. The CDN distributes the content from the origin server to dozens or hundreds of networked servers that serve the content to your viewing users. This distributed network ensures that content can be delivered to thousands or millions of viewing users simultaneously.

### Channel

A MediaLive channel ingests and transcodes (decodes and encodes) source content from the inputs that are attached to that channel, and packages the new content into outputs.

### Channel class

Each channel belongs to one of the following classes:
- Standard class – a channel has two processing pipelines
- Single-pipeline class – a channel has one processing pipeline

### Channel configuration

A MediaLive channel configuration contains information about how the channel ingests, transcodes, and packages content into output.

### Downstream system

The *downstream system* is a set of one or more servers that is positioned after MediaLive in the workflow. The downstream system handles the content that is output from MediaLive.

### Encode

An encode exists within an output. There are three types of encodes: video, audio, and captions. Each encode contains the instructions for one video stream, one audio stream, or one captions track that the transcoding process will create. Different encodes have different characteristics. For example, one video encode produced from the input might be high resolution while another is low resolution.

### Input

A MediaLive input holds information that describes how the upstream system and the MediaLive channel are connected. The input identifies endpoints (IP addresses) in MediaLive (for a push input, where the upstream system pushes to MediaLive) or source IP addresses on the upstream system (for a pull input, where MediaLive pulls from the upstream system). MediaLive has different input types for different formats and protocols of the source content. For example, HLS input and RTMP Push input.

### Input security group

A MediaLive input security group is a set of one or more ranges of IP addresses that define an allow list. You associate one or more input security groups with a push input in order to identify a range of IP addresses that are allowed to push content to the input.

### Output

An output exists within an output group. It is a collection of encodes that you want to handle as one set.
Origin service

An origin service might be part of the downstream system that is positioned after MediaLive in the workflow. It accepts the video output from MediaLive.

Output Group

An output group is a collection of outputs within the MediaLive channel.

Packager

A packager might be part of the downstream system. It accepts the video output from MediaLive and repackages it. AWS Elemental MediaPackage is a packager.

Pipeline

In MediaLive, there are one or two separate and independent pipelines that perform the processing within the MediaLive input and the MediaLive channel.

Playback device

A playback device is the final component of the downstream system. It is the device that the people who are your audience use to view the video.

Schedule

Each MediaLive channel has an associated schedule. The schedule contains a list of actions to perform in the channel at a specific time.

Source content

The video content that MediaLive transcodes. The content typically consists of video, audio, captions, and metadata.

Upstream system

The system that is in front of MediaLive in the workflow and that holds the source content. Examples of an upstream system are a streaming camera or appliance that is directly connected to the internet, or a contribution encoder that is located in a stadium at a sports event.

Related Services

**Amazon CloudWatch** is a monitoring service for AWS Cloud resources and the applications that you run on AWS. Use CloudWatch to track MediaLive events about the progress of running channels and to view metrics about your resources.

**AWS Identity and Access Management (IAM)** is a web service that helps you securely control access to AWS resources for your users. Use IAM to control who can use your AWS resources (authentication) and what resources users can use in which ways (authorization).

**AWS Elemental MediaPackage** is a just-in-time video packaging and origination service that runs in the AWS Cloud. You can use AWS Elemental MediaPackage to package content that has been encoded by MediaLive.

**AWS Elemental MediaConnect** is a transport service for live video that runs in the AWS Cloud. You can use AWS Elemental MediaConnect as a source for video to transcode.

**AWS Elemental MediaStore** is a video origination and storage service that offers the high performance and immediate consistency required for live and on-demand media. You can use AWS Elemental MediaStore to store assets that MediaLive retrieves and uses when transcoding, and as a destination for output from MediaLive.

**AWS Resource Groups** includes a tagging editor that lets you assign metadata to AWS resources. You can use Tag Editor to assign metadata to MediaLive channels and other resources.
Amazon Simple Storage Service (Amazon S3) is storage for the internet. You can use Amazon S3 to store assets that MediaLive retrieves and uses when transcoding, and as a destination for output from MediaLive.

AWS Systems Manager lets you store passwords in MediaLive in a secure manner, rather than storing them as plaintext. If you connect to external servers that you provide user credentials for, it is likely that you will have to use Systems Manager.

Amazon Virtual Private Cloud lets you set up your own virtual network within the AWS Cloud. Use Amazon VPC as the location for an upstream system, so that the transfer of source content is within a private cloud.

Accessing AWS Elemental MediaLive

You can access AWS Elemental MediaLive using any of the following methods:

- **AWS Management Console** – The procedures throughout this guide explain how to use the AWS Management Console to perform tasks for AWS Elemental MediaLive.
- **AWS SDKs** – If you're using a programming language that AWS provides an SDK for, you can use an SDK to access AWS Elemental MediaLive. SDKs simplify authentication, integrate easily with your development environment, and provide easy access to AWS Elemental MediaLive commands. For more information, see Tools for Amazon Web Services.
- **AWS Elemental MediaLive API** – If you're using a programming language that an SDK isn't available for, see the AWS Elemental MediaLive API Reference for information about API actions and about how to make API requests.
- **AWS Command Line Interface** – For more information, see the AWS Command Line Interface User Guide.
- **AWS Tools for Windows PowerShell** – For more information, see the AWS Tools for Windows PowerShell User Guide.

Pricing

As with other AWS products, there are no contracts or minimum commitments for using AWS Elemental MediaLive.

There are two components to pricing: pricing based on the input of the channel that is being processed, and pricing based on the outputs of the channel:

- The input pricing is based on a combination of the input codec, the bitrate of the input, and the resolution of the input. You specify these three characteristics in the input specification when you create the channel. For more information, see the section called “Input Specifications Settings” (p. 88).
- The output pricing is based on a combination of the output codec, the output frame rate, and the output resolution. You specify these values in the codec, frame rate, width, and height fields in the video settings of each output in the channel. For more information, see the section called “Step 7: Set Up the Video Encode” (p. 116). Note that it is possible to set up the output frame rate to match the frame rate of the input. In this case, the frame rate portion of the pricing calculation uses the rate for “30-60 fps” frame rate; it doesn’t use the actual input frame rate.

There are different charges for inputs and outputs when the channel is running compared to when the channel is idle.
As soon as you start a channel, running charges start accruing for inputs and outputs. Running charges continue if you pause one or both pipelines in a channel. Running charges stop accruing only when you stop the channel.

For more information about pricing, see https://aws.amazon.com/medialive/pricing/.
Setting Up AWS Elemental MediaLive

This chapter provides procedures for setting up users to work with AWS Elemental MediaLive. It describes how to grant permissions that are appropriate for the period when you are experimenting with MediaLive, before you start using MediaLive in a production environment.

This chapter covers the following tasks:

- Setting up one or more administrators for the service
- Creating or modifying user identities that have permissions to access AWS Elemental MediaLive and ancillary services that MediaLive typically works with
- Setting up MediaLive as a trusted service

After you perform the procedures in this chapter, you and other users will have permissions that let you successfully follow the Getting Started Tutorial (p. 47).

Important
This chapter includes steps that grant broad permissions to AWS Elemental MediaLive and other services. These permissions are known as AWS Identity and Access Management (IAM) permissions. The permissions are intended to allow you and others in your organization to get started with MediaLive as quickly as possible. These permissions are not suitable for assigning to a wide group of users or for users working in a production environment.

To set up users for production use of AWS Elemental MediaLive, see Setting Up for Production (p. 16).

Topics
- Signing Up for AWS Elemental MediaLive (p. 8)
- Creating an Administrator IAM User (p. 9)
- Creating a Non-Administrator IAM User (p. 10)
- Setting up AWS Elemental MediaLive as a Trusted Service (p. 14)
- Data Retention and Personally Identifiable Information (p. 15)

Signing Up for AWS Elemental MediaLive

If you do not have an AWS account, complete the following steps to create one.

To sign up for an AWS account

2. Follow the online instructions.

Part of the sign-up procedure involves receiving a phone call and entering a verification code on the phone keypad.
Creating an Administrator IAM User

The procedures in this section show how to create an IAM user that has full read/write administrator permissions. This administrator might be you or another person. You set up an administrator by creating a group, and then creating a user that belongs to that group:

- If your organization is new to AWS, follow both steps in this procedure: create the group, and then create the users for that group.
- If your organization is not new to AWS, then the group probably has already been created. Follow only the second step to create users for that group.

To create a full-access administrator group

1. Use your AWS account email address and password to sign in to the AWS Management Console as the AWS account root user.
2. Open the IAM console at https://console.aws.amazon.com/iam/.
3. In the navigation pane, choose Groups, and then choose Create New Group.
4. On the Set Group Name page, for Group Name, enter a name such as Administrators. Choose Next Step.
5. On the Attach Policy page, choose Filter: Policy Type, and then choose Job function.
6. In the policy list, select the check box for AdministratorAccess, and then choose Next Step.
7. On the Review page, review the information, and then choose Create Group.

Now that you have an administrator group, you are ready to create an IAM user and add the user to your group.

To add an IAM user to the full-access administrator group

1. Open the IAM console at https://console.aws.amazon.com/iam/.
2. In the navigation pane, choose Users, and then choose Add user.
3. On the Add User page, for User name, enter a name such as Administrator or Admin_2 (if Administrator has already been created).
4. For Access type, select AWS Management Console access.
   For Console password, choose Custom password, and then enter a password.
   If this administrator is not you, we recommend that you select Require password reset.
5. Choose Next: Permissions.
6. On the Set permissions page, choose Add user to group.
7. Select the check box for the group that you created in the preceding procedure, and then choose Next: Review.
8. Review the information, and then choose Create user. To return to the navigation pane, choose Close.

After you create this IAM user with administrator permissions, sign out and sign in again using the administrator credentials.

We highly recommend that from this point forward you always sign in using the IAM administrator credentials instead of your root user credentials, unless AWS requires you to use your root user
credentials to perform certain operations. For more information, see AWS Tasks That Require AWS Account Root User Credentials.

Repeat the procedure to set up more administrators (as backups), if needed. Or anyone who is now set up as a full-access administrator can set up more administrators.

Creating a Non-Administrator IAM User

This section shows how to create non-administrator IAM users and grant those users the following permissions:

- Full read/write access to the following AWS services and features:
  - AWS Elemental MediaLive
  - AWS Elemental MediaConnect
  - AWS Elemental MediaPackage
  - Amazon CloudWatch
  - Amazon CloudWatch Events
  - Amazon CloudWatch Logs
  - Amazon EC2
  - Amazon EC2 Systems Manager
  - AWS Resource Groups
  - Amazon SNS
  - Amazon VPC

- Limited access to AWS IAM. Users of AWS Elemental MediaLive need some access to IAM in order to use the MediaLive console to set up MediaLive as a trusted entity. This setup is always required when using MediaLive. For more information, see the section called “Setting up as a Trusted Service” (p. 14).

Warning
These permissions are broad. You should set up only a few users with these permissions and only for the pre-production period of using MediaLive. For information about setting up users for standard production use, see Setting Up for Production (p. 16).

To set up an IAM user, you follow three main steps

1. Create customer managed policies.
2. Create a group and attach the policies to the group.
3. Create users and add the users to the group.

Policies grant permissions. Policies are attached to a group. Users belong to a group. Therefore, the users have the permissions of the policies that are attached to the group.

The following diagram shows this relationship.

Topics
- Step 1: Create Customer Managed Policies (p. 11)
- Step 2: Create an IAM Group (p. 12)
• Step 3: Create or Add an IAM User to Your Group (p. 13)

Step 1: Create Customer Managed Policies

The procedures in this section show how to create three IAM customer managed policies. A customer managed policy is one that you create and manage. (IAM also includes AWS managed policies, which you can't change.)

Anyone with IAM administrator-level credentials can perform the procedures.

The first procedure shows how to create a policy called **MediaLivePowerAccess** that gives full read/write access to AWS Elemental MediaLive.

The second procedure shows how to create a policy called **MediaConnectPowerAccess** that gives full read/write access to AWS Elemental MediaConnect.

The third procedure shows how to create a policy called **MediaLiveTrustedEntityAccess** that gives access to six operations in AWS IAM. These actions allow IAM users to create and update a trusted entity role for AWS Elemental MediaLive by setting the fields in the IAM role section on the Channel and input details page on the MediaLive console.

To create the MediaLivePowerAccess policy

1. Open the IAM console at https://console.aws.amazon.com/iam/.
2. In the navigation pane, choose Policies, and then choose Create policy. On the Visual editor tab, follow the prompts to create a policy with these options:
   - Service: MediaLive
   - Actions: All MediaLive actions (medialive.*)
   - Resources: This option is completed automatically
   - Request conditions: Omit this option
4. On the Create policy page, for Name, enter MediaLivePowerAccess.
5. For Description, optionally describe the purpose of this policy. This helps you identify the policy on the dashboard.
6. Choose Create policy.

To create the MediaConnectPowerAccess policy

1. Open the IAM console at https://console.aws.amazon.com/iam/.
2. In the navigation pane, choose Policies, and then choose Create policy. On the Visual editor tab, follow the prompts to create a policy with these options:
   - Service: MediaConnect
   - Actions: All MediaConnect actions (mediaconnect.*)
   - Resources: This option is completed automatically
   - Request conditions: Omit this option
4. On the Create policy page, for Name, enter MediaConnectPowerAccess.
5. For Description, optionally describe the purpose of this policy. This helps you identify the policy on the dashboard.
6. Choose Create policy.
To create the MediaLiveTrustedEntityAccess policy

1. Open the IAM console at https://console.aws.amazon.com/iam/.
2. In the navigation pane, choose Policies, and then choose Create policy. On the Visual editor tab, follow the prompts to create a policy with these options:
   - Service: IAM
   - Actions: In the filter box under Specify the actions allowed in IAM, search for and then select each of these actions:
     - ListRoles
     - GetRolePolicy
     - CreateRole
     - PassRole
     - AttachRolePolicy
     - PutRolePolicy
   - Resources: All resources
   - Request conditions: Omit this option
4. On the Create policy page, for Name, enter MediaLiveTrustedEntityAccess.
5. For Description, optionally describe the purpose of this policy. This helps you identify the policy on the dashboard.
6. Choose Create policy.

Step 2: Create an IAM Group

The procedure in this section shows how to create an IAM group and attach policies. Anyone with IAM administrator-level credentials can perform the procedure. Perform this procedure once, at initial setup. Before you start the procedure, you should have already created the two policies in Step 1: Create Customer Managed Policies (p. 11).

To create a group

1. Open the IAM console at https://console.aws.amazon.com/iam/.
2. In the navigation pane, choose Groups, and then choose Create New Group.
3. On the Set Group Name page, for Group Name, enter MediaLivePowerUsers, and then choose Next Step.
4. On the Attach Policy page, select the check boxes for the following policies:
   - MediaLivePowerAccess (customer managed policy)
   - MediaConnectPowerAccess (customer managed policy)
   - MediaLiveTrustedEntityAccess (customer managed policy)
   - CloudWatchReadOnlyAccess (AWS managed policy)
   - CloudWatchEventsFullAccess (AWS managed policy)
   - CloudWatchReadOnlyAccess (AWS managed policy)
   - AmazonEC2FullAccess (AWS managed policy for access to AWS Virtual Private Network)
   - AWSElementalMediaPackageFullAccess (AWS managed policy)
   - ResourceGroupsandTagEditorFullAccess (AWS managed policy)
   - AmazonSSMFullAccess (AWS managed policy for access to Amazon EC2 Systems Manager)
   - AmazonSNSFullAccess (AWS managed policy)
5. Choose Next Step, review your information, and then choose Create Group.

This diagram shows how the policies and group are associated.

![Diagram showing policies and group association]

**Step 3: Create or Add an IAM User to Your Group**

The procedure in this section shows how to create or edit an IAM user identity. Anyone with IAM administrator-level credentials can perform the procedure. Perform this step for each user.

**Note**

This procedure shows how to set up an IAM user for console access, but not for AWS CLI or AWS SDK access. To set up for programmatic access, see the IAM User Guide.

**Creating an IAM User and Adding the User to Your Group**

Typically, you create an IAM user identity for an AWS user only if a person doesn't have an existing identity. If the person already has an IAM user identity, you can modify their access (p. 14) instead.

**To create an IAM user and add the user to your group**

1. Sign in to the AWS Management Console as an administrator, and open the IAM console at https://console.aws.amazon.com/iam/.
2. In the navigation pane, choose Users, and then choose Add user.
3. On the Add User page, for User name, enter a name for the user.
   For Access type, select AWS Management Console access.
   For Console password, choose Custom password, and then enter a password.
   For Require password reset, we recommend that you select the check box.
4. Choose Next: Permissions.
5. On the Set permissions for user page, choose Add user to group.
6. Select the check box for the MediaLivePowerUsers group that you created in Step 2: Create a Group (p. 12), and then choose Next: Review.
7. Choose Create user.
8. Optionally choose Send email to send an email to this user. Your local email client opens with a draft email that includes the user name and sign-in URL.
9. Choose Close to return to the navigation pane.
10. Provide the user with their password (it is not included in the generated email). You must provide the password in a way that complies with your organization’s security guidelines.

Repeat the steps to add more IAM users. As an example, the following diagram shows three IAM users that are associated with the same group, MediaLivePowerUsers.
Adding an Existing IAM User to Your Group

You can add an existing IAM user to a group that you create for AWS Elemental MediaLive, even if the user is already a member of other groups. In this procedure, you add the user to the MediaLivePowerUsers group that you created in Step 2: Create a Group (p. 12).

For more information about IAM users and groups, see IAM User Guide.

To add an existing IAM user to your group

1. Sign in to the AWS Management Console as an administrator, and open the IAM console at https://console.aws.amazon.com/iam/.
2. In the navigation pane, choose Users.
3. In the list of users, choose the user name (don't choose the check box).
4. On the Summary page, choose the Groups tab. On the Groups tab, choose Add user to groups, and then select the MediaLivePowerUsers group that you created in Step 2: Create a Group (p. 12).
5. Choose Add to Groups.

You now have a setup where an IAM user belongs to more than one group: the original groups and the group that you added. If one of the groups has a policy that gives specific permissions to a given service or resource, and another group has a policy that gives different permissions, the policy with the least permission applies. One situation in which this rule might apply is if the existing user currently has permissions in IAM that are broader than those in the MediaLiveTrustedEntityAccess policy that you created.

Setting up AWS Elemental MediaLive as a Trusted Service

Every time a user creates a channel, they must attach an IAM role that sets up MediaLive as a trusted entity for that channel. You must give the user the permissions to set up this trusted entity.

You give this permission when you create the user. You create a policy called MediaLiveTrustedEntityAccess and attach it to the group that the users belong to. For detailed information, see the section called “Step 1: Create Customer Managed Policies” (p. 11).
Data Retention and Personally Identifiable Information

AWS Elemental MediaLive doesn't require that you supply any customer data. There are no fields in channels, inputs, or input security groups where there is an expectation that you will provide customer data.

Don't put sensitive identifying information such as customer account numbers into free-form fields such as a Name field. This includes when you work with MediaLive using the console, REST API, AWS CLI, or AWS SDKs. Any given piece of data that you enter into MediaLive might get picked up for inclusion in diagnostic logs.

When you provide a URL to an external server, don't include credentials information in the URL to validate your request to that server. MediaLive includes features such as the Amazon EC2 Systems Manager Parameter Store that provide you with a secure way to handle sensitive information. You should always use these features to pass a password; you should not circumvent them by including a password in a URL.

Deleting Data in AWS Elemental MediaLive

You can delete data from AWS Elemental MediaLive by deleting the object, for example, the channel or input. You can delete data using the console, REST API, AWS CLI, or AWS SDKs. The data will be deleted; no further steps are required after you delete data by completing a delete operation.

To delete data using the console, see the following sections:

- the section called “Deleting a Channel” (p. 135)
- the section called “Deleting an Input” (p. 85)
- the section called “Deleting an Input Security Group” (p. 68)
Setting Up AWS Elemental MediaLive Permissions for a Production Environment

This chapter provides procedures for setting up users and other AWS identities so that they can use AWS Elemental MediaLive in a production environment. It describes options for imposing restricted controls on users, so that you can set up permissions that conform with the security policies and procedures of your organization.

Before you follow these procedures, do the initial setup described in Setting Up. Those instructions show you how to grant broad permissions to users for non-production environments. Then return to this chapter to create limited permissions for a production environment.

Note
For part of the setup described in this chapter, you use the AWS Identity and Access Management (AWS IAM) service to create user and administrator identities. There might be features of IAM, such as cross-account access, that are not covered in this chapter but are appropriate and useful to your deployment. For information about all IAM features, see the AWS IAM User Guide.

In this chapter, we assume the following:

- You are now moving from experimenting with MediaLive to using MediaLive in a production environment.
- You have followed the procedures in Setting Up (p. 8) to sign up for MediaLive and to create a full-access administrator user.
- You have followed the procedures in the section called “Creating a Non-Administrator IAM User” (p. 10) and are therefore familiar with the process for creating IAM users and IAM groups using the IAM console.

This chapter also describes the AWS services that integrate with or depend on MediaLive. For some of these services, you must grant permissions so that users can access the services and use them with MediaLive. For other services, you don't need to grant permissions because the services are fully integrated with MediaLive. Following is a list of the AWS services that are covered in this chapter:

- AWS CloudTrail
- Amazon CloudWatch
- Amazon CloudWatch Events
- Amazon CloudWatch Logs
- Amazon Elastic Compute Cloud (Amazon EC2)
- AWS IAM
- AWS Elemental MediaConnect
- AWS Elemental MediaPackage
- AWS Elemental MediaStore
- AWS Resource Groups (Resource Group Tagging)
- Amazon Simple Notification Service (Amazon SNS)
- Amazon Simple Storage Service (Amazon S3)
• AWS Systems Manager

The following sections describe how to grant permissions to MediaLive and, if needed, the other AWS services.

Topics
• Setting Up Administrators and Users (p. 17)
• Creating an Administrator User with Limited Access (p. 17)
• Creating a Non-Administrator User (p. 20)
• Setting Up AWS Elemental MediaLive as a Trusted Service (p. 37)

Setting Up Administrators and Users

You must set up each person who will use AWS Elemental MediaLive as a IAM user. It is useful to split user identities into three general groups:

• Full-access administrator users. These users have full read/write access to all AWS services, users, and resources, including broad permissions in IAM.

You already created this user when you followed the procedure in the section called “Creating an Administrator IAM User” (p. 9).

• Administrators with limited access. Typically, these users have more permissions than a non-administrator user, but they don't have broad permissions in IAM.

See the section called “Creating an Administrator User with Limited Access” (p. 17).

• Non-administrator users or “regular users.” Typically, these users have broad permissions to MediaLive and to some of the services, such as AWS Elemental MediaConnect, that MediaLive interacts with. These users have very limited permissions in IAM.

See the section called “Creating a Non-Administrator User” (p. 20).

We recommend that you set up most users as non-administrator users. Set up only highly trusted users as administrator users.

Creating an Administrator User with Limited Access

If you are a full-access administrator, you can create other administrator users and assign each one a different level of access. These administrator users have more access than non-administrator users ("regular" users), but they have less access than full-access administrator users. They can use AWS Elemental MediaLive in the same way as regular users, but they can also create non-administrative users and set up some of the services that MediaLive integrates with.

For example, you might create an administrator user with the following access:

• For MediaLive and services that integrate with MediaLive, the administrator has the same access as regular users.
• For services that require some setup to work with MediaLive, the administrator has more access than regular users.
• For IAM, the administrator has more access than regular users, but less than full-access administrators.
The following procedure shows how to create an administrative user who has limited access. You start by creating a custom policy with a name such as MediaLiveAdminAccess, creating a group called MediaLiveAdministrators, and attaching the policy to the group. Next, you create the administrator user and add the user to the group. The procedure assumes that the new administrator user does not need permissions to troubleshoot issues with MediaLive other than access issues.

To create a custom policy for a MediaLive administrator

1. Sign in to the AWS Management Console as a full-access administrator, and open the IAM console at https://console.aws.amazon.com/iam/.

2. In the navigation pane, choose Policies, and then choose Create policy. On the Create policy page, choose the Visual editor tab. This tab is a policy generator that lets you build a policy by selecting actions from a list to add them to the policy.

3. Read the table at the end of this procedure, and create a policy that gives access to the actions that aren't already covered by an existing policy. You don't need to create a policy when we suggest using an existing policy. For information about the purpose of these actions, see the section called “Step 1: Requirements for Permissions” (p. 21).

4. To create the policy, follow the prompts on the console. Here are some tips for creating the policy:

   • You can create one policy that covers several services. You don't need to create a policy for each separate service. To create a policy for several services, choose the actions for one service, and then choose Add additional permissions at the bottom of the page to set up another service. You might need to move both of the vertical scroll bars to the bottom to display this link.

   • If you do choose to create one policy that covers several services, you might choose to create the policy with actions for one service, save it, then edit the policy to add permissions for another service, and so on.

   • You can choose the Import managed policy button to import an existing policy into this policy. The policy actions are copied over (the policy is not copied by reference), so after importing you can add and remove actions if you want.

For full instructions on creating a custom policy, see the IAM User Guide.

The following table shows which actions to include in the policy in order to grant the identified access to the user.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Corresponding Service in IAM</th>
<th>Type of Access</th>
<th>Actions to Include in the Policy</th>
</tr>
</thead>
<tbody>
<tr>
<td>MediaLive Features</td>
<td>MediaLive</td>
<td>Full access to MediaLive. It is a good idea for the administrator to be able to work with all MediaLive features.</td>
<td>Use the customer managed policy MediaLivePowerAccess. If you followed the procedures in Setting Up (p. 8), you created this policy in the the section called “Step 1: Create Customer Managed Policies” (p. 11) section.</td>
</tr>
<tr>
<td>Monitoring Channel Health</td>
<td>CloudWatch</td>
<td>Limited access to CloudWatch (the same access as non-administrator users).</td>
<td>See the section called “Summary of Step 1 Access Requirements” (p. 31).</td>
</tr>
<tr>
<td>Feature</td>
<td>Corresponding Service in IAM</td>
<td>Type of Access</td>
<td>Actions to Include in the Policy</td>
</tr>
<tr>
<td>----------------------------------------</td>
<td>-----------------------------</td>
<td>--------------------------------------------------------------------------------</td>
<td>------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Setting Up for Email Notification</td>
<td>CloudWatch Events</td>
<td>Full access to CloudWatch Events, to set up users for email notification. (To set up for email notification, users also need access to SNS. See later in this table.)</td>
<td>Use the managed policy <strong>CloudWatchEventsFullAccess</strong>. The administrator might not need all these actions, but giving full access is probably low risk.</td>
</tr>
<tr>
<td>Setting Up Channel Logging</td>
<td>CloudWatch Logs</td>
<td>Limited access to CloudWatch Logs (the same access as non-administrator users).</td>
<td>See the section called “Summary of Step 1 Access Requirements” (p. 31).</td>
</tr>
<tr>
<td>Creating a VPC Input</td>
<td>EC2</td>
<td>Limited access to Amazon EC2 (the same access as non-administrator users).</td>
<td>See the section called “Summary of Step 1 Access Requirements” (p. 31).</td>
</tr>
<tr>
<td>Setting Up User Identities for MediaLive</td>
<td>IAM</td>
<td>Limited access to manage users, groups, policies, and trusted entity roles.</td>
<td>The action ChangePassword And all actions that have any of these strings in their name: &quot;User&quot;, &quot;Group&quot;, &quot;Policy&quot;, &quot;Policies&quot;, &quot;Role&quot;, &quot;AccessKey&quot;, &quot;LoginProfile&quot;. Except don't include actions that also have the string &quot;Instance&quot;, or the string &quot;ContextKeys&quot;</td>
</tr>
<tr>
<td>Setting Up Email Notification</td>
<td>SNS</td>
<td>Full access to <strong>SNS</strong>, to set up email notification for users. (To set up for email notification, users also need access to CloudWatch Events. See earlier in this table.)</td>
<td>Use the managed policy <strong>AmazonSNSFullAccess</strong>. The administrator might not need all these actions, but giving full access is probably low risk.</td>
</tr>
</tbody>
</table>
Creating a Non-Administrator User

This section describes how to create non-administrator users ("regular users") by using IAM to create groups, attach policies to each group, and add the users to the group.

If you are new to AWS or if you have been using AWS for only a few weeks, we recommend that you read this entire section.

If you have more experience using other AWS services, read the section called “Step 1: Requirements for Permissions” (p. 21). Then, based on your decisions, modify or create policies and groups in the usual way.

Summary of Steps

To create IAM users with access to AWS Elemental MediaLive, you must perform several steps:

- Identify the permissions that users need for MediaLive and other services.
- Identify the different sets of users that you need. Each set will become an IAM group.

To create a group for your custom policy or policies

1. If necessary, sign in to the AWS Management Console as a full-access administrator, and open the IAM console at https://console.aws.amazon.com/iam/. (You might still be signed in.)
2. In the navigation pane, choose Groups. Follow the prompts to create a group with a name such as MediaLiveAdministrators.
3. Attach the policy or policies that apply to this administrator.

For full instructions on creating a group and attaching a policy, see the IAM User Guide.

To create an administrator user and add the user to your group

1. If necessary, sign in to the AWS Management Console as a full-access administrator, and open the IAM console at https://console.aws.amazon.com/iam/. (You might still be signed in.)
2. In the navigation pane, choose Users. Follow the prompts to create a user using the name of the person who will be the administrator.
3. In the step to set permissions for the user, choose Add user to group, and then select the group that you created.
4. Follow the prompts to finish creating the user.

For full instructions for creating an administrator, see the IAM User Guide.
• Identify the managed and custom policies that will provide the access required for the sets of users. Create the custom policies.
• Create the groups that you have identified, and attach the managed and custom policies.
• Create each user and add them to the appropriate group.

Topics
• Step 1: Identify Requirements for Permissions for Users (p. 21)
• Summary of Step 1 Access Requirements (p. 31)
• Step 2: Identify Categories of Users (p. 33)
• Step 3: Create the Custom Policies (p. 34)
• Step 4: Create the Groups (p. 34)
• Step 5: Create or Modify each IAM User (p. 35)
• Step 6: Setting Up Required Data (p. 37)

Step 1: Identify Requirements for Permissions for Users

You must identify the IAM permissions that you need to grant to users, for AWS Elemental MediaLive features and for ancillary services that MediaLive always interacts with.

To do that, you should understand the MediaLive workflows for your organization and the different AWS services that the workflows use.

You might not want all regular users to have the same permissions. For example, you might be able to group regular users into three sets: users who can start channels and watch channel activity, users who have some write capabilities, and advanced users who can do everything. As you identify these permissions, think about how many different sets of users you need.

Topics
• Requirements for AWS Elemental MediaLive Features (p. 21)
• Requirements for AWS CloudTrail (p. 22)
• Requirements for Amazon CloudWatch—Monitoring Channel Health (p. 23)
• Requirements for CloudWatch and Amazon SNS—Setting Up Email Notification (p. 23)
• Requirements for Amazon CloudWatch Logs—Setting Up Channel Logging (p. 23)
• Requirements for Amazon Elastic Compute Cloud—VPC Inputs (p. 24)
• Requirements for AWS Identity and Access Management—Trusted Entity Role (p. 24)
• Requirements for AWS Elemental MediaConnect (p. 28)
• Requirements for AWS Elemental MediaPackage (p. 28)
• Requirements for AWS Elemental MediaStore (p. 28)
• Requirements for AWS Resource Groups—Tagging (p. 28)
• Requirements for Amazon S3 (p. 29)
• Requirements for AWS Systems Manager—Creating Password Parameters in Parameter Store (p. 29)

Requirements for AWS Elemental MediaLive Features

You must give your users access to AWS Elemental MediaLive features. The permissions for MediaLive can be divided into three categories:
• Permissions to create – Permissions to create, modify, and delete channels, inputs, input security groups, or reservations
• Permissions to view – Permissions to view the details of channels, inputs, input security groups, and reservations
• Permissions to run – Permissions to start and stop channels

You might choose to give different access to different kinds of users. For example, you might decide that "basic operators" should not have create permissions.

In particular, you must decide whether to restrict the ability to work with reservations; you might decide to give this access only to administrators or advanced users. For more information about reservations, see Working with Reservations (p. 243).

The following table shows the operations in IAM that relate to access for MediaLive.

<table>
<thead>
<tr>
<th>Permissions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create, Modify, and Delete Channels, Inputs, and Input Security Groups</td>
</tr>
<tr>
<td>View Channels, Inputs, and Input Security Groups</td>
</tr>
<tr>
<td>Change the Class for a Channel</td>
</tr>
<tr>
<td>Run Channels</td>
</tr>
<tr>
<td>Attach Tags to Channels, Inputs, and Input Security Groups When Creating Those Resources</td>
</tr>
<tr>
<td>Create, Modify, Delete, and View Reservations and Offerings</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Service Name in IAM</th>
</tr>
</thead>
<tbody>
<tr>
<td>MediaLive</td>
</tr>
<tr>
<td>MediaLive</td>
</tr>
<tr>
<td>MediaLive</td>
</tr>
<tr>
<td>MediaLive</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>One or more write operations</td>
</tr>
<tr>
<td>One or more list operations</td>
</tr>
<tr>
<td>One or more read operations</td>
</tr>
<tr>
<td>UpdateChannelClass</td>
</tr>
<tr>
<td>StartChannel</td>
</tr>
<tr>
<td>StopChannel</td>
</tr>
<tr>
<td>CreateTag</td>
</tr>
<tr>
<td>DeleteTags</td>
</tr>
<tr>
<td>ListTagsForResources</td>
</tr>
<tr>
<td>DeleteReservation</td>
</tr>
<tr>
<td>DescribeOffering</td>
</tr>
<tr>
<td>DescribeReservation</td>
</tr>
<tr>
<td>ListOfferings</td>
</tr>
<tr>
<td>ListReservations</td>
</tr>
<tr>
<td>PurchaseOffering</td>
</tr>
</tbody>
</table>

**Requirements for AWS CloudTrail**

MediaLive is integrated with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service in MediaLive.

Users don't need special permissions for AWS CloudTrail.
Requirements for Amazon CloudWatch—Monitoring Channel Health

The AWS Elemental MediaLive console includes a page (Channel details) that collects CloudWatch metrics information about the health of channels and displays it directly on the MediaLive console.

You must decide if you want to give some or all of your users permission to view metrics on the console.

For a user to view this information on the MediaLive console, that user must have view permissions for metrics operations in Amazon CloudWatch. When users have these permissions, they can also view the information through the CloudWatch console, AWS CLI, or REST API.

The following table shows the actions in IAM that relate to access for monitoring channel health.

<table>
<thead>
<tr>
<th>Permissions</th>
<th>Service Name in IAM</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>View Metrics</td>
<td>CloudWatch</td>
<td>ListMetrics</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GetMetricData</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GetMetricStatistics</td>
</tr>
</tbody>
</table>

Requirements for CloudWatch and Amazon SNS—Setting Up Email Notification

MediaLive provides information about channels as they are running. It sends this information to Amazon CloudWatch as events. The details of these events can optionally be distributed to one or more users. Someone must set up this distribution. (For the setup procedure, see the section called “Monitoring a Channel Using Amazon CloudWatch Events” (p. 126).)

You must decide if you want to give some or all of your users these permissions. You might choose to allow each user to perform their own distribution setup. Or you might decide that an administrator must be responsible for performing the setup at startup for applicable users, and then again whenever a new user is added.

The following table shows the actions in IAM that relate to access for setting up email notification.

<table>
<thead>
<tr>
<th>Permissions</th>
<th>Service Name in IAM</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Write</td>
<td>CloudWatch Events</td>
<td>All actions</td>
</tr>
<tr>
<td>Write</td>
<td>SNS</td>
<td>All actions</td>
</tr>
</tbody>
</table>

Requirements for Amazon CloudWatch Logs—Setting Up Channel Logging

MediaLive produces channel logs that it sends to CloudWatch Logs, where users can view them. For more information about channel logs, see the section called “Monitoring Using Amazon CloudWatch Logs” (p. 128).

You must decide if you want to give some or all of your users permission to view the logs in CloudWatch Logs.
Step 1: Requirements for Permissions

You must also decide if you want to give some or all of your users permission to set the retention policy for logs. If you decide not to give this access to any user, an administrator must be responsible for setting the policy.

Users don't need special permission to enable logging from within MediaLive.

The following table shows the actions in IAM that relate to access for setting up channel logs.

<table>
<thead>
<tr>
<th>Permissions</th>
<th>Service Name in IAM</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>View Logs</td>
<td>CloudWatch Logs</td>
<td>FilterLogEvents</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GetLogEvents</td>
</tr>
<tr>
<td>Set Retention Policy</td>
<td>CloudWatch Logs</td>
<td>DeleteRetentionPolicy</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PutRetentionPolicy</td>
</tr>
</tbody>
</table>

**Requirements for Amazon Elastic Compute Cloud—VPC Inputs**

Your deployment might include push inputs that connect to MediaLive from a VPC that you created with Amazon VPC.

When a user creates this type of input on the MediaLive console, they have the option to choose the subnet and security group from a dropdown list. For the dropdown list to be populated with the resources in Amazon VPC, the user must have the appropriate permissions. For more information about Amazon VPC inputs, see the section called “Creating an Input” (p. 70).

The following table shows the actions in IAM that relate to access for populating the dropdown.

<table>
<thead>
<tr>
<th>Permissions</th>
<th>Service Name in IAM</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>View the VPC subnets and VPC security groups on the MediaLive console</td>
<td>EC2</td>
<td>DescribeSubnets</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DescribeSecurityGroups</td>
</tr>
</tbody>
</table>

**Requirements for AWS Identity and Access Management—Trusted Entity Role**

This requirements analysis must be performed by a person in your organization who understands your organization’s requirements for access to resources. This person must understand whether there is a requirement that AWS Elemental MediaLive channels should be restricted in their access to resources in other AWS services. For example, this person should determine whether channels should be restricted in their access to containers in AWS Elemental MediaStore so that a specified channel can access some containers and not others.

Every time a user creates a channel, they must attach an IAM role that sets up MediaLive as a trusted entity for that channel. The user makes this attachment using the IAM role pane on the Create channel page on the MediaLive console.

You must decide what access you need to give to users for working in this IAM role pane.

If you followed the procedures in the section called “Creating a Non-Administrator IAM User” (p. 10) to set up users for the period when you are experimenting with MediaLive, then you already set up this trusted entity role. You set it up by creating the MediaLiveAccessRole role. However, you should still
read this section to determine if `MediaLiveAccessRole` is suitable for your organization when you are working in a production environment.

**Topics**
- About the Trusted Entity Role (p. 25)
- Options for Implementing the Role (p. 25)
- Requirements for Permissions for the Simple Option (p. 26)

### About the Trusted Entity Role

AWS Elemental MediaLive must be set up so that when a channel is running, MediaLive itself has access to perform operations on resources that belong to your organization’s AWS account. For example, your deployment might use AWS Elemental MediaStore as a source for files, such as blackout images, that MediaLive requires during processing. For MediaLive to obtain these files, it must have read access to some or all containers in MediaStore.

To perform the required operations on those resources, MediaLive must be set up as a trusted entity on your account. MediaLive is set up as a trusted entity as follows: A role (that belongs to your AWS account) identifies MediaLive as a trusted entity. The role is attached to one or more policies. Each policy contains statements about allowed operations and resources. The chain between the trusted entity, role, and policies makes this statement:

"MediaLive is allowed to assume this role in order to perform the operations on the resources that are specified in the policies."

![Role for the service diagram](image)

After this role is created, the role must be attached to a specified channel. This attachment makes this statement:

"For this channel, MediaLive is allowed to assume this role in order to perform the operations on the resources specified in the policies."

Creating this attachment at the channel level allows each channel to give MediaLive access to different operations and, especially, different resources.

### Options for Implementing the Role

There are two options for setting up the trusted entity role in AWS Elemental MediaLive: a simple option and a complex option.

**Simple Option**

The simple option typically applies when users in your organization are using AWS Elemental MediaLive to encode the organization’s own assets (not assets belonging to customers), and you don’t have rigorous
rules about accessing assets (for example, you don't have video assets that can be handled only by specific users or departments).

With the simple option, there is only one role: MediaLiveAccessRole. All channels use this role and all users can attach that role to the channels that they work with.

The simple option works only on the MediaLive console. It can't be performed using the AWS CLI, for example.

The MediaLiveAccessRole role grants broad access to operations and complete access to all resources. It allows either read-only access or read/write access to all the services that MediaLive must access when a channel is running. And most significantly, it allows full access to all the resources associated with those services.

If the simple option is suitable to your deployment, see the section called "Requirements for Permissions for the Simple Option" (p. 26).

Complex Option

The complex option applies when the MediaLiveAccessRole role is too broad for your use, given that it allows broad access to operations and complete access to all resources.

For example, you might have the following requirements:

- A requirement that a given channel should be allowed to access only specific resources and another channel should be allowed to access only specific, different resources. Therefore, you need to create several access roles, each of which narrows down permissions to a different set of resources.
- A requirement that each user should be allowed to display only specific roles on the console, to prevent a user from viewing a role they should not know about or to prevent a user from selecting the wrong role.

If the complex option is applicable to your deployment, see the section called “Setting Up AWS Elemental MediaLive as a Trusted Service” (p. 37).

Requirements for Permissions for the Simple Option

Read this section if you decide that the simple option (p. 25) for the trusted entity is appropriate to your deployment.

(To set up for the complex option, see the section called “Setting Up AWS Elemental MediaLive as a Trusted Service” (p. 37).)

For users to work in the IAM Role section on the Channel and input details pane, they must have access to specific IAM actions.

The following screenshot shows the IAM Role section on the Channel and input details pane as it appears when you start to create a channel.
You must set up users as follows:

- Users must be able to choose `MediaLiveAccessRole` from the selection field that accompanies the **Use existing role** field.
- Users must be able to choose the **Create role from template** field. (The role needs to be created only once, by the first user to create a channel. But it is easiest to give all users these permissions.)
- Users do not need to be able to use the **Specify custom role ARN** field. They will use `MediaLiveAccessRole`. They will never use a custom role.
- Users must be able to choose the **Update** button, in order to update the `MediaLiveAccessRole` from time to time.

The following table shows the service and action in IAM that you must grant to regular users with the simple option.
Permissions | Service Name in IAM | Actions
--- | --- | ---
Choose the Create role from template option | IAM | CreateRole, PutRolePolicy, AttachRolePolicy
Choose MediaLiveAccessRole from the list in Use existing role | IAM | ListRole, PassRole
Choose Update | IAM | GetRolePolicy, PutRolePolicy, AttachRolePolicy

Requirements for AWS Elemental MediaConnect

Your deployment might include using a flow from AWS Elemental MediaConnect as an input to AWS Elemental MediaLive.

Users don't need special permissions to specify an AWS Elemental MediaConnect flow as input.

Requirements for AWS Elemental MediaPackage

Your deployment might send outputs to AWS Elemental MediaPackage, either by creating an HLS output group or by creating a MediaPackage output group (p. 64). (Note that both MediaLive and MediaPackage have "channels"; however, they are different objects.)

Regardless of the type of output group, users don't need special permissions to specify a MediaPackage channel as the destination for output.

Requirements for AWS Elemental MediaStore

Your deployment might include using files in an AWS Elemental MediaStore container. For example, your deployment might use files in the following ways:

- As the source for an HLS input
- As the destination for an HLS output group

Users don't need special permissions to specify a MediaStore container as the destination for output.

Requirements for AWS Resource Groups—Tagging

When users create channels, inputs, or input security groups, they can optionally attach tags to the resource during creation. Typically, your organization has a policy to tag or to omit tags. There are two services that control permissions for tagging, for two different scenarios:

- The ability to tag during channel creation is controlled by actions within AWS Elemental MediaLive. See the section called "MediaLive" (p. 21).
- The ability to modify tags in existing resources is controlled by actions within Resource Group Tagging. See Working with Tag Editor in Getting Started with the AWS Management Console.
Requirements for Amazon S3

Your deployment might include using files in an Amazon S3 bucket. For example, your deployment might use files in the following ways:

- As the source for an HLS input
- As the destination for an Archive output group
- As the destination for an HLS output group

Users don’t need special permissions to specify an Amazon S3 bucket in a field on the MediaLive console.

Requirements for AWS Systems Manager—Creating Password Parameters in Parameter Store

The AWS Elemental MediaLive console includes a feature that lets a user create a password parameter in the AWS Systems Manager Parameter Store. This feature is part of the Create Channel page. This feature does not exist in the AWS CLI or REST API.

You must decide if you want to give some or all of your users permission to use this feature. (If you don’t give this access to any users, then an administrator must be responsible for creating parameters.)

About the Feature for Creating Password Parameters

The AWS Systems Manager Parameter Store is used extensively in AWS Elemental MediaLive. It is likely that you will use this store. The store holds passwords that MediaLive needs to retrieve and store files externally.

Here are some of the MediaLive functions that use this store to hold passwords:

- An input of type RTMP Pull or type HLS Pull, if the connection is secure.
- Fields in the channel that hold the URL to an external file, if the connection is secure. An example of this type of field is Avail blanking image.
- The destination in an HLS output group or a Microsoft Smooth output group, if the connection is secure.

In all these cases, MediaLive needs the user name and the password. The password is always stored in a parameter. Therefore, the console includes a Username field and a Password parameter field. For an example of the relevant fields, open the MediaLive console, choose Create channel, General settings, Avail blanking, Avail blanking image, and then choose Credentials.

How Password Parameters Work

The password parameter feature ensures that when the user is creating a channel, AWS Elemental MediaLive does not store passwords in plaintext. It works as follows:

- First, a user or administrator creates a password parameter in AWS Systems Manager Parameter Store. The parameter is a name-value pair where the name is something like corporateStorageImagesPassword and the value is the actual password.
- Second, when a user is creating a channel or input in MediaLive and needs to enter a password, the user specifies the password parameter name instead of the password. That name is stored in MediaLive. The actual password is never stored in MediaLive.
- Finally, when the channel is running and MediaLive needs the password (to either read or write to the external location), it sends the password parameter name to Parameter Store and gets back the actual password in response.
Create Feature That Is Built into AWS Elemental MediaLive

When a password field appears on the console, AWS Elemental MediaLive includes a feature that lets the user do one of the following:

- Enter the name of an existing password parameter.
- Create a password parameter by entering the name-value pair (a parameter name and an actual password).

Required Permissions

Users must enter the name of a password parameter or select a name from the dropdown list. Some users might need permission to create a password parameter within AWS Elemental MediaLive.

Permission to Enter a Name

No special permission is required to enter the name of an existing password parameter on the AWS Elemental MediaLive console.

Permission to Select a Name

For the user to select a name from the dropdown list, the user must have permission for `GetParameters` in AWS Systems Manager.

Permission to Create

For any user to create a password parameter on the AWS Elemental MediaLive console, that user must have permission to specific operations in AWS Systems Manager Parameter Store. (With this permission, the user can also create these password parameters ahead of time on the AWS Systems Manager console. The user can choose the option that they prefer.)

You can give access to some or all users to create these password parameters. Typically, you give this access only to users who are trusted with sensitive passwords; these might be users whom you have identified as advanced users:

- If you give access only to advanced users, those users must be responsible for creating parameters at startup for the applicable assets and whenever a new asset is required by MediaLive. The users can perform the setup on the MediaLive console or on the AWS Systems Manager console.
- If you don't give this access to any users, an administrator must be responsible for creating parameters at startup for the applicable assets and whenever a new asset is required by MediaLive. An administrator might prefer to perform this setup on the AWS Systems Manager console.

Permission to Modify and Delete

If you want users to be able to modify and delete password parameters (as well as create them), give access to modify and delete operations. The users will be able to modify and delete from the AWS Systems Manager Parameter Store. (There is no feature on the AWS Elemental MediaLive console for modifying and deleting.)

You might choose to give this access to the users who have create permissions. Or you might choose to give this access only to administrators.

The following table shows the actions in IAM that relate to access for the Parameter Store.

<table>
<thead>
<tr>
<th>Permissions</th>
<th>Service Name in IAM</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Select</td>
<td>Systems Manager</td>
<td>GetParameters</td>
</tr>
</tbody>
</table>
### Permissions

<table>
<thead>
<tr>
<th>Permissions</th>
<th>Service Name in IAM</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create</td>
<td>Systems Manager</td>
<td>PutParameter</td>
</tr>
<tr>
<td>Modify and Delete</td>
<td>Systems Manager</td>
<td>DeleteParameter, DeleteParameters, DescribeParameters, GetParameter, GetParameterHistory, GetParameters, GetParametersByPath</td>
</tr>
</tbody>
</table>

### Summary of Step 1 Access Requirements

The following table shows all the types of permissions that you might need to assign to users.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Corresponding Service in IAM</th>
<th>Tasks</th>
<th>Actions to Include in the Policy</th>
</tr>
</thead>
<tbody>
<tr>
<td>MediaLive Features</td>
<td>MediaLive</td>
<td>Create, modify, and delete channels, inputs, and input security groups</td>
<td>One or more List operations, One or more Read operations, One or more Write operations</td>
</tr>
<tr>
<td>MediaLive</td>
<td>View channels, inputs, and input security groups</td>
<td>One or more List operations, One or more Read operations</td>
<td></td>
</tr>
<tr>
<td>MediaLive</td>
<td>Run channels</td>
<td>StartChannel, StopChannel</td>
<td></td>
</tr>
<tr>
<td>MediaLive</td>
<td>Create, modify, delete, and view offerings and reservations</td>
<td>DeleteReservation, DescribeOffering, DescribeReservation, ListOfferings, ListReservations, PurchaseOffering</td>
<td></td>
</tr>
<tr>
<td>MediaLive</td>
<td>Attach tags when creating a resource</td>
<td>CreateTags, DeleteTags</td>
<td></td>
</tr>
</tbody>
</table>
### Summary of Step 1 Access Requirements

<table>
<thead>
<tr>
<th>Feature</th>
<th>Corresponding Service in IAM</th>
<th>Tasks</th>
<th>Actions to Include in the Policy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Monitoring Channel Health</td>
<td>CloudWatch</td>
<td></td>
<td>ListTagsForResources, ListMetrics, GetMetricData, GetMetricStatistics</td>
</tr>
<tr>
<td>Setting Up Events</td>
<td>CloudWatch Events</td>
<td>All actions</td>
<td>The managed policy CloudWatchEventsFullAccess provides these permissions</td>
</tr>
<tr>
<td>Setting Up Channel Logging</td>
<td>Amazon CloudWatch Logs</td>
<td>View logs</td>
<td>FilterLogEvents, GetLogEvents</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Set retention policy, DeleteRetentionPolicy, PutRetentionPolicy</td>
</tr>
<tr>
<td>Creating a VPC input</td>
<td>EC2</td>
<td>View the VPC subnets and VPC security groups on the MediaLive console</td>
<td>DescribeSubnets, DescribeSecurityGroups</td>
</tr>
<tr>
<td>Simple Option for the Trusted Entity Role</td>
<td>IAM</td>
<td>Create the MediaLiveAccessRole</td>
<td>CreateRole, PutRolePolicy, AttachRolePolicy</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Choose the MediaLiveAccessRole, ListRole, PassRole</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Update the MediaLiveAccessRole, GetRolePolicy, PutRolePolicy, AttachRolePolicy</td>
</tr>
<tr>
<td>Setting Up Email Notification</td>
<td>Amazon SNS</td>
<td>All actions</td>
<td>The managed policy AmazonSNSFullAccess provides these permissions</td>
</tr>
</tbody>
</table>
Step 2: Identify Categories of Users

After you have identified the permissions that your users need, you must identify the different categories of users that you need. You identify different categories based on the different operations they should be allowed to perform.

This requirements analysis must be performed by a person in your organization who understands the AWS Elemental MediaLive workflows for your organization and the different AWS services that the workflows use.

To identify categories of users

1. Refer to the sections in the section called “Step 1: Requirements for Permissions” (p. 21), and decide whether all of your users should have the same permissions on all the services or whether some users should have one set of permissions while other users have another set.
2. Group these different categories into, giving each category a name.
3. When looking at operations, keep in mind that you could decide that no regular user should have certain permissions—only an administrator should have those permissions.

For example, perhaps you identify three categories of users:

- Basic users – These users can start and stop channels and view metrics for channels, but have no write permissions.
- Read/write users – These users have nearly full permissions, but they can’t create password parameters in AWS Systems Manager Parameter Store.
- Advanced users – These users have full permissions on the services identified in the section called “Step 1: Requirements for Permissions” (p. 21). They are nearly as powerful as a restricted administrator, except that they can’t set up users.
Step 3: Create the Custom Policies

Any person who is an administrator can perform this procedure. Follow this procedure once, when setting up users for production.

After you identify the different collections of operations that your sets of users require, you must determine which collections have a corresponding managed policy or custom policy that already exists, and which require a new custom policy.

To identify and create custom policies

1. Look at each set of users that you identified, and look at the collections of operations for those users. For each service, determine which collections have a corresponding managed policy and which require a custom policy.

   For example, for CloudWatch Events, there is a managed policy called CloudWatchEventsFullAccess that corresponds to "events:*". But there is no policy that contains only the operations required to create a password parameter. You must create a custom policy for that.

2. In IAM, create custom policies as applicable, using the IAM policy generator. This generator lets you choose the service from a list, and then choose operations from a list. As a best practice, give the policy a name that starts with the service name, medialive.

   To create the policy, follow the prompts on the console. Here are some tips for creating the policy:

   • You can create one policy that covers several services. You don't need to create a policy for each separate service. To create a policy for several services, choose the actions for one service, and then choose Add additional permissions at the bottom of the page to set up another service. (You might need to move both of the vertical scroll bars to the bottom to find Add additional permissions.)

   • You can choose Import managed policy to import an existing policy into this policy. The policy actions are copied over (the policy is not copied by reference), so after importing you can add and remove actions if you want.

   For detailed instructions for creating a policy, see IAM User Guide.

The following example assumes that you created two custom policies.

Step 4: Create the Groups

Any person who is an administrator can perform this procedure. Follow the procedure once, when setting up users for production.

After you identify the different sets of users that your deployment requires, you must create a group for each set.

To create the groups

1. In IAM, choose Groups, and then use the Create New Group Wizard to create a group for each set of users that you identified in the section called “Step 2: Identify Categories of Users” (p. 33). See
http://docs.aws.amazon.com/IAM/latest/UserGuide/id_groups_create.html and follow the steps for creating groups using the console.

As a best practice, assign group names that start with the service name, medialive.

2. The Create New Group Wizard includes a step for attaching policies to the group as you create it. Make sure to attach the managed and custom policies that you have identified.

The following example assumes that you created a group called medialivebasicusers and associated the two custom policies plus one managed policy.

Step 5: Create or Modify each IAM User

Any person who is an administrator can perform this step. Follow this procedure when setting up users for production. After the initial setup, perform this procedure whenever you need to set up a new user.

After you create the groups and attach the policies to each group, you must create the users and attach each to the appropriate group.

This procedure describes how to set up the user for console access, but not for AWS CLI or AWS SDK access. To set up for programmatic access, see IAM User Guide.

Topics
- Create a User (p. 35)
- Modify an Existing User (p. 36)

Create a User

Typically, you create a new user identity for an AWS user only if a person does not have an existing identity. If the person already has a user identity, modify their access instead.

To create a user

1. Make sure that you know which group (the section called “Step 4: Create the Groups” (p. 34)) that you want to add each user to. Make sure that you have already created this group (the section called “Step 4: Create the Groups” (p. 34)).
2. Sign in to the AWS Management Console as an administrator, and open the IAM console at https://console.aws.amazon.com/iam/.
3. In the navigation pane, choose Users, and then choose Add user.
4. On the Add User page, for User name, enter a name for the user.
   - For Access type, select AWS Management Console access.
   - For Console password, select Custom password and enter a password.
   - For Require password reset, we recommend that you select the check box.
5. Choose Next: Permissions.
6. On the Set permissions for user page, choose Add user to group.
7. Select the check box for the appropriate group for this user, and then choose Next: Tags.
8. Add tags if your organization has a policy to create tags for users. For more information, see Tagging Resources (p. 227). Then choose Next: Review.

9. Choose Create user.

10. Optionally, choose Send email to send an email to this user. Your local email client opens with a draft email that includes the user name and sign-in URL.

11. Choose Close to return to the navigation pane.

12. Provide the user with their password (it is not included in the generated email). You must provide the password in a way that complies with your organization's security guidelines.

The following example assumes that you created three users and associated all of them with the same group, medialivebasicusers.

 Modify an Existing User

If a user identity already exists for someone who will use AWS Elemental MediaLive, you can set them up for use in a production environment by modifying their user identity to make them a member of the relevant group (in addition to the group or groups where they are already members.)

You need to know the group that each user needs to belong to; see the section called “Step 4: Create the Groups” (p. 34).

One situation in which it is very useful to modify an existing non-administrator user is if you followed the procedures in Setting Up (p. 8) to set up users for the period when you are experimenting with MediaLive. Following that procedure, you created a policy called MediaLiveAccessUser and a group called MediaLivePowerUsers. You can now take away the broad permissions that you gave those users and "move" the users from the MediaLivePowerUsers group to one of the groups that you created in the section called “Step 4: Create the Groups” (p. 34). There is no need to delete these users and create them again.

To modify a user

1. Sign in to the AWS Management Console as an administrator, and open the IAM console at https://console.aws.amazon.com/iam/.

2. In the navigation pane, choose Users.

3. In the list of users, choose the user name (don't select the check box).

4. On the Summary page, choose the Groups tab:

   - To add this user to another group, on the Groups tab, choose Add user to groups and follow the prompts to choose the group.
   - To remove this user from a group, choose the X icon beside the group name and follow the prompts.

   You might want to remove the user from the MediaLivePowerUsers group but keep them in groups that give access to other AWS services.
You might now have the setup where a user belongs to more than one group, the original groups and the groups that you added. If one of the groups has a policy that gives specific permissions to a given service or resource, and another group has a policy that gives different permissions, the policy with the least permission applies.

### Step 6: Setting Up Required Data

After you set up users with the appropriate access, you or another administrator should provide users with the information that they need to use MediaLive:

- Provide each user with a list of the MediaLive operations that they have access to. To prevent user frustration, make sure that users know which console pages they can't display. Make sure to include information about the channel metrics that users can't display on the Channel details page.
- If some external servers require user credentials, and only one or two users or administrators are responsible for creating password parameters in the AWS Systems Manager Parameter Store, make sure that those users are aware of their responsibility.

Also make sure that those users provide other users with the password parameters that those other users need.

- If a user or administrator is responsible for setting up other users for email notification, let that user know. Or if each user is responsible for setting up their own email notification, let each user know. Users can read the section called "Monitoring a Channel Using Amazon CloudWatch Events" (p. 126) for instructions.
- For the MediaLive role, if you chose the simple option (p. 25), make sure users know that the only role they will ever choose is the MediaLiveAccessRole.

If you set up for the complex option (p. 26), let the user or administrator who must create roles know that they must give other users a list of the roles (the list of role ARNs) that each user can use.

### Setting Up AWS Elemental MediaLive as a Trusted Service

You need to read this section only if you determined in the section called “Options for Implementing the Role” (p. 25) that the simple option for setting up the trusted entity role does not work for your deployment.

This section describes how to implement the complex option. It provides the following information:

- Background information about how the trusted entity role is created and used in the complex option.
- Instructions for identifying the trusted entity role or roles that your deployment needs and creating these roles.
- Instructions for granting limited permissions to regular users so that they can use only specific trusted entity roles.

If you are not familiar with the purpose of the trusted entity role, first read the section called “About the Trusted Entity Role” (p. 25) and the section called “Options for Implementing the Role” (p. 25).

**Topics**
- How the Trusted Entity Is Created and Attached (p. 38)
How the Trusted Entity Is Created and Attached

This section applies if you have determined that your deployment should implement the complex option for the trusted entity role, as described in the section called “Options for Implementing the Role” (p. 25).

With complex options, the process for creating trusted entity roles and attaching a specific role when creating a channel typically works as follows:

- Process for creating a role – An administrator creates the roles using IAM. They don't use the IAM role pane on the Create channel page on the AWS Elemental MediaLive console. They create these roles as part of the initial deployment.
- Process for attaching a role – After the required roles are created, the administrator gives each regular user a list of the roles and the channels that each role applies to.

Each user might have a different list of roles; they will have only the roles that apply to the channels that they work with.

When a user who has permission to create a channel is working on the Create channel page, they will display the Channel and input details pane. In the IAM Role section, the user will choose Specify custom role ARN and enter the role name in the field by typing or pasting.

An administrator therefore must perform the following setup:

- Set up all the trusted entity roles that your deployment requires. See the section called “Creating Trusted Entity Roles” (p. 38).
- Set up regular users with restricted permissions for working with roles. You must also make sure that you have not granted certain permissions; granting those permissions would give the regular users permissions that are too broad for the complex option. See the section called “Setting Up Permissions for Non-Administrator Users” (p. 44).

Creating Trusted Entity Roles

This section applies if you have determined that your deployment requires the complex option for the trusted entity role, as described in the section called “Options for Implementing the Role” (p. 25).

This section describes how to create a role, policy, and trust relationship, as described and illustrated in the section called “About the Trusted Entity Role” (p. 25).

Topics

- Step 1: Determine the Access Requirements (p. 38)
- Step 2: Create Policies (p. 42)
- Step 3: Create Roles (p. 43)
- Step 4: Revise the Trust Relationship (p. 43)

Step 1: Determine the Access Requirements

This requirements analysis must be performed by a person in your organization who understands your organization’s requirements for access to resources. This person must understand whether there is a
requirement that MediaLive channels should be restricted in their access to resources in other AWS services. For example, this person should determine whether channels should be restricted in their access to containers in MediaStore so that a specified channel can access some containers and not others.

You must identify the services that MediaLive will interact with in your deployment. Then within each service, you must identify the operations and resources that MediaLive needs access to.

**To determine the access requirements for MediaLive**

1. See the table at the bottom of this section for information about the services that MediaLive typically needs access to. Determine which of those services your deployment uses and which operations it needs.

2. Within a service, determine the number of policies that you need to create. Do you need several different combinations of objects and operations for different workflows, and do you need to keep those combinations separate from each for security reasons?

Specifically, determine whether you need access to different resources for different workflows, and whether it’s important to restrict access to specific resources. For example, in AWS Systems Manager Parameter Store you might have passwords that belong to different workflows, and you might want to allow only specific users to access the passwords for any given workflow.

If different workflows have different requirements for objects, operations, and resources, then for that service you need separate policies for each workflow.

After you perform this analysis, you might determine that you need three different policies for MediaStore, four different policies for Amazon S3, and three policies for AWS Systems Manager Parameter Store.

3. Design each policy: identify the allowed (or not allowed) objects, operations, and the allowed (or not allowed) resources in the policy.

4. Determine if any of the policies that you have identified are covered by a managed policy.

5. For each workflow, identify the policies that you need for all the services that the workflow uses.

   For example, for one workflow, you might need policy X for MediaStore, policy A for Amazon S3, and policy 1 for AWS Systems Manager Parameter Store. For the second workflow, you might need policy Y for MediaStore, policy B for Amazon S3, and policy 1 for AWS Systems Manager Parameter Store. For the third workflow, you need the same policies as for the first workflow.

6. Identify the number of roles that you need. You need one role for each unique combination of policies. Following our example, you need two roles: one role for the first and third workflows, and another for the second workflow.

7. Assign names to all the policies and roles that you have identified. Take care not to include sensitive identifying information (such as a customer account name) in these names.

**Summary Of Requirements for the MediaLive Trusted Entity**

The following table lists the services that are often used in MediaLive deployment. The third column specifies whether MediaLive itself needs access to these services. If yes, then the fourth column suggests an existing policy that provides the required access, or else lists the operations and resources that you would typically include in a custom policy.

<table>
<thead>
<tr>
<th>Service</th>
<th>Tasks</th>
<th>Type of Access Required</th>
<th>Suggested Actions or Policy</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWS Elemental MediaLive</td>
<td>Working with MediaLive</td>
<td>MediaLive doesn't need</td>
<td>MediaLive doesn't need access to itself. Only the users need access.</td>
</tr>
<tr>
<td></td>
<td>features.</td>
<td>access to itself.</td>
<td></td>
</tr>
<tr>
<td>Service</td>
<td>Tasks</td>
<td>Type of Access Required</td>
<td>Suggested Actions or Policy</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>----------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------</td>
<td>------------------------------------------------------------------</td>
</tr>
<tr>
<td>AWS CloudTrail</td>
<td>Capturing MediaLive activity.</td>
<td>MediaLive doesn't need IAM access for this task.</td>
<td></td>
</tr>
<tr>
<td>CloudWatch</td>
<td>Displaying CloudWatch metrics information on the console, to monitor channel health.</td>
<td>MediaLive doesn't need IAM access for this task. Only the users need access.</td>
<td></td>
</tr>
<tr>
<td>CloudWatch Events and Amazon SNS</td>
<td>Setting up email notification so that users can be notified about MediaLive alerts that are sent to CloudWatch Events.</td>
<td>MediaLive doesn't need access for this task. Only the users need access.</td>
<td></td>
</tr>
<tr>
<td>CloudWatch Logs</td>
<td>Sending channel log information to CloudWatch Logs when a channel is running.</td>
<td>When the channel is running, MediaLive must be able to send log messages to CloudWatch Logs.</td>
<td>CreateLogGroup, CreateLogStream, PutLogEvents, PutMetricFilter, PutRetentionPolicy, DescribeLogStreams, DescribeLogGroups, And these resources: arn:aws:logs:* arn:aws:log-group:*</td>
</tr>
<tr>
<td>Amazon EC2</td>
<td>Creating an RTP VPC input or RTMP VPC push input.</td>
<td>When the user is creating a VPC input, MediaLive must have write access to Amazon EC2 in order to create network interfaces for the input.</td>
<td>CreateNetworkInterface, CreateNetworkInterfacePermission, DescribeNetworkInterfaces, DescribeSecurityGroup, DescribeSubnets</td>
</tr>
<tr>
<td></td>
<td>Deleting an RTP VPC input or RTMP VPC push input.</td>
<td>When the user deletes a VPC input, MediaLive must have write access to Amazon Elastic Compute Cloud in order to delete the network interfaces for the input.</td>
<td>DeleteNetworkInterface, DeleteNetworkInterfacePermission, DescribeNetworkInterfaces, DescribeSubnets</td>
</tr>
<tr>
<td>Service</td>
<td>Tasks</td>
<td>Type of Access Required</td>
<td>Suggested Actions or Policy</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>------------------------------------------------</td>
<td>------------------------------------------------------------------------------------------</td>
<td>----------------------------</td>
</tr>
<tr>
<td>AWS Elemental MediaConnect</td>
<td>Creating an AWS Elemental MediaConnect input.</td>
<td>When the user creates an MediaConnect input, MediaLive must have read/write access to</td>
<td>ManagedDescribeFlow</td>
</tr>
<tr>
<td></td>
<td></td>
<td>the AWS Elemental MediaConnect flow, in order to add an output to that flow.</td>
<td>ManagedAddOutput</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Deleting an AWS Elemental MediaConnect input.</td>
<td>When the user creates an AWS Elemental MediaConnect input, MediaLive should have read/write access to the AWS Elemental MediaConnect flow, in order to delete the outputs on the flow, because the outputs are no longer needed.</td>
<td>ManagedDescribeFlow</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>ManagedDeleteOutput</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AWS Elemental MediaPackage</td>
<td>Sending channel output to MediaPackage when a channel is running, if your deployment uses this service.</td>
<td>When the user creates a MediaPackage output group, MediaLive must have read access to the AWS Elemental MediaPackage channel, in order to obtain the credentials required to send to that channel.</td>
<td>DescribeChannel</td>
</tr>
<tr>
<td>AWS Elemental MediaStore</td>
<td>Sending and retrieving assets from a MediaStore container when a channel is running, if your deployment uses this service.</td>
<td>When the channel is running, MediaLive must have read access (for a source) or read/write access (for a destination).</td>
<td>ListContainers</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>DescribeObject</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>PutObject</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>GetObject</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>DeleteObject</td>
</tr>
<tr>
<td>Resource Group Tagging</td>
<td>Attaching tags when creating resources—channels, inputs, and input security groups —and revising tags on existing resources.</td>
<td>MediaLive doesn't need IAM access for this task. Only the users need access.</td>
<td></td>
</tr>
</tbody>
</table>

41
### Service Type of Access Required

<table>
<thead>
<tr>
<th>Service</th>
<th>Tasks</th>
<th>Suggested Actions or Policy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amazon S3</td>
<td>Sending and retrieving assets from an Amazon S3 bucket when a channel is running, if your deployment uses this service.</td>
<td>When the channel is running, MediaLive must have read access (for a source) or read/write access (for a destination) to the buckets.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ListBucket, PutObject, GetObject, DeleteObject</td>
</tr>
<tr>
<td>AWS Systems Manager</td>
<td>Creating a password parameter on the MediaLive console.</td>
<td>MediaLive doesn't need IAM access for this task. Only the users need access.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The managed policy AmazonSSMReadOnlyAccess</td>
</tr>
</tbody>
</table>

### Step 2: Create Policies

Any person who is an administrator can create a policy.

In the section called “Step 1: Determine Requirements” (p. 38), someone in your organization identified the policy or policies that you need to create.

Create those policies now in IAM.

**To create a custom policy for the MediaLive trusted entity role**

1. If necessary, sign in to the AWS Management Console as a full-access administrator, and open the IAM console at https://console.aws.amazon.com/iam/.
2. In the navigation pane, choose Policies, and then choose Create policy. On the Create policy page, choose the Visual editor tab. This tab is a policy generator that lets you build a policy by selecting actions from a list to add them to the policy.

   To create the policy, follow the prompts on the console. Here are some tips for creating the policy:

   - You can create one policy that covers several services. You don't need to create a policy for each separate service. To create a policy for several services, choose the actions for one service, and then choose Add additional permissions at the bottom of the page to set up another service. (You might need to move both of the vertical scroll bars to the bottom to find Add additional permissions.)
   - You can choose the Import managed policy button to import an existing policy into this policy. The policy actions are copied over (the policy is not copied by reference), so after importing you can add and remove actions if you want.

   For full instructions on creating a custom policy, see the IAM User Guide.

To create the policy, follow the prompts on the console. Here are some tips for creating the policy:
Creating Trusted Entity Roles

- You can create one policy that covers several services. There is no need to create a policy for each separate service. To create a policy for several services, choose the actions for one service, and then choose the **Add additional permissions** button at the bottom of the screen to set up another service. You may need to move both the vertical scroll bars to the bottom to reveal this button.
- You can choose the **Import managed policy** button to import an existing policy into this policy. The policy actions are copied over (the policy is *not* copied by reference), so after importing you can add and remove actions if you want.

For full instructions on creating a custom policy, see the **IAM User Guide**.

**Step 3: Create Roles**

Any person who is an administrator can perform the procedure to create a role and attach policies to the role.

In the section called “Step 1: Determine Requirements” (p. 38), someone in your organization identified the roles that you need to create. Create those roles now using IAM.

**To create a role and attach a policy to it**

1. Sign into the AWS Management Console as an administrator, and open the IAM console at [https://console.aws.amazon.com/iam/](https://console.aws.amazon.com/iam/).
2. In the navigation pane, choose **Roles**.
3. On the **Role** page, choose **Create role**.
4. On the **Create role** page, in the **Select type of trusted entity** section, choose **AWS service** (the default).
5. In **Choose the service that will use this role**, choose **EC2**.
   You choose EC2 because MediaLive is not currently included in this list. Choosing EC2 lets you create a role; in a later step, you will change this role to mention MediaLive instead of EC2.
6. Choose **Next: Permissions**.
7. In the **Attach permissions policies** section, select all the policies that apply for this role, and then choose **Next: Tags**.
8. Add tags if your organization has a policy to create tags for resources. For more information, see **Tagging Resources** (p. 227). Then choose **Next: Review**.
9. Choose **Next: Review**.
10. For **Role name**, enter a name. We recommend that you don't use the name MediaLiveAccessRole because it is reserved for the simple option (p. 25). Instead, use a name that includes medialive and describes this role's purpose.
11. For **Trusted entities**, Amazon EC2 (ec2.amazonaws.com) is displayed as the trusted entity, but you will modify that line in the next procedure.
12. Choose **Create role**.

**Step 4: Revise the Trust Relationship**

Any person who is an administrator can perform this procedure.

When you created the role and established the trusted relationship, you chose EC2 as the service. You must now modify the role so that the trusted relationship is between your AWS account and MediaLive.

**To change the trust relationship to MediaLive**

1. On the **Summary** page for the role (which should still be displayed), choose **Trust relationships**.
2. Choose **Edit trust relationship**.


   The policy document should now look like this:

   ```json
   {
     "Version": "2012-10-17",
     "Statement": [
       {
         "Effect": "Allow",
         "Principal": {
           "Service": "medialive.amazonaws.com"
         },
         "Action": "sts:AssumeRole"
       }
     ]
   }
   ```

4. Choose **Update Trust Policy**.

5. On the **Summary** page, make a note of the value in **Role ARN**. It looks like this:

   `arn:aws:iam::111122223333:role/MediaLiveAccessRole`

   In the example, `111122223333` is your AWS account number.

6. Make a list of all the role ARNs, and include a description of the workflow and users for each. You will need this list in the section called “Step 6: Setting Up Required Data” (p. 37).

### Setting Up Permissions for Non-Administrator Users

This section applies if you have determined that your deployment requires the complex option for the trusted entity role, as described in the section called “Options for Implementing the Role” (p. 25).

This section describes how you set up permission for regular users to work with the trusted entity roles that you created in the section called “Creating Trusted Entity Roles” (p. 38).

**Topics**

- Identifying Permissions (p. 44)
- Setting Up Permissions (p. 46)

### Identifying Permissions

With the complex option, regular users don't create trusted entity roles. But they will attach existing roles to the channels that they create by completing the **IAM Role** section in the **Channel and input details** pane.

To work with this section, regular users therefore must have access to specific IAM actions.

The following screenshot shows the **IAM Role** section on the **Channel and input details** pane as it appears when you start to create a channel.
You must set up permissions for regular users so that they can access only specific fields on the IAM Role section in the Channel and input details pane. Typically, you must set up as follows:

- Users must *not* be able to choose the selection field that accompanies the **Use existing role** field. You probably want to disable this selection field because you don’t want users to choose a role from the list that accompanies this field. If this field is enabled for a user, that user can view all the roles that are created in the account, which would defeat the requirement to restrict access so that users can view and attach only specific roles.

- Users must *not* be able to choose the **Create role from template** field. Regular users do not create roles.

- Users must be able to enter values into the entry field that accompanies the **Specify custom role ARN** field. When this entry field is enabled, the user can enter or paste one of the role names that you provide.

- Users do *not* need to be able to choose the **Update** button because this button only ever appears in implementations that use the MediaLiveAccessRole. The complex option does not use this role; therefore, this button never appears.

To ensure that users interact with this section of the console in this restricted way, you must grant access to only one IAM action, as shown in the following table.
### Setting Up Permissions

This section applies if you have determined that your deployment requires the complex option for the trusted entity role, as described in the section called “Options for Implementing the Role” (p. 25).

You must create a policy for the IAM service to set up these regular users with the permissions that they need for the complex option. You must also attach that policy to the group that those regular users belong to.

#### To set up permissions

1. Follow the steps in the section called “Step 3: Create the Custom Policies” (p. 34) with these differences:
   - Create a policy with a name such as `MediaLiveTrustedEntityRegularUserAccess`.
   - Include only the actions that you identified in the section called “Identifying Permissions” (p. 44).

2. In the group that you created or will create for regular users (see the section called “Step 4: Create the Groups” (p. 34)), include this policy.

3. Identify any other policies that relate to the IAM service, and detach them from this group.

---

**Permissions**

<table>
<thead>
<tr>
<th>Permissions</th>
<th>Service Name in IAM</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attach</td>
<td>IAM</td>
<td>iam:PassRole</td>
</tr>
</tbody>
</table>

Equally important, you must make sure that you do not grant access to the following actions:

- `iam:ListRole`
- `iam:CreateRole`
- `iam:PutRolePolicy`
- `iam:AttachRolePolicy`
Getting Started with AWS Elemental MediaLive

This tutorial describes how to ingest a video source from an RTP source and generate one HLS output that contains one H.264 video encode and one audio encode. MediaLive will send the output to AWS Elemental MediaPackage. The output will consist of the following:

- One master manifest: channel.m3u8
- One rendition manifest: channel_1.m3u8
- TS files for each output: channel_1.00001.ts, channel_1.00002.ts, channel_1.00003.ts, and so on

This tutorial uses the default values for most configuration fields in the channel.

Note
All the text marked as an example in this tutorial is just that—a sample that shows what a piece of information typically looks like. You must replace each example with the information that is valid for your situation.

Prerequisites

Before you can use MediaLive, you need an AWS account and the appropriate permissions to access, create, and view MediaLive components. Complete the steps in Setting Up (p. 8), and then return to this tutorial. You can't use MediaLive, even as an administrator with full access, until you perform those steps.

Step 1: Set Up the Upstream System

The upstream system is the system that streams the video to MediaLive. The upstream system can be anything from an on-premises appliance that is serving as a "contribution encoder" to an application running on a smart phone. You must perform some setup of your upstream system before you start working with MediaLive.

For the purposes of this tutorial, the upstream system must be capable of sending a video stream via RTP push.

In a "push" delivery, the upstream system is pushing the stream from two IP addresses on the upstream system (for example, from 203.0.113.111 and from 203.0.113.112). The upstream system will push to two IP addresses on MediaLive (for example, rtp://198.51.100.10:5000 and rtp://192.0.2.131:5000). In the following steps, you will set up MediaLive so that the two from IP addresses are white listed. Furthermore, MediaLive will generate the two to IP addresses. You will set up the upstream system to push to those addresses.

To set up the upstream system

1. Set up your upstream system to perform an RTP push from two different IP addresses. You must push from two addresses because MediaLive always expects redundant inputs.
2. Make a note of the IP addresses. For example, 203.0.113.111 and from 203.0.113.112. You will need these addresses when you set up the input security group in a later step.

Step 2: Set Up the Downstream System

In this tutorial, the downstream system (the destination for the output from MediaLive) is AWS Elemental MediaPackage.

You must set up a channel in AWS Elemental MediaPackage, and you must set it up now because you need the two input URLs that AWS Elemental MediaPackage generates. You enter these input URLs into MediaLive.

To set up the downstream system

1. Sign in to the AWS Management Console and open the MediaPackage console at https://console.aws.amazon.com/mediapackage/.
2. In a new web browser tab or window, display the Getting Started for AWS Elemental MediaPackage and follow steps 1 to 3 to create one channel and its endpoint.
3. Make a note of the data that AWS Elemental MediaPackage has generated: two input URLs and their associated names and passwords. For example, the data for one input URL might be:
   - https://39fu04.mediapackage.us-east-1.amazonaws.com/in/v1/88dpee/channel
   - ue739wut
   - due484u

   Your channel might be in a different Region from the example.
4. Keep the web browser open; don't close it yet.

Step 3: Create an Input

You must create an input. The input defines how the upstream system provides the source video stream to MediaLive. In this tutorial, you create an RTP input.

You must also create an input security group for the input. This input security group applies the rule "only this specific IP address (an IP address that you own) can push to this input on MediaLive." Without the protection of this rule, any third party could push content to a MediaLive input if they know the IP address and port of the input.

To create an input and input security group

1. Sign in to the AWS Management Console and open the MediaLive console at https://console.aws.amazon.com/medialive/.
2. In the navigation pane, choose Inputs.
3. On the Inputs page, choose Create input.
4. In the Input details section, for Input name, enter My RTP push.
5. For Input type, choose RTP.
6. In the Input security group section, choose Create.
7. In the text box, enter the IP address that you noted in the section called "Step 1: Set Up the Upstream System" (p. 47) of this tutorial. Enter the address as a CIDR block. For example, 203.0.113.111/32 and 203.0.113.112/32.
8. Choose Create input security group.
Step 4: Set up Key Information

The first step to creating a channel from scratch is to choose the IAM role that MediaLive will use to access the channel when the channel is running (started) and specify key characteristics of the input. Now you are ready to start creating a channel. The first step is to identify the input. The channel contains the details that instruct MediaLive how to transcode (decode and encode) and package that input into specific outputs.

The first step to creating a channel from scratch is to choose the IAM role that MediaLive will use to access the channel when the channel is running (started) and specify key characteristics of the input.

To specify key information for the channel

1. On the MediaLive console, in the navigation pane, choose Channels.
2. In the Channels section, choose Create channel.
3. In the Channel and input details pane, in General info, for Channel name, enter Test channel.
4. For IAM role, choose Create role from template and choose Create IAM role. The Use existing role list now shows the role MediaLiveAccessRole.
5. Choose Remember role.

Step 5: Attach the Input

Now you are ready to identify the input that the channel will ingest.

To attach the input to the channel

1. On the Create channel page, in the navigation pane, for Input attachments, choose Add.
2. In Attach input, for Input, My RTP push (the input that you created.)

The Attachment name field is automatically populated with the name of the input itself. You can leave this name as is.

Step 6: Set up Input Video, Audio, Captions

You can create "selectors" to identify the specific video, audio, and captions that you want to extract from the input.

In this tutorial, you don't create a video selector. Instead, when the channel starts, MediaLive will automatically select the video (or the first video) in the input. You also don't create a captions selector. Typically, you include captions in the channel configuration, but in this tutorial we omit them.
Step 7: Create an HLS Output Group

Once you have set up the input, you continue with the channel creation by creating an output group. In this tutorial, you set up an HLS output group.

To create an output group

1. On the Create channel page, in the Output groups section, choose Add.
2. In the Add output group section, choose HLS, and then choose Confirm.
3. In the HLS group destination A section, for URL, enter the first input URL that AWS Elemental MediaPackage created for you in the section called “Step 2: Set Up the Downstream System” (p. 48). For example, https://39fu04.mediapackage.us-east-1.amazonaws.com/in/v1/88dpie/channel.
4. For Credentials:
   - For Username, enter the user name that corresponds to this URL. For example, ue739wuty.
   - For Password, choose Create parameter. For Name, enter DestinationA_MyHLS. For Password, enter the password that corresponds to the URL. For example, due484u.
5. Choose Create parameter.

You have created a parameter called DestinationA_MyHLS that holds the password due484u. The parameter is stored in the AWS Systems Manager Parameter Store. For more information, see the section called “About the Feature for Creating Password Parameters” (p. 29).

6. For HLS group destination B, for URL, enter the second input URL that AWS Elemental MediaPackage created for you in the section called “Step 2: Set Up the Downstream System” (p. 48). For example, https://mgu654.mediapackage.us-east-1.amazonaws.com/in/v1/xmm9s/channel.
7. For Credentials:
   - For Username, enter the user name that corresponds to this URL. For example, 883hdux.
   - For Password, choose Create parameter. For Name, enter DestinationB_MyHLS. For Password, enter the password that corresponds to the URL. For example, 634hjik.
8. Choose Create parameter.

You have created a parameter called DestinationB_MyHLS that holds the password 634hjik. The parameter is stored in the AWS Systems Manager Parameter Store.

9. In the HLS settings section, for Name, enter MyHLS.
10. For CDN settings, choose Hls webdav. This is the connection that AWS Elemental MediaPackage (the downstream system for the channel output) uses.

Leave the defaults for all the other CDN settings fields.
Step 8: Set Up the Output and Encodes

Now that you have defined one output group in the channel, you can set up an output in that output group, and specify how you want to encode the video output and the audio output.

To set up the output

1. In the **Output groups** section, choose **Output 1**. MediaLive automatically added this output when you created the output group. In addition, MediaLive automatically set up the output with one video and one audio, as shown in the **Stream settings** section.
2. In **Stream settings**, choose **Video**.
3. For **Video description name**, change the default name to **H264 video**.
4. For **Codec settings**, choose **H264**.
   
   Leave the remaining fields with the default values. Specifically, keep **Width** and **Height** empty to use the same width as the input.
5. In **Stream settings**, choose **Audio 1**.
6. For **Audio description name**, change the default name to **AAC audio**.
7. For **Audio selector name**, enter **My audio source**, which is the audio selector that you created in the section called “Step 6: Set up Input Video, Audio, Captions” (p. 49).
8. For **Codec settings**, choose **AAC**.
9. Leave the remaining fields with the default values.

Step 9: Create Your Channel

You have entered the minimum required information, so you are ready to create the channel.

To create the channel

- On the **Create channel** page, under the **Channel** section, choose **Create channel**.

  The **Channel** section reappears and shows the newly created channel, named **MyHLS**. The state changes to **Creating**, then **Ready**.

Step 10: Start the Upstream System and the Channel

You can now start the upstream system in order to push the streaming content to MediaLive, encode the content, and send it to AWS Elemental MediaPackage. You can preview the output on MediaPackage.

To start the upstream system

1. In your upstream system, start streaming the video sources that you set up in the section called “Step 1: Set Up the Upstream System” (p. 47). Set them up to push to the two destinations that you noted in the section called “Step 3: Create an Input” (p. 48). These are two addresses in the input in MediaLive. For example, `rtp://198.51.100.10:5000` and `rtp://192.0.2.131:5000`.
2. On the **Channels** list, choose the channel.
3. Choose **Start**. The channel state changes to **Starting**, then to **Running**.
4. Switch to the web browser tab or window where the AWS Elemental MediaPackage is displayed.
5. Choose the channel link (not the radio button). On the details page, under **Endpoints**, choose **Play**.
   A preview window appears.
6. Start the video. The output from AWS Elemental MediaLive starts playing.

**Step 11: Clean Up**

To avoid extraneous charges, delete this channel and input when you have finished working with it.

**To delete the channel**

1. On the **Channels** page, choose the channel.
2. If needed, choose **Stop**.
3. Choose **Delete**.
4. On the **Inputs** page, choose the input.
5. Choose **Delete**.
Components of AWS Elemental MediaLive

The key building blocks of AWS Elemental MediaLive are inputs, channels, and input security groups. A channel in turn consists of output groups, which contain outputs, which contain video, audio, and captions "encodes."

When a channel is started (run), AWS Elemental MediaLive ingests the input. It then transcodes that video (and the related audio, captions, and metadata) and creates output assets. The information about how to transcode a given input is contained in a channel.

An input security group (p. 67) is a mechanism to prevent unauthorized third parties from pushing content into a channel that is associated with a "push" input.

Inputs

An input is a video asset that is to be transcoded and packaged. It may be associated with an input security group, which provides protection to the input, and with a channel, which provides details about the transcoding and packaging to perform.

AWS Elemental MediaLive supports different types of stream and file inputs (for example, RTP and HLS). The service also provides two ways to ingest the inputs, either through a push model or a pull model. For more information, see Reference: Supported Containers and Codecs (p. 236).

Channels

In MediaLive, a channel is attached to one or more inputs (video sources). If the channel is attached to more than one input, the inputs are processed one after the other. A channel contains the details that instruct MediaLive how to transcode (decode and encode) and package the inputs into specific outputs. The key components of a channel are an encode, an output, and an output group.

Encodes

An encode is the smallest component on the output side of a channel. Each encode contains the instructions for one video asset, one audio asset, or one captions asset that will be created by the transcoding process. Different encodes have different characteristics. For example, one video encode produced from the input might be high resolution while another is low resolution. Or one audio encode might use the AAC audio codec while another uses the Dolby Digital audio codec.

A channel can contain multiple video, audio, and captions encodes.

In the following illustration, the red circle represents a video output, the blue circle represents an audio output, and the green circle represents a captions output.
Outputs

An output contains the encodes that belong together. For example, one output will contain the combination of video, audio, and captions encodes that make sense for one purpose, while another output will contain a different combination.

The output holds packaging instructions that apply to all the encodes in that output. For example, the packaging instructions for a UDP output are different from those for an Archive output. The encodes inside the outputs might be the same or different. But the packaging instructions are different.

Output Groups

An output group contains related outputs. An output group might contain only one output or it might contain several outputs. The output group holds details about the destination for all the outputs in that group.

Input Security Groups

An input security group is a group that you create and associate with specific input types, to prevent unauthorized third parties from pushing content into a channel. For more information, see Working with Input Security Groups (p. 67).

How Components Are Associated

The association between inputs and a channel is defined in the channel. In other words, to associate a channel with one or more inputs, you set up the channel to point to those inputs.

After you create this association, you can do the following:

- View the channel details to identify the associated input.
- View the input details to identify the associated channel.

The association between an input and an input security group is defined in the input. In other words, to associate an input with an input security group (or with several groups), you set up the input to point to a specific input security group (or groups).

After you create this association, you can do the following:

- View the channel details to identify the associated input.
- View the input details to identify the associated channel.
Planning Your Workflow for AWS Elemental MediaLive

To use AWS Elemental MediaLive to transcode a video asset, you follow this basic workflow:

1. Assess the video source to ensure that MediaLive can accept it.
2. Decide on the channel class—decide if you want to create a standard channel that supports redundancy or a single-pipeline channel that doesn't support redundancy.
3. Set up the upstream system that will provide the video source to MediaLive.
4. Create an input for your video asset.
5. Optionally associate the input with an input security group (required only for certain types of inputs).
6. Create a channel in which you identify the input to transcode and specify how MediaLive should ingest and encode that input.
7. Start (run) your channel. MediaLive ingests the input, encodes the input (and any associated audio, captions, and metadata), and then creates the output.
8. Send the output to a downstream system—for example, send the output to an origin service or a packager such as AWS Elemental MediaPackage.

Topics
- Assessing the Video Source (p. 55)
- Determining the Channel Class (p. 56)
- Setting Up the Upstream System (p. 57)
- Planning the Channel (p. 60)
- Examples of Channel Designs (p. 61)
- Setting Up the Downstream System (p. 63)
- Next Steps (p. 66)

Assessing the Video Source

The upstream system is the system that streams the video to AWS Elemental MediaLive. Examples of an upstream system are a streaming camera or appliance that is directly connected to the internet, or a contribution encoder that is located in a stadium at a sports event. The upstream system can be on the public internet or in a VPC that you created in Amazon Virtual Private Cloud (Amazon VPC).

Assess the video source to ensure that it delivers what you expect:

- Make sure that the upstream system is capable of streaming to your MediaLive input using one of the supported protocols. See the section called “Supported Input Types and Upstream Systems” (p. 236).
- Determine if your chosen protocol uses a “push” or a “pull.” With a push protocol, the upstream system pushes to MediaLive. With a pull protocol, MediaLive pulls from the upstream system. See the section called “Supported Input Types and Upstream Systems” (p. 236).
Determining the Channel Class

AWS Elemental MediaLive supports two classes of channel—a standard channel and a single-pipeline channel.

You set up each AWS Elemental MediaLive channel as one of these classes:

- A standard channel has two processing pipelines, for redundancy
- A single-pipeline channel has one pipeline.

You must decide on the channel class now because the choice affects the setup you must perform on the upstream system and on the downstream system.

To decide on the channel class

1. Contact the owner of the upstream system and the owner of the downstream system and find out what kind of redundancy they support. The possibilities are the following.

   - Both systems support only redundancy. The upstream system always sends two source streams. The downstream system always expects delivery at two destinations. In this case, you must set up the channel as a standard channel.
   - Both systems support only a single pipeline. The upstream system can provide only one source stream. The downstream system can handle delivery at only one destination. In this case, you must set up the channel as a single-pipeline channel.
   - Both systems can support either one pipeline or two pipelines. In this case, continue reading.

2. If you can choose between a standard and single-pipeline channel, then you should weigh the benefit of a standard channel (support for high availability) against the difference in processing charges for a standard channel compared to a single-pipeline channel. For information about charges for channels, see https://aws.amazon.com/medialive/pricing/.

   We strongly recommend that you set up channels as standard channels.

   - With two pipelines in the channel, if one of the pipelines fails, the other pipeline continues encoding and delivering output. (The failed pipeline automatically restarts within a few minutes.) The downstream system can be set up to detect failure of an output at one of its destinations and switch to using the output from the other destination.
   - With only one pipeline in the channel, there is no second pipeline to switch to if the first pipeline fails. No output is delivered to the downstream system until the failed pipeline restarts itself.

3. If you are sending output to AWS Elemental MediaPackage, you might want to implement input redundancy, which MediaPackage supports by default. In this case, you should set up channels as standard channels. In this way, MediaLive will send two identical outputs to the two inputs on the MediaPackage channel. You will achieve redundancy in both MediaLive and MediaPackage.

   If you don't want to implement input redundancy in MediaPackage, you might choose to set up a single-pipeline channel. There is no way to implement input redundancy in MediaPackage using a single-pipeline MediaLive channel.
Setting Up the Upstream System

You must set up your upstream system before you start working in AWS Elemental MediaLive.

**Important**
Typically, you set up the MediaLive channel as a standard channel (p. 56). The procedures in this section assume that you have decided to set up in this way. The procedures therefore describe how to set up the upstream system to provide two sources. If you have decided to set up a single-pipeline channel, then follow these procedures, but set up the upstream system to provide only one source.

**Topics**
- MediaConnect Push (p. 57)
- MP4 Pull (p. 57)
- RTP Push (p. 58)
- RTMP Push (p. 58)
- RTMP Pull (p. 59)
- HLS Pull (p. 59)
- Planning the Inputs (p. 60)

**MediaConnect Push**

MediaLive can accept a flow from AWS Elemental MediaConnect as an input:

- The AWS Elemental MediaConnect flow and the MediaLive input must be in the same AWS Region. If possible, set up AWS Elemental MediaConnect and MediaLive in the same Region. If that is not possible, then set up a distribution in AWS Elemental MediaConnect to move the source to the same Region as the MediaLive input.
- In AWS Elemental MediaConnect, make sure that the streams in the two flows are identical in terms of resolution and bitrate.
- Keep in mind that AWS Elemental MediaConnect must be pushing its two flows to the MediaLive input before you start the channel. All push inputs are live inputs, and a live input must be already pushing even if it is not the first input in the channel.

For detailed information about setting up AWS Elemental MediaConnect and MediaLive, see the section called “Creating a MediaConnect Push Input” (p. 72).

**MP4 Pull**

- MediaLive works with redundant sources, so you provide two video streams. For optimized redundancy, MediaLive runs each source on different encoder pipelines in different Availability Zones. You don’t have to set up these Availability Zones because MediaLive does it for you.
- Make sure that the two files are identical in terms of resolution and bitrate.
- Keep in mind that the video source must be ready to be pulled before the channel starts to ingest the input.
  - If this input is the only input or the first input in the channel, it must be ready before you start the channel.
  - If this input is not the first input, it must be ready approximately 30 seconds before the channel switches to this input.
RTP Push

- Make sure that the source is a streaming source, not a file source. A push input works only with a streaming source.
- We recommend that you enable FEC in the source. A source that includes FEC is less likely to result in an output that has visual disruptions.
- Make sure that the upstream system is set up to send over RTP, not UDP. The UDP protocol is not supported as an input into MediaLive.
- MediaLive works with redundant sources, so the upstream system provide two video streams. For optimized redundancy, MediaLive runs each source on different encoder pipelines in different Availability Zones. You don't have to set up these Availability Zones because MediaLive does it for you.
- Make sure that the two streams are identical in terms of resolution and bitrate.
- Keep in mind that the upstream system must be pushing the video source to the input before you start the channel. All push inputs are live inputs, and a live input must be already pushing even if it is not the first input in the channel.
- This information applies if the upstream system is on the public internet. From the upstream system, obtain the IP addresses that the two streams will push from and make a note of them. You will need this information to set up the required input security groups for the MediaLive inputs that you will create.
- This information applies if the upstream system is in a VPC that you created in Amazon VPC, and you plan to follow the typical plan of creating a VPC input in MediaLive. From the upstream system, obtain the IP addresses of the two streams that the upstream system will push, and make a note of them. You will need this information when you identify the VPC security groups for the MediaLive inputs that you will create.

RTMP Push

- Make sure that the source is a streaming source, not a file source. A push input works only with a streaming source.
- MediaLive works with redundant sources, so you provide two video streams. For optimized redundancy, MediaLive runs each source on different encoder pipelines in different Availability Zones. You don't have to set up these Availability Zones because MediaLive does it for you.
- Make sure that the two streams are identical in terms of resolution and bitrate.
- Determine the application name and application instance for this video source.

The upstream system might have already assigned these names. These names might be provided to you separately (for example, the application name is livestream and the application instance is curling) or as a path (livestream/curling). Make a note of these names.

If the upstream system has not assigned names, you could request names that work for you. We recommend that you use live as the application name and a name of your choosing as the application instance. Make sure that you and the operator of the upstream system agree on these names.

- Keep in mind that the upstream system must be pushing the video source to the input before you start the channel. All push inputs are live inputs, and a live input must be already pushing even if it is not the first input in the channel.
- From the upstream system, obtain the IP addresses that the two streams will push from and make a note of them. You will need this information to set up the required input security groups for the MediaLive inputs that you will create.
RTMP Pull

- MediaLive works with redundant sources, so you provide two video streams. For optimized redundancy, MediaLive runs each source on different encoder pipelines in different Availability Zones. You don’t have to set up these Availability Zones because MediaLive does it for you.
- Make sure that the two streams are identical in terms of resolution and bitrate.
- For a pull input, keep in mind that the video source must be ready to be pulled before you start the channel. This rule applies for both HLS VOD inputs and HLS live inputs.

HLS Pull

- MediaLive works with redundant sources, so you provide two video streams. For optimized redundancy, MediaLive runs each source on different encoder pipelines in different Availability Zones. You don’t have to set up these Availability Zones because MediaLive does it for you.
- Make sure that the two streams are identical in terms of resolution and bitrate.
- The sources can be encrypted or unencrypted. For information on setting up encrypted content, see later in this section.
- For a pull input, keep in mind that the video source must be ready to be pulled before you start the channel. This rule applies for both HLS VOD inputs and HLS live inputs.

Handling Encrypted Source Content

MediaLive can ingest an HLS input that is encrypted according to the HTTP Live Streaming specification. MediaLive supports AES-128 but not AES-SAMPLE. MediaLive supports encryption using either static or rotating keys.

How Decryption Works

The content owner sets up the main manifest to include the #EXT-X-KEY with the method (AES-128), the URL to the license server, and the initialization vector (IV). The content owner places the HLS manifests on the upstream system (an HTTP(S) server, an AWS Elemental MediaStore container, or an Amazon S3 bucket), and places the encryption key on the license server. When the channel that contains this input starts, MediaLive obtains the main manifest, reads the #EXT-X-KEY tag for the URL of the encryption key, and obtains the encryption key from that location. MediaLive decrypts the input using the encryption key and the IV.

Get Ready

Contact the owner of the upstream system and verify that:

- The content is encrypted with AES-128.
- The manifest includes the #EXT-X-KEY tag with these attributes:
  - The METHOD attribute specifies AES-128
  - The URL specifies the license server for the encryption key.
  - The IV is blank or specifies the IV to use. If the IV is blank, MediaLive uses the value in EXT-X-MEDIA-SEQUENCE tag as the IV.
  - If both the upstream system and the license server require authentication credentials (user name and password), make sure that the same credentials are used on both servers. MediaLive does not support having different credentials for these two servers.
Planning the Inputs

After you ensure that the upstream system is set up correctly, there is no further setup for you to perform. There is no special setup to perform in the input or in the channel. When the channel starts, MediaLive reads the manifest and determines from the #EXT-X-KEY that the input is encrypted. It follows the process described earlier in this section to decrypt the content as it is ingested.

Planning the Inputs

To plan your input or inputs, identify which individual video, audio, and captions assets you want to extract from each input and which ones that you want to omit. For example, you must extract one video file, but you can choose to omit some captions languages.

The rules for extracting input are the following:

- You must extract one and only one video file from each input.

  If the channel will have multiple inputs, there is no requirement for the video properties in the various inputs to be identical in terms of codec, resolution, frame rate, color space, scan type, and so on. So, for example, the video in one input might be HEVC, while the video in another input might be H.264.

- You can extract zero or more audio files from each input. Typically, you extract multiple audio files so that you can include multiple languages in the output. But you can also extract multiple audio files to extract different audio formats, for example, AAC and Dolby Digital.

  If the channel will have multiple inputs, there is no requirement for the audio files in the various inputs to be identical in terms of codec, sample rate, bitrate, and so on.

- You can extract zero or more captions files.

  If the channel will have multiple inputs, read the information about setting up for captions in the section called “Captions in Channels with Multiple Inputs” (p. 169).

Planning the Channel

To plan your channel, follow these guidelines:

1. Identify the output protocols (for streaming outputs) or the number of different output file types (for archive and frame capture outputs).

   For example, you could create a streaming ABR HLS output asset, a streaming non-ABR HLS output asset, and an archive version of the HLS output asset (containing the highest bitrate video). You could also create a streaming ABR Smooth output asset.

2. For the first output asset, identify the number of video encodes that you need:

   - Some output assets consist of one video encode (one set of encoding settings). In this case, you should plan to create an output group that contains one video output.

   - An ABR output asset will have more than one video encode, for example, one high-bitrate video, one medium-bitrate video, and one low-bitrate video. The encoding instructions are identical (for example, they all use H.264) except for the bitrate. In this case, you should plan to create an output group with more than one video output.

3. For the first output asset, identify the audio encodes that you need.

   Typically, you need one encode for each language (English, French, and so on).

4. For the first output asset, identify the captions that you need.

   Typically, you need one encode for each captions language (English, French, and so on).

5. Group these encodes into outputs, as described in the section called “Examples of Channel Designs” (p. 61). Make sure that the groupings follow the rules for encodes in outputs (p. 61).
6. Group the outputs into one output group. For example, group the outputs for the ABR HLS outputs into one HLS output group.

7. Repeat the design of encodes, outputs, and output group for each output asset.

Here's the result of running a channel that follows this design:

- Each output that you create in the channel becomes one media asset.
- If a media asset includes manifests, one master manifest is created for each output group and one "variant manifest" is created for each output.

Rules for Encodes in an Output

The following rules apply to the organization of encodes into outputs:

- **Video** – Each video encode goes in its own output. One output can't contain two videos.
- **Audio** – In a non-ABR asset, each audio encode goes in its own output. In an ABR asset, the audio encodes don't go in their own output; they are contained in the same output as the video encode.
- **Captions** – Embedded-type captions encodes always go in the same output as the video encode, inside (embedded) in the video encode. Object-style captions encodes go in their own object in the same output as the video encode. Sidecar captions encodes always go in their own output.

These rules mean that one output can contain the following:

- A video encode
- A video encode and one or more audio encodes
- A video encode and one embedded-type captions
- A video encode and one or more object-type captions
- A video encode, one or more audio encodes, and one embedded-type captions
- A video encode, one or more audio encodes, and one or more object-type captions
- An audio encode
- A sidecar-type captions

Examples of Channel Designs

The following sections show examples of channel designs. The designs progress from simple designs that contain only one video offering to more complex designs for adaptive bitrate streaming (ABR) assets that contain several video offerings.

Topics

- **Non-ABR Asset with Captions Embedded in the Video** (p. 62)
- **Non-ABR Segmented Asset with Captions as Separate Objects** (p. 62)
- **Non-ABR Segmented Asset with Captions as Sidecars** (p. 62)
- **ABR Asset with Captions Embedded in the Video** (p. 63)
- **ABR Asset with Captions in Sidecars** (p. 63)
Non-ABR Asset with Captions Embedded in the Video

For a non-ABR asset, you create the following outputs and encodes in the channel: one output that contains one video asset, as many audio assets as you require, and as many captions assets as you require.

The following illustration shows one output in one output group. The output contains one video asset, one captions asset, and two audio assets.

Running this channel produces one segmented media file that contains the video, captions, and audio encodes.

In outputs that have manifests, it also produces one manifest file and one variant manifest file.

Non-ABR Segmented Asset with Captions as Separate Objects

In this example, captions are separate objects, but they are contained within the media asset (they are not sidecar files). Captions such as DVB-Sub are set up as separate objects.

Running this channel produces one segmented media file that contains the video, audio, and captions encodes.

Non-ABR Segmented Asset with Captions as Sidecars

The following illustration shows a non-ABR asset with one video asset and two audio assets in one output group, and two captions assets, each in their own output.

Running this channel produces one media file that contains video and audio, and one media file for each captions asset.
ABR Asset with Captions Embedded in the Video

For an ABR asset, you create the following outputs and encodes:

- Several video outputs, each containing one video encode (for example, one high-bitrate video, one medium-bitrate video, and one low-bitrate video) and the same embedded captions encode.
- One or more audio encodes (for example, one for each language).

The following illustration shows an example configuration.

Running this channel produces five sets of segmented media files, one set for each video output and each audio output.

In outputs that have manifests, it also produces one master manifest and five variant manifests.

ABR Asset with Captions in Sidecars

In this example, each captions asset (one for each language) is in its own output.

Running this channel produces seven sets of segmented media files.

In outputs that have manifests, it also produces one manifest file and seven variant manifests.

Setting Up the Downstream System

You must set up the device or application that will be downstream of MediaLive. The downstream system is different for different outputs.

The output from MediaLive is considered input to this downstream system. You must set up this downstream input now, because when you create the MediaLive channel you need the location of that input.
Important
Typically, you set up the MediaLive as a standard channel (p. 56), so that the channel has two pipelines. The procedures in this section assume that you have decided to set up in this way. The procedures therefore describe how to set up the downstream system to expect outputs at two destinations.
If you have decided to set up a single-pipeline channel, then there is only one pipeline in the channel. Follow these procedures, but set up the downstream system to expect outputs at only one destination.

Setting Up the Downstream System for an Archive Output

The downstream system is always an Amazon S3 bucket.
In Amazon S3, create two buckets, one for each MediaLive channel pipeline. Make a note of the full path of the buckets.

Setting Up the Downstream System for an HLS Output

If you want to deliver HLS output, you must decide if you want to create an HLS output group or a MediaPackage output group. If you choose to create an HLS output group, you must set up the downstream system.

Choosing between HLS and MediaPackage Output Groups

For most output destinations, you create an HLS output in order to deliver HLS output.
But if your destination is a channel in AWS Elemental MediaPackage, you can choose to create an HLS output or a MediaPackage output. There are differences in the setup of each type:

- When you decided on the channel class, you decided whether you wanted a standard channel (to support input redundancy in AWS Elemental MediaPackage) or a single-pipeline channel. If you chose single-pipeline, then you should create an HLS output group because you will be able to control the behavior when output is lost (p. ).
- The MediaPackage output requires less setup. AWS Elemental MediaLive is already set up with most of the information that it needs to package and deliver the output to the AWS Elemental MediaPackage channel that you specify.
- For a MediaPackage output, the MediaLive channel and the AWS Elemental MediaPackage channel must be in the same AWS Region. For an HLS output, the two channels can be in different Regions (although we recommend that they are in the same Region).
- In a MediaPackage output, the output is always a live stream, not a VOD stream. In an HLS output, you can choose whether to create a live or a VOD stream.
- In a MediaPackage output, there are some restrictions on setting up ID3 metadata. For details, see Working with ID3 Metadata (p. 198).
### Setting Up the Downstream System for a MediaPackage Output

A MediaPackage output is an HLS output where the destination is always a channel in AWS Elemental MediaPackage. AWS Elemental MediaPackage is typically serving as an origin server that a CDN such as Amazon CloudFront can pull from.

For information on the differences between HLS and MediaPackage outputs, see the section called “Choosing between HLS and MediaPackage Output Groups” (p. 64).

**To set up AWS Elemental MediaPackage**

1. Make sure that the AWS Elemental MediaPackage channel and the AWS Elemental MediaLive channel are in the same AWS Region.
2. In AWS Elemental MediaPackage, create one channel. See Creating a Channel in the AWS Elemental MediaPackage User Guide.
3. View the details of the MediaPackage channel. See Viewing Channel Details in the AWS Elemental MediaPackage User Guide. Make a note of the channel ID. The channel ID is case sensitive.

### Setting Up the Downstream System for a Frame Capture Output

The downstream system is always an Amazon S3 bucket.

In Amazon S3, create two buckets, one for each MediaLive channel pipeline. Make a note of the full path of the buckets.
Setting Up the Downstream System for a Microsoft Smooth Output

The downstream system is always a CDN that uses HTTP (or HTTPS) PUT. Typically, the downstream system is a Microsoft IIS server.

Set up the CDN so that the CDN expects MediaLive output at two inputs, one for each MediaLive channel pipeline. Make a note of the input addresses, and of the input user name and input password, if applicable.

Setting Up the Downstream System for an RTMP or RTMPS Output

The downstream system is always a server that uses RTMP or RTMPS. The server might be a CDN.

Set up the RTMP server so that it expects MediaLive output at two inputs, one for each MediaLive channel pipeline.

Perform the necessary setup on the RTMP server to obtain the following information for each input:

- URL for the output to send to.
- Port number.
- Application name. (The application name might be identical for both inputs.)
- Stream name.

Make a note of the URLs and port numbers.

The URL might include a path portion in the format `<string>/<string>`. In this case, the first string is the application name and the second string is the stream name. In addition, the RTMP server might refer to the stream name as the application instance or the stream key.

Setting Up the Downstream System for a UDP Output

The downstream system is an address that can communicate over UDP or RTP.

Set up the UDP destination so that it expects MediaLive output at two inputs, one for each MediaLive channel pipeline.

Perform the necessary setup at the UDP destination to obtain the following information for each input:

- URL for the output to send to
- Port number

Make a note of the URLs and port numbers.

Next Steps

Now that you have set up the upstream and downstream systems and have created the inputs that you need, you are ready to create the channel itself. See Creating a Channel from Scratch (p. 86).
Working with Input Security Groups

An *input security group* is a set of one or more whitelist rules. Each rule is a range of IP addresses. These IP addresses are allowed to push traffic to the input destinations of a channel (to push traffic to the channel's input).

Purpose of an Input Security Group

Input security groups are used with specific "push" inputs where the upstream system for the source is on the public internet:

- They are used for RTP inputs and RTMP push inputs that don't use a VPC.
- They aren't used for RTP VPC inputs, RTMP VPC push inputs, or MediaConnect inputs. These inputs implement security in other ways.

For the relevant push inputs, an input security group restricts access to the input and prevents unauthorized third parties from pushing content into a channel that is associated with that input. Without the protection of this feature, any third party could push content to an MediaLive input if they know the IP address and port. Note that setting permissions on the account that owns the channel does not prevent this third-party push; only an input security group prevents it.

You can attach an input security group to more than one input; one input security group can "serve" several inputs.

Creating an Input Security Group

You create an input security group and then attach it to a "push" input when you create or edit that input.

**To create an input security group**

2. In the navigation pane, choose *Input security groups*.
3. On the *Input security groups* page, choose *Create input security group*.
4. For *New security group*, type one or more IPv4 CIDR blocks. Each CIDR block must include a subnet mask. Separate the entries with commas, or type each entry on a separate line.

   Each item in the list represents one whitelist rule, even if it encompasses several individual addresses. For example, each of the following examples counts as one rule:

   192.0.2.0/24
   192.0.2.111/32

5. In the *Tags* section, create tags if you want to associate tags with this input security group. For more information, see *Tagging Resources* (p. 227).
6. Choose *Create*. 
Editing an Input Security Group

You can edit any of the fields in an input security group. You can perform these edits at any time, even if the input security group is attached to an input that is attached to a channel that is running.

To edit an input security group
2. In the navigation pane, choose Input Security Groups.
3. On the Input security groups page, choose the input security group, and then choose Edit.
4. Change any fields as appropriate, and then choose Update.

Wait for the input security State to return to In use or Idle before performing another action with this input security group.

To add, delete, or edit tags in an input security group
2. In the navigation pane, choose Input Security Groups.
3. On the Input security groups page, choose the name of the input security group. Do not choose Edit.
4. On the Input security group page for this input security group, in the Tags section, add or delete tags. To edit the value of an existing tag, delete the tag and add it again. For more information, see Tagging Resources (p. 227).

Wait for the input security State to return to In use or Idle before performing another action with this input security group.

Deleting an Input Security Group

You can delete an input security group so long as it is not attached to any inputs.

To delete an input security group
2. In the navigation pane, choose Input security groups.
3. On the Input security groups page, look at the State for the group to delete:
   - If the State is Idle, choose the group, and then choose Delete.
   - If the State is In use, continue with this procedure.
4. Make a note of the ID of the input security group. For example, 1234567.
5. Choose the group, and then choose Edit.
6. On the Edit input security group page, look at the Inputs on the right side and count how many inputs are attached to this input security group.
7. Choose the first input. Then on the page for that input, choose Edit. On the Edit page, in the Input security group, either create a new input security group for this input or choose another group (make sure you don't rechoose the same group; check the ID that you noted earlier). Choose Update so that this input is no longer attached to the input security group that you want to delete.
8. If there are still more inputs associated with this input group, then in the navigation pane, choose Input security groups, and repeat these steps to detach this input security group from all the inputs.
9. After detaching the last input from this input security group, wait for the State of the input security group to specify Idle. Then choose the group, and choose Delete.
Working with Inputs in AWS Elemental MediaLive

An input is a video asset that is to be transcoded and packaged. The source of the video asset is the upstream system (p. 55)—the system in your end-to-end workflow whose activities occur before those of AWS Elemental MediaLive. The upstream system can be on the public internet or in a virtual private cloud (VPC) that you created using Amazon Virtual Private Cloud (Amazon VPC).

MediaLive supports specific input types. For information about the types, see the section called “Supported Input Types and Upstream Systems” (p. 236).

**Important**
Typically, you set up the MediaLive channel as a standard channel (p. 56). The procedures in this chapter assume that you have set up in this way. The procedures therefore refer to an upstream system having two sources that deliver content to two endpoints on the MediaLive input. If you will set up the channel as a single-pipeline channel, then your upstream system has only one source. MediaLive creates two endpoints on an input, but only the first endpoint is used.

Inputs, Input Security Groups, and Channels

The input is one of the components of a workflow. The others are the input security group (p. 54) and the channel (p. 53). These three components are linked together. An input security group is attached to an input (if the input requires an input security group; not all inputs have this requirement). An input is attached to a channel.

The following rules apply to the linking to an input:

- The association between an input and an input security group is defined in the input side. You set up the association when you create or edit the input.
- The association between an input and a channel is defined on the channel side. You set up the association when you create or edit the channel.
- An input can have only one input security group attached to it. But that input security group can be already attached to another input; one input security group can "serve" several inputs.
- An input can be attached to only one channel; several channels can't be attached to the same input.
- If your channel has multiple inputs, you can attach the same input more than once to a channel. For example, you might have a file to use to fill the time between live events. You can use this file as often as you want.

Creating an Input

To provide information about the source of the video asset, you must create an AWS Elemental MediaLive input.

**Topics**

- Creating an HLS Pull Input (p. 71)
Creating an HLS Pull Input

Create your input before you create the channel that ingests the input.

The HLS input can be encrypted or unencrypted. For information on preparing encrypted content to ensure that MediaLive can successfully unencrypt it, see the section called “HLS Pull” (p. 59).

To create an HLS pull input

2. In the navigation pane, choose Inputs.
3. On the Inputs page, choose Create input.
4. In the Input details section, for Input name, enter a name.
5. For Input type, choose HLS.
6. From the upstream system, obtain the full URLs of the locations where MediaLive will pull the M3U8 manifest source from. The upstream system will provide two URLs (for a standard channel (p. 56)) or one URL (for a single-pipeline channel).

The URLs will have one of these formats:

- For a location that supports HTTP or HTTPS, enter an HTTP or HTTPS URL. For example, https://203.0.113.13/newschannel/anytownusa.m3u8 and https://203.0.113.54/newschannel/anytownusa.m3u8.
- For a manifest that is stored on AWS Elemental MediaStore, the URL must include the data endpoint for the container. For example, the M3U8 file is called mlaw.m3u8, and it is stored in the container "movies" in the folder path premium/canada. The URL for the container might be eri39n.data.mediastore.us-west-2.amazonaws.com. The value that you enter in this field would be mediastoressl://eri39n.data.mediastore.us-west-2.amazonaws.com/premium/canada/mlaw.m3u8.
- For a manifest that is stored on Amazon S3, enter the protocol as s3 or s3ssl, and then enter the bucket name and object for the manifest. For example, s3://movies/mlaw.m3u8 and s3://movies/redundant/mlaw.m3u8.

7. In the Input sources section, enter these URLs in one or both fields:

- If the channel for this input will be set up as a standard channel (p. 56), complete both fields, to provide two URLs.
- If the channel for this input will be set up as a single-pipeline channel, complete the first field with the URL that you obtained and leave the second field empty.

8. If you use a secure connection (s3ssl), you must also enter the user name and Amazon EC2 password key for accessing the location. These credentials are stored on the Amazon EC2 Systems Manager Parameter Store. For more information, see the section called “About the Feature for Creating Password Parameters” (p. 29).

9. In the Tags section, create tags if you want to associate tags with this input. For more information, see Tagging Resources (p. 227).
10. Choose **Create**.

MediaLive creates the input and adds it to the list of inputs. The input specifies either one or two sources. The sources don't appear in the list, but if you choose the **Name** link, the details page shows them.

When you start the channel, MediaLive will connect to the upstream system at this source location or locations and pull the content:

- For a channel set up as a standard channel, MediaLive expects the upstream system to provide two sources and will therefore attempt to pull from both source locations.
- For a channel set up as a single-pipeline channel, MediaLive expects the upstream system to provide one source and will therefore attempt to pull from one source location.

## Creating a MediaConnect Push Input

There are three stages to creating a MediaConnect input in AWS Elemental MediaLive. First, you set up flows in AWS Elemental MediaConnect, then you create an input in MediaLive, then you verify that the output has been automatically created in AWS Elemental MediaConnect.

### To create flows in AWS Elemental MediaConnect

1. Follow the procedure in **Creating a Flow** in the *AWS Elemental MediaConnect User Guide* to create one or two MediaConnect flows:

   - If the channel for this input will be set up as a standard channel (p. 56), create two flows.
   - If the channel for this input will be set up as a single-pipeline channel, create one flow.

   Complete the fields as follows:

   - **Name**: enter a name that includes A (for the first flow) and B (for the second flow). For example, `sports_event_A` and `sports_event_B`. Using these letters will help you match the flows to the input pipelines in AWS Elemental MediaConnect.
   - **Availability Zone**: choose a different Availability Zone for each flow. (If you choose the same Availability Zone for both flows, MediaLive doesn't allow you to create the input.)
   - **Source**: keep in mind that the source here is the source into AWS Elemental MediaConnect. It has nothing to do with MediaLive.
   - **Don't create outputs or entitlements. In the next stage, MediaLive will automatically create outputs.**
   - **When you create the flow, AWS Elemental MediaConnect creates an ARN for each flow. For example:**

     ```
     arn:aws:mediaconnect:us-west-1:111122223333:flow:1bgf67:sports_event_A
     ```

     Note that the ARNs include the flow names as the last portion.

2. Make a note of the ARN or ARNs. You need them to set up the MediaLive input.

### To create an input in MediaLive


2. In the navigation pane, choose **Inputs**.

3. On the **Inputs** page, choose **Create input**.
4. In the **Input details** section, for **Input name**, enter a name.

5. For **Input type**, choose **MediaConnect**.

6. In the **MediaConnect** section, for **ARN for flow A**, specify the ARN for the flow that you identified as flow A.

   If you created a second flow, then for **ARN for flow B**, specify the ARN for flow B.

7. Complete the **Role ARN** section to choose a role for MediaLive to use with this input. For information, see the section called “IAM Role and ARN” (p. 73).

8. In the **Tags** section, create tags if you want to associate tags with this input. For more information, see **Tagging Resources** (p. 227).

9. Choose **Create**.

   MediaLive creates the input and automatically creates two endpoints on that input. MediaLive always creates two endpoints, even if you specified only one flow (flow A) for the input.

10. At the same time, MediaLive automatically connects to the AWS Elemental MediaConnect flows.

   - If you specified two flows for the input, MediaLive instructs AWS Elemental MediaConnect to create two outputs and attach them to the two flows that you created in the first stage.
   - If you specified only one flow for the input (to support a single-pipeline channel), MediaLive instructs AWS Elemental MediaConnect to create one output and to attach it to the single flow that you created in the first stage.

   If AWS Elemental MediaConnect has two flows for the channel, it runs the flows in different Availability Zones—one zone for flow A, another zone for flow B. Similarly, MediaLive runs each pipeline in a different Availability Zone—one zone for pipeline A, another zone for pipeline B. MediaLive and AWS Elemental MediaConnect coordinate so that both services choose the same Availability Zone for flow and pipeline A and for flow and pipeline B. This setup ensures maximum redundancy if one flow fails.

**To verify the output in AWS Elemental MediaConnect**

1. To view the list of outputs for the flow or flows that you created, follow the procedure in Viewing a List of Outputs of a Flow in the **AWS Elemental MediaConnect User Guide**.

   When you created the input in MediaLive, AWS Elemental MediaConnect automatically created outputs. Therefore, you end up with the following:

   Flow A (in MediaConnect) – output for flow A (in MediaConnect) – source for A (in MediaLive)
   Flow B (in MediaConnect) – output for flow B (in MediaConnect) – source for B (in MediaLive)

2. If you created two flows, make sure that there is one output for flow A and one output for flow B.

   Each output is assigned a name that consists of **MediaLive dash random characters**. For example:

   MediaLive-ace74fa23

**IAM Role and ARN**

This section describes how to complete the **Role ARN** section on the **Create input** pane of the MediaLive console.

You must choose a role for MediaLive to assume when it creates any input. The role ensures that MediaLive succeeds in its request to AWS Elemental MediaConnect to create outputs on the flows. MediaLive sends this request as soon as you choose **Create** for this input.
Note
This section on the MediaLive console is identical to the IAM role section on the Create channel page (also on the MediaLive console). The difference in the two usages is that on the Create input page, you are attaching the role to the input. On the Create channel page, you are attaching the role to the channel. You can use the same role (for example, the MediaLiveAccessRole) in both usages.

There are two general scenarios for choosing a role, depending on whether your organization has a designated administrator.

Your Organization Has a Designated Administrator
Your organization might have an administrator who manages this service. That administrator has likely set up one or more roles:

• Ask the administrator or your manager which role to use. Or if only one role is listed in Use existing role, choose that role.
• If the only role that is listed is MediaLiveAccessRole, choose that role. In addition, if the Update button is displayed beside this role name, choose the button. (The button does not always appear, but whenever it does appear, choose it to refresh the role.)
• If you want the selected role to appear first in the list next time, select Remember ARN.

Your Organization Has No Administrator
Your organization might not have a designated service administrator. In this case, if none of your colleagues have set up a suitable role, you might have to create one yourself and then choose it.

• You can create the default role, called MediaLiveAccessRole. To first check if someone else has already created this role (only one person needs to create it for all users in your AWS account), look at Create role from template:
  • If this option is grayed out, this task has been done. In that case, choose Use existing role, and then choose MediaLiveAccessRole from the list.
  • If this option is not grayed out, choose Create role from template, and then choose Create IAM role. Next, choose that role from the list. If MediaLive does not let you create the role, speak to an AWS IAM administrator in your organization about your permissions.
• If the MediaLiveAccessRole has already been created and the Update button is displayed beside it, choose the button. (The button does not always appear, but whenever it does appear, choose it to refresh the role.)
• If you want the selected role to appear first in the list next time, select Remember ARN.

Creating an MP4 Pull Input
Create your input before you create the channel that ingests the input.

To create an MP4 pull input
2. In the navigation pane, choose Inputs.
3. On the Inputs page, choose Create input.
4. In the Input details section, for Input name, enter a name.
5. For Input type, choose MP4.
6. From the upstream system, obtain the full URLs of the locations where MediaLive will pull the MP4 file from. The upstream system will provide two URLs (for a standard channel (p. 56)) or one URL (for a single-pipeline channel):
Creating an RTMP Push Input

Create your input before you create the channel that ingests the input.

To create an RTMP push input

1. Obtain the ID of the input security group (p. 67) that you will use with this input. (Or if you will create an input security group at the same time as you create this input, obtain the IP addresses for the input security group.) Obtain the application name and application instance (p. 58) that you identified when planning the workflow.
3. In the navigation pane, choose Inputs.
4. On the Inputs page, choose Create input.
5. In the Input details section, for Input name, enter a name.
6. For Input type, choose RTMP (push).
7. In the Network mode section, choose Public.
8. In the **Input security group** section, specify the group to attach to this push input. You can choose an existing group, or you can create a group. For more information about security groups, see *Working with Input Security Groups* (p. 67).

9. In the **Input destinations** section, for **Destination A**, enter the application name and application instance for one of the sources. For example, `live` and `curling`. The names that you enter must match the names that the upstream system assigned to the source. If they do not, MediaLive might detect the source but does not process it.

10. In the **Input destinations** section, for **Destination B**, enter the application name and application instance for the other source. (If the channel is set up as a single-pipeline channel, leave this destination empty.)

11. In the **Tags** section, create tags if you want to associate tags with this input. For more information, see *Tagging Resources* (p. 227).

12. Choose **Create**.

   MediaLive creates the input and automatically creates two endpoints on that input. The endpoints include the application name, the application instance, and the port 1935. For example:

   ```
   ```

   MediaLive always creates two endpoints:
   - If the channel for this input will be set up as a standard channel, both endpoints will be used.
   - If the channel for this input will be set up as a single-pipeline channel, only the first endpoint will be used. MediaLive won't expect to receive content at the second endpoint.

13. Provide the upstream system with the following information:
   - If the channel for this input will be set up as a standard channel, provide both locations. The upstream system must push the video streams to these locations.
   - If the channel for this input will be set up as a single-pipeline channel, provide only the first location. The upstream system must push its one stream to this location.

---

### Creating an RTMP Push Input in Amazon VPC

You create an RTMP push input in Amazon Virtual Private Cloud (Amazon VPC) to push content from an upstream system that is in your VPC to MediaLive. Create your input before you create the channel that ingests the input.

**To set up the upstream system and the VPC**

1. Identify a VPC. Make a note of the VPC ID. For example:

   ```
   vpc-3f139646
   ```

   **Note**
   We recommend that your organization create a VPC specifically for all AWS Media Services. A single VPC will help to ensure the availability of IP addresses, help in setting up appropriate rules in the security groups, help in adhering to the two Availability Zone rule, and help to ensure that a network administrator doesn't accidentally delete elastic network interfaces.

2. Identify two subnets in the VPC. These rules apply to the subnets:
   - The two subnets must be in different Availability Zones.
   - Each subnet must have a private CIDR block (a range of IP addresses).
• Each subnet must have at least two unused addresses in that block—one for the upstream system and one for the MediaLive input.
• Any other RTP VPC inputs or RTMP VPC inputs that you create for use in the same channel as this input must be in subnets that are in the same Availability Zones as this input. The two subnets of the new inputs can be different from the existing input, but the Availability Zones of those two subnets must be the same as the Availability Zones of this input.

Make a note of the subnet IDs. For example:

subnet-1122aabb
subnet-4455ccdd

3. On your upstream system, set up two identical source streams (if the channel for this input will be set up as a standard channel (p. 56)) or one source stream (if it will be set up as a single-pipeline channel). Set up one source stream so that it has an output interface in one of the subnets, and the other so that it has an output interface in the other subnet.

4. Make sure that the two streams are identical in terms of resolution and bitrate.

5. Identify at least one security group for each subnet.

Note
Don’t confuse the security groups that belong to Amazon VPC with the input security groups that belong to MediaLive.

These rules apply to the security groups for each subnet:

• The combined rules of the security groups must allow inbound traffic from the IP addresses of the upstream system in that subnet.
• The combined rules of the security groups must allow outbound traffic to port 3500.

6. Make a note of the IDs of the security group or groups. For example:

sg-51530134

7. Make a note of the following three characteristics of the source stream. You will need this information to set up the channel:

• The video codec
• The resolution of the video—SD, HD, or UHD
• The maximum input bitrate

To create an RTMP VPC push input
2. In the navigation pane, choose Inputs.
3. On the Inputs page, choose Create input.
4. In the Input details section, for Input name, enter a name.
5. For Input type, choose RTMP (push).
6. In the Network mode section, choose VPC.
7. In the VPC settings section, choose Use existing VPC. For Subnets, choose one of the subnets that you identified. The dropdown list shows subnets in all VPCs, identified as follows:

<subnet ID> <Availability Zone of subnet> <IPv4 CIDR block of subnet> <VPC ID> <Subnet tag called "Name", if it exists>

For example:
subnet-1122aabb us-west-2a 10.1.128.0/24 vpc-3f139646 Subnet for MLive push inputs

If the list of subnets is empty, choose Specify custom VPC and enter the subnet ID in the field. (You need to enter only the subnet ID, for example, subnet-1122aabb.)

8. Choose the second subnet. This second time, the dropdown list shows only the subnets in the same VPC as the first subnet.

9. For Security groups, choose the security group or groups that you identified, following the same process as for the subnets. The dropdown list shows security groups belonging to the VPC that you chose, identified as follows:

<security group ID> <description attached to this security group> <VPC ID>

For example:

sg-51530134 Security group for MLive push inputs vpc-3f139646

10. Complete the Role ARN section to choose a role for MediaLive to use with this input. For more information, see the section called “IAM Role and ARN” (p. 83).

11. In the Tags section, create tags if you want to associate tags with this input. For more information, see Tagging Resources (p. 227).

12. Choose Create.

MediaLive creates the input and automatically creates two endpoints on that input. These endpoints have a private IP address from the subnet range, and they specify port 1935. For example:

rtmp://10.99.20.15:1935

rtmp://192.0.2.53:1935.

13. Provide the upstream system with these endpoints:

- If the channel for this input will be set up as a standard channel, provide both endpoints. The upstream system must push the content to both endpoints.
- If the channel for this input will be set up as a single-pipeline channel, provide only the first endpoint. The upstream system must push to this one endpoint.

Result of These Procedures

As a result of this setup, each output of the upstream system has an IP address in one of the specified subnets in your VPC. The RTMP input has two IP addresses. Each address is in one of those same subnets. In this way, the delivery of the content from the upstream system to MediaLive takes place within the security of the VPC.

Keep in mind that with a push input, the upstream system must be pushing the video source to the input when you start the channel. The upstream system does not need to be pushing before then.

IAM Role and ARN

This section describes how to complete the Role ARN section on the Create input pane of the MediaLive console.

You must choose a role for MediaLive to assume when it creates an RTMP Push input. In order to create the input, MediaLive must obtain the network interfaces for the two endpoints in the input. These endpoints are in the CIDR range of the subnets that you identified. As soon as you choose Create for this input, MediaLive requests these network interfaces from Amazon VPC. The role that you choose ensures that MediaLive succeeds in its request to Amazon VPC.
Note
This section on the MediaLive console is identical to the IAM role section on the Create channel page (also on the MediaLive console). The difference in the two usages is that on the Create input page, you are attaching the role to the input. On the Create channel page, you are attaching the role to the channel. You can use the same role (for example, the MediaLiveAccessRole) in both usages.

There are two general scenarios for choosing a role, depending on whether your organization has a designated administrator.

Your Organization Has a Designated Administrator
Your organization might have an administrator who manages this service. That administrator has likely set up one or more roles:

- Ask the administrator or your manager which role to use. Or if only one role is listed in Use existing role, choose that role.
- If the only role that is listed is MediaLiveAccessRole, choose that role. In addition, if the Update button is displayed beside this role name, choose the button. (The button does not always appear, but whenever it does appear, choose it to refresh the role.)
- If you want the selected role to appear first in the list next time, select Remember ARN.

Your Organization Has No Administrator
Your organization might not have a designated service administrator. In this case, if none of your colleagues have set up a suitable role, you might have to create one yourself and then choose it.

- You can create the default role, called MediaLiveAccessRole. To first check if someone else has already created this role (only one person needs to create it for all users in your AWS account), look at Create role from template:
  - If this option is grayed out, this task has been done. In that case, choose Use existing role, and then choose MediaLiveAccessRole from the list.
  - If this option is not grayed out, choose Create role from template, and then choose Create IAM role. Next, choose that role from the list. If MediaLive does not let you create the role, speak to an AWS IAM administrator in your organization about your permissions.
  - If the MediaLiveAccessRole has already been created and the Update button is displayed beside it, choose the button. (The button does not always appear, but whenever it does appear, choose it to refresh the role.)
  - If you want the selected role to appear first in the list next time, select Remember ARN.

Creating an RTMP Pull Input
Create your input before you create the channel that ingests the input.

To create an RTMP pull input
2. In the navigation pane, choose Inputs.
3. On the Inputs page, choose Create input.
4. In the Input details section, for Input name, enter a name.
5. For Input type, choose RTMP (pull).
6. From the upstream system, obtain the full URLs of the locations where MediaLive will pull the source from. The upstream system will provide two URLs (for a channel set up as a standard channel (p. 56)) or one URL (for a single-pipeline channel). For example:
Creating an RTP Push Input

Create your input before you create the channel that ingests the input.

To create an RTP push input

2. In the navigation pane, choose Inputs.
3. On the Inputs page, choose Create input.
4. In the Input details section, for Input name, enter a name.
5. For Input type, choose RTP.
6. In the Network mode section, choose Public.
7. In the Input security group section, specify a group to attach to this "push" input. You can choose an existing group, or you can create a group. For more information about security groups, see Working with Input Security Groups (p. 67).
8. In the Tags section, create tags if you want to associate tags with this input. For more information, see Tagging Resources (p. 227).
9. Choose Create.

MediaLive creates the input and automatically creates two endpoints on that input. These endpoints include the port 5000. For example:

rtp://203.0.113.19:5000
rtp://203.0.113.131:5000.

MediaLive always creates two endpoints:

- If the channel for this input will be set up as a standard channel, both endpoints will be used.
- If the channel for this input will be set up as a single-pipeline channel, only the first endpoint will be used. MediaLive won't expect to receive content at the second endpoint.

10. Provide the upstream system with the following information:

- If the channel for this input will be set up as a standard channel, provide both locations. The upstream system must push the video streams to these locations.
- If the channel for this input will be set up as a single-pipeline channel, provide only the first location. The upstream system must push its one stream to this location.

Creating an RTP Push Input in Amazon VPC

You create an RTP push input in Amazon Virtual Private Cloud to push content from an upstream system that is on your VPC to MediaLive. Create your input before you create the channel that ingests the input.

To set up the upstream system and the VPC

1. Identify a VPC. Make a note of the VPC ID. For example:

   vpc-3f139646

   Note
   We recommend that your organization create a VPC specifically for all AWS Media Services. A single VPC will help to ensure the availability of IP addresses, help in setting up appropriate rules in the security groups, help in adhering to the two Availability Zone rule, and help to ensure that a network administrator doesn't accidentally delete elastic network interfaces.

2. Identify two subnets in the VPC. These rules apply to the subnets:

   - The two subnets must be in different Availability Zones.
   - Each subnet must have a private CIDR block (a range of IP addresses).
   - Each subnet must have at least two unused addresses in that block—one for the upstream system and one for the MediaLive input.
   - Any other RTP VPC inputs or RTMP VPC inputs that you create for use in the same channel as this input must be in subnets that are in the same Availability Zones as this input. The two subnets of the new inputs can be different from the existing input, but the Availability Zones of those two subnets must be the same as the Availability Zones of this input.

   Make a note of the subnet IDs. For example:

   subnet-1122aabb

   subnet-4455ccdd

3. On your upstream system, set up two identical source streams (if the channel for this input will be set up as a standard channel (p. 56)) or one source stream (if it will be set up as a single-pipeline channel). Set up one source stream so that it has an output interface in one of the subnets, and the other so that it has an output interface in the other subnet.

4. Make sure that the two streams are identical in terms of resolution and bitrate.

5. Identify at least one security group for each subnet.
Note
Don’t confuse the security groups that belong to Amazon VPC with the input security
groups that belong to MediaLive.

These rules apply to the security groups for each subnet:

- The combined rules of the security groups must allow inbound traffic from the IP addresses of the
  upstream system in that subnet.
- The combined rules of the security groups must allow outbound traffic to port 5000.

6. Make a note of the IDs of the security group or groups. For example:
   
   sg-51530134

7. Make a note of the following three characteristics of the source stream. You will need this
   information to set up the channel:

   - The video codec
   - The resolution of the video: SD, HD, or UHD.
   - The maximum input bitrate

To create an RTP VPC push input from Amazon VPC

2. In the navigation pane, choose Inputs.
3. On the Inputs page, choose Create input.
4. In the Input details section, for Input name, enter a name.
5. For Input type, choose RTP.
6. In the Network mode section, choose VPC.
7. In the VPC settings section, choose Use existing VPC. For Subnets, choose one of the subnets that
   you identified. The dropdown list shows subnets in all VPCs, identified as follows:

   <subnet ID> <Availability Zone of subnet> <IPv4 CIDR block of subnet> <VPC
   ID> <Subnet tag called "Name", if it exists>

   For example:

   subnet-1122aabb us-west-2a 10.1.128.0/24 vpc-3f139646 Subnet for MLive push inputs

   If the list of subnets is empty, choose Specify custom VPC, and enter the subnet ID in the field. (You
   need to enter only the subnet ID, for example, subnet-1122aabb.)
8. Choose the second subnet. This second time, the dropdown list shows only the subnets in the same
   VPC as the first subnet.
9. For Security groups, choose the security group or groups that you identified, following the same
   process as for the subnets. The dropdown list shows security groups belonging to the VPC that you
   chose, identified as follows:

   <security group ID> <description attached to this security group> <VPC ID>

   For example:

   sg-51530134 Security group for MLive push inputs vpc-3f139646

10. Complete the Role ARN section to choose a role for MediaLive to use with this input. For more
    information, see the section called “IAM Role and ARN” (p. 83).
11. In the Tags section, create tags if you want to associate tags with this input. For more information,
    see Tagging Resources (p. 227).
12. Choose **Create**.

   MediaLive creates the input and automatically creates two endpoints on that input. These endpoints have a private IP address from the subnet range, and they specify port 5000. For example:


   rtp://192.0.2.44:5000.

13. Provide the upstream system with these endpoints:

   - If the channel for this input will be set up as a standard channel, provide both endpoints. The upstream system must push the content to both endpoints.
   - If the channel for this input will be set up as a single-pipeline channel, provide only the first endpoint. The upstream system must push to this one endpoint.

**Result of These Procedures**

As a result of this setup, each output of the upstream system has an IP address in one of the specified subnets in your VPC. The RTP input has two IP addresses. Each address is in one of those same subnets. In this way, the delivery of the content from the upstream system to MediaLive takes place within the security of the VPC.

Keep in mind that with a push input, the upstream system must be pushing the video source to the input when you start the channel. The upstream system does not need to be pushing before then.

**IAM Role and ARN**

This section describes how to complete the **Role ARN** section on the **Create input** pane of the MediaLive console.

You must choose a role for MediaLive to assume when it creates an RTP Push input. In order to create the input, MediaLive must obtain the network interfaces for the two endpoints in the input. These endpoints are in the CIDR range of the subnets that you identified. As soon as you choose **Create** for this input, MediaLive requests these network interfaces from Amazon VPC. The role that you choose ensures that MediaLive succeeds in its request to Amazon VPC.

**Note**

This section on the MediaLive console is identical to the **IAM role** section on the **Create channel** page (also on the MediaLive console). The difference in the two usages is that on the **Create input** page, you are attaching the role to the input. On the **Create channel** page, you are attaching the role to the channel. You can use the same role (for example, the **MediaLiveAccessRole**) in both usages.

There are two general scenarios for choosing a role, depending on whether your organization has a designated administrator.

**Your Organization Has a Designated Administrator**

Your organization might have an administrator who manages this service. That administrator has likely set up one or more roles:

- Ask the administrator or your manager which role to use. Or if only one role is listed in **Use existing role**, choose that role.
- If the only role that is listed is **MediaLiveAccessRole**, choose that role. In addition, if the **Update** button is displayed beside this role name, choose the button. (The button does not always appear, but whenever it does appear, choose it to refresh the role.)
- If you want the selected role to appear first in the list next time, select **Remember ARN**.
Your Organization Has No Administrator

Your organization might not have a designated service administrator. In this case, if none of your colleagues have set up a suitable role, you might have to create one yourself and then choose it.

- You can create the default role, called MediaLiveAccessRole. To first check if someone else has already created this role (only one person needs to create it for all users in your AWS account), look at Create role from template:
  - If this option is grayed out, this task has been done. In that case, choose Use existing role, and then choose MediaLiveAccessRole from the list.
  - If this option is not grayed out, choose Create role from template, and then choose Create IAM role. Next, choose that role from the list. If MediaLive does not let you create the role, speak to an AWS IAM administrator in your organization about your permissions.
- If the MediaLiveAccessRole has already been created and the Update button is displayed beside it, choose the button. (The button does not always appear, but whenever it does appear, choose it to refresh the role.)
- If you want the selected role to appear first in the list next time, select Remember ARN.

Editing an Input

The rules for editing an input are as follows:

- You can attach a different input security group.
- For an RTP input or an RTMP push input that isn't for a VPC, you can edit the fields in the input endpoint.
- For an RTP VPC input or an RTMP VPC push input, you can't edit the IP addresses input endpoint. To change these addresses, you must delete the input and create it again.
- For a MediaConnect push input, you can edit the ARNs to refer to different MediaConnect flows. The outputs for the former ARNs will be deleted in AWS Elemental MediaConnect, and new outputs (with new IDs) for the new ARNs will be created.
- For a pull input, you can edit the fields in an input source.
- You can't change the type of an input. For example, if you set up an input as an RTMP push but it is actually an HLS input, delete the input and create it again.

There are constraints on performing these edits, as follows:

- If an input is attached to a channel, you can edit the input only if the channel is idle.
- If an input is attached to a channel and an input security group, you can edit the input only if the channel is idle.
- If an input is not attached to a channel, you can edit it at any time, even if it is attached to an input security group.

To edit an input

2. In the navigation pane, choose Inputs.
3. Choose the name of the input, and then choose Edit.
4. On the Inputs page, make the following changes as appropriate:
   - You can change the Name.
   - You can't change the Input type. If the input has the wrong type, delete it and create it over again.
Deleting an Input

If an input is not attached to a channel, you can delete it, even if it is attached to an input security group.

**Note**
If the input is attached to a channel, you can't delete the input. This rule exists to ensure that you don't remove an input from a channel and therefore make the channel unusable.

When you delete an input, the attached input security group (if any) is not deleted.

**To delete an input**

2. In the navigation pane, choose **Inputs**.
3. On the **Inputs** page, find the input that you want to delete, and then look at the **State** column.
4. If the state is **Detached**, then choose **Delete**. If the state is **Attached** and you want to delete both the input and its channel, then delete the channel first. For more information, see the section called "Deleting a Channel" (p. 135).

If the input is a MediaConnect push input, the corresponding outputs in AWS Elemental MediaConnect are automatically deleted; you don't have to delete the outputs.

If the input is an RTP VPC input or an RTMP VPC push input, the elastic network interfaces of the endpoints are deleted and the IPv4 addresses in the subnet are released for use by another resource. You don't have to delete the network interfaces.
Creating a Channel from Scratch

A channel contains the details that instruct AWS Elemental MediaLive how to transcode (decode and encode) and package your input into specific outputs.

Before you start the process of creating a channel, you should plan your channel (p. 55) to identify the following elements:

- Inputs that the channel will use
- Output groups for the channel
- Outputs (within each output group)
- Video, audio, and captions encodes (in each output) that the channel will produce

There are three ways to create a channel:

- **Create from scratch.** See the topics (steps 1-9) in this chapter.
- **Use a built-in or custom template.** See Creating a Channel from a Template or by Cloning (p. 120).
- **Clone an existing channel.** See Creating a Channel from a Template or by Cloning (p. 120).

**Topics**

- Step 1: Complete the Channel and Input Details (p. 86)
- Step 2: Attach Inputs to the Channel (p. 89)
- Step 3: Complete the Settings for Each Input (p. 92)
- Step 4: Complete the General Settings (p. 94)
- Step 5: Create Output Groups (p. 95)
- Step 6: Create Outputs (p. 109)
- Step 7: Set Up the Video Encode (p. 116)
- Step 8: Set Up the Audio Encodes (p. 118)
- Step 9: Set up the Captions Encodes (p. 118)
- Step 10: Save the Channel (p. 119)

**Step 1: Complete the Channel and Input Details**

The first step to creating a channel from scratch is to choose the IAM role that AWS Elemental MediaLive will use to access the channel when the channel is running (started) and specify key characteristics of the input.

**To provide channel and input details**

2. Before creating a channel, make sure that you have created the input (p. 70) that you will attach to the channel.
3. On the MediaLive home page, choose **Create channel**, and in the navigation pane, choose **Channels**.

   If you've created a channel before, you won't see the home page. In that case, in the MediaLive navigation pane, choose **Channels**, and then choose **Create channel**.
4. On the Create channel page, choose Channel and input details.
5. In the General info section, do the following:
   - For Channel name, enter a name for your channel.
   - Complete IAM role. See the section called “IAM Role and ARN” (p. 87).
6. For information about the Channel template section, see Creating a Channel from a Template or by Cloning (p. 120).
7. In the Channel class section, choose the class. See the section called “Channel Class” (p. 88).
8. In the Input specifications section, complete the fields to match your input. See the section called “Input Specifications Settings” (p. 88).
9. In the Tags section, create tags if you want to associate tags with this channel. For more information, see Tagging Resources (p. 227).
10. When ready, go to the next step (p. 89).

IAM Role and ARN

This section describes how to complete the IAM role section in the General info section of the Channel and input details pane.

You must choose a role for MediaLive to assume when it works with this channel. If you don’t choose a role, you can’t create the channel. There are two general scenarios, depending on whether your organization has a designated administrator.

Note
This section on the MediaLive console is identical to the IAM role section on the Create input page for a MediaConnect push input (also on the MediaLive console). The difference in the two usages is that on the Create channel page, you attach the role to the channel. On the Create input page, you attach the role to the MediaConnect input. You can use the same role (for example, the MediaLiveAccessRole) in both usages.

There are two general scenarios for choosing a role, depending on whether your organization has a designated administrator.

Your Organization Has a Designated Administrator

Your organization might have an administrator who manages this service. That administrator has likely set up one or more roles:

- Ask the administrator or your manager which role to use. Or if only one rule is listed in Use existing role, choose that role.
- If the only rule that is listed is MediaLiveAccessRole, choose that role. In addition, if the Update button is displayed beside this role name, choose the button. (The button does not always appear, but whenever it does appear, choose it to refresh the role.)
- If you want the selected ARN to appear first in the list next time, select Remember ARN.

Your Organization Has No Administrator

Your organization might not have a designated service administrator. In this case, if none of your colleagues have set up a suitable role, you might have to create one yourself and then choose it.

- You can create the default role, called MediaLiveAccessRole. To first check if someone else has already created this role (only one person needs to create it for all users in your AWS account), look at Create role from template:
Channel Class

When you planned the workflow (p. 56), you decided whether to set up the channel as a standard channel (with two pipelines) or a single-pipeline channel. You must now specify the class in the channel configuration.

For Channel class, choose STANDARD or SINGLE_PIPELINE.

Standard Class

With this class, the channel contains two pipelines. The input for the channel has two entry points. The upstream system sends identical source streams to these two entry points, to provide content to two pipelines within the channel. MediaLive performs identical processing on both pipelines. For each output that you configure (for example, for both HLS output and RTMP output), the two pipelines deliver identical content to two destinations on the downstream system.

We strongly recommend that you set up all channels in as standard channels.

Single Pipeline Class

With this class, the channel contains one pipeline. For each output that you configure, the channel delivers content to one destination on the downstream system.

Input Specifications Settings

The Input Specifications settings include three fields that characterize the video in the input that you intend to use with this channel. The values in these fields are used to calculate the charges that you will incur on the input side. The values also ensure that MediaLive allocates sufficient processing resources when you run this channel. The fields are the following:

- Input codec
- Input resolution
- Maximum input bitrate

All the fields provide options that cover ranges, with the lowest range shown first and the highest shown last. Lower ranges imply lower processing requirements, and higher ranges imply higher requirements.

For each field, choose an option that meets or exceeds the requirements of your input. If your plan is for your channel to have more than one input, choose the option that meets or exceeds the most demanding of your inputs.

If you don't choose the correct option, MediaLive might not allocate sufficient processing resources. If you aren't sure about the processing requirements of your input, choose a higher option. For example, if you aren't sure of the bitrate and you are trying to choose between 10 Mbps and 20 Mbps, then choose
20 Mbps, to be on the safe side. Even with codecs, this advice applies. For example, if you aren’t sure if your input is AVC (H.264) or HEVC (H.265), then choose HEVC.

MediaLive uses these values for billing and resource allocation purposes: you pay for the option that you specify. For example, if you specify HD but the input is actually SD, you will be charged for HD.

MediaLive doesn’t use these values for determining what is actually in the video for decoding purposes. At ingest time, it still inspects the video to detect the source codec, resolution, and bitrate.

Step 2: Attach Inputs to the Channel

You must choose the input or inputs to attach to the channel, and then configure how AWS Elemental MediaLive will handle this input.

You can attach multiple inputs to the channel. For detailed information about setting up a channel with more than one input, see Input Switching (p. 189). There are specific rules about the number and type (push versus pull, for example) of inputs that you can attach to one channel.

To attach one input

1. On the Create channel page, for Input attachments, choose Add.
2. On the Attach input page, for Input, choose an existing input. More fields appear.
   Complete the fields that apply to your input. See the following sections:
   • the section called “Channel Input—HLS Pull Input” (p. 89)
   • the section called “Channel Input—MediaConnect Push Input” (p. 90)
   • the section called “Channel Input—MP4 Pull Input” (p. 90)
   • the section called “Channel Input—RTMP Push Input” (p. 90)
   • the section called “Channel Input—RTMP Pull Input” (p. 91)
   • the section called “Channel Input—RTP Push Input” (p. 91)
3. For Attachment name, enter a name for the attachment. The default name is the name of the input itself. You will need to change this default if you have already attached this input to the channel because names must be unique in the channel.
5. For information about completing the fields in the General input settings section, go to the next step (p. 94).

Topics
• Channel Input—HLS Pull Input (p. 89)
• Channel Input—MediaConnect Push Input (p. 90)
• Channel Input—MP4 Pull Input (p. 90)
• Channel Input—RTMP Push Input (p. 90)
• Channel Input—RTMP Pull Input (p. 91)
• Channel Input—RTP Push Input (p. 91)

Channel Input—HLS Pull Input

To verify that the input is set up correctly, look at the Input sources section. It shows the locations of the source video. You specified these locations when you created the input:
• If the channel is set up as a standard channel, you specified two locations.
• If the channel is set up as a single-pipeline channel, you specified one.

For example, for an HTTPS pull:

https://203.0.113.13/newschannel/anytownusa.m3u8 and https://203.0.113.54/newschannel/anytownusa.m3u8

Or, for an AWS Elemental MediaStore pull:

mediastoressl://eri39n.data.mediastore.us-west-2.amazonaws.com/premium/canada/mlaw.m3u8 and mediastoressl://eri39n.data.mediastore.us-west-2.amazonaws.com/redundant/premium/canada/mlaw.m3u8

Channel Input—MediaConnect Push Input

To verify that the input is set up correctly, look at the MediaConnect flows section. It shows the ARNs of the AWS Elemental MediaConnect flows that are the source for this input. These ARNs were automatically generated when you created the input:

• If the channel is set up as a standard channel, two ARNs are generated.
• If the channel is set up as a single-pipeline channel, one ARN is generated.

For example:

arn:aws:mediaconnect:us-west-1:111122223333:flow:1bgf67:sports_event_A and

Channel Input—MP4 Pull Input

To verify that the input is set up correctly, look at the Input destinations section. It shows the locations of the source video. You specified these locations when you created the input:

• If the channel is set up as a standard channel, you specified two locations.
• If the channel is set up as a single-pipeline channel, you specified one.

The format of the location depends on the type of upstream system:

• For an upstream system that uses HTTP or HTTPS, the location is an HTTP or HTTPS URL. For example:
  https://203.0.113.31/fillervideos/oceanwaves.mp4 and
  https://203.0.113.52/fillervideos/oceanwaves.mp4
• For a file that is stored on Amazon S3, the location is the bucket name and object for the file. For example:
  s3://fillervideos/main/oceanwaves.mp4 and
  s3://fillervideos/redundant/oceanwaves.mp4

Channel Input—RTMP Push Input

Follow these guidelines to verify that the input is set up correctly.
To verify the setup of the input

1. Look at the Input destinations section. It shows the two locations on MediaLive that the upstream system will push the source to when the channel is running. These locations were automatically generated when you created the input:
   - If the channel is set up as a standard channel, two locations are generated.
   - If the channel is set up as a single-pipeline channel, one location is generated.

Each location consists of an address portion that was automatically generated, appended by a folder that you specified when you created the input.

For example, for an RTMP Public push input:

rtmp://203.0.113.111:1935/movies/classic
rtmp://203.0.113.22:1935/movies/classic

For example, for an RTMP VPC push input:

rtmp://192.0.2.131:1935

2. Look again at the Input destinations section.
   - If the section has an Input security group with a number beside it, then the input is an RTMP Public input that has a MediaLive security group. The input is correctly set up and you can continue.
   - If the section has an Input security group without a number beside it, then the input is an RTMP Public input that is missing a MediaLive input security group. This input isn't correctly set up. Typically, this situation occurs if, for example, you have input A attached to input security group B and then you delete B. Input A is no longer useable. You must recreate the input and attach an input security group to it before you can associate it with a channel that you are creating.
   - If the section doesn't have an Input security group, then the input is an RTMP VPC push input. The input is correctly set up and you can continue.

Channel Input—RTMP Pull Input

To verify that the input is set up correctly, look at the Input destinations section. It shows the locations of the source video. You specified these locations when you created the input:

- If the channel is set up as a standard channel, you specified two locations.
- If the channel is set up as a single-pipeline channel, you specified one.

For example:

rtmp://203.0.113.20:1935/movies/classic
rtmp://203.0.113.88:1935/movies/classic

Channel Input—RTP Push Input

Follow these guidelines to verify that the input is set up correctly.
To verify the setup of the input

1. Look at the **Input destinations** section. It shows the two locations on MediaLive that the upstream system will push the source to when the channel is running. These locations were automatically generated when you created the input:
   - If the channel is set up as a standard channel, two locations are generated.
   - If the channel is set up as a single-pipeline channel, one location is generated.

   For example, for an RTP Public input:

   ```
   rtp://203.59.21.50:5000
   rtp://203.59.21.131:5000
   ```

   For example, for an RTP VPC input:

   ```
   rtp://10.99.39.87.30.100:5000
   rtp://192.0.2.123:5000
   ```

2. Look again at the **Input destinations** section.
   - If the section has an **Input security group** with a number beside it, then the input is an RTP Public input that has a MediaLive security group. The input is correctly set up and you can continue.
   - If the section has an **Input security group** without a number beside it, then the input is an RTP Public input that is missing a MediaLive input security group. This input isn't correctly set up. Typically, this situation occurs if, for example, you have input A attached to input security group B and then you delete B. Input A is no longer useable. You must recreate the input and attach an input security group to it before you can associate it with a channel that you are creating.
   - If the section doesn’t have an **Input security group**, then the input is an RTP VPC input. The input is correctly set up and you can continue.

**Step 3: Complete the Settings for Each Input**

As soon as you attach the input on the **Attach** input pages, the **Input attachment** section closes and the **General input settings** section appears. You must complete these fields to configure the input.

**To configure the input**

1. Complete the fields as required. See the topics links below. For details about a field, choose the **Info** link next to the field:
   - For most fields, the default values are sufficient.
   - However, if you want to include audio and captions in the outputs, you must complete the **Audio selectors** and **Caption selectors** sections; the defaults do not specify enough information.

2. If you are setting up the channel with multiple inputs, add more inputs to the channel. For guidelines about implementing input switching, see **Input Switching** (p. 189).

3. When ready, go to the **next step** (p. 94).

**Topics**
- **Input Settings—Network Input Settings** (p. 93)
- **Input Settings—Other Settings** (p. 93)
- **Input Settings—Video Selector** (p. 93)
Input Settings—Network Input Settings

Complete this section only if the input is HLS.

Input Settings—Other Settings

The fields that are not within the Network input settings section apply to all inputs.

Input Settings—Video Selector

This section lets you identify the video to extract from the input, and lets you enable the optional color space feature.

- In Video selector, choose Video selector. More fields appear.
- Selector settings: This field lets you identify the video to ingest.

With RTP input, this field is optional but highly recommended because the input might contain more than one video. If you don’t identify the video, MediaLive selects the first video it finds, which might result in undesired content, especially in a live streaming context. If you want to explicitly identify the video but you don’t know its program ID or PID, speak to the content provider.

With all other input types, this field is optional because the input only ever contains one video.

You can specify the video by specifying a program ID (which typically exists in an MPTS input) or by specifying a PID (which exists in both MPTS and SPTS inputs).

Keep in mind that there is no button to add more video selectors because you can extract only one video asset from the input.

- Color space and Color space usage: These fields let you configure the optional color space feature.

Input Settings—Audio Selectors

If you want to extract audio from the input, this section is required. You create one or more audio selectors to identify the audio to extract. Typically, you identify different languages from the input, but you could also extract different audio codecs (such as AAC and Dolby).

For each audio that you want to extract, choose Add audio selector. Complete the fields that appear to identify the location of the audio and to specify optional handling of the audio.

Input Settings—Caption Selectors

If you want to extract captions from the input or to specify an external file as the source of the captions, this section is required. You create one or more captions selectors to identify the captions to extract. Typically, you identify different languages in each selector, but you could also identify different captions formats.

For each captions item that you want to extract or include, choose the Add captions selector. For detailed information about setting up input for captions, see Working with Captions (p. 167), specifically the section called “Step 1: Create Captions Selectors in the Input” (p. 172).
Step 4: Complete the General Settings

AWS Elemental MediaLive has several settings that apply globally to all outputs. MediaLive also has features that are optional but that apply globally to all outputs if they are enabled.

These settings and features apply to all outputs. Therefore, they appear on the General settings page, rather than in individual output groups and outputs.

To complete the general settings

1. On the Create channel page, in the Channel section, choose General settings.
2. In the General channel settings section, set the global settings and optional features as needed. For information about each setting or feature, see the topics at the end of this procedure.
3. When you have finished working with these fields, go to the next step (p. 95).

Avail Blanking

Optional feature. You can set this to blank out the output video during ad avails. For more information, see SCTE-35 Message Processing (p. 202).

Avail Configuration

Optional feature. You can modify the way that MediaLive handles SCTE-35 ad avail messages, or you can keep the default behavior. For information about the default behavior and how to modify that behavior, see SCTE-35 Message Processing (p. 202).

Blackout Slate

Optional feature. You can black out the output video as specified by program metadata, if that metadata is present in the input. For more information, see SCTE-35 Message Processing (p. 202).

Global Configuration

Global configuration settings. In this section, complete the first three fields as appropriate. For details about each field, choose the Info link next to the field.

Global Configuration - Input Loss Behavior

Global configuration settings. The Input Loss Behavior fields change how MediaLive handles input loss.

The behavior that you configure here applies to all the inputs attached to the channel.

When MediaLive detects that the input has not arrived within the expected time, it repeats the previous frame for a configurable number of milliseconds (from zero to forever). When that time expires, it displays a black frame for a configurable number of milliseconds (from zero to forever). When that time expires, it switches to a specified slate or to a specified color. When input resumes, the normal ingest continues.

You can change this behavior: for Input loss behavior, choose Input Loss Behavior. The default values are shown in the fields that appear. Change the fields as needed.
Timecode Configuration

Global configuration settings. This section lets you specify the timecode for the output. This timecode does not have to be the same as the timecode that you specified for the input. The input timecode fields are used to notify MediaLive what timecode is present in the input, so that MediaLive can find it, and read it or apply it correctly. The output timecode fields are used to specify the timecode format that MediaLive must apply to the output.

Logging

Optional feature. You can enable logging of activity on this individual channel. For detailed information about this feature, see the section called “Monitoring Using Amazon CloudWatch Logs” (p. 128).

To enable logging, choose a log level other than **DISABLED**. The levels are listed from least to most verbose.

To disable logging, choose **DISABLED**.

Step 5: Create Output Groups

In this step, you create the output groups that you identified when you planned the channel (p. 55). AWS Elemental MediaLive supports different output types. For more information, see the section called “Supported Codecs for Outputs” (p. 240).

**Important**
Typically, you set up the MediaLive channel as a standard channel (p. 88). The procedures in this chapter assume that you have set up in this way. The procedures therefore refer to setting up an output group with two destinations.

If you will set up the channel as a single-pipeline channel, then your upstream system has only one source. The output group has only one destination.

Topics

- Creating an Archive Output Group (p. 95)
- Creating a Frame Capture Output Group (p. 98)
- Creating an HLS Output Group (p. 100)
- Creating a MediaPackage Output Group (p. 105)
- Creating an RTMP Output Group (p. 106)
- Creating a Microsoft Smooth Output Group (p. 106)
- Creating a UDP Output Group (p. 108)

Creating an Archive Output Group

Follow these steps if, when you were planning the channel (p. 55), you determined that you want to include an Archive output group.

To create an Archive output group

1. On the Create channel page, under Output groups, choose Add. The Add output group section appears.
2. Choose Archive, and then choose Confirm. More sections appear.
3. Complete the fields as described in the section called “Fields for the Archive Group” (p. 96).
4. After you enter all the information for one output group, you can create another output group if your plan requires it. Otherwise, go to the next step (p. 109).

**Topics**
- Fields for the Archive Group (p. 96)
- About Archive Locations and File Names (p. 96)
- Archive Examples (p. 97)

**Fields for the Archive Group**

You must provide information about the destination for each Archive output group. This destination information applies to all the outputs in the individual Archive output group.

**Archive Destinations**

For **Archive group destination A** and **Archive group destination B**, specify two destinations when the channel is set up as a standard channel (p. 88), or one destination when it is set up as a single-pipeline channel. The URL is one piece of the information that is used for the location and file names of the output file. For more information, see the section called “About Archive Locations and File Names” (p. 96).

Each destination is a bucket and object in an Amazon S3 account.

**Archive Settings**

- For **Name**, enter a name for the output group. For example, **Sports Game 10122017 ABR** or **tvchannel59**.
- For **Additional settings**, optionally complete the **Rollover Interval** field. This value is one piece of the information that is used for the location and file names (p. 96) of the media files.

**Archive Outputs**

This section contains fields that relate to the encoding of the video, audio, and captions in the output, and that relate to the packaging and delivery of the output.

- If you want more than one output in this output group, choose **Add output**. An **Output** line is added for each output. Setup of the individual outputs is described in the section called “Step 6: Create Outputs” (p. 109).
- For **Name modifier** for each output, enter a modifier, if appropriate. For uses for this field, see the section called “About Archive Locations and File Names” (p. 96).

**About Archive Locations and File Names**

The location of archive output files is controlled by several fields in the **Archive** output group and the individual outputs:

- The two **URL** fields in the **Archive group destinations** section. The URL consists of a **protocol** portion, a **path** portion and a **base filename** portion.

  For example, assume that the URL is `s3ssl://interviews/3series/Delivery/3633_WangXiuLan`.

  The protocol is always `s3ssl://`, to indicate that the destination is an Amazon S3 bucket.
• `interviews/3series/Delivery/` is the path. The path is required and consists of a bucket and folders, terminated by a slash.

`3633_WangXiuLan` is the base file name. Do not terminate the base file name with a slash.

• The **Name modifier** field in the **Archive outputs** section. Required. The string forms part of the file name.

• The **Extension** field in the **Archive outputs** section. The extension for the file name. Required only if you don't want to use the default (.ts).

• The **Rollover interval** field in the **Archive settings** section. Required. For example, **600** divides the output into separate files, each 600 seconds (10 minutes) long. Each file name includes a six-digit sequential counter, such as 000000, 000001, and so on.

Each time the rollover expires, MediaLive closes the current file on Amazon S3 and starts a new file. The current file is visible on Amazon S3 only after it has closed.

The values from these fields are put together to form the location:

```
protocol path base_filename name_modifier sequential_counter extension
```

For more information, see the examples (p. 97).

**Archive Examples**

These examples show how to set up the fields that relate to locations. They don't show how to set up other fields such as fields in the individual outputs.

**Example 1**

You want to create an archive of the streaming output from TV channel 59. You want to store the output in the S3 bucket named "channel59", and you want to break up the stream into 10-minute chunks.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Rollover interval</strong> field in <strong>Archive settings</strong> section</td>
<td><strong>600</strong></td>
</tr>
<tr>
<td><strong>URL in Archive group destination A section</strong></td>
<td><code>s3ssl://channel59/delivery/program</code></td>
</tr>
<tr>
<td><strong>URL in Archive group destination B section</strong></td>
<td><code>s3ssl://channel59/backup/program</code></td>
</tr>
<tr>
<td><strong>Name modifier in Archive outputs section</strong></td>
<td><code>$_dt$</code></td>
</tr>
<tr>
<td><strong>Extension in Archive outputs section</strong></td>
<td>Leave blank to use the default (.ts).</td>
</tr>
</tbody>
</table>

result: the output will be broken into files of 10 minutes (600 seconds) each. Each file will have a file name of **program**, plus the time that the channel started plus a counter (000000, 000001, and so on), plus the file name extension. For example:

• The first file will be `program_20171012T033162.000000.ts`. 
• The second file will be program_20171012T033162.000001.ts.

Each file will be stored in both s3ssl://channel59/delivery and s3ssl://channel59/backup.

A given file is not visible in Amazon S3 while it is being written. As soon as the rollover happens (or if the user stops the channel), MediaLive closes the current file. At that point, the file becomes visible.

**Example 2**

You want to create an archive of highlights from the curling game that are also being streamed (in a separate HLS output group). You want to create three outputs: one that has audio languages for Europe, one for audio languages for Asia, and one for audio languages for Africa. You want to store the outputs in the Amazon S3 bucket named "sports/highlights/curling". You want to break up the stream into five-minute chunks.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Rollover interval</strong> field in <strong>Archive settings</strong> section</td>
<td>300</td>
</tr>
<tr>
<td><strong>URL in Archive group destination A</strong> section</td>
<td>s3ssl://sports/delivery/highlights/curling/10312017 In this example, the 10312017 folder is set to match today's date.</td>
</tr>
<tr>
<td><strong>URL in Archive group destination B</strong> section</td>
<td>s3ssl://sports/backup/highlights/curling/10312017 Using &quot;delivery&quot; and &quot;backup&quot; as folder names is only an example.</td>
</tr>
<tr>
<td><strong>Name modifier in Archive outputs</strong> section</td>
<td>Choose Add output twice: two more Output lines are added to this section, for a total of three lines. In each line, enter a modifier: _audiogroup1, _audiogroup2, and _audiogroup3.</td>
</tr>
<tr>
<td><strong>Extension in Archive outputs</strong> section</td>
<td>Leave blank to use the default (.ts).</td>
</tr>
</tbody>
</table>

Result: three separate sets of files are created for each output. Each file has a file name of 10312017, plus the modifier, plus the sequence counter, plus the file name extension. For example:

• 10312017_audiogroup1.000000.ts, 10312017_audiogroup2.000000.ts, and 10312017_audiogroup3.000000.ts.
• 10312017_audiogroup1.000001.ts, 10312017_audiogroup2.000001.ts, and 10312017_audiogroup3.000001.ts.

Each file will be stored in both s3ssl://sports/delivery/highlights/curling and s3ssl://sports/backup/highlights/curling.

A given file is not visible in Amazon S3 while it is being written. As soon as the rollover happens (or if the user stops the channel), MediaLive closes the current file. At that point, the file becomes visible.

**Creating a Frame Capture Output Group**

Follow these steps if, when you were planning the channel (p. 55), you determined that you want to include a Frame Capture output group.
A frame capture output lets you capture the video as a series of files, with each file containing one still JPEG image. For example, the output might capture every 10th output frame. You save the files to an Amazon S3 bucket.

**To create a Frame Capture output group**

1. On the Create channel page, under Output groups, choose Add. The Add output group section appears.
2. Choose Frame capture, and then choose Confirm. More sections appear.
3. Complete the fields as described in the section called “Fields for the Frame Capture Group” (p. 99).
4. After you enter all the information for one output group, you can create another output group if your plan requires it. Otherwise, go to the next step (p. 109).

**Fields for the Frame Capture Group**

You must provide information about the destination for each Frame Capture output group.

**Frame Capture Destinations**

For Frame Capture group destination A and Archive group destination B, specify two destinations when the channel is set up as a standard channel (p. 88), or one destination when it is set up as a single-pipeline channel. The URL is one piece of the information that is used for the location and file names of the output file. For more information, see the section called “About Frame Capture Locations and File Names” (p. 99).

Each destination is a bucket and object in an Amazon S3 account.

**Archive Settings**

For Name, enter a name for the output group. The name must be unique in this channel. For example, Sports Game Frame Capture.

**Archive Outputs**

For Name modifier for each output, enter a modifier, if appropriate. For uses for this field, see the section called “About Frame Capture Locations and File Names” (p. 99).

**About Frame Capture Locations and File Names**

The location of frame capture output files is constructed from several parts. Some of these parts are derived from fields in the Frame capture output group and the individual outputs, and some are set automatically by MediaLive:

- The two URL fields in the Frame capture group destinations section. The URL consists of a protocol portion, a path portion, and a base filename portion.

  For example, assume that the URL is s3ssl://sports_delivery/highlights/20180820/curling_. These are the parts of that URL:

  - s3ssl:// is the protocol. The protocol is always s3ssl://, to indicate that the destination is an Amazon S3 bucket.
  - /sports_delivery/highlights/20180820/curling_ is the path and base filename portion.
sports_delivery/highlights/20180820/ is the path. The path is required and consists of a bucket and folders, terminated by a slash.

curling_ is the base file name. Do not terminate the base file name with a slash.

- The **Name modifier** field in the **Frame capture outputs** section. Optional. This modifier forms part of the output file name.
- A sequential counter, which is automatically set to a 5-digit number starting with 00001.
- An extension, which is always .jpg.

The values from these components are put together to form the location:

```
protocol + path + base_filename + name_modifier + sequential_counter + .extension
```

The protocol and path point to an Amazon S3 bucket.

The base filename, name modifier, sequential counter, and extension form the name of each file in the bucket.

For example, the path for the bucket for the output might be the following:

`s3ssl://sports_delivery/highlights/20180820/`

The first file for an output might be the following:

`curling_low_00001.jpg`

**Creating an HLS Output Group**

Follow these steps if, when you were planning the channel (p. 55), you determined that you want to include an HLS output group.

**To create an HLS output group**

1. On the **Create channel** page, in the **Output groups** section, choose **Add**. The content pane changes to show the **Add output group** section.
2. Choose **HLS**, and then choose **Confirm**. More sections appear.
3. Complete the fields as described in the section called “**Fields for the HLS Group**” (p. 100).
4. After you enter all the information for one output group, you can optionally create another output group. Otherwise, go to the **next step** (p. 109).

**Topics**

- **Fields for the HLS Group** (p. 100)
- **About HLS Group Destinations and File Names** (p. 102)
- **HLS Example** (p. 104)

**Fields for the HLS Group**

You must provide information about the destination and the structure and contents of the manifest for each HLS output group. This destination and manifest information applies to all the outputs in the individual HLS output group.
HLS Group Destinations

For the destination URLs, specify two destinations when the channel is set up as a standard channel (p. 88), or one destination when it is set up as a single-pipeline channel.

The URL is one piece of the information that is used for the destination and file names (p. 102) of the manifest and media files.

HLS Settings

- For **Name**, enter a name for the output group. For example, *Sports Game 10122017 ABR* or *tvchannel159*.
- For **CDN settings**, set the value to specify the type of connection that is being used to write to the destination URLs (specified in the section called "HLS Group Destinations" (p. 101)). The options are the following:
  - **Hls basic put**: To send to a content delivery network (CDN) that uses HTTP or HTTPS PUT. Or to send to an Amazon S3 bucket (`s3://` or `s3ssl://`).
  - **Hls media store**: To send to an MediaStore container (`mediastoressl://`).
  - **Hls akamai**: To send to an Akamai CDN (this always uses HTTP or HTTPS).
  - **Hls webdav**: To send to AWS Elemental MediaPackage. Or to send to a downstream system that uses HTTP WebDAV or HTTPS WebDAV.

When you select the CDN type, more fields appear, appropriate to the type of connection. For details about a field, choose the *Info* link next to the field.

The CDN is one piece of the information that is used for the destination and file names (p. 102) of the manifest and media files.

- Optionally change the value of **Input loss action**. This field applies only if you have set up the channel as a standard channel. It is ignored for a single-pipeline channel; no switching occurs.

If you’re sending output to AWS Elemental MediaPackage, set this field to match how you set the channel class (p. 88):

- If the channel is a standard channel (to support input redundancy on AWS Elemental MediaPackage), set this field to **PAUSE_OUTPUT**. With this setup, if MediaLive stops producing output on one pipeline, MediaPackage detects the lack of content on its current input and switches to the other input. Content loss is minimized. (If you set this field to **EMIT_OUTPUT**, MediaLive sends filler frames to MediaPackage. MediaPackage doesn’t consider filler frames to be lost content, and therefore doesn’t switch to its other input.)
- If the channel is a single-pipeline channel, set this field to **EMIT_OUTPUT**. In this way, if the pipeline fails in MediaLive then AWS Elemental MediaPackage continues delivering to its own downstream system (although the content will be filler frames). If you set this field to **PAUSE_OUTPUT**, AWS Elemental MediaPackage stops updating its endpoint, which might cause problems at the downstream system.

For other destinations, the appropriate value for this field depends on the behavior of the downstream system.

- Complete the **Caption language mappings** fields only if your plan is to include at least one embedded captions asset in the output in this output group. See the section called “HLS Manifests (Embedded Captions)” (p. 181).

HLS Outputs

This section contains fields that are related to the encoding of the video, audio, and captions in the output, and that are related to the packaging and delivery of the output.
If you want more than one output in this output group, choose Add output. An Output line is added for each output. Setup of the individual outputs is described in the section called “Step 6: Create Outputs” (p. 109).

For the Name modifier field for each output, enter a modifier, if appropriate. For uses for this field, see the section called “About HLS Group Destinations and File Names” (p. 102).

Location

Complete this section to specify the location and organization of the manifest and asset files at the publishing point. The fields in this section provide some of the information that is used for the destination and file names (p. 102) of the manifest and media files.

Manifests and Segments

Complete this section to change the default setup of the HLS manifest and the segmentation of outputs.

DRM

Complete this section only if you are setting up for DRM using a static key to encrypt the output. In Key provider settings, choose Static key, and then complete all the other fields as appropriate. For details about a field, choose the Info link next to the field.

In a static key setup, you enter an encryption key in this section (along with other configuration data) and then give that key to the other party (for example, by sending it in an email). A static key is not really a DRM solution and is not highly secure.

MediaLive supports only a static key as an encryption option. To use a DRM solution with a key provider, you must deliver the output to AWS Elemental MediaPackage (in other words, set up AWS Elemental MediaPackage as the destination for the output) and then encrypt the video using AWS Elemental MediaPackage. For more information, see the AWS Elemental MediaPackage User Guide.

Ad Markers

Complete this section if you want to include SCTE-35 ad messages in the output. See SCTE-35 Message Processing (p. 202) and specifically the section called “Enabling Decoration – HLS” (p. 209).

Captions

If your plan is to include at least one embedded captions asset in the output in this output group, then you can optionally set up the HLS manifest to include information about the captions languages. See the section called “HLS Manifests (Embedded Captions)” (p. 181).

ID3

You can use these fields to insert timed ID3 metadata into all the outputs in this output group. For detailed information, see the section called “Inserting ID3 Metadata When Creating the Channel” (p. 200).

About HLS Group Destinations and File Names

HLS output consists of a manifest, one rendition manifest for each output in the output group, and media files: one set of .ts files for each output, and optionally one or more captions files for each output.

For example, one manifest file called curling.m3u8, one rendition manifest called curling_high.m3u8, and many .ts files that contain the video and audio (each file contains one segment of a specified number of seconds) and three .vtt files for English, French, and Spanish WebVTT captions.

Several fields in the HLS group section and the individual outputs control the location of these files.
• The **CDN settings** fields in the **HLS settings** section.

The main field specifies the type of connection to the content delivery network (CDN), which is the downstream system that is the destination for the HLS output. For example, if the destination is AWS Elemental MediaPackage, the CDN is of type **hls webdav**. The other fields in this section provide connection details.

• The two **URL** fields in the **HLS group destinations** section.

The URL consists of a protocol portion, a path portion, and a base filename portion.

For example, assume that the URL is **https://sports/curling**.

The protocol portion is **https://**. The protocol is required and must be correct for the CDN that you specify. For example, **https://** is correct if the CDN type is **Hls basic put** or **Hls akamai** or **Hls webdav**:

- **http://** or **https://** if you select **Hls basic put** in the **CDN** field and you are sending to a CDN that uses HTTP or HTTPS PUT
- **s3://** or **s3ssl://** if you select **Hls basic put** in the **CDN** field and you are sending to an Amazon S3 bucket
- **mediastoressl://** if you select **Hls media store** in the **CDN** field
- **http://** or **https://** if you select **Hls akamai** in the **CDN** field
- **http://** or **https://** if you select **Hls webdav** in the **CDN** field and you are sending to a server using WebDAV or you are sending to AWS Elemental MediaPackage

The path portion is **sports/**. The path is required and consists of the folders, terminated by a slash. It identifies the location of the manifest and media files.

The base file name is **curling**. It is used in the manifest file names and media file names. The base file name is optional. If you omit it, MediaLive uses the name of the input as the base file name.

• The **Name modifier** field in the **HLS outputs** section.

Required only in output groups with more than one output. For example, **_high**. Used in the rendition manifest file names and in media file names.

Following from our example, the manifest file would be **curling**, and the rendition manifest files would be **curling_high** and **curling_low**. The media video files would be **curling_high.00001.ts**, **curling_high.00002.ts**, and so on for output 1, and **curling_medium.00001.ts**, **curling_medium.00002.ts**, and so on for output 2.

• The **Segment modifier** field in the **Output settings** section of each individual output.

Always optional. For example, **_high**. Used only in the media file names. Typically used instead of **Name modifier**, when you have only one output in the output group and you want a modifier in the media but not in the manifest.

Following from our example, the manifest file would be **curling**, and the rendition manifest file would be **curling**, and the media video files would be **curling_high.00001.ts** and **curling_high.00002.ts**.

• The **Base URL manifest** field and **Base URL** field in the **Location** section.

Optional. These fields are typically used only for non-standard manifests.

• The **Directory structure** field in the **Location** section.

Optional. Used only to create subdirectories for the media files. Creates one subdirectory for each output, and then creates sub-subdirectories according to the **Segments per subdirectory** field.

For example, the high-resolution media files will go in subdirectories with the same name as each rendition manifest: **curling_high** and **curling_low**. Inside each subdirectory there will be a sub-
subdirectory named 00001 (for the first set of media files), 00002 (for the next set of media files), and so on.

**HLS Example**

This example shows how to set up the fields that relate to destinations. They don’t show how to set up other fields such as fields in the individual outputs.

You want to stream the curling game to MediaPackage. You want to create three outputs: high, medium, and low bitrate.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CDN settings in HLS settings section</strong></td>
<td><strong>hls webdav</strong></td>
</tr>
<tr>
<td></td>
<td>This is the type of connection that MediaPackage uses. Change the other CDN fields that appear, or keep the defaults.</td>
</tr>
<tr>
<td><strong>Fields in Location section</strong></td>
<td>Keep the defaults in all the fields; these fields are not used in this example.</td>
</tr>
<tr>
<td><strong>URL in HLS group destination A section</strong></td>
<td>For example, <code>https://62e3c93793c034c.mediapackage.us-west-2.amazonaws.com/in/v1/9378dje8/channel</code>.</td>
</tr>
<tr>
<td></td>
<td>The URLs are the Input URLs from the channel in AWS Elemental MediaPackage. As discussed in the section called “Setting Up the Downstream System” (p. 63), the input in MediaPackage is identical to the output from AWS Elemental MediaLive. This input URL must already exist in MediaPackage.</td>
</tr>
<tr>
<td></td>
<td>Note that in MediaPackage, URLs always end in &quot;channel&quot;, so the base file name in MediaLive must be &quot;channel&quot;.</td>
</tr>
<tr>
<td><strong>Credentials in HLS group destination A section</strong></td>
<td>MediaPackage accepts only authenticated requests, so you must enter a user name and a password that is known to MediaPackage. For the password, enter the name of the password stored on the AWS Systems Manager Parameter Store. Don’t enter the password itself. For more information, see Access to Amazon EC2 Systems Manager Parameter Store.</td>
</tr>
<tr>
<td><strong>URL in HLS group destination B section</strong></td>
<td>For example, <code>https://60dei849783734c.mediapackage.us-west-2.amazonaws.com/in/v1/6da5ba717b357a/channel</code>.</td>
</tr>
<tr>
<td><strong>Credentials in HLS group destination B section</strong></td>
<td>Enter a user name and password as for the URL for destination A. The credentials are probably the same for both URLs, but they might not be.</td>
</tr>
</tbody>
</table>
Creating a MediaPackage Output Group

Follow these steps if, when you were planning the channel (p. 55), you determined that you want to include a MediaPackage output group.

To create a MediaPackage output group

1. On the Create channel page, in the Output groups section, choose Add. The content pane changes to show the Add output group section.
2. Choose MediaPackage, and then choose Confirm. More sections appear.
3. For MediaPackage channel ID, enter the channel ID for that channel. You made a note of this ID when you set up the channel (p. 65) on AWS Elemental MediaPackage.

MediaLive automatically sets up the output group as follows:

- The output is delivered to AWS Elemental MediaPackage using WebDAV. The output is always a live stream, not a VOD stream.
- For a MediaPackage output group, if input into MediaLive is lost then the behavior is for MediaLive to pause delivery. AWS Elemental MediaPackage expects this behavior and handles the loss by switching to the other input.
- The codec specification is RFC 4281. The playout device might use this information.
- SCTE-35 ad markers are always enabled. If you don’t want SCTE-35 markers, you will be able to remove them in the AWS Elemental MediaPackage channel. For information about SCTE-35 handling in a MediaPackage output, see SCTE-35 Message Processing (p. 202).
- The ability to insert ID3 markers through the output group is disabled. However, you can set up to pass through ID3 markers that are in the input, and you can insert ID3 markers using the MediaLive schedule. For information about ID3 handling in a MediaPackage output, see Working with ID3 Metadata (p. 198).
- The output name or names are automatically set to Output n, where n is an integer starting at 1.
- The name modifier for each output is automatically set to match the output name.
- The program date time (PDT) period is set to 1 second.
4. When you are ready, go to the next step (p. 109).
Creating an RTMP Output Group

Follow these steps if, when you were planning the channel (p. 55), you determined that you want to include an RTMP output group.

To create an RTMP output group

1. On the Create channel page, under Output groups, choose Add. The Add output group section appears.
2. Choose RTMP, and then choose Confirm. More sections appear.
3. Complete the fields as described in the section called “Fields for the RTMP Group” (p. 106).
4. After you enter all the information for one output group, you can create another output group if your plan requires it. Otherwise, go to the next step (p. 109).

Fields for the RTMP Group

You must provide information that applies to all the outputs in the individual RTMP output group.

- For Name, enter a name for the output group. For example, My Video Podcast.
- If you want to connect to the destination over RTMPs, for Authentication Scheme, specify the type of scheme. Typically, choose Common. Choose Akamai only if instructed to do so by the people responsible for the downstream system. If you connect over RTMP, the value in this field is ignored.
- In Additional Settings, optionally change the Cache length, Restart delay, and Cache full behavior. These fields relate to reconnecting to the RTMP server. For more information, see the section called “Reconnection Settings” (p. 114).
- In Additional settings, for Caption Data, complete this field only if at least one of your outputs includes captions with embedded as the source captions format and RTMPCaptionInfo as the output format. If there are no captions in any output, the value in this field is ignored. For details about a field, choose the Info link next to the field. For detailed information about setting up for captions, see Working with Captions (p. 167).
- The RTMP outputs section contains fields that are different for each output. Setup of the individual outputs is described in the section called “Step 6: Create Outputs” (p. 109).

If you want more than one output in this output group, choose Add output. An Output line is added for each output.

Creating a Microsoft Smooth Output Group

Follow these steps if, when you were planning the channel (p. 55), you determined that you want to include a Microsoft Smooth output group.

To create a Microsoft Smooth output group

1. On the Create channel page, in the Output groups section, choose Add. The Add output group section appears.
2. Choose Microsoft Smooth, and then choose Confirm. More sections appear.
3. Complete the fields as described in the section called “Fields for the Microsoft Smooth Group” (p. 107).
4. After you enter all the information for one output group, you can optionally create another output group. Otherwise, go to the next step (p. 109).
Fields for the Microsoft Smooth Group

You must provide information about the destination for all the outputs in each Microsoft Smooth output group.

You also must provide information about the destination for each Microsoft Smooth output group. This destination information applies to all the outputs in the individual Microsoft Smooth output group.

Microsoft Smooth Settings

- For **Name**, enter a name for the output group. For example, *Sports Game 10122017 ABR* or *tvchannel159*.
- If you want to change the default setup of the Microsoft Smooth manifest and fragments, choose **General configuration**. For details about a field, choose the **Info** link next to the field.
- If you want to change the configuration of the event information that is sent to the Microsoft IIS server, choose **Event configuration**. For details about a field, choose the **Info** link next to the field.
- If you want to change the default setup of the timecode and timestamp that are used in all the outputs in this output group, choose **Timecode configuration**. For details about a field, choose the **Info** link next to the field.
- If you want all the outputs in this output group to include the SCTE-35 messages that are already present in the input, choose **Sparse track**. The messages will be included in a sparse track. For details about a field, choose the **Info** link next to the field. For more information, see *SCTE-35 Message Processing* (p. 202).

Microsoft Smooth Group Destinations

For the destinations (A and B), specify two destinations when the channel is set up as a standard channel (p. 88), or one destination when it is set up as a single-pipeline channel. The URLs must use the HTTP or HTTPS protocol. Do not include the port.

Specify the path portion of the destination as */folders/basefilename*. The basefilename will be used as the first part of the file name of all the files for all outputs in this output group. Or specify it as */folders/*. In this case, the name of the input will be used for the file names.

Microsoft Smooth Outputs

The **Microsoft Smooth Outputs** section on the console contains fields that relate to the encoding of the video, audio, and captions in the output. The section also includes fields that relate to the packaging and delivery of the output:

- If you want more than one output in this output group, choose **Add output**. An **Output** line is added for each output. Setup of the individual outputs is described in the section called “Step 6: Create Outputs” (p. 109).

- For the **Name modifier** field for each output, enter a modifier, if appropriate. For uses for this field, see the examples (p. 107).

Microsoft Smooth Example

The following examples show how to set up the fields that relate to destination. They do not show how to set up other fields such as fields in the individual outputs.
You want to stream the curling game to an origin server that supports Microsoft Smooth. You want to create three outputs: high, medium, and low bitrate.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>URL in the <strong>Microsoft Smooth group destination A section</strong></td>
<td>For example, <a href="https://203.0.113.18/sports/curling/$d$">https://203.0.113.18/sports/curling/$d$</a>. For information about identifiers for variable data (such as $d$), see Reference: Identifiers for Variable Data (p. 241).</td>
</tr>
<tr>
<td>Credentials in the <strong>Microsoft Smooth group destination A section</strong></td>
<td>The Microsoft origin server accepts only authenticated requests, so you must enter a user name and a password that is known to that server. For the password, enter the name of the password stored on the AWS Systems Manager Parameter Store. Don't enter the password itself. For more information, see Access to Amazon EC2 Systems Manager Parameter Store.</td>
</tr>
<tr>
<td>URL in the <strong>Microsoft Smooth group destination B section</strong></td>
<td>For example, <a href="https://203.0.113.77/sports/curling/$d$">https://203.0.113.77/sports/curling/$d$</a>.</td>
</tr>
<tr>
<td>Credentials in the <strong>Microsoft Smooth group destination B section</strong></td>
<td>Enter a user name and password, as for the URL for destination A. The credentials are probably the same for both URLs, but they might not be.</td>
</tr>
<tr>
<td>Name modifier in the <strong>Microsoft Smooth outputs section</strong></td>
<td>Leave blank; in this example, a name modifier is not used.</td>
</tr>
</tbody>
</table>

Result: If the channel is run on November 30, 2017, files are created with the following names:

- One master manifest: 20171130.isml
- A set of video, audio, and captions for output 1: 20171130_high.ismv, 20171130_high.isma, 20171130_high.ismt
- A set of video, audio, and captions for output 2: 20171130_medium.ismv, 20171130_medium.isma, 20171130_medium.ismt
- A set of video, audio, and captions for output 3: 20171130_low.ismv, 20171130_low.isma, 20171130_low.ismt

The files will be published to both destinations: https://203.0.113.18/sports/curling/$d$ and https://203.0.113.77/sports/curling/$d$.

**Creating a UDP Output Group**

Follow these steps if, when you were planning the channel (p. 55), you determined that you want to include a UDP output group.

**To create a UDP output group**

1. On the **Create channel** page, in the **Output groups** section, choose **Add**. The **Add output group** section appears.
2. Choose **UDP**, and then choose **Confirm**. More sections appear.
3. Complete the fields in each section (p. 109).
4. After you enter all the information for one output group, you can optionally create another output group. Otherwise, go to the next step (p. 109).

**UDP Settings**

- Enter a name for the output group. For example, *Sports Game 10122017 ABR* or *tvchannel159*.
- Optionally change the value of *Input loss action*.
- Optionally complete the *ID3* fields. You use these fields to insert timed ID3 metadata into all the outputs in this output group. For detailed information, see the section called “Inserting ID3 Metadata When Creating the Channel” (p. 200).

**UDP Destinations**

For the URLs for the destinations (A and B), specify two destinations when the channel is set up as a standard channel (p. 88), or one destination when it is set up as a single-pipeline channel. The URLs must use the RTP or UDP protocol and must include a port number.

If FEC is enabled (this field is in the *Output* pane, not the *Output group* pane), then leave space between the port numbers for the two destinations. For example, if one destination is `rtp://203.0.113.28:5000`, assume that FEC also uses port 5002 and 5004. So the lowest possible port number for the other destination is 5005: `rtp://203.0.113.33:5005`.

**UDP Outputs**

This section contains fields that relate to the encoding of the video, audio, and captions in the output, and that relate to the packaging and delivery of the output.

If you want more than one output in this output group, choose *Add output*. An *Output* line is added for each output. Setup of the individual outputs is described in the section called “Step 6: Create Outputs” (p. 109).

**Step 6: Create Outputs**

Every output group contains one or more outputs. Each output contains the individual video, audio, and captions encodes and configuration information for the type of output. For example, it contains configuration information that is specific to an HLS output.

This section describes how to create and configure the outputs and assumes that you have created the output group (p. 95).

**To create an output in any output group**

1. On the *Create channel* page, in the *Output groups* section, the *Output* section by default has one output. If you need to create more outputs (as determined when you planned the channel (p. 60)), then choose *Add Output* as many times as required. A new output line is added.
2. Choose the *Settings* link beside the first output line. The content pane shows the fields for that output.
   - The pane for all output types shows two sections: *Output settings* and *Stream settings*.
3. Complete both the *Output settings* and *Stream settings* for the output type: the section called “Settings for an Archive Output” (p. 110), the section called “Settings for an HLS
Settings for an Archive Output

On the **Output** page for an Archive output group, you provide information about the container for this output, and the video, audio, and captions encodes to create in that container.

**Output Settings Section**

- For **Name modifier**, enter a name as described in the section called “Archive Destinations” (p. 96).
- Leave **Extension** blank; it is always set to **m2ts**.
- In the fields under **Container Settings**, optionally change any values. For details about a field, choose the **Info** link next to the field.
- In the fields under **PID settings**, optionally change any values. For details about a field, choose the **Info** link next to the field.

**Streams Section**

In **Stream settings**, decide if you need to create more encodes for this output, based on the workflow that you planned (p. 60). By default, each output is set up with one video encode and one audio encode. Choose the appropriate **Add** button or **Delete** button to set up the output with the encodes—video, audio, and captions—that you planned for this output.

For example, in one output you might want one video asset and two audio assets, in another output you might want one captions asset for French captions, and in a third output you might want one captions asset for Spanish captions. For information about why you would set up like this (for example, with an output that contains only one captions asset), see the section called “Examples of Channel Designs” (p. 61).

For information about the fields in each type of encode, see the section called “Step 7: Set Up the Video Encode” (p. 116), the section called “Step 8: Set Up the Audio Encodes” (p. 118), and the section called “Step 9: Set up the Captions Encodes” (p. 118).

**Settings for a Frame Capture Output**

On the **Output** page for a Frame Capture output group, you provide information about the still images to create.
Output Settings Section

For Name modifier, enter a name as described in the section called “Frame Capture Destinations” (p. 99).

Streams Section

For information about the fields in the video encode, see the section called “Step 7: Set Up the Video Encode” (p. 116).

Settings for an HLS Output

In the Output page for an HLS output group, you provide information about the transport for this output and the video, audio, and captions encodes to create in the output.

Output Settings Section

- For Name modifier and Segment modifier, complete the fields as described in the section called “HLS Group Destinations” (p. 101).
- Set the HLS Settings field as appropriate:
  - If this output is a regular output with video and audio encodes, choose Standard HLS. More fields appear. Keep the default for Audio rendition sets (the value is ignored), and complete the fields under M3U8 as needed or keep the default value, which often is a good choice. For details about a field, choose the Info link next to the field.
  - If this output is an output that contains only a video encode and the output is part of an output group that includes an audio rendition group, choose Standard HLS. More fields appear. Complete Audio rendition sets as required. Complete the fields under M3U8 as needed or keep the default value, which often is a good choice. For details about a field, choose the Info link next to the field.
  - If this output is an output that contains only an audio encode and the output is part of an output group that includes an audio rendition group or is part of an output group that has only audio encodes (no video at all), choose Audio only HLS. A special set of fields appears for configuring the audio. For details about a field, choose the Info link next to the field.

Streams Section

In Stream settings, decide if you need to create more encodes for this output, based on the workflow that you planned (p. 60). By default, each output is set up with one video encode and one audio encode. Choose the appropriate Add button or Delete button to set up the output with the encodes—video, audio, and captions—that you planned for this output.

For example, in one output you might want one video asset and two audio assets, in another output you might want one captions asset for French captions, and in a third output you might want one captions asset for Spanish captions. For information on why you would set up like this (for example, with an output that contains only one captions asset), see the section called “Examples of Channel Designs” (p. 61).

For information about the fields in each type of encode, see the section called “Step 7: Set Up the Video Encode” (p. 116), the section called “Step 8: Set Up the Audio Encodes” (p. 118), and the section called “Step 9: Set up the Captions Encodes” (p. 118).

Settings for a MediaPackage Output

In the Output page for a MediaPackage output group, you provide information about the video, audio, and captions encodes to create in the output.
Output Settings Section

There are no fields in the Output page for a MediaPackage output group.

MediaLive automatically sets up the output as follows:

- The output name or names are automatically set to Output n, where n is an integer starting at 1.
- The name modifier for each output is automatically set to match the output name.
- Passthrough of SCTE-35 messages is enabled. For information about SCTE-35 handling in a MediaPackage output, see SCTE-35 Message Processing (p. 202).
- ID3 metadata is enabled. For information about ID3 handling in a MediaPackage output, see Working with ID3 Metadata (p. 198).
- The PAT interval is set to 0, which means a single PAT is inserted at the beginning of each segment.
- The PMT interval is set to 0, which means a single PMT is inserted at the beginning of each segment.

Streams Section

In Stream settings, decide if you need to create more encodes for this output, based on the workflow that you planned (p. 60). By default, each output is set up with one video encode and one audio encode. Choose the appropriate Add button or Delete button to set up the output with the encodes—video, audio, and captions—that you planned for this output.

For example, in one output you might want one video asset and two audio assets, in another output you might want one captions asset for French captions, and in a third output you might want one captions asset for Spanish captions. For information about why you would set up like this (for example, with an output that contains only one captions asset), see the section called “Examples of Channel Designs” (p. 61).

Packaging of Video Encodes and Audio-only Encodes

MediaLive handles the packaging of encodes within each output as follows:

- If an output contains both video and audio (and optionally captions), the audio rendition is marked as "program audio".
- If an output doesn't contain video, the audio rendition is marked as "audio only" and each audio encode is marked as ALTERNATE_AUDIO_NOT_AUTO_SELECT.

Setting the Width and Height of the Video

This section refers to the fields in Stream settings, Video.

You must specify values in Width and Height. The MediaPackage output group doesn't support leaving this fields blank in order to use the width and height from the source video.

Setting the Aspect Ratio of the Video

This section refers to the fields in Stream settings, Video, Aspect ratio.

You must set PAR control to SPECIFIED. The MediaPackage output group doesn't support setting the aspect ratio of the output to follow the source video. When you choose SPECIFIED, you must complete PAR numerator and PAR denominator. You can set the AFD fields as you want.

Setting the Frame Rate of the Video

This section refers to the fields in Stream settings, Video, Frame rate.
You must set **Framerate control** to **SPECIFIED**. The MediaPackage output group doesn't support setting the frame rate of the output to follow the source video. When you choose **SPECIFIED**, you must complete **Framerate numerator** and **Framerate denominator**. You can set the scan type as you want; it doesn’t relate directly to the frame rate.

**Setting Up for GOPs and Segments**

This section refers to the fields in **Stream settings, Video, GOP structure**.

For the video, you must set the GOP size to ensure that the output from AWS Elemental MediaPackage has a segment size that is close to the segment size that you specify in AWS Elemental MediaPackage. MediaLive and AWS Elemental MediaPackage work together to obtain a final segment size. The logic is as follows:

- In MediaLive, you specify the **GOP size** and **GOP size units** fields.
- MediaLive calculates the GOP duration, taking into account the frame rate that you specify in the **Video** section of the **Output** page.
- In AWS Elemental MediaPackage, you specify the segment duration. You always specify a whole number. This segment duration is the **desired** minimum duration.
- When MediaPackage receives the video from MediaLive, it determines how much it must adjust the segment duration to fit a whole number of GOPs into the segment. The segment duration can only be adjusted up, never down. This adjusted segment duration appears in the manifest that AWS Elemental MediaPackage produces.

**Example 1:**

Assume that in MediaLive, you set the GOP size to 60 frames. You set the frame rate to 29.97. These two values result in a GOP duration of 2.002 seconds.

Assume that in MediaPackage, you set the segment duration to 6 seconds. This segment duration is the **desired** minimum duration.

When MediaPackage receives the video from MediaLive, it determines how much it must adjust the segment duration to fit a whole number of GOPs into the segment. In this case, the segment duration must be adjusted to 6.006 seconds (3 GOPs, where each GOP is 2.002 seconds long).

**Example 2:**

Assume that in MediaLive, you set the GOP size to 90 frames. You set the frame rate to 30. These two values result in a GOP duration of 3 seconds.

Assume that in MediaPackage, you set the segment duration to 4 seconds. This segment duration is the **desired** minimum duration.

When MediaPackage receives the video from MediaLive, it determines how much it must adjust the segment duration to fit a whole number of GOPs into the segment. In this case, the segment duration must be adjusted to 6 seconds (2 GOPs, where each GOP is 3 seconds long).

**Other Encode Fields**

For information about the fields in each type of encode, see the section called **“Step 7: Set Up the Video Encode”** (p. 116), the section called **“Step 8: Set Up the Audio Encodes”** (p. 118), and the section called **“Step 9: Set up the Captions Encodes”** (p. 118).

**Settings for an RTMP Output**

In each **Output** page for an RTMP output group, you provide information about the packaging and delivery of the output, and about the video, audio, and captions encodes to create in the output.
RTMP Destination Section

For RTMP destination A and RTMP destination B, specify two destinations when the channel is set up as a standard channel (p. 88), or one destination when it is set up as a single-pipeline channel.

Complete each destination section as follows:

- For URL, enter some of the information that you obtained when you set up the downstream system (p. 63). Enter the URL in this format:
  \[<protocol>://<IP address or domain>:<port>/<application name>\]
  
  For example:
  \[rtmp://203.0.113.28:1935/live\]

- For Stream, enter the stream name (also known as the application instance or stream key). For example, MyStream or 03a3bb838e0a.

- If the server requires that you authenticate with a user name and password, complete the Credentials section. Make sure that you have obtained the user name and password from the RTMP server and that you have already set up the user name and password in the Parameter Store. For more information, see Access to Amazon EC2 Systems Manager Parameter Store.

Note that these credentials relate to user authentication, not to the protocol. User authentication is about whether the RTMP server will accept your request. The protocol is about whether the request is sent over a secure connection.

Output Settings Section

- For Connection retry interval and Num retries, see the section called “Reconnection Settings” (p. 114).
- If you want to connect to the destination over RTMPS, then for Certificate Mode, choose the option that is required by the RTMP server (the downstream system). If you connect over RTMP, the value in this field is ignored.

Reconnection Settings

There are several fields that control how MediaLive behaves if the connection to the RTMP server seems to drop:

- Cache length (on the output group page) specifies how long to hold the output in memory, waiting for the RTMP server to respond.

  When that time expires, Cache full behavior (on the output group page) specifies whether to disconnect immediately or wait 5 minutes.

- If MediaLive disconnects, then Restart delay (on the output group page) specifies how long to wait before trying to reconnect.

  When MediaLive tries to reconnect, Connection retry interval (on the output page) specifies how often to retry. Num retries (on the output page) specifies how many times to retry. When the retries expire, this output stops. The channel stops running only if all outputs lose their connections.

Stream Settings Section

In the Stream settings section, decide if you need to create more encodes for this output, based on the workflow that you planned (p. 60). By default, each output is set up with one video encode and
Settings for a Microsoft Smooth Output

On the **Output** page for a Microsoft Smooth output group, you provide information about the video, audio, and captions encodes to create in the output.

**Output Settings Section**

For **Name modifier**, complete the field as described in the section called “Microsoft Smooth Group Destinations” (p. 107).

**Streams Section**

In the **Stream settings** section, decide if you need to create more encodes for this output, based on the workflow that you planned (p. 60). By default, each output is set up with one video encode and one audio encode. Choose the appropriate **Add** button or **Delete** button to set up the output with the encodes—video, audio, and captions—that you planned for this output.

For example, in one output you might want one video asset and two audio assets, in another output you might want one captions asset for French captions, and in a third output you might want one captions asset for Spanish captions. For information about why you would set up like this (for example, with an output that contains only one captions asset), see the section called “Examples of Channel Designs” (p. 61).

For information about the fields in each type of encode, see the section called “Step 7: Set Up the Video Encode” (p. 116), the section called “Step 8: Set Up the Audio Encodes” (p. 118), and the section called “Step 9: Set up the Captions Encodes” (p. 118).

Settings for a UDP Output

On the **Output** page for a UDP output group, you provide information about the individual output container and the video, audio, and captions encodes to create in that container.

**Output Settings Section**

- For **FEC output settings**, choose a value.
- In the fields under **Network settings**, change any values as needed. For details about a field, choose the **Info** link next to the field.
- In the fields under **PID settings**, change any values as appropriate. For details about a field, choose the **Info** link next to the field.

**Streams Section**

- In **Stream settings**, decide if you need to create more encodes for this output, based on the workflow that you planned (p. 60). By default, each output is set up with one video encode and one audio encode.
• Choose the appropriate **Add** button or **Delete** button to set up the output with the encodes—video, audio, and captions—that you planned for this output.

For example, in one output you might want one video and two audio assets, in another output you might want one captions asset for French captions, and in a third output you might want one captions asset for Spanish captions.

For information about the fields in each type of encode, see the section called “Step 7: Set Up the Video Encode” (p. 116), the section called “Step 8: Set Up the Audio Encodes” (p. 118), and the section called “Step 9: Set up the Captions Encodes” (p. 118).

---

**Step 7: Set Up the Video Encode**

The output section for every type of output group contains a **Stream settings** section. In this section, you create **encodes** (p. 53) for the video, audio (if applicable), and captions (if applicable) in the output and specify the details of how you want these assets encoded.

The following procedure describes how to set up a video encode and assumes that you have created the output (p. 109) that will hold the video. The fields for a video encode are identical for all output group types. For example, the fields for video for an HLS group are identical to those for a UDP group.

**To set up the video encodes in most output types**

1. On the **Create channel** page, find the output group that you **created** (p. 95).
2. Under that output group, find the output (or the first output) where you want to set up a video encode.
3. Choose the link for the video encode.
4. For **Codec settings**, choose the codec to use for this output. More fields appear.
5. Complete each field as appropriate. For details about a field, choose the **Info** link next to the field.
6. Continue setting up the **audio encodes** (p. 118), video encodes, and **captions encodes** (p. 118) for all outputs in all output groups. When you finish with the encodes for all outputs, go to **save the channel** (p. 119).

**To set up the video encodes in the Frame Capture output type**

1. On the **Create channel** page, find the output group that you **created** (p. 95).
2. Under that output group, find the output and choose the link for the video encode.
3. Complete each field as appropriate. For details about a field, choose the **Info** link next to the field.
4. When you are ready, go to **save the channel** (p. 119).

---

**Setting the Rate Control Mode**

This feature does not apply to the video in a Frame Capture output.

The **Rate control** section in the **Video** section of each output on the console lets you control the quality and bitrate of the video.

When encoding visually complex video (such as high-motion sports events with brightly dressed crowds in the background), there is always a trade-off between high video quality and low bitrate. Higher video quality requires higher bitrate. There is less trade-off with visually simple video such as cartoons.
AWS Elemental MediaLive offers several options that provide different balances of video quality versus bitrate.

To set the rate control mode and bitrate for the output

1. On the Stream settings pane, for Video, for Codec settings, choose H264.
2. In the Rate Control section, for Rate control mode, choose QVBR or CBR or VBR. For information about choosing the best option, see the sections below.

   - If you chose QVBR, complete Max bitrate and Quality level.
   - If you chose VBR, complete Bitrate (average bitrate) and Max bitrate.
   - If you chose CBR, complete Bitrate.

Quality-defined Variable Bitrate Mode (QVBR)

With quality-defined variable bitrate mode (QVBR), you specify a maximum bitrate and a quality level. Video quality will match the specified quality level except when it is constrained by the maximum bitrate; this constraint occurs when the video is very complex so that it is not possible to reach the quality level without exceeding the maximum bitrate.

We recommend this mode if you or your viewers pay for bandwidth, for example, if you are delivering to a CDN such as Amazon CloudFront or if your viewing users are on mobile networks.

Values to use: When choosing QVBR, you should set the quality level and maximum bitrate for your most important viewing devices. Set the buffer size to twice the maximum bitrate, and set the initial buffer to 90%.

<table>
<thead>
<tr>
<th>Viewing Device</th>
<th>Quality Level</th>
<th>Max Bitrate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Primary Screen</td>
<td>8 to 10</td>
<td>4,000,000 to 6,000,000</td>
</tr>
<tr>
<td>PC or Tablet</td>
<td>7</td>
<td>1,500,000 to 3,000,000</td>
</tr>
<tr>
<td>Smartphone</td>
<td>6</td>
<td>1,000,000 to 1,500,000</td>
</tr>
</tbody>
</table>

How it works: The bitrate can change with each frame (in order to obtain at least the specified quality), but it can't exceed the maximum bitrate. The encoder does not attempt to maintain an average bitrate. It always reaches the maximum bitrate if that is necessary to obtain the specified quality. On the other hand, if the quality can be obtained with lower bitrates, the encoder doesn't use a higher bitrate.

Variable Bitrate Mode (VBR)

With variable bitrate mode (VBR), you specify an average bitrate and a maximum bitrate. Video quality and bitrate vary, depending on the video complexity.

Choose VBR instead of QVBR if you want to maintain a specific average bitrate over the duration of the channel. If bitrate does not need to be constrained, then consider using QVBR.

Values to use: When choosing VBR, you should try to assess the expected complexity of the video, and set a suitable average bitrate. Set the maximum bitrate to accommodate expected spikes. Set the buffer size to twice the maximum bitrate, and set the initial buffer to 90%.

How it works: The bitrate can change with each frame (in order to obtain the best quality) but it can't exceed the specified maximum bitrate. The encoder also ensures that as the channel progresses, the stream meets the specified average bitrate. This mode is useful when you expect short spikes in the
Step 8: Set Up the Audio Encodes

The output section for every type of output group (Archive, Frame Capture, HLS, Microsoft Smooth, and UDP) contains a Stream settings section. In this section, you create encodes (p. 53) for the video, audio, and captions in the output and specify the details of how you want these assets encoded.

The following procedure describes how to set up an audio encode and assumes that you have created the output (p. 109) that will hold the audio. The configuration options for an audio encode are identical for all output group types.

Note that a Frame Capture output group contains video but doesn't contain audio or captions.

**To set up the audio encodes in the outputs**

1. On the Create channel page, find the output group that you created (p. 95).
2. Under that output group, find the output (or the first output) where you want to set up the audio encode.
3. Choose the link for one of the audio encodes. (You might have created more than one encode.)
4. For Codec settings, choose the codec to use to encode this audio asset. The remaining fields change to match this codec.
5. Complete each field as appropriate. For details about a field, choose the Info link next to the field.
6. Optionally complete the fields in the Remix settings section, or keep the defaults (to omit remixing).
7. Optionally complete the fields in the Audio normalization settings section, or keep the defaults (to omit normalization).
8. Repeat for each audio encode in this output, if any.
9. Continue setting up the audio encodes, video encodes (p. 116), and captions encodes (p. 118) for all outputs in all output groups. When you have finished with the encodes for all outputs, go to save the channel (p. 119).

Step 9: Set up the Captions Encodes

The output section for every type of output group (Archive, HLS, MediaPackage, Frame Capture, Microsoft Smooth, and UDP) contains a Stream settings section. In this section, you create
encodes (p. 53) for the video, audio, and captions in the output and specify the details of how you want these assets encoded.

Note that a Frame Capture output group contains video but doesn't contain audio or captions.

Before setting up captions, see Reference: Supported Captions (p. 229). There are rules about the following:

- What captions formats can be generated, depending on the format of the input captions and container. For example, ARIB can be generated only from ARIB input captions, which can appear only in a TS input.
- What captions type (embedded, object, sidecar) applies to which captions format. For example, DVB-Sub is an object type of format.
- Which format can be included in which type of output asset. For example, HLS can take WebVTT (a sidecar type), 608 (embedded), and burn-in.

To set up the captions encodes in the outputs
1. For detailed information about setting up captions, see Working with Captions (p. 167).
2. After you set up the captions for one output, continue setting up the audio encodes (p. 118), video encodes (p. 116), and captions encodes for all outputs in all output groups. When you have finished with the encodes for all outputs, go to save the channel (p. 119).

Step 10: Save the Channel

To save (create) the channel, choose Create channel in the navigation pane.

You can save the channel only after you have configured and created everything that you require. As soon as you save the channel, MediaLive validates the configuration of the channel and displays messages for any errors. You can't save a draft of the channel, and you can't save a channel that contains error messages.

To find your newly created channel, in the navigation pane, choose Channel. (The navigation pane might be collapsed. To open it, choose the menu icon in the upper-left corner of the console).

The Channel pane appears and shows the newly created channel in the list of channels. The state changes to Creating, and then to Ready.
Creating a Channel from a Template or by Cloning

A channel contains the details that instruct AWS Elemental MediaLive how to transcode (decode and encode) and package your input into specific outputs.

To create a channel, you provide details about inputs, about one or more output groups and their destinations, about the outputs in each output group, and about the video, audio, and caption encodes in each output.

There are three ways to create a channel:

- **From scratch.** The Create form on the MediaLive console contains some fields that display system defaults and other fields that are empty. You can create a channel from scratch by modifying the system defaults and by completing the appropriate empty fields. For more information, see *Creating a Channel from Scratch* (p. 86).

- **Using a built-in template or custom template.** You can use a template to create a channel, and reuse the template to create more channels. For more information, see the section called “Creating a Channel from a Template” (p. 121).

- **By cloning an existing channel.** You can clone an existing channel, and then edit the settings for the new (cloned) channel. For more information, see the section called “Creating a Channel by Cloning” (p. 122).

The procedures in the following topics show how to create a channel by using a template or by cloning. Before you use the procedures, you should understand how to create a channel from scratch. For more information, see *Creating a Channel from Scratch* (p. 86).

Topics
- About Templates (p. 120)
- About Cloning (p. 121)
- Creating a Channel from a Template (p. 121)
- Creating a Channel by Cloning (p. 122)
- Creating a Custom Template (p. 122)

About Templates

Using Built-in Templates

MediaLive includes built-in templates that you can access on the console. Each template includes data for output groups and outputs, and most importantly, data for encoding video to meet specific use cases (as specified in the template description).

When you use a built-in template, all sections of the Create channel page are populated with data except for the inputs and output destinations sections.

Even though the templates are built-in, you can choose to edit the existing fields and complete the empty fields.
Using Custom Templates

You or another person in your organization may have created custom templates. A custom template might contain nearly all the data that is required to create a complete channel, or it might contain only portions of the data. To create a custom template, see the section called “Creating a Custom Template” (p. 122).

Typically, templates are created in order to be shared among different users.

If your organization uses templates, you must obtain the templates you will use from the person who created the templates. You must store them in a folder on the computer where you are working on the MediaLive console. This folder is the "custom template location." You perform this task in your computer's filesystem, outside of MediaLive.

When you use a custom template, MediaLive populates all sections of the Create channel page are populated with data from the template, except for the input data. Even if the template includes input data, that data will not be pulled into the Create channel page.

You can edit the existing fields and complete the empty fields as needed.

About Cloning

Cloning lets you use an existing channel as the basis for a new channel.

When you clone an existing channel, all sections of the Create channel page are populated with the data from the cloned channel, except for the input data. Input data is always left blank.

You can edit the existing fields and complete the empty fields as needed.

Creating a Channel from a Template

You can create a channel by using a custom template or by using one of the built-in templates that MediaLive provides.

To create a channel from a template (console)

1. If you plan to use a custom template, make sure you have set up to use them. See the section called “Using Custom Templates” (p. 121).
3. In the navigation pane, choose Channels.
4. On the Channels page, choose Create channel.
5. On the Create channel page, in the Channel and input details section, in the Channel template section, do one of the following:
   - To use a built-in template: For Template, from the Channel templates section of the drop-down list, choose a template. (The Existing channels section does not list templates.)
   - To use a custom template: Choose Select custom template. Navigate to the "custom template" folder and choose the template. For information on the custom template location, see the section called “Using Custom Templates” (p. 121).
6. Complete the fields, such as the input fields, that must always be completed. You can also edit other fields as needed.
Creating a Channel by Cloning

You can clone a channel that is in the Channels list. (You can also clone a channel after choosing Create channel; for more information, see the section called “Creating a Channel from a Template” (p. 121).)

To create a channel by cloning (console)

2. In the navigation pane, choose Channels.
3. On the Channels page, choose the radio button next to the channel name.
4. Choose Clone.

The Create channel page appears. It replicates all the data from the base channel except for the input sections, which are always empty.
5. Give the channel a new name and complete the input sections. Change other fields as needed.

Creating a Custom Template

You create a custom template by exporting the data from an existing (and therefore validated) channel. MediaLive exports the data to a JSON file that you can use on the console.

To create a custom template (console)

2. In the navigation pane, choose Channels.
3. On the Channels page, choose the channel name (not the radio button).
4. On the details page, choose Create custom template. Follow the prompts to save the channel as a template. The template is a JSON file with the same name as the channel.
5. (Optional) Open the file in a suitable editor and make changes. For example, you can change field values, add fields, and remove fields. Be careful to maintain valid JSON.
6. Make the custom template available to the users who will need them. Each user must store the template in a folder that is accessible from the computer where the user will work on the MediaLive console. This task is performed outside of MediaLive.

Users of MediaLive can use the template file on the console.
Starting, Stopping, and Pausing an AWS Elemental MediaLive Channel

After you create a channel, you can start it. The channel never starts automatically except when it is already running and attempts to recover from a failure.

You can stop a running channel at any time.

You can also pause one or both the pipelines in a channel by adding a Pause action to the schedule for the channel. For more information, see the section called “Types of Schedule Actions for Pause” (p. 138).

For information about charges when a channel is running, see the section called “Pricing” (p. 6).

To start a channel

2. In the navigation pane, choose Channels, and then on the Channels page, choose the channel that you want to start.
3. Choose Start. The channel state changes to one of the following:
   • Starting
   • Running (encoding on both destinations)
4. Choose the channel name. The details for the channel appear.

To stop a channel

2. In the navigation pane, choose Channels, and then on the Channels page, choose the channel that you want to stop.
3. Choose Stop.
Viewing and Monitoring Channels

You can view the configuration of an idle or running channel at any time.

You can monitor activity on channels from the AWS Elemental MediaLive console, from Amazon CloudWatch Events, or from Amazon CloudWatch Logs.

Topics
- Viewing a Channel Configuration (p. 124)
- Monitoring a Channel Using the AWS Elemental MediaLive Console (p. 124)
- Monitoring a Channel Using Amazon CloudWatch Events (p. 126)
- Monitoring Using Amazon CloudWatch Logs (p. 128)
- Logging MediaLive API Calls with AWS CloudTrail (p. 130)

Viewing a Channel Configuration

You can view information about the configuration of a channel on the Channel details page on the AWS Elemental MediaLive console. This page is useful for viewing information when the channel is running. (When a channel is running, you can't view details by choosing Edit).

To view configuration information (AWS Elemental MediaLive console)

2. In the navigation pane, choose Channels. (For information about the buttons on this page, see the section called “Editing a Channel” (p. 133), Starting, Stopping, and Pausing a Channel (p. 123), and the section called “Creating a Channel by Cloning” (p. 122).)
3. To view more details about a channel, choose the name of that channel. The Channel details page appears.
4. View configuration information in one of these places:
   - For information about the input specification for the channel, choose the Details tab and look at the Input specifications pane.
   - For a one-click view of the destination for the channel (on the downstream system), choose the Destinations tab.
   - For basic information about the configuration of the channel, choose the Details tab.
   - For a read-only view of the complete configuration of the channel (which you specified when you created or edited the channel), choose the Settings tab.
   - For a view of the raw JSON code for the channel configuration, choose the Details tab, and then choose Advanced details. You can copy this JSON code to your clipboard.

Monitoring a Channel Using the AWS Elemental MediaLive Console

You can view the activity of your channel and its current state.
To monitor activity on a channel (AWS Elemental MediaLive console)

2. In the navigation pane, choose Channels. (For information about the buttons on the page, see the section called “Editing a Channel” (p. 133), Starting, Stopping, and Pausing a Channel (p. 123), and the section called “Creating a Channel by Cloning” (p. 122).)
3. The Channels page shows a list of your channels. Each line in the list provides basic information about the channel, including its state:
   - Creating.
   - Deleting.
   - Idle: The channel isn't running. For information about charges that you accrue when a channel is idle, see the section called “Pricing” (p. 6).
   - Recovering: One or both pipelines in the channel failed, but MediaLive is restarting it.
   - Running.
   - Starting.
   - Stopping.
   - Updating: You modified a channel, and MediaLive is updating the channel information.
4. To view more details about a channel, choose the name of that channel. The Channel details page appears.

Viewing Status Information

For basic status information, look at the Status pane.

For information about the inputs in the channel, choose the Details tab.

For detailed information about the status, choose the Health tab. This tab provides information for the pipelines in the channel:

- Pipeline 0 and pipeline 1, if the channel is set up as a standard channel and therefore has two pipelines
- Pipeline 0, if the channel is set up as a single-pipeline channel

You can specify the period of time for the health information.

Viewing Alerts

AWS Elemental MediaLive generates alerts for a channel when an issue or potential issue occurs in either pipeline in a channel. These alerts are displayed in two ways:

- On the right side of the Status pane, there is a count of active alerts for each pipeline.
- On the Alerts tab, details about each alert are displayed.

If the alert is still active, the Cleared column is blank. If the alert has cleared, the column shows the timestamp for when it cleared.

Handling Alerts

When an alert occurs, look at the Alerts tab to determine possible causes of the issue. Take steps to resolve the issue.

After you resolve the issue, AWS Elemental MediaLive automatically clears the alert.
If you stop a channel, alerts always automatically clear.

Monitoring a Channel Using Amazon CloudWatch Events

AWS Elemental MediaLive automatically turns alert information into events in CloudWatch Events. You can use Amazon CloudWatch Events to manage these events. For example, you can create event rules and deliver the events in emails or SMS messages. You can deliver events to a number of destinations. This chapter describes how to deliver them through Amazon Simple Notification Service (SNS).

For complete information about the options for managing events using Amazon CloudWatch Events, see the Cloudwatch Events User Guide.

For complete information about using Amazon SNS, see the SNS Developer Guide.

Topics
- Option 1: Send all MediaLive Events to an Email Address (p. 126)
- Option 2: Send Events for Specific Channels to an Email Address (p. 127)

Option 1: Send all MediaLive Events to an Email Address

This option shows how to set up to send all events to a single email address. The drawback of this setup is that the email account will receive a large volume of emails. Therefore, we recommend that you don't use this setup in a production environment.

You must perform the following procedure in each region where channels are running.

Step 1: Create a Subscription

Create a subscription to set up a specific email address so that it automatically receives email notifications when any event occurs in MediaLive. You must identify an email recipient for the emails.

In the following procedure, we use the example of "MediaLive_alert" as the subject line and "MediaLive" as the sender of the email. We create the subscription using the Amazon Simple Notification Service (Amazon SNS) console.

To create a subscription for email notifications (Amazon SNS console)

1. Sign in to the AWS Management Console and open the Amazon SNS console at https://console.aws.amazon.com/sns/v2/home.
2. In the navigation pane, choose Topics, and then choose Create new topic.
3. In the Create new topic dialog box, for Topic name, type the name that you want for the subject line of the email, such as MediaLive_alert.
4. For Display name, type the name that you want for the sender of the email, such as MediaLive.
5. Choose Create topic.
6. Amazon SNS creates the topic and displays the ARN in the list of topics. For example, arn:aws:sns:us-west-2:111122223333:MediaLive, where 111122223333 is your AWS account.
7. Copy this ARN to your clipboard.

126
8. In the navigation pane, choose Subscriptions, and then choose Create subscription.
10. In the Create subscriptions dialog box, for Topic ARN, type or paste the ARN.
11. For Protocol, choose Email.
12. For Endpoint, type the email address of the recipient. You must be able to log on to this email account because Amazon SNS sends a confirmation email to this address.
13. Choose Create subscription.

Amazon SNS sends a confirmation email to the address that you specified.
14. Log on to that email account, and display the email. Choose the "Confirm subscription" link in the email to enable the subscription. A confirmation window appears in a web browser. You can close this window.

Step 2: Create a Rule

You now create a rule in Amazon CloudWatch that says, "When CloudWatch receives any event from aws.medialive, invoke the specified SNS topic." In other words, you create a rule that sends an email to the subscribed email address.

To create a rule (Amazon CloudWatch console)

1. Sign in to the AWS Management Console and open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
2. In the navigation pane, choose Events.
3. On the Welcome to CloudWatch Events page, choose Create rule.
4. On the Step 1 page, in Event Source, choose Event Pattern.
5. Change Build event pattern to match to Custom event pattern.
6. In the box, type the following:

```json
{
    "source": [
        "aws.medialive"
    ]
}
```

7. On the pane on the right, choose Add target.
8. Choose SNS topic.
9. For Topic, choose the topic that you created, for example, MediaLive_alert.
10. In Configure input, choose Matched event.
11. Choose Configure details.
12. Type a name and optional description, and then choose Create rule.

Now, whenever an alert occurs in MediaLive, an event will be sent to Amazon CloudWatch. This event will trigger the rule that instructs CloudWatch to send an email to the email address that you specified in the SNS subscription.

Option 2: Send Events for Specific Channels to an Email Address

You can set up a rule to send all events for a specific channel (or channels) to one email address. You must perform this setup in each region where channels are running.
Create as many subscriptions and rules combinations as you need. Follow the steps for option 1 (p. 126), with these differences:

- When creating the SNS subscription, you might want to add more detail to the topic, for example, `MediaLive_notifications_channel_1234567`.
- When creating the CloudWatch rule, you create an event pattern that identifies `aws.medialive` as the event source and the ARN for the specific channel as the resource within that event source, as follows:

```json
{
    "source": ["aws.medialive"],
    "resources": [
    ]
}
```

The resource is the ARN for the channel. You can obtain this ARN from the channels list on the MediaLive console.

This rule says, "When CloudWatch receives any event from `aws.medialive` for channel 1234567, invoke the specified SNS topic." In other words, the rule triggers an email that is sent to the subscribed email address.

You can choose to include more than one channel in the resources section, as shown in the following example:

```json
"resources": [
]
```

## Monitoring Using Amazon CloudWatch Logs

MediaLive produces channel logs that contain detailed information about activity in a channel. The logs provide a sequential description of activity that occurs in the channel. These logs can be useful when the information in alerts (the section called "Monitoring a Channel Using Amazon CloudWatch Events" (p. 126)) does not provide enough information to resolve an issue on the channel.

Channel logs are sent to Amazon CloudWatch Logs. You can use the standard features of CloudWatch Logs to view and manage the logs. For more information, see Amazon CloudWatch Logs User Guide.

You should not automate any processing based on the wording in logs because that wording is subject to change. (By comparison, you can automate based on the wording in alerts, which are accessed using CloudWatch Events, because the wording of alerts does not change.)

There is a cost for channel logging. Logging of MediaLive channel activity forms part of your charges for Amazon CloudWatch Logs. See Amazon CloudWatch Pricing.

### Topics
- Enabling Channel Logs (p. 129)
### Enabling Channel Logs

You enable the capture of logging information for an individual channel on the MediaLive console. You enable logging and set the logging level (error, warning, info, or debug) on a per channel basis. The channel must be idle in order to enable or disable logging.

**To enable a channel log (MediaLive console)**

1. If you are a returning user of MediaLive, check with your administrator that your deployment has been set up in AWS IAM to support channel logs.
2. Your administrator might instruct you to update the `MediaLiveAccessRole` permission in one of the channels. If you are given this instruction, you must edit a channel (p. 133) (choose any idle channel), display the Channel and input details page (p. 87), and choose the **Update** button. When the role is updated in one channel, the change applies to all channels.
3. To enable logging in a new channel, set up logging during creation (p. 86). To enable logging in an existing channel, edit the channel (p. 133); this channel must be idle. In both cases, on the General settings page, in the Channel logging section, choose **Logging**. Choose a level other than **DISABLED**. For more information, see the section called “Logging” (p. 95).
4. You or an administrator can also go into CloudWatch Logs and set an expiry date for the logs.

### Disabling Channel Logs

You disable the capture of logging information for an individual channel on the MediaLive console. Edit the channel, and on the General settings page, on the Channel logging section, choose **Logging**. Set the level to **DISABLED**.

### Working with Logs

You view MediaLive logs on the CloudWatch Logs console, in the same way that you view logs for any service.

You don't have to set up the logs, logging groups, or log streams on the CloudWatch Logs console because MediaLive automatically sets them up for you.

The log group is always the following: **ElementalMediaLive**.

The log stream is named after the ARN/pipeline. For example, `arn:aws:medialive:us-west-2:111122223333:channel:5106412_0`, where 5106412 is the channel ID and 0 is the pipeline.

### Content of EML Logs

The MediaLive logs are in JSON format:

```json
{
    "encoder_pipeline": 0,
    "severity": "I",
    "timestamp": "2018-05-21T16:36:41.650318",
    "logger_name": "",
    "message": "Probing input media..."
}
```
The data is the following:

- **encoder_pipeline**: 0 or 1 (if the channel is set up as a standard channel (p. 88) and therefore has two pipelines).
- **severity**: A letter. The logging level (which you set when you enable logging) controls which severities could appear in logs. For more information, see Log Levels and Verbosities (p. ).
- **timestamp**: The time in ISO 8601 format: yyyy - mm - dd T hh : mm : ss : decimal fraction of second.
- **channel_arn**: The ARN plus the channel ID. In the preceding example, the channel has ID 5106412.
- **logger_name**: This might be blank or might specify a name that ties a series of related messages together.
- **message**: The message. Remember that the wording is subject to change, so you should not automate against it.

### Log Levels and Verbosities

To use this table, find a level in the first column then read across to identify the message severities that will appear in the logs with this logging level.

<table>
<thead>
<tr>
<th>Level</th>
<th>Debug Messages</th>
<th>Info Messages</th>
<th>Warning Messages</th>
<th>Critical Messages</th>
<th>Fatal Messages</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEBUG</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>INFO</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>WARNING</td>
<td></td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>ERROR</td>
<td></td>
<td>Yes</td>
<td></td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

### Managing Log Storage

When you delete a channel, the associated logs remain in CloudWatch Logs. You will continue to be charged for their storage until you delete them. To delete logs, change the log data retention. All the data that is older than the retention setting that you specify will be deleted. For more information, see Amazon CloudWatch Logs User Guide. The **Log group** for the logs is **ElementalMediaLive**.

### Logging MediaLive API Calls with AWS CloudTrail

AWS Elemental MediaLive is integrated with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service in MediaLive. CloudTrail captures all API calls for MediaLive as events. The calls captured include calls from the MediaLive console and code calls to the MediaLive API operations. If you create a trail, you can enable continuous delivery of CloudTrail events to an Amazon S3 bucket, including events for MediaLive. If you don't configure a trail, you can still view the most recent events in the CloudTrail console in **Event history**. Using the information collected by CloudTrail, you can determine the request that was made to MediaLive, the IP address from which the request was made, who made the request, when it was made, and additional details.

To learn more about CloudTrail, see the AWS CloudTrail User Guide.
MediaLive Information in CloudTrail

CloudTrail is enabled on your AWS account when you create the account. When activity occurs in MediaLive, that activity is recorded in a CloudTrail event along with other AWS service events in Event history. You can view, search, and download recent events in your AWS account. For more information, see Viewing Events with CloudTrail Event History.

For an ongoing record of events in your AWS account, including events for MediaLive, create a trail. A trail enables CloudTrail to deliver log files to an Amazon S3 bucket. By default, when you create a trail in the console, the trail applies to all AWS Regions. The trail logs events from all Regions in the AWS partition and delivers the log files to the Amazon S3 bucket that you specify. Additionally, you can configure other AWS services to further analyze and act upon the event data collected in CloudTrail logs. For more information, see the following:

- Overview for Creating a Trail
- CloudTrail Supported Services and Integrations
- Configuring Amazon SNS Notifications for CloudTrail
- Receiving CloudTrail Log Files from Multiple Regions and Receiving CloudTrail Log Files from Multiple Accounts

All MediaLive actions are logged by CloudTrail and are documented in the https://docs.aws.amazon.com/medialive/latest/apireference/.

Every event or log entry contains information about who generated the request. The identity information helps you determine the following:

- Whether the request was made with root or AWS Identity and Access Management (IAM) user credentials.
- Whether the request was made with temporary security credentials for a role or federated user.
- Whether the request was made by another AWS service.

For more information, see the CloudTrail userIdentity Element.

Understanding MediaLive Log File Entries

A trail is a configuration that enables delivery of events as log files to an Amazon S3 bucket that you specify. CloudTrail log files contain one or more log entries. An event represents a single request from any source and includes information about the requested action, the date and time of the action, request parameters, and so on. CloudTrail log files aren't an ordered stack trace of the public API calls, so they don't appear in any specific order.

The following example shows a CloudTrail log entry. The example shows the entry for one API call. The call is made by the identity that is specified in userIdentity, in this case an IAM user with the user name santosp. The call was a CreateInput operation coming from the AWS CLI (as specified in userAgent) running on a computer with the IP address 203.0.113.33:

```json
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
    "arn": "arn:aws:iam::111122223333:user/santosp",
    "accountId": "111122223333",
    "accessKeyId": "AKIAOSFODNNYEXAMPLE",
```
"userName": "santosp",
"eventTime": "2019-01-17T21:21:17Z",
"eventSource": "medialive.amazonaws.com",
"eventName": "CreateInput",
"awsRegion": "us-west-2",
"sourceIPAddress": "203.0.113.33",
"userAgent": "aws-cli/1.16.86 Python/2.7.15 Darwin/17.7.0 botocore/1.12.76",
"requestParameters": {
  "mediaConnectFlows": [],
  "inputSecurityGroups": [
    "9999999"
  ],
  "sources": [],
  "roleArn": "MediaLiveAccessRole",
  "requestId": "1111aaaa-9604-4459-a160-46a28ae166",
  "name": "live_studio_feed",
  "type": "RTP_PUSH",
}
},
"responseElements": {
  "input": {
    "id": "7780651",
    "name": "live_studio_feed",
    "type": "RTP_PUSH",
    "sources": [],
    "destinations": [
      {
        "url": "rtp://198.51.100.10:1935",
        "ip": "198.51.100.10:1935",
        "port": "1935"
      },
      {
        "url": "rtp://192.0.2.131:1935",
        "ip": "192.0.2.131:1935",
        "port": "1935"
      }
    ],
    "mediaConnectFlows": [],
    "state": "DETACHED",
    "attachedChannels": [],
    "securityGroups": [
      "9999999"
    ],
    "roleArn": ""
  }
},
"requestID": "d2f882ac-1a9d-11e9-a0e5-afe6a8c88993",
"eventID": "ebbe0290-7a1b-4053-a219-367404e0fe96",
"readOnly": false,
"eventType": "AwsApiCall",
"recipientAccountId": "111122223333"}
Editing and Deleting a Channel

You can edit an existing (saved) channel to change how it processes the input, and you can delete a channel. However, you can edit or delete a channel only when it is not running.

Editing a Channel

You can edit any existing channel by editing, adding, or deleting output groups and outputs. You can also edit, add, or delete the channel's video, audio, and caption encodes.

The channel must be idle (not running).

**Note**

You can't edit a channel to change the input associated with that channel. Instead, you must clone (p. 122) the channel and associate a different input.

You can't edit a channel to change the channel class. Instead, you must update the channel class as its own operation. See the section called “Changing the Channel Class” (p. 133), later in this chapter.

To edit a channel

1. On the **Channels** page, choose the option by the channel name.
2. Choose Actions, and then choose **Edit**. The Edit channel page appears. The details on this page are identical to those on the **Create channel** page. For information about working with this page, see *Creating a Channel from Scratch* (p. 86).
3. When done, choose Update channel.

Wait for the channel **State** to return to **Idle** before performing another action with this channel.

Changing the Channel Class

When you create a channel, you choose a channel class: a standard channel (two pipelines) or a single-pipeline channel. For information about the role of the channel class, see the section called “Determining the Channel Class” (p. 56). You can change the channel class of an existing channel in order to change the resiliency model of the channel:

- You can change a standard channel to single-pipeline, to remove one of the pipelines in the channel and to remove pipeline resiliency.
- You can change a single-pipeline channel to a standard channel, to upgrade the channel to two pipelines.

To change the channel class, the channel must be idle (not running).

**To change the channel class to a single-pipeline channel**

1. On the **Channels** page, choose the channel. (Don't choose the channel name.)
2. On the menu, choose Actions, Other channel actions, Update channel class to SINGLE_PIPELINE.
3. In the dialog box, choose **Confirm**. MediaLive performs the following actions:
   - It removes the second pipeline (pipeline 1) in the channel.
• It removes the second destination address in each output group.
• It doesn't remove the second endpoint on the inputs. The inputs aren't changed in any way. Instead, when you restart the channel, MediaLive simply ignores the second endpoint.

While MediaLive is performing these actions, the channel has a status of **UPDATING**. When the update is completed, the status changes to **IDLE**.

4. You might want to notify the upstream system for each push input that it no longer needs to push input to the second endpoint. You might want to notify the downstream system for each output group that it should no longer expect output at its second destination.

### To change the channel class to a standard channel

1. Obtain a second destination address for each output group. Each address is at the downstream systems of each output group.

   For example, if the channel has an HLS output group (with an HTTPS server as its downstream system) and an archive output group (with an Amazon S3 bucket as its downstream system), you must enter the URL to a new destination address at the HTTPS server, and the URL to a new folder in the Amazon S3 bucket.

   Plan these destinations now, in the same way as you planned the destination addresses when you originally set up the channel. You might need to contact the owner of each downstream system.

2. For the inputs that are attached to the channel, look at the endpoints (for push inputs) or the sources (for pull inputs). If any endpoint or source doesn't have a second address, you must set that up now:

   • For a push input, edit the input (p. 84) to include an address in for the second input source. Give that address to the owner of the upstream system, so that they can push source content to that address. You should also find out from the upstream system the address that the new source will be pushed from. Make sure that this address is covered by the input security group for the channel.

   • For a pull input, obtain a new address from the owner of the downstream system. Edit the input (p. 84) to include that address. When the second pipeline is created, MediaLive will be able to pull the second source content (for the second pipeline).

3. On the **Channels** page, choose the channel. (Don't choose the channel name.)

4. On the menu, choose Actions, Other channel actions, Update channel class to STANDARD.

5. In the dialog box, choose **Confirm**.

6. On the **Update channel class to STANDARD** page, enter the destination addresses that you identified in step 1. There is one field for each output group in the channel.

7. Choose **Submit**. MediaLive updates the channel and creates a new pipeline called pipeline 1. When you start the channel, MediaLive sends the output from this pipeline to the new destinations in every output group.

---

**Editing the Tags Associated with a Channel**

You can edit the tags associated with a channel at any time, when the channel is running or when it is idle. You can add more tags (up to the limit (p. 227)), and you can delete tags.

### To edit the tags in a channel

1. On the **Channels** page, choose the channel name.
2. Choose the Tags tab. Add or delete tags. To edit the value of an existing tag, delete the tag and add it again. For more information, see Tagging Resources (p. 227).
3. When done, choose Save.

Deleting a Channel

You can delete a channel from the Channels list or the details view.

The channel must be idle (not running).

To delete a channel

1. On the Channels page, choose the option by the channel name.
2. If the channel is running, choose Stop.
3. Choose Delete.
Working with the AWS Elemental MediaLive Schedule

In AWS Elemental MediaLive, you can manipulate the processing of a channel while it is running. You perform this manipulation by inserting actions into the schedule that is associated with the channel.

You can use actions to do the following:

- Switch the input that the running channel is ingesting.
- Insert a static image overlay (an image layered over the underlying video) into the running channel.
- Insert SCTE-35 messages into the running channel.
- Insert ID3 metadata into the running channel.
- Pause one or both of the pipelines in the channel.

This chapter describes how to set up the schedule with the actions that you want. Before you get started, we recommend that you do the following:

- If you want to switch inputs, read the information about setting up a channel to include multiple inputs, as described in Input Switching (p. 189).
- If you want to activate and deactivate image overlays, set up the images that you will use. See Working with Image Overlays (p. 225).
- If you want to insert SCTE-35 messages, read the information that explains how MediaLive handles SCTE-35 messages in transport stream (TS) outputs. See SCTE-35 Message Processing (p. 202).
- If you want to insert ID3 metadata, read the information about the options for ID3 metadata in MediaLive. See Working with ID3 Metadata (p. 198).

Actions in the Schedule

The schedule is a list of actions that a channel executes as it is running. You create the actions in the schedule, and you set either a fixed or relative time that specifies when you want the channel to execute each action.

MediaLive stores the actions in the schedule. Shortly before its start time, each action is received by the channel. At the action start time, the channel executes the action.

You can delete an action from the schedule up to the moment that the channel receives it.

Types of Schedule Actions for Input Switching

- Action to switch the input that the running channel is ingesting. The action specifies when to execute the switch, either at a specific (fixed) time, or to follow when the currently running input is finished.

  When the channel executes this action, the channel stops ingesting the current input and starts ingesting the specified input.

  With input switching, the channel must already be set up with all the inputs that you want to switch to. You can't use a switch action to insert an input into the list of inputs in the channel. You can use it only to switch among the inputs that are already in the list.
After you create an action, the action waits in the schedule. The channel receives the action 15 seconds before the action's designated start time. At the start time, the channel executes the action.

Before you add input switching actions to the schedule, read Input Switching (p. 189).

**Types of Schedule Actions for Image Overlay**

- Action to activate a static image overlay. An image overlay is an image that is layered over the underlying video.

  When a channel executes this action, the image overlay is superimposed on the video. If the image overlay information includes a duration, then at the appropriate time the image overlay is removed.

- Action to deactivate a static image overlay.

  When the channel executes the deactivate action, the image overlay is removed. You therefore use this action to remove a currently running image overlay before the specified duration, or remove it when no duration is specified.

After you create an action, the action waits in the schedule. The channel receives the action 15 seconds before the action's designated start time. At the start time, the channel executes the action.

**Note**

You remove the active image from the video by creating a deactivate action. You don't deactivate the image by deleting the activate action from the schedule. (In fact, deleting the action has no effect because its start time has passed.)

Before you add image overlay actions to the schedule, read Working with Image Overlays (p. 225).

**Types of Schedule Actions for SCTE-35**

- Action to insert a splice_insert into the channel: a SCTE-35 message with splice_command_type set to splice_insert.

- Action to insert a time_signal into the channel: a SCTE-35 message with splice_command_type set to time_signal.

- Action to insert a SCTE-35 return-to-network message into the schedule in order to end a splice_insert that either has a duration or has no duration.

After you create an action, the action waits in the schedule. The channel receives the action 15 seconds before the action's designated start time. At the start time, the channel executes the action.

Before you add SCTE-35 actions to the schedule, read SCTE-35 Message Processing (p. 202).

**Types of Schedule Actions for ID3 Metadata**

- Action to insert ID3 metadata in outputs where ID3 passthrough is enabled (p. 198). You must insert a fully formed ID3 metadata item (including both a header and a frame, as per the ID3 specification) and encode it as base64. You specify a start time for inserting the metadata.

After you create an action, the action waits in the schedule. The channel receives the action 15 seconds before the action's designated start time. At the start time, the channel executes the action.

Before you add ID3 metadata actions to the schedule, read Working with ID3 Metadata (p. 198).
Types of Schedule Actions for Pause

- Action to pause the specified pipeline or pipelines in the channel. By implication, any pipelines that you don't include in the action are automatically set to unpause. You must specify a start time for the pause action.
- Action to unpause a pipeline or pipelines that is currently paused. You must specify a start time for the unpause action.

After you create an action, the action waits in the schedule. The channel receives the action 15 seconds before the action's designated start time. At the start time, the channel executes the action.

**Note**
You unpause (stop the pause) on the pipeline by creating an unpause action. You don't stop the pause by deleting the pause action from the schedule. (In fact, deleting the action has no effect because its start time has passed.)

Working with the Schedule (Console)

You can use the AWS Elemental MediaLive console to create or delete any of the schedule actions (p. 136) in a channel. You can work with the schedule when the channel is running or when it is idle.

The actions are executed in the channel when the channel is running. The channel executes the actions at the start times that you specify.

The console provides two views for working with actions:

- A list view that lists actions in tabular format
- A timeline view that shows a timeline representation of the actions

In either view, you can do the following:

- Create individual actions
- Delete individual actions
- Delete several actions in one request (a batch command)
- View the actions currently in the schedule

**Topics**
- Creating Actions in the Schedule (Console) (p. 138)
- Deleting Actions from the Schedule (Console) (p. 144)
- Modifying Actions in the Schedule (Console) (p. 145)
- Viewing the Schedule (Console) (p. 147)

Creating Actions in the Schedule (Console)

You can create actions to switch the input that the channel is ingesting, activate or deactivate image overlays on the video, insert SCTE-35 messages in the output, insert ID3 metadata in the output, or pause or unpause a pipeline in the channel.

You can create an action in the schedule when the channel is running or when it is idle.
The action that you create must have a UTC start time or a follow start time (for input switches) that is at least 15 seconds in the future. (In other words, it must not be already received in the channel.) After that cutoff, AWS Elemental MediaLive rejects the create request.

The general procedure is the same to create any type of action.

**To create an action**

2. In the navigation pane, choose Channel, and then choose the channel that you want to work with.
3. On the Details pane, choose the Schedule tab.
4. Choose the Switch button to display the view that you want: List view or Timeline view. For information about the layout and color coding of the timeline view, see the section called “Viewing the Schedule” (p. 147).
5. For List view, choose Create. Or choose an existing action, choose Actions, and then choose Create follow actions from. This action displays the Create schedule action page with some fields already completed, so you can quickly create a follow switch for that existing action.
6. For Timeline view, choose the appropriate action:
   - To create a fixed input switch, an image overlay action, a SCTE-35 action, or an HLS timed metadata (ID3 metadata) action, choose Create.
   - To create a follow input switch, find the action for the preceding input switch, and then choose Create follow action in that card.
7. On the Create schedule action page, complete the fields. For information about completing the fields, see the following topics.
8. When you are done, choose Create.

   MediaLive adds the action to the list or the timeline at its appropriate time slot.

When you create a follow input switch, you effectively create a follow chain. The follow chain starts with the input above the first follow and ends with the last follow input. For more information about follow chains, see the section called “Types of Switches—Fixed, Follow, and Follow Chains” (p. 192).

**Topics**

- Fields for an Input Switch (p. 139)
- Fields for Activating an Image Overlay (p. 140)
- Fields for Deactivating an Image Overlay (p. 141)
- Fields for a Splice_Insert Message (p. 141)
- Fields for a Time_Signal Message (p. 142)
- Fields for a Return-to-Network Message (p. 142)
- Fields for ID3 Metadata (p. 143)
- Fields for Pause (p. 143)
- Fields for Unpause (p. 144)

**Fields for an Input Switch**

This table shows the fields that apply for an action to switch the input.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action type</td>
<td>Input Switch.</td>
</tr>
</tbody>
</table>
## Creating Actions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action name</td>
<td>A name for this input switch.</td>
</tr>
<tr>
<td>Start type</td>
<td>Fixed or Follow.</td>
</tr>
<tr>
<td>Date and time</td>
<td>If the Start type is Fixed, specify the date and time (in UTC format) that the channel must switch to this new input. This time must be at least 30 seconds in the future. Note that the time is the wall clock time, not the timecode in the input.</td>
</tr>
<tr>
<td>Reference action name</td>
<td>If the Start type is Follow, choose the input to switch from, which is the input that precedes this new input. The dropdown list shows all existing input switches that are file inputs. Remember that input B can follow input A only if input A is a file input. For information about these switching rules, see the section called &quot;Types of Switches—Fixed, Follow, and Follow Chains&quot; (p. 192). For example, if you want to switch from input A to input B, specify input A in this field.</td>
</tr>
<tr>
<td>Follow point</td>
<td>Applies only to a Follow start type. The follow point is always End, to indicate that the switch will occur when the input in Reference action name has finished.</td>
</tr>
<tr>
<td>Input attachment</td>
<td>The input to switch to. The input must already be set up as an input attachment (p. 89) in this channel. It can be a live input or a file input. For example, if you want to switch from input A to input B, specify input B in this field.</td>
</tr>
</tbody>
</table>

### Fields for Activating an Image Overlay

This table shows the fields that apply for an action to activate an image overlay.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action type</td>
<td>Static Image Activate.</td>
</tr>
<tr>
<td>Action name</td>
<td>A name for this activation action. For example, the name of the image to overlay.</td>
</tr>
<tr>
<td>Start type</td>
<td>Fixed.</td>
</tr>
<tr>
<td>Date and time</td>
<td>The date and time (in UTC format) that the channel must activate the image overlay. The time should be at least 60 seconds later than the time that you submit the action.</td>
</tr>
</tbody>
</table>
### Fields for Deactivating an Image Overlay

This table shows the fields that apply for an action to deactivate an image overlay.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action type</td>
<td>Static Image Deactivate.</td>
</tr>
<tr>
<td>Action name</td>
<td>A name for this deactivation action. For example, the name of the image. Or a name that ties back to the activation action plus the term &quot;deactivate&quot;.</td>
</tr>
<tr>
<td>Start type</td>
<td>Fixed.</td>
</tr>
<tr>
<td>Date and time</td>
<td>The date and time (in UTC format) that the channel must deactivate the image overlay. The time should be at least 60 seconds later than the time that you submit the action. Note that the time is the wall clock time, not the timecode in the input.</td>
</tr>
</tbody>
</table>

### Fields for a Splice_Insert Message

This table shows the fields that apply for an action to insert a splice_insert SCTE-35 message.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action type</td>
<td>SCTE-35 Splice Insert.</td>
</tr>
<tr>
<td>Action name</td>
<td>A name for this splice_insert action. For example, splice_insert actions could be numbered sequentially, restarting every day or every month.</td>
</tr>
<tr>
<td>Start type</td>
<td>Fixed.</td>
</tr>
<tr>
<td>Date and time</td>
<td>The UTC start time for the splice_insert action. The time should be at least 15 seconds later than the time that you submit the action. Note that the time is the wall clock time, not the timecode in the input.</td>
</tr>
<tr>
<td>Splice event id</td>
<td>The ID for the splice event. Enter an ID for the splice event that is unique among all scheduled and active splice_insert messages in this channel. A message is active if the schedule action is in process in the channel and has not completed.</td>
</tr>
</tbody>
</table>
### Creating Actions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Duration</td>
<td>The duration for the splice event. Enter the duration, in 90 kHz ticks. For example, 1350000, which is equal to 15 seconds.</td>
</tr>
</tbody>
</table>

The splice_insert inserted in the transport stream will have the following:

```plaintext
segmentation_event_cancel_indicator = 0
out_of_network = 1
duration_flag = 1
duration = the specified time
```

Or

```plaintext
segmentation_event_cancel_indicator = 0
out_of_network = 1
duration_flag = 0
```

### Fields for a Time_Signal Message

This table shows the fields that apply for an action to insert a time_signal SCTE-35 message.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action type</td>
<td>SCTE-35 Time Signal.</td>
</tr>
<tr>
<td>Action name</td>
<td>A name for this time_signal action. For example, you might name time_signal actions with a sequential number, restarting every day or every month.</td>
</tr>
<tr>
<td>Start type</td>
<td>Fixed.</td>
</tr>
<tr>
<td>Date and time</td>
<td>The UTC start time for the time_signal. The time should be at least 15 seconds later than the time that you submit the action. Note that the time is the wall clock time, not the timecode in the input.</td>
</tr>
<tr>
<td>Add Scte35 descriptors</td>
<td>Choose this button and complete the fields that appear. The descriptors are a standard component of a time_signal message.</td>
</tr>
</tbody>
</table>

### Fields for a Return-to-Network Message

This table shows the fields that apply for an action to insert a return-to-network SCTE-35 message.
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action type</td>
<td>SCTE-35 Return to Network.</td>
</tr>
<tr>
<td>Action name</td>
<td>A name for this return-to-network action. For example, &quot;splice0003_return_early&quot;.</td>
</tr>
<tr>
<td>Start type</td>
<td>Fixed.</td>
</tr>
<tr>
<td>Date and time</td>
<td>The UTC start time for the return. The time should be at least 15 seconds later than the time that you submit the action.</td>
</tr>
<tr>
<td></td>
<td>Note that the time is the wall clock time, not the timecode in the input.</td>
</tr>
<tr>
<td>Splice event id</td>
<td>The ID of the splice_insert that the return-to-network should end. You assigned this ID when you created the splice_insert.</td>
</tr>
</tbody>
</table>

**Fields for ID3 Metadata**

This table shows the fields that apply for an action to insert one ID3 metadata.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action type</td>
<td>HLS Timed Metadata.</td>
</tr>
<tr>
<td>Action name</td>
<td>A name for the metadata item. You might want to design a convention for naming ID3 metadata items, such as &quot;id3_metadata-&lt;UTC time&gt;&quot;.</td>
</tr>
<tr>
<td>Start type</td>
<td>Fixed.</td>
</tr>
<tr>
<td>Date and time</td>
<td>The UTC start time for the ID3 metadata item. The time should be at least 15 seconds later than the time that you submit the action.</td>
</tr>
<tr>
<td></td>
<td>Note that the time is the wall clock time, not the timecode in the input.</td>
</tr>
<tr>
<td>Id3</td>
<td>The ID3 metadata. The metadata must be fully formed (including both a header and a frame, as per the ID3 specification) and must be encoded as base64.</td>
</tr>
</tbody>
</table>

**Fields for Pause**

In Schedule action settings, complete the following fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action type</td>
<td>Pause</td>
</tr>
<tr>
<td>Action name</td>
<td>A name for the action.</td>
</tr>
</tbody>
</table>
### Deleting Actions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start type</td>
<td>Fixed.</td>
</tr>
<tr>
<td>Date and time</td>
<td>The UTC start time for the action. The time should be at least 15 seconds later than the time that you submit the action. Note that the time is the wall clock time, not the timecode in the input.</td>
</tr>
<tr>
<td>Actions</td>
<td>Choose <strong>Add actions</strong>, then for <strong>Pipeline id</strong>, choose the pipeline that you want to pause: <strong>PIPELINE_0</strong> or <strong>PIPELINE_1</strong>.</td>
</tr>
</tbody>
</table>

When you choose **Create**, MediaLive adds an action to the schedule to pause the specified pipeline and to unpause any pipeline that isn't specified. As a result, only the specified pipeline will be paused after the action is executed.

### Fields for Unpause

In **Schedule action settings**, complete the following fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action type</td>
<td>Pause</td>
</tr>
<tr>
<td>Action name</td>
<td>A name for the action.</td>
</tr>
<tr>
<td>Start type</td>
<td>Fixed.</td>
</tr>
<tr>
<td>Date and time</td>
<td>The UTC start time for the action. The time should be at least 15 seconds later than the time that you submit the action. Note that the time is the wall clock time, not the timecode in the input.</td>
</tr>
<tr>
<td>Actions</td>
<td>Leave this section empty. Don't add any actions.</td>
</tr>
</tbody>
</table>

When you choose **Create**, MediaLive adds an action to the schedule to unpause all pipelines.

### Deleting Actions from the Schedule (Console)

You can delete actions that you previously created in the schedule, so long as the action has not been received by the channel. You can delete an action when the channel is running or when it is idle.

The action that you delete must have a UTC start time or a follow start time (for input switches) that is at least 15 seconds in the future. (In other words, it must not be already received in the channel.) After that cutoff, AWS Elemental MediaLive rejects the delete request.

You can delete any number of actions in one request, or any combination of types of actions in one request. For example, you can mix the deletion of SCTE-35 message actions and image overlay actions.

The general procedure is the same to delete any type of action.
To delete actions in List view
2. In the navigation pane, choose Channel, and then choose the channel that you want to work with.
3. On the Details pane, choose the Schedule tab.
4. If necessary, choose the Switch button to display the List view. For information about the layout and color coding of the timeline view, see the section called “Viewing the Schedule” (p. 147).
5. Choose one or more actions to delete. Make sure that you choose actions with a UTC start time in the future.

   If you choose an input switch that is in a follow chain, a prompt appears to notify you that the follow actions below this action (up to the next fixed input switch) will also be deleted. You can cancel or continue.

   Choose Actions, and then choose Delete.

To delete actions in Timeline view (console)
2. In the navigation pane, choose Channel, and then choose the channel that you want to work with.
3. On the Details pane, choose the Schedule tab.
4. If necessary, choose the Switch button to display the Timeline view. For information about the layout and color coding of the timeline view, see the section called “Viewing the Schedule” (p. 147).
5. In each action section, choose the X to delete the action. Make sure that you choose actions with a UTC start time in the future.

   If you choose an input switch that is in a follow chain, a prompt appears to notify you that the follow actions below this action (up to the next fixed input switch) will also be deleted. You can cancel or continue.

Modifying Actions in the Schedule (Console)

You can't modify an action in the schedule, even if it hasn't been received by the channel. However, you can sometimes achieve a modify effect using a create action, a delete action, or both.

Modifying Actions Not Yet Started

You can't modify an action in the schedule, even if it hasn't been received by the channel.

To change an action that hasn't yet started, delete the action and create it again. See the following sections for important tips on deleting and recreating.

Modifying an Input Switch Action

You can't modify input switches in the schedule. But you can achieve the same result by deleting the action and creating it again.

Keep in mind that you can't delete or create an action that has a start time less than 15 seconds in the future. Give yourself enough time to delete and recreate the action before these deadlines.
Modifying an Input Switch Action in a Follow Chain

When you delete an action in a follow chain (in order to delete and recreate it), you must also delete and recreate the follow actions below this one. You must do this because each action refers to the previous action. If you delete the previous action, the next action becomes an orphan. Orphans aren't permitted.

**Example 1: Modify an Action**

For example, assume this is the follow chain:

<table>
<thead>
<tr>
<th>Input A</th>
<th>Fixed</th>
<th>File</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input B</td>
<td>Follow</td>
<td>File</td>
</tr>
<tr>
<td>Input C</td>
<td>Follow</td>
<td>File</td>
</tr>
<tr>
<td>Input D</td>
<td>Follow</td>
<td>File or Live</td>
</tr>
<tr>
<td>Input E</td>
<td>Fixed</td>
<td>File or Live</td>
</tr>
</tbody>
</table>

To delete and modify input A, you must also delete inputs B, C, and D. You must delete input B to prevent it becoming an orphan. The same rule applies until the next fixed input (input E), which isn't chained to another input. Therefore, you aren't required to delete input E.

When you delete input A using the console, a prompt appears to notify you that the follow actions below this action (up to Input E, which is the next fixed input switch) will also be deleted. You can cancel or continue. You must then recreate inputs A to D. Recreate them in order going down the chain: input A, input B, input C, input D.

**Example 2: Delete an Action**

This example shows how to delete input B:

<table>
<thead>
<tr>
<th>Input A</th>
<th>Fixed</th>
<th>File</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input B</td>
<td>Follow</td>
<td>File</td>
</tr>
<tr>
<td>Input C</td>
<td>Follow</td>
<td>File</td>
</tr>
<tr>
<td>Input D</td>
<td>Follow</td>
<td>File or Live</td>
</tr>
<tr>
<td>Input E</td>
<td>Fixed</td>
<td>File or Live</td>
</tr>
</tbody>
</table>

When you delete input B using the console, a prompt appears to notify you that the follow actions below this action (up to Input E, which is the next fixed input switch) will also be deleted. You can cancel or continue. You must then recreate inputs C and D. Recreate them in order going down the chain: input C, input D. Remember to set up input C to follow input A instead of input B.

Inserting an Input Switch Action into a Chain

To insert an action in a follow chain (between two existing follow switch actions), you must delete and recreate the follow actions below the insertion. You do that to prevent two actions from following one previous action. Branching is not allowed in the chain.

For example, suppose that you want to insert input X between input B and input C. Input C already refers to input B. You also want input X to refer to input B, but that is not allowed. Therefore, you must delete input C and the inputs that follow. You then recreate the chain in the following order: input X (refers to input B), input C (refers to input X), input D (refers to input C).

Modifying an Image Overlay That Is in Progress

You can't directly modify an image overlay that is running in the channel. But you can achieve the same result by creating a new action with the same layer specified. You can do the following:
• Shorten or extend the duration of an image overlay.
• Change one or more attributes.
• Specify an attribute that isn't currently specified in an image overlay. For example, you might want to specify a fadeout where there is no fadeout in the current overlay.
• Create a new action (with a new action name) that inserts an image overlay 15 seconds in the future. Make sure to specify the following:
  • The same layer as the image that you want to modify.
  • A duration that is appropriate (the image doesn't inherit the duration of the current image).
  • All the attributes that you want.

The new action replaces the current action because you specified the same layer.

Modifying a SCTE-35 Message That Is in Progress
You can't modify a SCTE-35 message that is active in the channel. Specifically, you can't shorten the duration of a splice_insert. But you can achieve the same result by creating a return-to-network action.

Viewing the Schedule (Console)
You can display the list of actions currently in the schedule and view them in list or timeline view.

To view actions (console)
2. In the navigation pane, choose Channel, and then choose the channel that you want to work with.
3. On the Details pane, choose the Schedule tab.

   The Schedule actions pane shows the actions in the schedule's current window of time.
   You can switch between the two views of the schedule by choosing the Switch button.

List View
The actions are mostly listed in chronological order.

Input switches in a follow chain are grouped together starting from the top of the chain (which is the action above the first follow action) to the last follow action. Other actions, such as actions for SCTE-35 and image overlay, might occur between two follow actions. MediaLive can't predict whether a SCTE-35 or image overlay action will occur between two follow actions and doesn't attempt to show it in the list view.

Timeline View
The actions are arranged in cards along a vertical axis. The card titles are color-coded by the action type. For example, SCTE-35 time_signal messages are yellow.

One card might contain several input switches. The first input switch is always a fixed input switch, and the remaining input switches in that card are always follow input switches.

Window of Time for the Schedule
The searchable window of time for the schedule extends to approximately one hour in the past. Actions with a start time more than one hour in the past aren't displayed when you view the schedule on the console, and aren't included in the response to a DescribeChannel command.

However, a running action that is no longer visible continues to run for its full duration.
The existence of this searchable window of time means the following:

- Actions that relate to SCTE-35: for messages that have a duration (splice_insert specified with a duration), the message could still be active, but it is no longer possible to query MediaLive for the duration.
- Actions that relate to image overlays: for actions to activate a static image overlay for a duration of time, the overlay might still be active, but it is no longer be possible to query MediaLive for the duration.
- Actions that relate to input switching: there is no impact.
- Actions that relate to ID3 metadata: there is no impact.

Working with the Schedule (AWS CLI)

You can use the AWS CLI to work with the schedule programmatically. The sections later in this chapter describe how to enter the appropriate commands. These sections assume that you are familiar with the basics of using the AWS CLI. For information about the basics, see the AWS CLI Command Reference.

The following sections describe each command and provide this additional information:

- A description of the AWS CLI command syntax.
- A description of the schema for the request or response JSON payload. This payload is shown using the syntax for the AWS CLI.
- An example of the request or response JSON payload. This payload is also shown using the syntax for the AWS CLI.

For details on the JSON contents, we recommend that you read the AWS Elemental MediaLive API Reference. This guide is easy to use because it includes links from elements in the JSON payload to tables that describe the element. But you need to adjust the syntax of the elements in the JSON code because the AWS CLI uses one form of capitalization for elements (for example, SubSegmentNum) and the API uses another form (for example, subSegmentNum).

Topics
- Creating and Deleting Using a Batch Command (p. 148)
- Submitting a Batch Update Schedule Command (p. 150)
- JSON Payload for Create Actions (p. 151)
- JSON Payload for Delete Actions (p. 162)
- JSON Payload for Combining Create and Delete (p. 163)
- Viewing the Schedule (AWS CLI) (p. 164)

Creating and Deleting Using a Batch Command

To create and delete actions in the schedule for a channel, you use the batch update schedule command. This command lets you perform multiple actions in one request. There is not one command for creating actions and another for deleting actions.

You can use the command as follows:

- Submit a single request such as a request to do the following:
  - Create one action.
  - Delete one action.
- Submit a batch request such as one request to do the following:
• Create several actions.
• Delete several actions.
• Create one or more actions and delete one or more actions.

**Important**
In a command that combines create actions and delete actions, the delete actions are *always* performed before the create actions. This means that MediaLive removes the delete actions from the schedule before it adds the create actions to the schedule.

**Topics**
• How a Batch Request Works (p. 149)
• Batch Command in Different Interfaces (p. 150)
• JSON Payload in Different Interfaces (p. 150)

**How a Batch Request Works**

The intention of batching is to pass or fail all the actions together. Therefore, AWS Elemental MediaLive validates batch actions together. MediaLive performs the following validation:

• It ensures that each action that is created or deleted has a UTC start time that is at least 15 seconds in the future.
• If an action refers to an existing action in the schedule, it ensures that the reference to the existing action is correct. For example, a follow input switch includes a reference to the action that it follows. That action must exist.

If the validation fails for any one action, it fails for all the actions in the batch.

If you don't want the actions to pass or fail together, don't submit a batch. Instead, create each action in its own batch update schedule command.

If the validation succeeds, MediaLive processes all the delete requests before the create requests, regardless of the start times of the actions.

**Example 1**

An important use of batching is to execute several actions that must pass or fail together. For example, suppose that you want to remove the corporate logo and immediately insert a splice_insert (in order to go to an ad avail). To do that, you must create an action to remove the logo and another action to insert the splice_insert. However, you don't want MediaLive to insert the remove action if the splice_insert action fails, or vice versa. It's better if both actions fail because that allows you to fix the badly formed action, and then submit both actions again.

You therefore submit the two actions together, in one batch update schedule command.

**Example 2**

Another important use of batching is to fix an error in an action in the schedule. For example, you might want to fix an image overlay that hasn't started yet and that was created with the wrong start time. To do that, you submit one batch update schedule command with JSON that contains the following:

• A payload to remove the original action to activate the image overlay. This action has the incorrect start time.
• A payload to add a new action to activate the same image overlay. This action has the correct start time.
Batch Command in Different Interfaces

The batch update schedule command is represented differently in different interfaces:

- In the AWS CLI, the command is `batch-update-schedule`.
- In the API, the command is represented by an HTTP PUT on `channels/channelId/schedule`.
- In the AWS SDKs, the command is represented by constructs that are suitable to that SDK language.

JSON Payload in Different Interfaces

The JSON payload for the command is different for the different interfaces:

- In the AWS CLI, the contents of the payload depend on how you use the command:
  - You can enter a command with two parameters: `channel-id` and `--cli-input-json`. In this case, you create a file that repeats the channel ID and includes the JSON payload.
  - You can enter a command with three parameters: one for the channel ID, one for the JSON payload for the create actions (if applicable), and one for the JSON payload for the delete actions (if applicable). You pass the payloads in the command. If both parameters are present, each parameter takes a separate payload. But the two payloads are validated and performed as a batch.

  The payload for the AWS CLI is always Pascal case (upper camel case).

- In the API, there is one payload with two sections, a `creates` section and a `deletes` section. A request can contain one or both sections.

  The payload for the API is always camel case for variable names and Pascal case for classes.

- In the AWS SDKs, the JSON payload is represented by constructs that are suitable to that SDK language.

To get more familiar with individual actions, we recommend that you use the MediaLive console to create an action. After you create the action, use the `DescribeSchedule` command in the appropriate interface (for example, the AWS CLI or an SDK) to obtain the raw JSON payload for the entire schedule. You can then copy individual actions and save them as models to use when working programmatically.

Submitting a Batch Update Schedule Command

The command for a batch update schedule command is identical for creating actions, deleting actions, or submitting a combination of create and delete actions. The command is identical. Only the contents of the JSON payload differ.

There are different ways to enter the command to create an action. We recommend that you follow this usage:

- Enter the command with two parameters: `channel-id` and `--cli-input-json`. In this case, you create a file that repeats the channel ID and includes the JSON payload.

The instructions and examples in this section illustrate this usage.

The following rules apply to create actions:

- You can create actions when the channel is running or when it is idle.
- The action must have a UTC start time that is at least 15 seconds in the future (and no more than 14 days in the future). After that cutoff, AWS Elemental MediaLive rejects the create request.
• You can create any number of actions in one request, or any combination of types of actions in one request. For example, you can mix the creation of SCTE-35 message actions and image overlay actions.
• If you create several actions in one request and one of the create requests fails (usually because the start time isn't sufficiently in the future), then they will all fail.

The following rules apply to delete actions:
• You can delete an action when the channel is running or when it is idle.
• The action that you delete must have a UTC start time or a follow start time (for input switches) that is at least 15 seconds in the future. (In other words, it must not be already received in the channel.) After that cutoff, AWS Elemental MediaLive rejects the delete request.
• You can delete any number of actions in one request, or any combination of types of actions in one request. For example, you can mix the deletion of SCTE-35 message actions and image overlay actions.
• If you delete several actions in one request and one of the delete requests fails (usually because the start time isn't sufficiently in the future), then they will all fail.

To submit a batch command
1. Prepare a file that contains the channel ID and the appropriate JSON payload for the actions. For the structure and examples of the JSON payload for different actions, see the sections that follow.
2. Give the file a suitable name with a .txt extension. For example, the file name for a payload that creates only actions might be schedule-create-actions.txt.
3. Save the file to the folder where you are running the AWS CLI.
4. On the command line, enter this command:

   aws medialive batch-update-schedule --channel-id value --cli-input-json value

   • In the value for --channel-id, enter the channel ID as a number.
   • In the value for --cli-input-json, enter the file name in this format:

      file://filename.txt

   For example:

   aws medialive batch-update-schedule --channel-id 999999 --cli-input-json schedule-create-actions.txt

5. To submit the command, press Enter. The response appears on the screen. The response repeats the data from the request.

   If the JSON payload includes create actions and you didn't enter ActionName elements for an action, AWS Elemental MediaLive generates one and includes it in the response.

JSON Payload for Create Actions

The following sections show the structure of the payload and an example of the payload for every type of create action for a MediaLive schedule.

Topics
• Payload for an Input Switch Action (p. 152)
• Payload for an Activate Image Action (p. 154)
• Payload for a Deactivate Overlay Action (p. 155)
Payload for an Input Switch Action

The following sections show the payload for fixed and follow input switch actions.

Payload for a Fixed Input Switch Action

For information about the meaning and values for the fields in the following JSON, see the AWS Elemental MediaLive API Reference:

```json
{
    "ChannelId": "string",
    "Creates": {
        "ScheduleActions": [
            {
                "ScheduleActionStartSettings": {
                    "FixedModeScheduleActionStartSettings": {
                        "Time": "string"
                    }
                },
                "ActionName": "string",
                "ScheduleActionSettings": {
                    "InputSwitchSettings": {
                        "InputAttachmentNameReference": "string"
                    }
                }
            }
        ]
    }
}
```

Payload for a Follow Input Switch Action

For information about the meaning and values for the fields in the following JSON, see the AWS Elemental MediaLive API Reference:

```json
{
    "ChannelId": "string",
    "Creates": {
        "ScheduleActions": [
            {
                "ScheduleActionStartSettings": {
                    "FollowModeScheduleActionStartSettings": {
                        "FollowPoint": "string",
                        "ReferenceActionName": "string"
                    }
                },
                "ActionName": "string",
                "ScheduleActionSettings": {
                    "InputSwitchSettings": {
                        "InputAttachmentNameReference": "string"
                    }
                }
            }
        ]
    }
}
```
Example 1

This example of a request switches to an input at a fixed time. The input is the input attachment with the name `vod_ward_cars_ad`. This name was given to the input when the input was set up as an input attachment in this channel.

```json
{
  "ChannelId": "999999",
  "Creates": {
    "ScheduleActions": [
      {
        "ScheduleActionStartSettings": {
          "FixedModeScheduleActionStartSettings": {
            "Time": "2018-05-21T20:42:19.000Z"
          }
        },
        "ActionName": "ad_002",
        "ScheduleActionSettings": {
          "InputSwitchSettings": {
            "InputAttachmentNameReference": "vod_ward_cars_ad"
          }
        }
      }
    ]
  }
}
```

Example 2

This example shows a request that contains two input switches. The first switch is the same as example 1. The second switch (which starts at the second "ScheduleActionStartSettings" line) is set up to follow the first switch. The switch will occur when the previous action ("ad_002") ends. In order for a follow switch to work, the previous action must have an input that is a file input. The second switch can be a file input or live input. In this case, it is a live input, as the input name implies:

```json
{
  "ChannelId": "999999",
  "Creates": {
    "ScheduleActions": [
      {
        "ScheduleActionStartSettings": {
          "FixedModeScheduleActionStartSettings": {
            "Time": "2018-05-21T20:42:19.000Z"
          }
        },
        "ActionName": "ad_002",
        "ScheduleActionSettings": {
          "InputSwitchSettings": {
            "InputAttachmentNameReference": "vod_ward_cars_ad"
          }
        }
      }
    ],
    {
      "ScheduleActionStartSettings": {
        "FollowModeScheduleActionStartSettings": {
          "FollowPoint": "END"
        }
      }
    }
  }
}
```
"ReferenceActionName": "ad_002"
}
},
"ActionName": "end_ad_block",
"ScheduleActionSettings": {
"InputSwitchSettings": {
"InputAttachmentNameReference": "live_studio_feed"
}
}
"
]
}
"

Payload for an Activate Image Action

For information about the meaning and values for the fields in the following JSON, see the AWS Elemental MediaLive API Reference:

```
{
  "ChannelId": "string",
  "Creates": {
    "ScheduleActions": [
    {
      "ScheduleActionStartSettings": {
        "FixedModeScheduleActionStartSettings": {
          "Time": "string"
        }
      },
      "ActionName": "string",
      "ScheduleActionSettings": {
        "StaticImageActivateSettings": {
          "Duration": "integer",
          "Image": {
            "PasswordParam": "string",
            "Uri": "string",
            "Username": "string"
          },
          "FadeOut": "integer",
          "ImageY": "integer",
          "FadeIn": "integer",
          "ImageX": "integer",
          "Width": "integer",
          "Opacity": "integer",
          "Layer": "integer",
          "Height": "integer"
        }
      }
    }
    ]
  }
}
```

**Example**

This example of a request creates an image overlay using a file that is stored in an Amazon S3 bucket. The request doesn't include a duration and therefore doesn't include a fadeout. Instead, the intention is to send a separate deactivate request at the appropriate time. All the times are in milliseconds, and all the positioning values are in pixels:

```
{
  "ChannelId": "999999",
```
"Creates": {
  "ScheduleActions": [
    {
      "ScheduleActionStartSettings": {
        "FixedModeScheduleActionStartSettings": {
          "Time": "2018-05-21T20:42:19.000Z"
        }
      },
      "ActionName": "corporate_logo_030",
      "ScheduleActionSettings": {
        "StaticImageActivateSettings": {
          "Image": {
            "PasswordParam": "corplogo!2312",
            "Uri": "s3ssl://logos/corporate/high-res.bmp",
            "Username": "medialiveoperator"
          },
          "ImageY": 300,
          "FadeIn": 1500,
          "ImageX": 200,
          "Width": 800,
          "Opacity": 60,
          "Layer": 1,
          "Height": 900
        }
      }
    }
  ]
}

Payload for a Deactivate Overlay Action

For information about the meaning and values for the fields in the following JSON, see the AWS Elemental MediaLive API Reference:

{
  "ChannelId": "string",
  "Creates": {
    "ScheduleActions": [
      {
        "ScheduleActionStartSettings": {
          "FixedModeScheduleActionStartSettings": {
            "Time": "string"
          }
        },
        "ActionName": "string",
        "ScheduleActionSettings": {
          "StaticImageDeactivateSettings": {
            "FadeOut": "integer",
            "Layer": "integer"
          }
        }
      }
    ]
  }

Example

This example of a request creates an action to end an image overlay at 20:42:04.000 (UTC) with a 500 millisecond fadeout that is added onto the end time, which means that the overlay will be invisible at 20:42:04.500:
Payload for a Splice_Insert Message

For information about the meaning and values for the fields in the following JSON, see the AWS Elemental MediaLive API Reference:

```json
{
  "ChannelId": "string",
  "Creates": {
    "ScheduleActions": [
      {
        "ScheduleActionStartSettings": {
          "FixedModeScheduleActionStartSettings": {
            "Time": "string"
          }
        },
        "ActionName": "string",
        "ScheduleActionSettings": {
          "Scte35SpliceInsertSettings": {
            "SpliceEventId": "integer",
            "Duration": "integer"
          }
        }
      }
    ]
  }
}
```

Example

This example of a request creates an action for a splice_insert with a UTC start time of 20:42:04.000. It also has an ActionName that perhaps references an ad avail from your database, a unique integer for the splice event ID, and a duration of 1,350,000 kHz ticks (15 seconds):

```json
{
  "ChannelId": "999999",
  "Creates": {
    "ScheduleActions": [
      {
        "ScheduleActionStartSettings": {
          "FixedModeScheduleActionStartSettings": {
            "Time": "2018-05-21T20:42:04.000Z"
          }
        },
        "ActionName": "stop_overlay_029",
        "ScheduleActionSettings": {
          "StaticImageDeactivateSettings": {
            "FadeOut": 500,
            "Layer": 1
          }
        }
      }
    ]
  }
}
```
"FixedModeScheduleActionStartSettings": {  
  "Time": "2018-05-21T20:42:04.000Z"  
},  
"ActionName": "adavail_3708",  
"ScheduleActionSettings": {  
  "Scte35SpliceInsertSettings": {  
    "SpliceEventId": 3708,  
    "Duration": 1350000  
  }  
},  
"ScheduleActionSettings": {  
  "Scte35TimeSignalSettings": {  
    "Scte35Descriptors": [  
      "Scte35DescriptorSettings": {  
        "SegmentationDescriptorScte35DescriptorSettings": {  
          "SubSegmentsExpected": "integer",  
          "SegmentationEventId": "integer",  
          "SegmentationDuration": "integer",  
          "SegmentationCancelIndicator": "enum",  
          "SubSegmentNum": "integer",  
          "SegmentationUpidType": "integer",  
          "SegmentNum": "integer",  
          "SegmentationCancelIndicator": "enum",  
          "DeliveryRestrictions": {  
            "DeviceRestrictions": "enum",  
            "WebDeliveryAllowedFlag": "enum",  
            "NoRegionalBlackoutFlag": "enum",  
            "ArchiveAllowedFlag": "enum"  
          },  
          "SegmentationUpid": "string",  
          "SegmentationTypeId": "integer",  
          "SegmentsExpected": "integer"  
        }  
      }  
    ]  
  }  
}  
}
Example

This example of a request creates an action for a time_signal with a UTC start time of 20:42:04.000 and with a unique integer for SegmentationEventId. For the restrictions fields, NoRegionalBlackoutFlag has a restriction set (regional blackouts are in place):

```json
{
  "ChannelId": "999999",
  "Creates": {
    "ScheduleActions": [
      {
        "ScheduleActionStartSettings": {
          "FixedModeScheduleActionStartSettings": {
            "Time": "2018-05-21T20:42:04.000Z"
          }
        },
        "ActionName": "adavail_3708",
        "ScheduleActionSettings": {
          "Scte35TimeSignalSettings": {
            "Scte35Descriptors": [
              {
                "Scte35DescriptorSettings": {
                  "SegmentDescriptorScte35DescriptorSettings": {
                    "SubSegmentsExpected": 0,
                    "SegmentationEventId": 7054,
                    "SegmentationDuration": 1350000,
                    "SegmentationCancelIndicator": 0,
                    "SubSegmentNum": 0,
                    "SegmentationUpidType": 12,
                    "SegmentNum": 0,
                    "SegmentationCancelIndicator": "SEGMENTATION_EVENT_NOT_CANCELED",
                    "DeliveryRestrictions": {
                      "DeviceRestrictions": "NONE",
                      "WebDeliveryAllowedFlag": "WEB_DELIVERY_ALLOWED",
                      "NoRegionalBlackoutFlag": "REGIONAL_BLACKOUT",
                      "ArchiveAllowedFlag": "ARCHIVE_ALLOWED"
                    },
                    "SegmentationUpid": "4a414e3136494e4155303031",
                    "SegmentationTypeId": 52,
                    "SegmentsExpected": 0
                  }
                }
              }
            ]
          }
        }
      }
    ]
  }
}
```

Payload for a Return-to-Network Message

For information about the meaning and values for the fields in the following JSON, see the AWS Elemental MediaLive API Reference:

```json
{
  "ChannelId": "string",
  "Creates": {
```
Example

This example of a request creates a return-to-network with a UTC start time of 20:42:19:

```
{
   "ChannelId": "999999",
   "Creates": {
      "ScheduleActions": [
         {
            "ScheduleActionStartSettings": {
               "FixedModeScheduleActionStartSettings": {
                  "Time": "2018-05-21T20:42:19.000Z"
               }
            },
            "ActionName": "end_adavail_3708",
            "ScheduleActionSettings": {
               "Scte35ReturnToNetworkSettings": {}
            }
         }
      ]
   }
}
```

Payload for an ID3 Metadata Item

For information about the meaning and values for the fields in the following JSON, see the AWS Elemental MediaLive API Reference:
Example

This example of a request creates ID3 metadata to be inserted at 13:35:59 UTC:

```json
{
  "ChannelId": "999999",
  "Creates": {
    "ScheduleActions": [
      {
        "ScheduleActionStartSettings": {
          "FixedModeScheduleActionStartSettings": {
            "Time": "2019-01-02T13:35:59Z"
          }
        },
        "ActionName": "id3_metadata.2019-01-02T13:35:59Z",
        "ScheduleActionSettings": {
          "HlsTimedMetadataSettings": {
            "Id3": "SUQzBAAAAAAAFVRYWFgAAAALAABIZWxsbyBXb3JsZVAA=="
          }
        }
      }
    ]
  }
}
```

Payload for a Pause Pipeline Action

For information about the meaning and values for the fields in the following JSON, see the [AWS Elemental MediaLive API Reference](#):
Example: Pausing One Pipeline

This example of a request pauses pipeline 0 at 20:42:19 UTC. MediaLive always reads the command as "set the specified pipeline or pipelines to pause and set all other pipelines as unpaused."

```
{
    "ChannelId": "999999",
    "Creates": {
        "ScheduleActions": [
            {
                "ScheduleActionStartSettings": {
                    "FixedModeScheduleActionStartSettings": {
                        "Time": "2018-05-21T20:42:19Z"
                    }
                },
                "ActionName": "pause_pipeline_0_now",
                "ScheduleActionSettings": {
                    "PauseStateSettings": {
                        "Pipelines": [
                            {
                                "PipelineId": "PIPELINE_0"
                            }
                        ]
                    }
                }
            }
        ]
    }
}
```

Example: Unpausing Both Pipelines

This example of a request unpauses all pipelines that are currently paused. Note that the Pipelines array is empty. MediaLive interprets this empty array as "set all pipelines to unpaused."

```
{
    "ChannelId": "999999",
    "Creates": {
        "ScheduleActions": [
            {
                "ScheduleActionStartSettings": {
                    "FixedModeScheduleActionStartSettings": {
                        "Time": "2018-05-21T20:52:00Z"
                    }
                },
                "ActionName": "unpause_pipeline_0",
                "ScheduleActionSettings": {
                    "PauseStateSettings": {
                        "Pipelines": [
                            {
                                "PipelineId": "PIPELINE_0"
                            }
                        ]
                    }
                }
            }
        ]
    }
}
```
Combination of Create Actions

Here is an example of a JSON body to pass into the \texttt{--creates} parameter of the \texttt{batch-update-schedule} AWS CLI command. It contains two actions to create. In this example, both actions are \texttt{splice_inserts}, but in fact you can combine any number and any type of create actions:

```

``` `{ 
   "ScheduleActions": [ 
      { 
        "ScheduleActionSettings": { 
          "Scte35SpliceInsertSettings": { 
            "Duration": 1350000, 
            "SpliceEventId": 3 
          } 
        }, 
        "ActionName": "SpliceInsert-01", 
        "ScheduleActionStartSettings": { 
          "FixedModeScheduleActionStartSettings": { 
            "Time": "2018-11-05T16:10:30.000Z" 
          } 
        } 
      }, 
      { 
        "ScheduleActionSettings": { 
          "Scte35SpliceInsertSettings": { 
            "Duration": 2700000, 
            "SpliceEventId": 3 
          } 
        }, 
        "ActionName": "SpliceInsert-02", 
        "ScheduleActionStartSettings": { 
          "FixedModeScheduleActionStartSettings": { 
            "Time": "2018-11-05T16:30:45.000Z" 
          } 
        } 
      } 
   ] 
}``

JSON Payload for Delete Actions

In the \texttt{Deletes} section, include the list of actions to delete by entering an array of \texttt{ActionNames}. The array contains one or more action names. You can obtain these action names using the \texttt{DescribeChannel} command (see the section called “Viewing the Schedule” (p. 164):

```

``` `{ 
   "ChannelId": "string", 
   "Deletes": { 
      "ActionNames": [ 
      ] 
   } 
}``

Example

This example of a request deletes the three actions identified by \texttt{ActionNames} that were assigned when you created the actions:

```

``` { }
"ChannelId": "999999",
"Deletes": {
  "ActionNames": [
    "stop_overlay_33",
    "adavail_3711",
    "end_adavail_3711"
  ]
}
}

JSON Payload for Combining Create and Delete

To combine a batch of creates and deletes, include both a Creates section and a Deletes section in the JSON payload.

In this example, the payload in the Deletes section removes an action to activate an image overlay because it has an incorrect start time. The action is named overlay-21. The payload in the Creates section inserts that action again, this time with the correct start time.

Even though the Creates section appears first in the JSON payload, MediaLive always executes the delete actions first.

In this action, the delete action and the create action have the same ActionName. The name is being reused because the batch is a "delete and replace." But you could assign a different name to the create action:

```json
{
  "ChannelId": "999999",
  "Creates": {
    "ScheduleActions": [
      { "ScheduleActionStartSettings": {
        "FixedModeScheduleActionStartSettings": {
          "Time": "2018-05-21T20:42:19.000Z"
        }
      },
      "ActionName": "overlay-21",
      "ScheduleActionSettings": {
        "StaticImageActivateSettings": {
          "Image": {
            "PasswordParam": "imagespassword",
            "Uri": "s3ssl://banners/banner_A/high-res.bmp",
            "Username": "medialiveoperator"
          },
          "ImageY": 300,
          "FadeIn": 1500,
          "ImageX": 200,
          "Width": 800,
          "Opacity": 60,
          "Layer": 1,
          "Height": 900
        }
      }
    ]
  },
  "Deletes": {
    "ActionNames": ["overlay-21"]
  }
}
```
Viewing the Schedule (AWS CLI)

You can use the AWS CLI to view a list of the actions that are currently in the schedule for one channel:

- Actions that have not yet been executed in the channel
- Actions that have been executed within the last hour

To view the schedule, you use the DescribeSchedule command. This command is represented differently in different interfaces:

- In the AWS CLI, the command is `describe-schedule`.
- In the API, the command is represented by an HTTP GET on `channels/channelId/schedule`.
- In the AWS SDKs, the command is represented by constructs that are suitable to that SDK language.

To view actions (AWS CLI)

1. Enter this command:
   ```bash
   aws medialive describe-schedule --channel-id value --max-results value
   ```
2. To submit the command, press Enter. The response appears on the screen.
3. If you used the `--max-results` option and the response included NextToken, enter the DescribeChannel command and pass the value of NextToken in `--next-token`. For example:
   ```bash
   aws medialive describe-schedule --channel-id value --next-token 3jhrprd0
   ```
4. To submit the command, press Enter. The response appears on the screen.

Example

The JSON body of the command response is similar to that of the BatchUpdateSchedule command request.

This example of a response shows the following actions:

- An action with the ActionName "corporate_logo_029" to activate an image overlay in layer 1 at 20:30:00 UTC
- An action with the ActionName "stop_overlay_029" to deactivate the overlay in layer 1 at 20:42:04 UTC
- An action with the ActionName "adavail_3708" to insert a splice_insert at the same time as the deactivate action
- An action with the ActionName "end_adavail_3708" to return-to-network 15 seconds later, at 20:42:19 UTC
- An action with the ActionName "corporate_logo_030" to reactivate the same overlay in layer 1 at the same time as the return

This schedule describes a workflow where you generally show your corporate logo, but you remove it at the start of each ad avail and then display it again at the end of the ad avail:

```json
{
  "NextToken": "3jhrprd0",
  "ScheduleActions": [
    {
      "ScheduleActionStartSettings": {
```
"FixedModeScheduleActionStartSettings": {  
  "Time": "2018-05-21T20:30:00.000Z"
},
"ActionName": "corporate_logo_029",
"ScheduleActionSettings": {  
  "StaticImageActivateSettings": {  
    "Image": {  
      "PasswordParam": "corplogo!2312",
      "Uri": "s3ssl://logos/corporate/high-res.bmp",
      "Username": "medialiveoperator"
    },
    "ImageY": 300,
    "FadeIn": 1500,
    "ImageX": 200,
    "Width": 800,
    "Opacity": 60,
    "Layer": 1,
    "Height": 900
  }
},
"ScheduleActionStartSettings": {  
  "FixedModeScheduleActionStartSettings": {  
    "Time": "2018-05-21T20:42:04.000Z"
  }
},
"ActionName": " stop_overlay_029",
"ScheduleActionSettings": {  
  "StaticImageDeactivateSettings": {  
    "FadeOut": 1500,
    "Layer": 1
  }
},
"ScheduleActionStartSettings": {  
  "FixedModeScheduleActionStartSettings": {  
    "Time": "2018-05-21T20:42:04.000Z"
  }
},
"ActionName": "adavail_3708",
"ScheduleActionSettings": {  
  "Scte35SpliceInsertSettings": {  
    "SpliceEventId": 3708,
    "Duration": 1350000
  }
},
"ScheduleActionStartSettings": {  
  "FixedModeScheduleActionStartSettings": {  
    "Time": "2018-05-21T20:42:19.000Z"
  }
},
"ActionName": "end_adavail_3708",
"ScheduleActionSettings": {  
  "Scte35ReturnToNetworkSettings": {  
    "SpliceEventId": 3708
  }
},
"ScheduleActionStartSettings": {  
  "FixedModeScheduleActionStartSettings": {  
    "Time": "2018-05-21T20:42:19.000Z"
  }
}
Working with Captions

You can set up the AWS Elemental MediaLive channel to extract captions when it ingests the source, and to include those captions in the output in either the same or a different format. You can include several captions in the output. For example, you can include captions for several languages. You can take a source captions asset and convert it to one format in one output and to another format in a different output.

You perform the setup for captions in your AWS Elemental MediaLive channel.

By default, AWS Elemental MediaLive does not ingest any captions (not even captions that are embedded in the video). You must explicitly identify the captions to ingest and the captions to output.

Note
The information in this captions section assumes that you are familiar with the general steps for creating a channel, as described in Creating a Channel from Scratch (p. 86). It also assumes that you have started creating a channel, including associating an input with the channel.

Supported Features

This section provides information on the different features of captions that AWS Elemental MediaLive supports.

Topics
- Supported Features (p. 167)
- Typical Scenarios (p. 170)
- Setting Up for Captions (p. 172)
- Examples (p. 182)

Supported Formats

AWS Elemental MediaLive supports specific formats in inputs and specific formats in outputs. See the section called “General Information About Supported Formats” (p. 229) for a table that lists the supported captions formats, with a reference to the standard that defines that format. The table specifies whether the format is supported as input or output or both.

Format Support by Output Container

There are several factors that control your ability to include captions of a specific format in your outputs:
• **The type of input container** – A given input container can contain captions in some formats and not in others.

• **The format of the input captions** – A given format of captions can be converted to some formats and not to others.

• **The type of output containers** – A given output container supports some captions formats and not others.

For example, if your input container is an MP4 container and your output is HLS, and you want to include Web-VTT captions in the HLS output, you can do so if the MP4 container holds 608 embedded captions. But you can't include Web-VTT captions if the MP4 container holds Ancillary captions.

For more information about all the supported combinations of input container, input format, and output container, see [Reference: Supported Captions](p. 229).

**Support for Languages**

If the source includes captions in multiple languages, you can include multiple languages in the output as follows:

• **Embedded passthrough**– For any of the embedded source formats, if you specify embedded as the output format, all languages that are in the input are included in the output. You can't remove any of the languages.

• **Embedded In, Other Out**– For any of the embedded source formats, if you are setting up for “embedded in, other out,” you can specify which languages to extract and include in an output.

• **Teletext passthrough**– For a Teletext source, if you specify Teletext as the output, then all languages (pages) are included in the output. You can't strip out any languages. In fact, the entire Teletext content is included in the output; you can't strip out any of the pages. Furthermore, Teletext passthrough is supported only in TS outputs.

• **Teletext In, Other Out**– For a Teletext source, if you are setting up for “Teletext in, other out,” you can specify which languages (Teletext pages) to extract and which languages to include in an output.

• **ARIB passthrough** – For an ARIB source, the only possible output is ARIB. All the languages that are in the input are included in the output. You can't strip out any languages.

• **Any Other Combination**– For all other sources, you always specify the language to extract from the input and the language to include in an output, regardless of the source format and output format.

**Support for Font Styles in Output Captions**

Depending on the scenario, there are three possibilities for the font style for output captions:

• You can specify the style that you want for fonts, including color, outline, and background color.

• The font styles in the input are passed through.

• The font styles are controlled by the downstream player.

**Font Style Options**

<table>
<thead>
<tr>
<th>Source Captions</th>
<th>Output Captions</th>
<th>Options for Font Style</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARIB</td>
<td>ARIB</td>
<td>None. The font styles in the input are automatically passed through in the output.</td>
</tr>
<tr>
<td>Any supported captions format</td>
<td>Burn-in</td>
<td>You can specify font styles in the output. If you don't specify</td>
</tr>
</tbody>
</table>
Captions in Channels with Multiple Inputs

If your channel includes multiple inputs, these rules apply to the handling of captions:

- The captions formats in one input can be different from the captions formats in another input. For example, you can have 608 embedded captions in one input and Teletext in another. Or you can have DVB-Sub in one input and no captions in another.
- There is no requirement that all the inputs have captions that are capable of producing the specified captions in any given output.

If the captions from an input cannot produce the specified captions in one of the outputs, the captions will be omitted for the duration of that input. The channel will not fail. When the channel switches to a different input, the captions will be included again if the captions from that input can produce the specified captions in that output.

Example 1

You might have an RTP input as your primary live feed, and an MP4 as one of the other inputs. The RTP input might have Teletext captions. The MP4 input might have 608 embedded captions.
• You might want to produce DVB-Sub captions in a UDP output. This is possible for both the input that has a Teletext source and the input that has a 608 embedded source.
• You might want to produce Web-VTT captions in an HLS output. This is possible for both the input that has a Teletext source and the input that has a 608 embedded source.
• You might want to produce 608 embedded captions in an Archive output. This is not possible when the RTP input is active because Teletext captions can't be converted to 608 embedded captions. For that period, there will be no captions included in the Archive output.

Example 2

You might have an RTP input as your primary live feed, and an MP4 as one of the other inputs. The RTP input might have Teletext captions. The MP4 input might have DVB-Sub captions.

• You might want to produce Teletext captions in a UDP output. This is possible when the RTP input is active, but it is not possible when the MP4 input is active because DVB-Sub captions can't be converted to Teletext captions. For that period, there will be no captions included in the UDP output.

General Observations

The captions that are most restricted are the following:

• **ARIB in the input or output** – ARIB can only be passed through (ARIB in, ARIB out).
• **SCTE-27 in the input** – It can be converted only to SMPTE-TT. However, it is not the only captions format that can be converted to SMPTE-TT, so it might work for your workflow.
• **Teletext in the output** – It can be produced only from Teletext input (Teletext in, Teletext out).

For the recommended procedure for setting up multiple inputs, see the section called “Creating a Channel with Multiple Inputs” (p. 195).

Typical Scenarios

Following are some sample use cases. The use cases are ordered from less to more complicated. They are intended to illustrate many of the capabilities of MediaLive.

Topics

• Use Case A: One Input Format to One Output and Not Converted (p. 170)
• Use Case B: One Input Format Converted to One Different Format in One Output (p. 171)
• Use Case C: One Input Format Converted to Different Formats, One Format for Each Output (p. 171)
• Use Case D: One Captions Output Shared by Multiple Video Encodes (p. 172)

Use Case A: One Input Format to One Output and Not Converted

The input is set up with one format of captions and two or more languages. Assume that you want to maintain the format in the output, and that you want to produce only one type of output and to include all the languages in that output.

For example, the input has embedded captions in English and French. You want to produce HLS output that includes embedded captions in both English and French.
Use Case B: One Input Format Converted to One Different Format in One Output

The input is set up with one format of captions and two or more languages. You want to convert the captions to a different format in the output. You want to produce only one type of output and include all the languages in that output.

For example, the input has embedded captions in German and French. You want to convert the captions to DVB-Sub and include these captions in both languages in a UDP output.

Use Case C: One Input Format Converted to Different Formats, One Format for Each Output

The input is set up with one format of captions and two or more languages. Assume that you want to produce several different types of output, and that in each output you want to convert the captions to a different format but include all the languages.

For example, the input has Teletext captions in Czech and Polish. You want to produce a Microsoft Smooth output and an HLS output. In the Microsoft Smooth output, you want to convert both captions to TTML. In the HLS output, you want to convert both captions to Web-VTT.
Use Case D: One Captions Output Shared by Multiple Video Encodes

This use case deals with captions in an ABR workflow.

For example, assume that there are three video/audio media combinations: one for low-resolution video, one for medium, and one for high. Assume that there is one output captions asset (English and Spanish embedded) that you want to associate with all three video/audio media combinations.

Setting Up for Captions

When you create a channel, you must specify the format of the input captions, then specify the desired format of the captions for every output. When you save the channel, your choices are validated according to the supported combinations of input container, source captions format, and output container.

Topics
- Step 1: Create Captions Selectors in the Input (p. 172)
- Step 2: Plan Captions for the Outputs (p. 176)
- Step 3: Match Formats to Categories (p. 177)
- Step 4: Create Captions Encodes (p. 179)

Step 1: Create Captions Selectors in the Input

You must identify the captions that you want to use and assign each to a captions selector. If you don't create any captions selectors, you can't include captions in the output. All the captions will be removed from the media.

Then you must extract the captions that you want by adding a captions selector in the channel. Each extracted captions asset is contained in one captions selector. For example, one selector contains the Teletext captions in Czech.

To identify the captions that you want

1. Identify which captions are in the input (the provider of the input should provide you with this information). Identify the captions formats and, for each format, the languages.
2. Identify which of those formats and languages that you want to use.
3. Determine how many captions selectors to create in the input in the channel, using the following guidance:

- For embedded passthrough, create a single captions selector for all languages. All languages are passed through; there is no other option. For details, see the section called “Information for Embedded” (p. 174).
- For Teletext passthrough, create a single captions selector for all languages (in fact, one captions selector for the entire content). All languages are passed through; there is no other option. For details, see the section called “Information for Teletext” (p. 175).
- In all other cases, create one captions selector for each language and format combination.

You end up with a list of captions selectors to create. For example:

- Captions Selector 1: Teletext captions in Czech
- Captions Selector 2: Teletext captions in Polish

To create a captions selector

1. In the channel that you are creating, in the navigation pane, in Input attachments, choose the input.
2. For General input settings, choose Add captions selectors.
3. For Captions selector name, enter a suitable name. For example, Teletext Czech.
4. For Selector settings, choose the format of the source captions.
5. For most formats, more fields appear. For details about a field, choose the Info link next to the field. In addition, see DVB-Sub or SCTE-27 (p. 173), Embedded (p. 174), or Teletext (p. 175).
6. Create more captions selectors, as required.

Information for DVB-Sub or SCTE-27

DVB-Sub and SCTE-27 formats are supported only in TS inputs. You must specify the location of the captions by completing the PID or Language code fields in one of these ways.

<table>
<thead>
<tr>
<th>PID</th>
<th>Language Code</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specified</td>
<td>Blank</td>
<td>Extracts captions from the specified PID.</td>
</tr>
<tr>
<td>Blank</td>
<td>Specified</td>
<td>Extracts the specified language, whichever PID that happens to be in.</td>
</tr>
<tr>
<td>Specified</td>
<td>Specified</td>
<td>Extracts captions from that PID; the language is informational.</td>
</tr>
<tr>
<td>Blank</td>
<td>Blank</td>
<td>Valid only if the source is DVB-Sub and the output is DVB-Sub. With this combination of PID and Language, all input DVB-Sub PIDs are included in the output. Not valid for SCTE-27.</td>
</tr>
</tbody>
</table>
Information for Embedded

Read this section if the input captions are any of the following: embedded (EIA-608 or CEA-708), embedded+SCTE-20, SCTE-20+embedded, or SCTE-20.

How Many Captions Selectors?

- If you are setting up for embedded passthrough, create only one captions selector. With this scenario, all languages are automatically extracted and are automatically included in the output.
- If you are setting up for embedded-to-other, create one captions selector for each language that you want to include in the output, to a maximum of four selectors.
- If you are setting up for embedded passthrough in some outputs and embedded-to-other in other outputs, create one captions selector for each language that you want to include in the output, to a maximum of four selectors. Don’t worry about a selector for the embedded passthrough output. MediaLive extracts all the languages for that output, even though there is not a selector to explicitly specify this action.

Captions Selector Fields

- **Selector settings:**
  - Choose embedded if the source captions are embedded (EIA-608 or CEA-708), embedded+SCTE-20, or SCTE-20+embedded.
  - Choose SCTE-20 if the source captions are SCTE-20 alone.

- **EIA-608 track number** – This field specifies the language to extract. Complete as follows:
  - If you are setting up for embedded passthrough only (you are creating only one captions selector for the input embedded captions), this field is ignored, so keep the default.
  - If you are converting embedded to another format (you are creating several captions selectors, one for each language), specify the number of the CC instance (from the input) that holds the language that you want.

- **Convert 608 to 708:** The embedded source captions can be EIA-608 captions, CEA-708 captions, or both EIA-608 and CEA-708. You can specify how you want these captions to be handled when AWS Elemental MediaLive is ingesting content. The following table describes the behavior for various scenarios.

<table>
<thead>
<tr>
<th>EIA-608 in Source</th>
<th>CEA-708 in Source</th>
<th>Convert Field</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
<td>No</td>
<td>Upconvert</td>
<td>CEA-708 data is created based on the EIA-608 data. EIA-608 data is added as 608-compatibility bits in the CEA-708 data.</td>
</tr>
<tr>
<td>Yes</td>
<td>No</td>
<td>Disabled</td>
<td>Original EIA-608 is preserved.</td>
</tr>
<tr>
<td>No</td>
<td>Yes</td>
<td>Upconvert</td>
<td>Original CEA-708 is preserved.</td>
</tr>
<tr>
<td>No</td>
<td>Yes</td>
<td>Disabled</td>
<td>Original CEA-708 is preserved.</td>
</tr>
</tbody>
</table>
### Step 1: Create Captions Selectors in the Input

<table>
<thead>
<tr>
<th>EIA-608 in Source</th>
<th>CEA-708 in Source</th>
<th>Convert Field</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
<td>Yes</td>
<td>Upconvert</td>
<td>CEA-708 data is discarded. New CEA-708 data is created based on the EIA-608 data. EIA-608 data is added as 608-compatibility bits in the CEA-708 data. The new CEA-708 data will not include any CEA-708 formatting features. Not recommended.</td>
</tr>
<tr>
<td>Yes</td>
<td>Yes</td>
<td>Disabled</td>
<td>Original EIA-608 is preserved and original CEA-708 is preserved.</td>
</tr>
</tbody>
</table>

- **SCTE-20 detection** – If the source captions combine embedded (EIA-608 or CEA-708) and SCTE-20, you might want to set this field to **Auto**. AWS Elemental MediaLive gives preference to the 608/708 embedded captions but switches to use the SCTE-20 captions when necessary. If you set this field to **Off**, AWS Elemental MediaLive never uses the SCTE-20 captions.

### Information for Teletext

Teletext is a form of data that can contain several types of information, not just captions. Teletext can be handled in one of the following ways:

- If you want to include the entire Teletext input, you must set up for Teletext passthrough. The entire Teletext can never be converted to another format. Teletext passthrough is supported only in a TS output.
- Individual captions pages (the captions in a specific language) can be extracted and converted to another captions format.
- Individual captions pages (the captions in a specific language) **cannot** be extracted and kept in Teletext. If you want to extract individual captions pages, you must convert them to another format.

### How Many Captions Selectors?

- If you are setting up for Teletext passthrough captions, create only one captions selector, even if you want to include multiple languages in the output. With this scenario, all languages are automatically extracted and are automatically included in the output.
- If you are setting up for Teletext-to-other, create one captions selector for each language that you want to include in the output. For example, one selector to extract English Teletext, and one selector to extract Swedish Teletext.
- If you are setting up for Teletext passthrough in some outputs and Teletext-to-other in other outputs, create one captions selector for each language that you want to include in the output. Don't worry about a selector for the passthrough output. MediaLive passes through all the data, even though there is not a selector to explicitly specify this action.

### Captions Selector Fields
• **Selector settings** – Choose Teletext.

• **Page number** – This field specifies the page of the desired language. Complete as follows:
  - If you are setting up for Teletext passthrough captions (you are creating only one captions selector for the input captions), keep the field blank. The value is ignored.
  - If you are converting Teletext to another format (you are creating several captions selectors, one for each language), specify the page for the language that you want. If you leave this field blank, you get a validation error when you save the channel.

---

**Step 2: Plan Captions for the Outputs**

If you followed the instructions in the section called “Step 1: Create Captions Selectors in the Input” (p. 172), you should have a list of the captions formats and languages that are available for inclusion in the outputs.

You must now plan the captions information for the outputs.

**To plan the captions for the output**

1. Identify the types of output media that you plan to create in the channel, for example, Microsoft Smooth and HLS.
2. Identify the combinations of video and audio that you plan to create for each output media.
3. For each output media, identify which input captions will be converted to which output formats. For example, you will convert Teletext captions to TTML for the Microsoft Smooth output media, and those same Teletext captions to Web-VTT for the HLS output media.

   The output formats that are possible depend on the input formats and the type of output media. To determine which output captions are possible given the input format, see Reference: Supported Captions (p. 229).

4. Identify the languages for each output format:
   - In general, count each language separately.
   - Exception: For embedded passthrough, count all languages as one.
   - Exception: For Teletext passthrough, count all languages as one.

**The Result**

You end up with a list of outputs, and the captions formats and languages for each output. For example:

- Microsoft Smooth output with TTML captions in Czech
- Microsoft Smooth output with TTML captions in Polish
- HLS output with Web-VTT captions in Czech
- HLS output with Web-VTT captions in Polish

**Outputting Multiple Formats**

You can include captions from two or more different formats in an output. For example, you can include both embedded captions and Web-VTT captions in an HLS output, to give the downstream system more choices about which captions to use. The only rules for multiple formats are the following:

- The output container must support all the formats. See Reference: Supported Captions (p. 229).
- The font styles in all the captions that are associated with an output must match. This means that the end result must be identical, not that you must use the same option to get that result. For example,
all captions that are associated with the output must be white for the first language and blue for the second language.

Managing this style matching can be a little tricky. For information about the font style options, see Support for Font Styles in Output Captions (p. 168).

**Step 3: Match Formats to Categories**

There are different procedures to follow to create captions encodes in the output. The correct procedure depends on the “category” that the output captions belong to. There are five categories of captions, described after the table.

On the list of outputs that you have created, make a note of the category that each captions option belongs to.

<table>
<thead>
<tr>
<th>Captions Format</th>
<th>Category of This Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARIB</td>
<td>Object</td>
</tr>
<tr>
<td>Burn-in</td>
<td>Burn-in</td>
</tr>
<tr>
<td>DVB-Sub</td>
<td>Object</td>
</tr>
<tr>
<td>Embedded</td>
<td>Embedded</td>
</tr>
<tr>
<td>Embedded+SCTE-20</td>
<td>Embedded</td>
</tr>
<tr>
<td>RTMP CaptionInfo</td>
<td>Object</td>
</tr>
<tr>
<td>SCTE-20+Embedded</td>
<td>Embedded</td>
</tr>
<tr>
<td>SCTE-27</td>
<td>Object</td>
</tr>
<tr>
<td>SMPTE-TT</td>
<td>Stream in Microsoft Smooth</td>
</tr>
<tr>
<td>Teletext</td>
<td>Object</td>
</tr>
<tr>
<td>TTML</td>
<td>Sidecar</td>
</tr>
<tr>
<td>Web-VTT</td>
<td>Sidecar</td>
</tr>
</tbody>
</table>

For example, your list of outputs might now look like this:

- Microsoft Smooth output with TTML captions (sidecar) in Czech
- Microsoft Smooth output with TTML captions (sidecar) in Polish
- HLS output with Web-VTT captions (sidecar) in Czech
- HLS output with Web-VTT captions (sidecar) in Polish

**Embedded in Video**

The captions are carried inside the video encode, which is itself in an output in the output group. There is only ever one captions asset within that video encode, although that asset might contain captions for several languages.
Captions Object

All the captions encodes for a given output group are in the same output as the corresponding video and audio.

Sidecar

Each captions encode for a given output group is in its own "captions-only" output. The output group can contain more than one captions output, for example, one for each language.

Stream

Each captions encode for a given output group is in its own "captions-only" output. The output group can contain more than one captions output, for example, one for each language.
Burn-in

The captions are converted into text and then overlaid on the picture directly in the video encode. Strictly speaking, once the overlay occurs, these are not really captions because they are indistinguishable from the video.

Step 4: Create Captions Encodes

Go through the list of outputs that you created and set up the captions in each output group, one by one.

Follow the procedure that applies to the format category of the captions output:

- the section called “All Captions Except Sidecar or SMPTE-TT in Microsoft Smooth” (p. 179)
- the section called “Sidecar Captions and SMPTE-TT in Microsoft Smooth” (p. 180)

All Captions Except Sidecar or SMPTE-TT in Microsoft Smooth

Follow this procedure if the format of the captions asset that you want to add belongs to the category of embedded, burn-in, or object. You set up the captions and video and audio in the same output.

To set up the output captions

1. In the channel that you are creating, in the navigation pane, find the output group (which you have already created). For example, find the HLS output group.
2. If you have already set up this output group with video and audio, find the outputs where you want to add the captions. Or if you have not set up with video and audio, create a new output in this output group. You set up the captions now, and then set up the video and audio later.
3. Choose the output.
4. For Stream settings, choose Add captions. You now have an undefined captions encode inside this output.
5. For Captions description name, enter a name for this captions asset that is unique in the channel, for example, Embedded. Or accept the default (which is automatically generated).
6. For Captions selector name, enter the name of the captions selector that you created in step 1 (p. 172). Specify the selector that identifies the captions asset that is the source for the captions in this output.
7. For Captions settings, choose the captions format for the output captions.
8. Complete the fields that appear for the selected format. For details about a field, choose the Info link beside the field. For tips about font styles in DVB-Sub or burn-in, see Font Styles for Burn-in or DVB-Sub Output (p. 180).
9. If the output format is embedded and the output group is HLS, you can include captions language information in the manifest. You perform this setup in the output settings (separate from the captions encode). See HLS manifest (p. 181).
10. If the output format is ARIB or DVB-Sub or SCTE-27, you must perform some extra setup in the output settings (separate from the captions encode). See PIDS for ARIB output (p. 180) or PIDs for DVB-Sub output (p. 181) or PIDs for SCTE-27 (p. 181) or PIDs for Teletext output (p. 181).
11. You now have a captions encode that is fully defined.
12. Repeat these steps to create captions in more outputs and output groups, as applicable.
Sidecar Captions and SMPTE-TT in Microsoft Smooth

Follow this procedure if the format of the captions asset that you want to add is a sidecar, or if the format is SMPTE-TT for a Microsoft Smooth output group. See the section called “Step 3: Match Formats to Categories” (p. 177).

You set up each captions asset in its own output within the output group.

To set up the captions

1. In the channel that you are creating, in the navigation pane, find the output group (which you have already created). For example, find the HLS output group.
2. Create an output in the usual way: in the HLS outputs pane, choose Add output.
3. Choose the output to show the Stream settings pane. The output is set up by default with one undefined video encode and one undefined audio encode.
4. For Stream settings, remove the video and audio encodes from this output by choosing the encode and selecting Remove video or Remove audio. The output is now empty.
5. Choose Add captions. You now have an undefined captions encode inside this output.
6. For Captions description name, enter a name for this captions asset that is unique in the channel, for example, Web-VTT Czech. Or accept the default (which is automatically generated).
7. For Captions selector name, enter the name of the captions selector that you created in step 1 (p. 172). Specify the selector that identifies the captions asset that is the source for the captions in this output.
8. For Captions settings, choose the appropriate format for the output captions.
9. Complete the fields that appear for the selected format. For details about a field, choose the Info link beside the field.
10. You now have one output that contains one captions encode that is fully defined.
11. Repeat these steps to create sidecar captions in this or another output group, as applicable.

Details for Specific Output Formats

Following is information that applies only to the specified captions format.

Font Styles for Burn-in or DVB-Sub

If the output captions are Burn-in or DVB-Sub, you can specify the look of the captions.

If you are using the same captions source in several outputs and both those outputs use the same format, then you must set up the font style information identically in each output. If you don't, you get an error when you save the channel.

For example, output A might use Captions Selector 1 with the Destination Type set to Burn-in. And output B might also use Captions Selector 1 with the Destination Type set to Burn-in. You set the font information once in output 1 and again in output 2. But you must set up all the font information identically in both outputs.

PIIDs for ARIB

Complete this step if the output group is UDP/TS and the output captions format is ARIB:

- In the relevant UDP output group, choose the output that has the ARIB captions.
• For **PID settings**, complete **ARIB captions PID control** and **ARIB captions PID** as shown in the following table.

<table>
<thead>
<tr>
<th>ARIB Captions PID Control</th>
<th>ARIB Captions PID</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Auto</td>
<td>Ignore</td>
<td>A PID is automatically assigned during encoding. This value could be any number.</td>
</tr>
<tr>
<td>Use Configured</td>
<td>Enter a decimal or hexadecimal</td>
<td>This PID is used for the captions.</td>
</tr>
</tbody>
</table>

**PIDs for DVB-Sub**

Complete this step if the output group is UDP and the output captions format is DVB-Sub.

• In the relevant UDP output group, choose the output that has the DVB-Sub captions.
• For **PID settings**, in **DVB-Sub PIDs**, enter the PID for the DVB-Sub captions in this output. Or keep the default.

**PIDs for SCTE-27**

Complete this step if the output group is UDP and the output captions format is SCTE-27:

• In the relevant UDP output group, choose the output that has the SCTE-27 captions.
• For **PID settings**, in **SCTE-27 PIDs**, enter the PID for the SCTE-27 captions in this output. Or keep the default.

**PIDs for Teletext**

Complete this step if the output group is UDP and the output captions format is Teletext:

• In the relevant UDP output group, choose the output that has the Teletext captions.
• For **PID settings**, in **DVB Teletext PID**, enter the PID for the Teletext captions in this output. Or keep the default.

**HLS Manifests (Embedded Captions)**

If the captions are embedded captions and the output is HLS, you must include captions language information in the manifest. If you don't include this information, the downstream player won't have information about the embedded captions.

• In the HLS output group in Output groups, for **Captions**, in **Captions language setting**, choose **Insert**. Choosing this option inserts lines in the manifest for each embedded captions language. It inserts as many lines as the mappings that you will add in the next step.
• Still in the HLS output group, for **HLS settings**, in **Captions language mappings**, choose **Add captions language mappings**.
• Choose **Add captions language mappings** again to add more mapping groups, one for each embedded captions asset, to a maximum of four groups. For example, if the output embedded languages contain English, French, and Spanish, you need three mapping groups.
• Complete each mapping group to identify the CC (caption channel) number and its language. Specify the language as a three-letter ISO language code, as per ISO 639-2. For example, if captions channel 1 is French, then set up the three fields with "1", "fre", and "French".
The order in which you enter the languages must match the order of the captions in the source. For example, if the captions are in the order French, then English, then Spanish, then Portuguese, then set up CC1 as French, CC2 as English, and so on. If you don't order them correctly, the captions in the manifest will be tagged with the wrong languages.

Examples

The following examples describe how to implement the use cases from the section called “Typical Scenarios” (p. 170).

Topics

- Use Case A: One Input Format to One Output and Not Converted (p. 182)
- Use Case B: One Input Format Converted to One Different Output Format (p. 183)
- Use Case C: One Input Format Converted to Different Formats, One Format for Each Output (p. 184)
- Use Case D: One Captions Output Shared by Multiple Video Encodes (p. 185)

Use Case A: One Input Format to One Output and Not Converted

This example shows how to implement the first use case (p. 170) from the typical scenarios. The input is set up with one format of captions and two or more languages. Assume that you want to maintain the format in the output, and that you want to produce only one type of output and include all the languages in that output.

For example, the input has embedded captions in English and French. You want to produce an HLS output that includes embedded captions in both English and French, plus one video and one audio.

This example illustrates two important features of an embedded passthrough workflow. First, you don't create separate captions selectors; all the languages are automatically included. Second, if you are outputting to HLS, there is an opportunity to specify the languages and the order in which they appear.

To set up for this use case

1. In the channel that you are creating, in the navigation pane, for Input attachments, choose the input.
2. For General input settings, choose Add captions selector to create one captions selector. Set Selector settings to Embedded source.
3. Create an HLS output group.
4. Create one output and set up the video and audio.
5. In that same output, create one captions asset with the following:
   - **Captions selector name**: Captions selector 1.
   - **Captions settings**: One of the Embedded formats.
   - **Language code** and **Language description**: Keep the field blank. With embedded captions, all the languages are included.
6. In the HLS output group, in **Captions**, for **Captions language setting**, choose **Insert**.
7. For **HLS settings**, in **Captions language mappings**, choose **Add captions language mappings** twice (once for each language).
8. Complete the first group of mapping fields with 1, ENG, and English and the second group with 2, FRE, and French.
9. Finish setting up the channel and save it.

**Use Case B: One Input Format Converted to One Different Output Format**

This example shows how to implement the second use case (p. 170) (p. 171) from the typical scenarios. The input includes two captions languages, and the single output converts those captions. For example, the input has embedded captions in German and French. You want to produce a UDP output with both captions converted to DVB-Sub, plus one video and one audio.

To set up for this use case

1. In the channel that you are creating, in the navigation pane, for **Input attachments**, choose the input.
2. For **General input settings**, choose **Add captions selector** twice, to create Captions selector 1 (for German) and Captions selector 2 (for French). In both cases, set **Selector settings** to **Embedded source**.
3. Create a UDP output group.
4. Create one output and set up the video and audio.
5. In this output, choose **Add captions** to create a captions encode.
   - **Captions selector name**: Captions selector 1.
   - **Captions settings**: DVB-Sub.
   - **Language code** and **Language description**: German.
   - Other fields: Keep the defaults or complete as desired.
6. Choose **Add captions** again to create another captions encode. Set up this encode for the French captions. Make sure that you set up the font fields for German and French in exactly the same way.

7. Finish setting up the channel and save it.

**Use Case C: One Input Format Converted to Different Formats, One Format for Each Output**

This example shows how to implement the third use case (p. 170) from the typical scenarios. The input is set up with one format of captions and two or more languages. You want to produce several different types of output. In each output, you want to convert the captions to a different format but include all the languages.

For example, the input has Teletext captions in Czech and Polish. Assume that you want to produce a Microsoft Smooth output and an HLS output. Assume that in the Microsoft Smooth output, you want to include one video and one audio and you want to convert the captions to TTML. In the HLS output, you want to include one video and one audio and you want to convert the captions to Web-VTT.

To set up for this use case

1. In the channel that you are creating, in the navigation pane, for **Input attachments**, choose the input.
2. For **General input settings**, choose **Add captions selector** twice to create the following captions selectors:
Use Case D: One Captions Output Shared by Multiple Video Encodes

This example shows how to set up captions in an ABR workflow.

The first setup shows how to set up an ABR workflow when the captions are in the same output as the video, meaning that the captions are either embedded or captions style.

The second setup shows how to set up an ABR workflow when the captions belong to the sidecar category, in which case each captions encode is in its own output.
Setup with Embedded or Object-style Captions

This example shows how to implement the fourth use case (p. 170) (p. 172) from the typical scenarios. For example, you want to produce an HLS output with three video encodes (one for low-resolution video, one for medium, one for high) and one audio. You also want to include embedded captions (in English and Spanish) and associate them with all three video encodes.

To set up for this use case

1. In the channel that you are creating, in the navigation pane, in Input attachments, choose the input.
2. For General input settings, choose Add captions selector to create one captions selector. Set Selector settings to Embedded source.
3. Create an HLS output group.
4. Create one output and set up the video and audio for low-resolution video.
5. In that same output, create one captions asset with the following:
   - Captions selector name: Captions selector 1.
   - Captions settings: One of the Embedded formats.
   - Language code and Language description: Leave blank; with embedded passthrough captions, all the languages are included.
6. Create a second output and set up the video and audio for medium-resolution video.
7. In that same output, create one captions asset with the following:
   - Captions selector name: Captions selector 1.
   - Captions settings: One of the Embedded formats.
8. Create a third output and set up the video and audio for high-resolution video.

9. In that same output, create one captions asset with the following:

   - **Captions selector name**: Captions selector 1.
   - **Captions settings**: One of the Embedded formats.
   - **Language code** and **Language description**: Keep blank. With embedded captions, all the languages are included.

10. Finish setting up the channel and save it.

### Setup with Sidecar Captions

This example shows an ABR workflow where the captions are in sidecars. For example, you want to produce a Microsoft Smooth output with three video encodes (one for low-resolution video, one for medium, one for high) and one audio. These encodes are in a Microsoft Smooth output. You want to ingest embedded captions (in English and Spanish) and convert them to TTML captions, one for English and one for Spanish.

![Diagram of Sidecar Captions setup]

**To set up for this use case**

1. In the channel that you are creating, in the navigation pane, for **Input attachments**, choose the input.
2. For **General input settings**, choose **Add captions selector** twice to create the following captions selectors:
   - Captions selector 1: for Embedded English.
   - Captions Selector 2: for Embedded Spanish.

3. Create a Microsoft Smooth output group.

4. Create one output that contains one video encode and set it up for low-resolution video.

5. Create a second output that contains one video encode and set it up for medium-resolution video.

6. Create a third output that contains one video encode and set it up for high-resolution video.

7. Create a fourth output that contains one audio encode and no video encode.

8. Create a fifth output that contains one captions encode and no video or audio encodes, and with the following settings for the captions encode:
   - **Captions selector name**: Captions selector 1.
   - **Captions settings**: TTML.
   - **Language code** and **Language description**: English.

9. Create a sixth output that contains one captions encode and no video or audio encodes, and with the following settings for the captions encode:
   - **Captions selector name**: Captions selector 2.
   - **Captions settings**: TTML.
   - **Language code** and **Language description**: Spanish.

10. Finish setting up the channel and save it.
Input Switching in AWS Elemental MediaLive

You can set up an AWS Elemental MediaLive channel to ingest multiple sequential inputs, rather than setting it up to ingest only one input. Here are the main steps:

- You create multiple inputs.
- You create a channel and associate those inputs by setting them up as input attachments.
- You create actions in the schedule that specify when to switch from one input to another.
- You start the channel. MediaLive immediately looks at the schedule to determine the first input to ingest. It then follows the schedule throughout the running of the channel and switches inputs as required.

Note
It is not enough to just add multiple inputs to the channel. You must create actions in the schedule that specify when to switch from one input to another. Even if you want the inputs to follow each other, you must create actions in the schedule. Otherwise, the MediaLive channel ingests only the first input.

Topics
- Typical Use Cases (p. 189)
- Recommended Procedure (p. 190)
- Rules and Limits for Input Switches (p. 191)
- Types of Switches—Fixed, Follow, and Follow Chains (p. 192)
- Planning the Schedule of Input Switches (p. 193)
- Creating Inputs for Input Switching (p. 194)
- Creating a Channel with Multiple Inputs (p. 195)
- Setting Up the Schedule with Input Switches (p. 196)
- Starting a Channel That Has Multiple Inputs (p. 196)

Typical Use Cases

Scheduled input switching supports the following use cases.

Use Case 1: One Live Feed and One File Input Alternating

You have a channel to process a live (streaming) feed from a specific source, perhaps for a sports tournament. Periodically (perhaps between individual sports events), the live feed should be replaced by file content (perhaps a filler such as a video of ocean waves). After a few minutes, the same live feed should be resumed.

You set up the channel with one live input and one file input. The first input is the live input.

Before you start the channel, you create a schedule that consists of actions to switch to the live input at the top of each hour—at 10:00 AM, 11:00 AM, and so on.
You then start the channel. As soon as each sports event has finished, you modify the schedule "on the spot" to switch to the video filler. The live feed continues for a few moments (perhaps showing the sports crowd or the players leaving the stadium), and then the channel switches to the filler video. At the top of each hour, the channel switches to the live feed.

Use Case 2: One Live Feed and File Inputs, and the Channel Starts with a File Input

You have the same requirements as for use case 1, except that you want to start the channel with a file clip, perhaps from the opening of the sports event. At the top of the first hour, you want to show the video filler. But at the top of the second and succeeding hours, you want to show highlights from earlier in the day.

You set up the channel with one live event (a live input) and several file inputs: one for the opening, one for the video filler, and several for the highlights. The first input is the file input for the opening event.

Before you start the channel, you create a schedule that contains one action to switch to the live input as soon as the file input has finished.

You then start the channel. As time goes on, you modify the schedule to add more actions, as for use case 1, to switch back and forth between the live input and the file inputs.

Use Case 3: Two Live Feeds

You have a channel to process live feed from two different sources. You want to insert ad content into the channel, as required. You want to insert this ad content using MediaLive. (You do not want to insert SCTE-35 messages that a downstream system will read in order to replace the avails with ad content.)

The live feeds might be the venue feed and the in-studio feed for the same sports event. You want to switch from one live feed to the other. You want to time the switches "on the spot" instead of according to a strict clock schedule. Occasionally, you want to switch from one live feed to an ad. When the ad is finished, you might want to return to one of the live feeds.

You set up the channel with two live inputs and several file inputs (one file for each ad).

Before you start the channel, you create a schedule that contains the first action in the schedule. That action is to switch to the first input, input A, that you want to the channel to ingest. You set the start time for input A to a time that is at least one minute earlier than the time that you start the schedule. You then start the channel. MediaLive immediately reads the schedule and switches to the input that is supposed to be the current action, which is input A. When appropriate, you modify the schedule on the spot to add actions to queue up one or more switches.

Recommended Procedure

Every channel has a schedule. In a channel that has multiple inputs, the schedule controls the order and timing of switches from one input to another. There is nothing in the channel that implies an order, other than that the first input listed is ingested first if the schedule does not override it with an input switch.

Therefore, when your channel includes input switching, you must add switch actions to the schedule. It is a good idea to plan the schedule before you create the channel, and to add actions to the schedule before you start the channel.

This is the recommended order of work for setting up for input switching in a channel. This order of work assumes that either you are already familiar with working with channels and inputs, or that you will read the chapters referred to in the following list, in order to understand how they work:
1. Plan the channel in the usual way, with these additions:
   - Assessing the upstream system: Make sure that you read the information for making sure the inputs are ready to be switched to. See the section called “Setting Up the Upstream System” (p. 57).
   - Planning the input: Make sure that you read the information about assessing video and audio (p. 60) for multiple inputs.

2. Plan the schedule of input switches. See the section called “Planning the Schedule of Input Switches” (p. 193).

3. Create the inputs. See the section called “Creating Inputs for Input Switching” (p. 194).

4. Create the channel and attach the inputs. See the section called “Creating a Channel with Multiple Inputs” (p. 195).

5. Add switch actions to the schedule. See the section called “Setting Up the Schedule with Input Switches” (p. 196).

6. Start the channel. See the section called “Starting a Channel That Has Multiple Inputs” (p. 196).

Rules and Limits for Input Switches

This section describes the rules and limits that apply to input switches.

Rules for Types of Inputs

There are three restrictions on the different input types that can be in the "pool" of inputs attached to one channel:

- You can't have both MediaConnect inputs and VPC inputs attached to one channel.
- You can have multiple MediaConnect inputs attached to one channel, but all those inputs must be in the same two Availability Zones.
- You can have multiple VPC inputs attached to one channel, including both RTP VPC inputs and RTMP VPC push inputs. But all those inputs must be in the same two Availability Zones.

For example:

- You can have both HLS inputs and MediaConnect inputs attached to one channel.
- You can have both RTMP push inputs used for a source from the public internet and an RTMP VPC push input.

Limit to Inputs in a Channel

Each channel that will implement input switching can contain a specific number of inputs, as follows:

- A maximum 20 inputs.
- From 0 to 2 of those inputs can be live inputs.
- The remainder of the 20 inputs can be file inputs. File inputs are file-based video inputs.

No Limits to the Number of Input Switches

The schedule for the channel can contain any number of scheduled input switching actions.

You can switch to a specific input as many times as you want.
Reusing a File Input

If you switch away from a file input and then switch back to it, the channel ingests the file from the start of the file. This rule applies even if you had switched away from the file input before the end of the file.

Types of Switches—Fixed, Follow, and Follow Chains

There are two types of input switch start modes:

- Fixed: These input switches start at a specific UTC time.
- Follows: These input switches start when the previous input has ended (when MediaLive has reached the end of the file).

Note that switches at a fixed time use UTC time. They do not use the timecode of the input.

The combination of types of switches and types of inputs (file and live) means that there are these types of switches:

- A file input with a fixed start. The previous input can be a file or live input. At the specified start time, MediaLive stops ingesting the previous input and switches to the new input.
- A file input that follows the previous input. The previous input must be a file input. It can’t be a live input because a live input doesn’t have an end, so the switch would never occur.
- A live input with a fixed start. The previous input can be a file or live input. At the specified start time, MediaLive stops ingesting the previous input and switches to the new input.
- A live input that follows the previous input. The previous input must be a file input. It can’t be a live input because a live input doesn’t have an end, so the switch would never occur.

The following table summarizes the inputs and start types.

<table>
<thead>
<tr>
<th>Current Input</th>
<th>Next Input</th>
<th>Possible Start Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>File</td>
<td>File</td>
<td>Fixed</td>
</tr>
<tr>
<td>File</td>
<td>File</td>
<td>Follow</td>
</tr>
<tr>
<td>File</td>
<td>Live</td>
<td>Fixed</td>
</tr>
<tr>
<td>File</td>
<td>Live</td>
<td>Follow</td>
</tr>
<tr>
<td>Live</td>
<td>File</td>
<td>Fixed</td>
</tr>
<tr>
<td>Live</td>
<td>Live</td>
<td>Fixed</td>
</tr>
</tbody>
</table>

A series of follow input switches is called a *follow chain*. When each input ends, MediaLive automatically starts ingesting the next input. Here is a diagram of a follow chain:

<table>
<thead>
<tr>
<th>Input A</th>
<th>Fixed</th>
<th>File</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input B</td>
<td>Follow</td>
<td>File</td>
</tr>
<tr>
<td>Input C</td>
<td>Follow</td>
<td>File</td>
</tr>
</tbody>
</table>
The follow chain starts with the input above the first follow and ends with the last follow input. In the preceding example, the chain starts with input A and ends with input D.

The last input can be file or live. Either of these types can come before a fixed input (input E).

The other inputs (inputs A, B, C) in the follow chain must be files because they must have a defined ending so that the next input can successfully follow.

Planning the Schedule of Input Switches

This section provides some guidelines for successfully setting up input switches in the channel.

Planning the Names

When you set up MediaLive for a channel with multiple inputs, you create inputs, you associate the inputs with a channel by setting them up as input attachments, and in the schedule you create actions that reference the input attachments.

In effect, the same video asset is referenced three times:

- As an input
- As an input attachment in a channel
- As an action of type input switch in a schedule

Each of these representations has a name that you assign. The default (recommended) input attachment name on the console is equal to the input name.

Given both of these points, you might want to consider using the same name for all these representations. You might also want to establish a naming convention. For example, for the naming convention you could include the type (VOD or live) plus a description, such as `vod_ward_cars_ad` and `live_studio_feed`.

You would then use the same name for all three representations. For example, use `vod_ward_cars_ad` as the name for the input, for the input attachment, and for the schedule action.

Handling the Transition When the Next Input is Fixed

When planning the schedule, you should ensure that there is no gap when switching from a file input (input A) to an input that starts at a fixed time (input B). Input B can be a file or a live input. If the current input ends before the switch start time, there is potential for a gap.

The `Source end behavior` field in each input attachment controls the gap. (This field appears in the Input attachments page, in the General input settings section.) There are two options to ensure a smooth transition in this situation:

- If you set the `Source end behavior` field for input A to `LOOP`, then when input A finishes, MediaLive goes back and ingests it again until the start time of input B occurs.
- If you set the `Source end behavior` field for input A to `CONTINUE`, then input A is ingested only once; when the input finishes, the channel follows the behavior specified in the Input Loss Behavior set of fields (although without the "repeat frames" logic). When the start time of input B occurs, the input loss behavior ends and the channel switches to input B.
Handling the Transition When the Next Input is Follow

When planning the schedule, you should ensure that a switch from one input to a “follow input" can succeed.

A follow input (input B) won’t succeed if the current input (input A) is set up to loop. When MediaLive reaches the file end, it starts to ingest again from the beginning of the file.

The Source end behavior field in each input attachment controls looping (This field appears in the Input attachments page, in the General input settings section.)

• Always set the Source end behavior for input A to CONTINUE. When input A finishes, the channel immediately switches to input B.

When you create the channel, it is important to set the Source end behavior to CONTINUE in every input attachment where the next planned input in the schedule will be a follow input. If you don’t set up the input with CONTINUE, you won’t be able to set up the schedule with the next input as a follow input. You will have to cancel the schedule action, modify the input attachment, and try the schedule action again.

Creating Inputs for Input Switching

This section is a supplement to the information in Working with Inputs (p. 70). It provides information specific to inputs that will be used in a channel that contains multiple inputs.

When you work with the schedule to set up the switching of inputs, all the inputs must already exist in the channel. You can’t use the schedule to add inputs to a channel after the channel has started. You can modify the channel itself to add more inputs, but you must stop the channel to do so.

Therefore, when you set up your workflow you need to do the following:

• Create all the inputs that you will need through the lifetime of the channel, or at least until the next planned maintenance period.
• When you create the channel, add all these inputs to the channel as input attachments.

Rules for Setup

To create a live input, create an input of one of these types:

• MediaConnect Push
• RTMP Push or Pull
• RTP Push
• HLS Pull of a live HLS input

MediaLive considers an HLS input to be a live stream if the Buffer segments field has a value from 3 to 10, inclusive.
It considers the input to be a VOD asset if that field has a value of 11 or more, or 0, or undefined.

(To display this field, in General input settings for Network input settings, choose Network input. For HLS input settings, choose Hls input. The Buffer segments field appears.)

To create a file input, create the type of input:

- MP4 Pull

For each file input, make sure that you have correctly set up the Source end behavior field. For more information, see the section called “Handling the Transition When the Next Input is Fixed” (p. 193).

Note
MediaLive does not support the use of HLS VOD assets as inputs in a channel that has multiple input attachments.

Creating a Channel with Multiple Inputs

This section is a supplement to the information in Creating a Channel from Scratch (p. 86). It provides information that applies specifically to creating a channel that contains multiple input attachments.

Follow the steps for creating a channel, as described in Creating a Channel from Scratch (p. 86).

Note the following:

- In the Input specifications section (p. 88) for the channel, set up each option to meet or exceed the most demanding of your inputs.
- In the Input attachments section (p. 89) for the channel, to attach multiple inputs to the channel, for Input attachments, choose Add once for each input, for up to 20 inputs.
  - From 0 to 2 of those input attachments can be live inputs. One input can be a pull and one a push, or both can be the same.
  - The remainder, up to the maximum, can be file inputs. File inputs are always Pull inputs.
- Do not include HLS VOD inputs in the channel. MediaLive considers an HLS input to be a VOD input if the Buffer segments field is set to 11 or higher, or 0, or undefined. (To display this field, in General input settings for Network input settings, choose Network input. For HLS input settings, choose Hls input. The Buffer segments fields appears.)
- The order in which you create the input attachments is not relevant. The order in which the input attachments appear in the Channel page on the console is not relevant. The schedule controls the order that the channel will follow to ingest inputs.
- In the General input settings section for each Input attachment, set Source end behavior to work correctly. For information, see the section called “Handling the Transition When the Next Input is Fixed” (p. 193).
- In the General input settings section for each Input attachment, set up the Video selector, Audio selectors, and Caption selectors according to the plan that you created when you assessed the sources.

The setup for video and audio is nearly the same as when the channel has only one input; the only difference is that you set up the video and audio in every input attachment.

The setup for captions is more complicated. For more information, see the section called “Captions in Channels with Multiple Inputs” (p. 169).

- In each Output group, set up the video, audio, and captions according to the plan that you created when you designed the channel, as described in the section called “Planning the Channel” (p. 60).
Set up other features of MediaLive as needed.

Setting Up the Schedule with Input Switches

After you create the channel, you must create actions in the schedule to set up the input switches that you want. For detailed information, see the section called “Creating Actions” (p. 138).

We recommend that you add input switch actions to the schedule before you start the channel.

Starting a Channel That Has Multiple Inputs

Before you start the channel, make sure that the following are set up:

- Create actions in the schedule, if appropriate.

You can start the channel before or after you have added actions to the schedule.

We recommend that you create at least the first few input switches before you start the channel.

If the first input attachment listed in the channel is not the input that you want to be ingested first, then you must create an action in the schedule to switch inputs. Create this action to start on or before the time that you plan to start the channel.

- Make sure the inputs attached to the channel are ready:
  - Live inputs (in other words, all push inputs) must be already pushing before you start the channel. A live input must be already pushing even if it is not the first input in the channel.
  - If the first input in the channel is a file input, it must be ready to be pulled.
  - A file input that is not the first input doesn't have to be ready to be pulled until approximately 30 seconds before the switch to the input occurs.

Topics

- What Happens at Runtime (p. 196)
- Restarting a Channel (p. 196)

What Happens at Runtime

When you start the channel, MediaLive takes a short time to get the channel ready to run.

As soon as the channel is ready, MediaLive looks at the schedule to determine if there is an input switch with a start time that is now or that is overdue:

- If it finds this action, it switches to that input and starts ingesting.
- If it does not find this action, it starts ingesting the first input attachment listed in the channel.

Restarting a Channel

If you restart a channel that has multiple inputs set up for scheduled input switching, MediaLive looks at the schedule to determine which input should currently be running. MediaLive then behaves as follows:

- If that input is a live input, then MediaLive starts ingesting that input at the current frame.
• If that input is a file input set to start at a fixed time, then MediaLive starts ingesting that input at the start of the file. It does not adjust for the difference between the scheduled time and the current time. For example, assume that it is now 13:10:00 UTC. The schedule specifies to switch to input X at 13:00:00. MediaLive starts ingesting the file from the start, not from 10 minutes into the file.
• If the current input is ambiguous because there is a chain of “follow” inputs, then MediaLive ignores the “follow” inputs. It finds the most recent “fixed” input that is in the past, relative to the UTC time at which you restart the channel. It starts ingesting the input at the start of the file.

For example, assume the schedule looks like this:
• Live input X with fixed start time of 11:00
• File input A with fixed start time of 11:06
• File input B with follow start time
• File input C with follow start time
• Live input D with fixed start time of 12:15

Scenario 1: Assume the channel stopped at 11:04, when input X was active. You restart the channel at 12:09. The most recent “fixed” input switch relative to the current time is at 11:06. It is a switch to file input A. MediaLive goes to input A and starts ingesting that input from the beginning.

Scenario 2: Assume the channel stopped at 11:04, when input X was active. You restart the channel at 12:16. The most recent “fixed” input switch relative to the current time is at 12:15. It is a switch to live input D. MediaLive goes to input D and starts ingesting.

Scenario 3: Assume the channel stopped at 11:08, when input A was active. You restart the channel at 12:14. The most recent “fixed” input switch relative to the current time is at 11:06. It is a switch to file input A. MediaLive goes back to input A and starts ingesting. It will ingest files A to C until 12:15, when it will switch to the live input. It will ingest at least part of file A. It might ingest files B and C. But at 12:15 it will definitely switch to input D.
Enabling ID3 Metadata

To include ID3 metadata in an output, you must enable ID3 metadata in that output when you create or edit the channel.

Enabling in Archive Outputs

To include ID3 metadata in Archive outputs, you must enable the feature in each applicable output.

To enable ID3 metadata in Archive outputs

1. On the Create channel page, in the Output groups section, in the Archive group, choose the output where you want to enable ID3 metadata.
2. For Container Settings, for PID Settings, for Timed Metadata Behavior, choose PASSTHROUGH.
3. For Timed Metadata PIDs, enter the PID where you want to insert the ID3 metadata.
4. Repeat for each applicable output.

For information about the results of enabling, see the section called “Results of Enabling ID3 Metadata” (p. 199) later in this section.

Enabling in HLS Outputs

To include ID3 metadata in HLS outputs, you must enable the feature in each applicable output.
To enable ID3 metadata in HLS outputs

1. On the Create channel page, in the Output groups section, in the HLS group, choose the output where you want to enable ID3 metadata.
2. Make sure that HLS Settings is set to Standard hls. Only standard outputs can contain ID3 metadata. The Audio-only outputs option (which is the other option in this field), is used to set up audio rendition groups and can't contain this metadata.
3. For PID Settings, Timed Metadata Behavior, choose PASSTHROUGH.
4. For Timed Metadata PIDs, enter the PID where you want to insert the ID3 metadata.
5. Repeat for each applicable output.

For information about the results of enabling, see the section called “Results of Enabling ID3 Metadata” (p. 199) later in this section.

Enabling in MediaPackage Outputs

To include ID3 metadata in MediaPackage outputs, you don't have to set up the output. MediaPackage outputs are automatically set up with this feature enabled.

For information about handling of ID3 metadata in MediaPackage outputs, see the section called “Results of Enabling ID3 Metadata” (p. 199) later in this chapter.

Enabling in UDP Outputs

To include ID3 metadata in UDP outputs, you must enable the feature in each applicable output.

To enable ID3 metadata in UDP outputs

1. On the Create channel page, in the Output groups section, in the UDP group, choose the output where you want to enable ID3 metadata.
2. For Network Settings, PID Settings, Timed Metadata Behavior, choose PASSTHROUGH.
3. For Timed Metadata PIDs, enter the PID where you want to insert the metadata.
4. Repeat for each applicable output.

For information about the results of enabling, see the section called “Results of Enabling ID3 Metadata” (p. 199) later in this section.

Results of Enabling ID3 Metadata

Here are the results of enabling ID3 metadata in the channel:

- ID3 metadata other than type TDRL or PRIV that is present in the input is automatically included in the eligible outputs.
- ID3 metadata of type TDRL or PRIV that is present in the input is passed through to eligible outputs as follows:
  - If the frame doesn't have “Elemental Technologies” included in the wording, the metadata is passed through.
  - If the frame has “Elemental Technologies” included in the wording, the metadata is not passed through. The metadata isn’t passed through because MediaLive assumes that the timestamp for this metadata has passed.
- ID3 metadata that you set up in the output group is inserted in those outputs where you enabled ID3 metadata, when you created the channel. For information about setting up ID3 metadata in the output group, see the section called “Inserting ID3 Metadata When Creating the Channel” (p. 200).
ID3 metadata that you set up by creating an action in the MediaLive schedule is included in the eligible outputs. For information about setting up ID3 metadata in the schedule, see the section called “Inserting ID3 Metadata Using the Schedule” (p. 201).

The eligibility of an output depends on the output group type, as shown in the following table.

<table>
<thead>
<tr>
<th>Type of Output Group</th>
<th>ID3 Metadata That Is Present in Input</th>
<th>ID3 Metadata That You Specify When Setting Up the Channel</th>
<th>ID3 Metadata That You Insert Using the Schedule</th>
</tr>
</thead>
<tbody>
<tr>
<td>Archive</td>
<td>Passed through</td>
<td>Not included in output</td>
<td>Included in output</td>
</tr>
<tr>
<td>HLS</td>
<td>Passed through</td>
<td>Included in output</td>
<td>Included in output</td>
</tr>
<tr>
<td>MediaPackage</td>
<td>Passed through</td>
<td>Not included in output</td>
<td>Included in output</td>
</tr>
<tr>
<td>UDP</td>
<td>Passed through</td>
<td>Included in output</td>
<td>Not included in output</td>
</tr>
</tbody>
</table>

### Passing Through ID3 Metadata

You can set up outputs so that ID3 metadata that is in the channel input is automatically passed through to the output. To pass through ID3 metadata, enable ID3 in the outputs. For information, see the section called “Enabling ID3 Metadata” (p. 198).

### Inserting ID3 Metadata When Creating the Channel

You can set up to insert ID3 metadata at a regular cadence (for example, every 10 seconds) into HLS or UDP outputs where you enabled ID3 metadata. You can’t insert ID3 metadata into Archive or MediaPackage outputs.

#### To insert ID3 metadata when creating the channel

1. Make sure that you enabled ID3 metadata. For detailed information, see the section called “Enabling ID3 Metadata” (p. 198).
2. On the Create channel page, in the Output groups section, choose the HLS group or the UDP group. (You can’t insert ID3 metadata in an Archive group or MediaPackage group.)
3. Choose ID3.
4. For Timed Metadata ID3 Frame, choose the ID3 frame type that you want to apply to the metadata.
   - Try to avoid using PRIV for metadata that you insert when creating the channel and for metadata from one of the other sources.
5. For Timed Metadata ID3 Period, enter the repeat interval for the ID3 metadata, in seconds.
   - For a UDP output group, set any length. For an HLS output group, we recommend that you set the period (interval) to half the segment length. To verify the segment length, in the HLS output group, choose Manifests and Segments, and look at Segment Length.

When you start the channel, the first ID3 metadata is inserted shortly after the output starts and then at the specified interval for the lifetime of the channel.
The timestamp in the ID3 metadata is derived from the output timecode. It indicates the time at which the ID3 frame is inserted into the output, when the channel is running. The timestamp is in the format that you specified for the Source field in the Timecode Configuration section of the General Settings page for the channel.

Inserting ID3 Metadata Using the Schedule

You can insert ID3 metadata at a specific time by creating an action in the MediaLive schedule. The metadata is inserted in each HLS output or MediaPackage output where you have enabled ID3 metadata. It is not inserted in UDP outputs.

Typically, you include ID3 metadata in accordance with instructions of the downstream system.

To insert ID3 metadata

1. Make sure that you enabled ID3 metadata. For detailed information, see the section called “Enabling ID3 Metadata” (p. 198).
2. Create actions in the schedule. For detailed information, see Working with the Schedule (p. 136).
### SCTE-35 Message Processing

SCTE-35 messages are messages that accompany the video input in an MPEG-2 transport stream (TS). These messages provide information about advertisement availability, also known as ad avail events, and other non-ad avail events.

On the input side, SCTE-35 messages can appear only in MPEG-2 transport stream (TS) inputs, which means that in MediaLive they can appear only in RTP or HLS inputs.

You can set up a channel so that if an input includes these messages, the messages are either processed during ingest (passed through) or ignored.

MediaLive doesn't support processing of ad avail decorations in input manifests. The ad avail decorations in input manifests are always ignored.

On the output side, if you set up to pass through the input (rather than ignore it), then you can set up each output so that the SCTE-35 messages from the input are turned into cueing information that is appropriate for that output type. This cueing information can be in the form of one or both of the following:

- SCTE-35 messages in a TS output
- Manifest (or sparse track) decoration

You set up each output separately, so that you can set up some outputs to include cueing information and some to exclude it.

As an adjunct to the ad avail information, you can also set up the outputs to blank out the video, audio, and captions within the cueing information.

**Note**

To use the ad avail features of MediaLive, you should be familiar with the SCTE-35 standard and optionally with the SCTE-67 standard. You should also be familiar with how the input that you are encoding implements those standards.

The information in this SCTE-35 section assumes that you are familiar with the general steps for creating a channel, as described in *Creating a Channel from Scratch* (p. 86). It also assumes that you have started creating a channel, including associating an input with the channel.

**Topics**

- About Message Processing (p. 202)
- Getting Ready: Set the Ad Avail Mode (p. 208)
- Enabling Manifest Decoration in the Output (p. 209)
- Enabling Ad Avail Blanking in the Output (p. 212)
- Enabling Blackout in the Output (p. 216)
- Enabling SCTE-35 Passthrough or Removal in the Output (p. 220)
- Sample Manifests - HLS (p. 222)

### About Message Processing

MediaLive works with the SCTE-35 messages in MPEG-2 transport stream (TS) inputs. These messages might or might not include segmentation descriptors.

**Topics**

- Supported Features by Input Type (p. 203)
Supported Features by Input Type

The following table shows which inputs might include ad avail information and how MediaLive handles that information.

<table>
<thead>
<tr>
<th>Input</th>
<th>Interpret SCTE-35 Messages in the Transport Stream</th>
<th>Interpret Ad Avail Information in the Input Manifest</th>
</tr>
</thead>
<tbody>
<tr>
<td>HLS</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>RTMP</td>
<td>No</td>
<td>Not applicable</td>
</tr>
<tr>
<td>RTP</td>
<td>Yes</td>
<td>Not applicable</td>
</tr>
</tbody>
</table>

Supported Output Features

Blanking and Blackout

The "cue out" and "cue in" instructions in SCTE-35 messages in TS inputs line up with specific content in the video, audio, and captions streams. You can set up so that this content is blanked out in the output:

- To blank out content for ad avails, use the ad avail blanking feature.
- To blank out content for other messages, use the blackout feature.

The behavior that you want must be set up in the channel.

For more information, see the section called “Enabling Ad Avail Blanking” (p. 212) and the section called “Enabling Blackout” (p. 216).

Manifest Decoration

You can set up an output so that its manifest is decorated with ad avail information. Manifest decoration works on two sources of ad avail information:

- Ad avail information found in the channel input, if the input is a transport stream (TS)
- Ad avail information from SCTE-35 messages added to the output using the MediaLive schedule

Manifest decoration applies only to HLS outputs, MediaPackage outputs, and Microsoft Smooth outputs:

- You can set up HLS outputs so that their manifests are decorated according to one of the following styles:
  - Adobe
  - Elemental
  - SCTE-35 enhanced
• MediaPackage outputs are always set up so that their manifests are decorated. The marker style is always SCTE-35 enhanced style. Keep in mind that if you don't actually want SCTE-35 messages in the output that you deliver from AWS Elemental MediaPackage, then on the AWS Elemental MediaPackage side you can set up the channel to strip out the markers.

• You can set up Microsoft Smooth outputs so that the sparse track includes instructions that correspond to the original SCTE-35 message content. The behavior that you want must be set up in the channel. For more information, see the section called “Enabling Manifest Decoration” (p. 209).

**SCTE-35 Passthrough**

You can set up TS outputs so that all the SCTE-35 messages from the input are passed through to the output. Or you can set up to remove these messages from the output.

The behavior that you want must be set up in the channel. For more information, see the section called “Enabling SCTE-35 Passthrough or Removal” (p. 220).

**Processing Features – Default Behavior**

The default handling of SCTE-35 by MediaLive is the following:

- No passthrough – Remove SCTE-35 messages in any data stream outputs. There is one exception: for MediaPackage outputs, passthrough is always enabled.
- No blanking or blackout – Do not blank out video content for any events. Leave the content as is.
- No manifest decoration – Do not convert any SCTE-35 messages to event information in any output manifests or data streams. There is one exception: for MediaPackage outputs, manifest decoration is always enabled and can't be disabled.

If this is the behavior that you want, you don't need to read any further in this SCTE-35 section.

**Scope of Processing by Feature**

The SCTE-35 features have different scopes in terms of the output groups and outputs that they affect:

**Blackout or ad avail blanking**

Blackout applies at the global output level. All the relevant content in every output in every output group is blanked.

Ad avail blanking also applies at the global output level. All the ad avails in every output in every output group are blanked.
Decoration

Manifest decoration applies at the *output group* level. All the outputs in that output group have their manifests decorated.

**SCTE-35 passthrough or removal**

SCTE-35 passthrough or removal applies at the *output* level. The messages are passed through or removed only in a specific output.

---

**Supported Features by Output Type**

This section describes which SCTE-35 features apply to the various types of output.

**Topics**

- Archive Output with MPEG-2 Container (p. 205)
- Frame Capture Output (p. 206)
- HLS Output (p. 206)
- MediaPackage Output (p. 207)
- Microsoft Smooth Output (p. 207)
- UDP Output (p. 208)

**Archive Output with MPEG-2 Container**

A transport stream in an MPEG-2 container supports passthrough of the SCTE-35 messages, but it doesn't support creation of a manifest. The following table shows the valid processing options.
AWS Elemental MediaLive User Guide
Supported Features by Output Type

<table>
<thead>
<tr>
<th>SCTE-35 Passthrough</th>
<th>Manifest Decoration</th>
<th>Blanking and Blackout</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enabled</td>
<td>Not applicable</td>
<td>Yes or No</td>
<td>Turns on passthrough of SCTE-35 messages. You could also implement blanking and blackout.</td>
</tr>
<tr>
<td>Disabled</td>
<td>Not applicable</td>
<td>No</td>
<td>Turns off passthrough, to remove SCTE-35 messages from the video stream. Do not implement blanking or blackout. Choose this option only if, in a downstream system, you don't want to replace video that was originally marked by cues.</td>
</tr>
</tbody>
</table>

**Frame Capture Output**

A frame capture output doesn't support passthrough of the SCTE-35 messages. However, if blanking or blackout has been enabled (at the channel level), then content that falls between the start and stop of the blackout will be blanked or blacked out, even though no SCTE-35 messages are present. The following table shows the valid processing options.

<table>
<thead>
<tr>
<th>SCTE-35 Passthrough</th>
<th>Manifest Decoration</th>
<th>Blanking and Blackout</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not applicable (SCTE-35 messages are never included in output)</td>
<td>Not applicable</td>
<td>Yes or No</td>
<td>If you implement blanking or blackout (in order to enable these features for other outputs in the channel), then the affected content will be blanked or blacked out in the frame capture.</td>
</tr>
</tbody>
</table>

**HLS Output**

HLS output supports both passthrough of the SCTE-35 messages and manifest decoration. With HLS outputs, passthrough and manifest decoration are either both enabled or both disabled.

The following table shows the valid processing options.

<table>
<thead>
<tr>
<th>SCTE-35 Passthrough</th>
<th>Manifest Decoration</th>
<th>Blanking and Blackout</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enabled</td>
<td>Enabled</td>
<td>Yes or No</td>
<td>Turns on passthrough of SCTE-35 messages and manifest decoration. You</td>
</tr>
</tbody>
</table>
AWS Elemental MediaLive User Guide  
Supported Features by Output Type

<table>
<thead>
<tr>
<th>SCTE-35 Passthrough</th>
<th>Manifest Decoration</th>
<th>Blanking and Blackout</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disabled</td>
<td>Disabled</td>
<td>No</td>
<td>Turns off passthrough, to remove SCTE-35 messages from the video stream. Turns off manifest decoration. Do not implement blanking or blackout. Choose this option only if, in a downstream system, you don't want to replace video that was originally marked by cues.</td>
</tr>
</tbody>
</table>

Note that with HLS you must either enable passthrough and decoration in all outputs in the same output group, or disable them in all outputs in the same output group.

**MediaPackage Output**

MediaPackage output, which is a type of HLS output, supports both passthrough of the SCTE-35 messages and manifest decoration. With MediaPackage outputs, passthrough and manifest decoration are always enabled and can't be disabled.

The following table shows the valid processing option.

<table>
<thead>
<tr>
<th>SCTE-35 Passthrough</th>
<th>Manifest Decoration</th>
<th>Blanking and Blackout</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enabled</td>
<td>Enabled</td>
<td>Yes or No</td>
<td>Passthrough of SCTE-35 messages and manifest decoration are always enabled and can't be disabled. You could also implement blanking and blackout.</td>
</tr>
</tbody>
</table>

**Microsoft Smooth Output**

Microsoft Smooth output does not support passthrough of the SCTE-35 messages, but does support instructions in the sparse track. The following table shows the valid processing options.

<table>
<thead>
<tr>
<th>SCTE-35 Passthrough</th>
<th>Manifest Decoration</th>
<th>Blanking and Blackout</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not applicable (SCTE-35 messages are never included in output)</td>
<td>Enabled</td>
<td>Yes or No</td>
<td>SCTE-35 messages are removed from the video stream. But instructions are included in the sparse track. You could also implement blanking and blackout.</td>
</tr>
</tbody>
</table>
**Getting Ready: Set the Ad Avail Mode**

You must set the Ad Avail mode to notify MediaLive of the ID type of SCTE-35 messages that the input is using to indicate ad avail events.

Follow this procedure if you want to support one or more of the following features:

- Manifest decoration

---

**UDP Output**

UDP output supports passthrough of the SCTE-35 messages, but it doesn't support creation of a manifest. The following table shows the valid processing options.

<table>
<thead>
<tr>
<th>SCTE-35 Passthrough</th>
<th>Manifest Decoration</th>
<th>Blanking and Blackout</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enabled</td>
<td>Not applicable</td>
<td>Yes or No</td>
<td>Turns on passthrough of SCTE-35 messages. You could also implement blanking and blackout.</td>
</tr>
<tr>
<td>Disabled</td>
<td>Not applicable</td>
<td>No</td>
<td>Turns off passthrough, to remove SCTE-35 messages from the video stream. Do not implement blanking or blackout. Choose this option only if, in a downstream system, you don't want to replace video that was originally marked by cues.</td>
</tr>
</tbody>
</table>

---

**SCTE-35 Passthrough**

- Not applicable: SCTE-35 messages are removed from the output. The sparse track doesn't include instructions. Don't implement blanking or blackout because without SCTE-35 messages in the video stream and without data in the sparse track, it will be impossible to find these blanks and blackouts programmatically in the output.

**Manifest Decoration**

- Disabled: Not applicable
- Yes or No

**Blanking and Blackout**

- No

---

**UDP Output**

UDP output supports passthrough of the SCTE-35 messages, but it doesn't support creation of a manifest. The following table shows the valid processing options.
• Ad avail blanking

If your processing does not involve at least one of these features, the ad avail mode is ignored.

**To set the ad avail mode**

1. In the channel that you are creating, in the navigation pane, choose **General settings**.
2. Choose **Avail configuration**.
3. Set the **Avail settings**:
   - SCTE-35 splice insert (default): Select this mode if the input uses splice inserts to indicate ad avails. The input might also contain messages for others events such as chapters or programs.
   - SCTE-35 time signal apos: Select this mode if the input contains time signals of segmentation type **Placement opportunity**. The input might also contain messages for other events such as chapters or programs.

The mode identifies which of all possible events are treated as triggers for ad avails and as triggers for blackouts. In turn, these triggers affect how manifests are decorated (p. 210), when video is blanked (p. 213), and when video is blacked out (p. 218).

4. In **Ad avail offset**, set a value, if desired. See the help for this field.
5. Leave **web_delivery_allowed_flag** and **no_regional_blackout_flag** as Follow for now. For information about these fields, see the section called “Ad Avail Blanking Restriction Flags” (p. 215).

### Enabling Manifest Decoration in the Output

You can choose to interpret SCTE-35 messages from the original input and insert corresponding instructions into the output manifest for the following outputs:

- HLS
- Microsoft Smooth (the instructions are inserted in the sparse track).

MediaPackage outputs, which are a type of HLS output, are set up with manifest decoration enabled. You can't disable decoration in these outputs.

Manifest decoration is enabled at the output group level. If you enable the feature in a specific output group, all the outputs in that group have their manifests decorated.

To include manifest decoration in some outputs and not others, you must create two output groups of the specified type, for example, two HLS output groups.

**Topics**

- Enabling Decoration – HLS (p. 209)
- Enabling Decoration – Microsoft Smooth (p. 210)
- How SCTE-35 Events Are Handled in Manifests and Sparse Tracks (p. 210)

### Enabling Decoration – HLS

Manifest decoration is enabled at the output group level, which means that the manifests for all outputs in that group include instructions based on the SCTE-35 content.
To enable decoration

1. In the channel that you are creating, make sure that you have set the ad avail mode. See the section called "Getting Ready: Set the Ad Avail Mode" (p. 208).
2. In the navigation pane, find the desired HLS output group.
3. In Ad Marker, choose Add ad markers.
4. For HLS ad markers, select the type of ad marker. For information about the different types of markers, see Sample Manifests - HLS (p. 222).
5. Repeat to add more types of markers, as desired.

The manifest for each output will include a separate set of tags for each type that you select.

Enabling Decoration – Microsoft Smooth

With Microsoft Smooth, if you enable manifest decoration, instructions are inserted in the sparse track. Manifest decoration is enabled at the output group level, which means that the sparse tracks for all outputs in that group will include instructions based on the SCTE-35 content.

To enable decoration

1. In the channel that you are creating, make sure that you have set the ad avail mode. See the section called "Getting Ready: Set the Ad Avail Mode" (p. 208).
2. In the navigation pane, find the desired Microsoft Smooth output group.
3. For Sparse track, for Sparse track type, choose SCTE_35.
4. Complete Acquisition point ID, only if encryption is enabled on the output. Enter the address of the certificate.

How SCTE-35 Events Are Handled in Manifests and Sparse Tracks

When manifest decoration or sparse track is enabled, MediaLive inserts up to three types of information. The triggers for inserting this information depend on the mode.

Types of Information

<table>
<thead>
<tr>
<th>Type of Instruction</th>
<th>When Inserted</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base64</td>
<td>Information about all SCTE-35 messages in the output is incorporated into the manifest; the entire SCTE-35 message is added in base64 format.</td>
</tr>
<tr>
<td>Cue-out, Cue-in</td>
<td>SCTE-35 messages that are ad avails result in the insertion of cue-out, cue-in instructions.</td>
</tr>
<tr>
<td>Blackout</td>
<td>Only applies to the SCTE-35 Enhanced ad marker style (for HLS output; see the section called &quot;Enabling Decoration – HLS&quot; (p. 209)).</td>
</tr>
<tr>
<td></td>
<td>SCTE-35 messages that are not ad avails result in the insertion of blackout start/end instructions,</td>
</tr>
</tbody>
</table>
How SCTE-35 Events Are Handled in Manifests and Sparse Tracks

<table>
<thead>
<tr>
<th>Type of Instruction</th>
<th>When Inserted</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>assuming that blackout is enabled. If blackout is not enabled, these instructions are not inserted.</td>
</tr>
</tbody>
</table>

## Splice Insert Mode

### Message Type ID: Splice Insert

<table>
<thead>
<tr>
<th>Segmentation Type ID</th>
<th>Base64</th>
<th>Cue-out, Cue-in</th>
<th>Blackout</th>
</tr>
</thead>
<tbody>
<tr>
<td>No segmentation descriptor present</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Provider advertisement</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Distributor advertisement</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Placement opportunity</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Other: Programs, Chapters, Network, Unscheduled</td>
<td>Yes</td>
<td></td>
<td>Yes</td>
</tr>
</tbody>
</table>

## Message Type ID: Time Signal

<table>
<thead>
<tr>
<th>Segmentation Type ID</th>
<th>Base64</th>
<th>Cue-out, Cue-in</th>
<th>Blackout</th>
</tr>
</thead>
<tbody>
<tr>
<td>Provider advertisement</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Distributor advertisement</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Placement opportunity</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Other: Programs, Chapters, Network, Unscheduled</td>
<td>Yes</td>
<td></td>
<td>Yes</td>
</tr>
</tbody>
</table>

For example, read the first line in the first table as follows: When a splice insert (with no segmentation descriptor) is encountered, the base64 and cue-out, cue-in information will be inserted in the manifest; blackout information will not be inserted.

## Timesignal APOS Mode

### Message Type ID: Splice Insert

<table>
<thead>
<tr>
<th>Segmentation Type ID</th>
<th>Base64</th>
<th>Cue-out, Cue-in</th>
<th>Blackout</th>
</tr>
</thead>
<tbody>
<tr>
<td>No segmentation descriptor present</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Provider advertisement</td>
<td>Yes</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Enabling Ad Avail Blanking in the Output

You can enable ad avail blanking to blank out the content for an SCTE-35 message that is considered an ad avail (as defined by the ad avail mode in Getting Ready: Set the Ad Avail Mode (p. 208)).

A similar feature is blackout (p. 216).

Blanking involves the following processing:

- Replace the video content associated with this event with an image that you specify or is with a black image.
- Remove the audio that is associated with this event.
- Remove the captions that are associated with this event.

### Comparison to Manifest Decoration and Passthrough

Ad avail blanking applies to all outputs. You cannot choose to blank out for some outputs (for example, the HLS output) and not blank out for others (for example, the Microsoft Smooth output). It is an all-or-nothing decision.

Manifest decoration and passthrough have a smaller scope: they apply only to outputs that support these features.
Important
Take note of this fact, because if you do not do passthrough and do not do manifest decoration in a specific output (because they are not supported or because you choose not to) but you do implement blanking, there will be no markers for where the blanked content occurs. The only way to identify where this blanking is occurring will be to look for the IDR i-frames that identify where the SCTE-35 message used to be.

Enabling Blanking

Follow this procedure if you want to enable the ad avail blanking feature.

To enable blanking

1. In the channel that you are creating, in the navigation pane, choose General settings.
2. In Avail configuration, set Avail settings, if you have not already done so:
   - SCTE-35 splice insert (default): Select this mode if the input uses splice inserts to indicate ad avails. The input might also contain messages for others events such as chapters or programs.
   - SCTE-35 time signal APOS: Select this mode if the input contains time signals of segmentation type Placement opportunity. The input might also contain messages for other events such as chapters or programs.

   The mode identifies which of all possible events are treated as triggers for ad avails and as triggers for blackouts. In turn, these triggers affect how manifests are decorated (p. 210), when video is blanked (p. 213), and when video is blacked out (p. 218).
3. For Ad avail offset, set a value, if desired. See the help for this field.
4. In web_delivery_allowed_flag and no_regional_blackout_flag, choose appropriate values. For information about these fields, see the section called “Triggers for Ad Avail Blanking” (p. 213).
   - Follow (default): Observe the restriction and blank the content for the ad avail event.
   - Ignore: Ignore the restriction and do not blank the content for the ad avail event.

   Warning
   Never set both fields to Ignore.

   In Avail blanking, in State, choose Enabled.
5. In Avail blanking image, choose the appropriate value:
   - Disable: To use a plain black image for blanking.
   - Avail blanking image: To use a special image for blanking. In the URL field, type the path to a file in an S3 bucket. The file must be of type .bmp or .png. Also enter the user name and Systems Manager password parameter for accessing the S3 bucket. See the section called “About the Feature for Creating Password Parameters” (p. 29).

Triggers for Ad Avail Blanking

For ad avail blanking, the ad avail mode that you set controls which SCTE-35 events result in the blanking of the content.

Triggers in Splice Insert Mode

This section describes which message type and segmentation type combination is blanked by ad avail blanking when the Ad Avail mode is Splice Insert mode.
Message Type ID: Splice Insert

<table>
<thead>
<tr>
<th>Segmentation Type ID</th>
<th>Blanked</th>
</tr>
</thead>
<tbody>
<tr>
<td>No segmentation descriptor present</td>
<td>Yes</td>
</tr>
<tr>
<td>Provider advertisement</td>
<td>Yes</td>
</tr>
<tr>
<td>Distributor advertisement</td>
<td>Yes</td>
</tr>
<tr>
<td>Placement opportunity</td>
<td>Yes</td>
</tr>
<tr>
<td>Other: Programs, Chapters, Network, Unscheduled</td>
<td>No</td>
</tr>
</tbody>
</table>

Message Type ID: Time Signal

<table>
<thead>
<tr>
<th>Segmentation Type ID</th>
<th>Blanked</th>
</tr>
</thead>
<tbody>
<tr>
<td>Provider advertisement</td>
<td>Yes</td>
</tr>
<tr>
<td>Distributor advertisement</td>
<td>Yes</td>
</tr>
<tr>
<td>Placement opportunity</td>
<td>Yes</td>
</tr>
<tr>
<td>Other: Programs, Chapters, Network, Unscheduled</td>
<td>No</td>
</tr>
</tbody>
</table>

Triggers in Timesignal APOS Mode

This section describes which message type/segmentation type combination is blanked by ad avail blanking when the Ad Avail mode is Timesignal with APOS mode.

Message Type ID: Splice Insert

<table>
<thead>
<tr>
<th>Segmentation Type ID</th>
<th>Blanked</th>
</tr>
</thead>
<tbody>
<tr>
<td>No segmentation descriptor present</td>
<td>No</td>
</tr>
<tr>
<td>Provider advertisement</td>
<td>No</td>
</tr>
<tr>
<td>Distributor advertisement</td>
<td>No</td>
</tr>
<tr>
<td>Placement opportunity</td>
<td>No</td>
</tr>
<tr>
<td>Other: Programs, Chapters, Network, Unscheduled</td>
<td>No</td>
</tr>
</tbody>
</table>
Message Type ID: Time Signal

<table>
<thead>
<tr>
<th>Segmentation Type ID</th>
<th>Blanked</th>
</tr>
</thead>
<tbody>
<tr>
<td>Provider advertisement</td>
<td>No</td>
</tr>
<tr>
<td>Distributor advertisement</td>
<td>No</td>
</tr>
<tr>
<td>Placement opportunity</td>
<td>Yes</td>
</tr>
<tr>
<td>Other: Programs, Chapters, Network, Unscheduled</td>
<td>No</td>
</tr>
</tbody>
</table>

Ad Avail Blanking Restriction Flags

Restrictions in the Input

SCTE-35 messages of type time_signal always contain segmentation descriptors.

SCTE-35 messages of type splice_insert might or might not include segmentation descriptors.

If the input has SCTE-35 messages that do include segmentation descriptors, these segmentation descriptors always include two types of flags. Each flag has a value of "true" or "false" and provides additional information as guidance for blanking in specific situations:

- **web_delivery_allowed_flag**
  - True means that there is no restriction on including the ad avail event's content in a stream that is intended for web delivery: there is no need to blank out content in streams intended for web delivery.
  - False means there is a restriction: the content should be blanked out.

- **no_regional_blackout_flag**
  (The wording of this flag is confusing. Think of it as the "regional_delivery_allowed_flag".)
  - True means that there is no restriction on including the ad avail event's video in a stream that is intended for regional markets: there is no need to blank out content in streams intended for regional markets.
  - False means there is a restriction: the content should be blanked out.

If neither flag is present (usually the case with splice_inserts), then both are considered to be false. Blanking should occur.

If both flags are present (which is usually the case; it is unusual to have only one flag present), then a "false" for one flag takes precedence over a "true" for the other flag. Blanking should occur.

Typically, in any message in the input only one of these flags is ever set to false, so only one restriction is ever in place. There would typically never be *both* a regional delivery restriction and a web delivery restriction. This is because if content is considered restricted for regional delivery, then it would not also be considered restricted for web delivery (where the concept of a region makes no sense).

To summarize, this is the blanking logic that applies to each ad avail event that is encountered.
Blanking Logic for Ad Avail Events

<table>
<thead>
<tr>
<th>Content of corresponding SCTE-35 message: Web delivery allowed?</th>
<th>Content of corresponding SCTE-35 message: Regional delivery allowed?</th>
<th>Result</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1 Flag is not present</td>
<td>Flag is not present</td>
<td>Blanking occurs</td>
<td>This combination can only occur in a message type splice_insert (where the segmentation descriptor is optional).</td>
</tr>
<tr>
<td>S2 Flag is set to &quot;true&quot;</td>
<td>Flag is set to &quot;true&quot;</td>
<td>Blanking doesn't occur</td>
<td></td>
</tr>
<tr>
<td>S3 Flag is set to &quot;true&quot;</td>
<td>Flag is set to &quot;false&quot;</td>
<td>Blanking occurs</td>
<td></td>
</tr>
<tr>
<td>S4 Flag is set to &quot;false&quot;</td>
<td>Flag is set to &quot;true&quot;</td>
<td>Blanking occurs</td>
<td></td>
</tr>
</tbody>
</table>

MediaLive Handling of Restrictions

You can modify this default blanking behavior by instructing MediaLive to ignore a restriction flag that is set to false, so that blanking does not occur for this ad avail event. In other words, use this logic: "Even if the message indicates to blank content because a regional blackout is in place, do not follow this instruction. Ignore the fact that a regional blackout is in place and do not blank content".

You modify the behavior by setting fields in the channel. See the section called “Enabling Blanking” (p. 213).

Restriction Flags with "Splice Insert"

If you select Splice Insert as the Ad Avail mode, then there is an assumption that the SCTE-35 ad avail message does not include the two restriction flags that are described earlier in this section. There is an assumption that every SCTE-35 ad avail message should result in an ad avail.

Therefore, if you know that the input contains splice inserts (not time signals), you should leave both restriction fields unchecked.

Enabling Blackout in the Output

You can enable blackout to blank out the content for an SCTE-35 message that is of type “other event” (as defined by the mode in Getting Ready: Set the Ad Avail Mode (p. 208)). For example, chapters and programs.

(A similar feature is described in the section called “Enabling Ad Avail Blanking” (p. 212).)

Blackout involves the following processing:

- Replace the video content associated with the event with an image that you specify or is with a black image.
• Remove the audio that is associated with the event.
• Remove the captions that are associated with the event.

Comparison to Manifest Decoration and Passthrough

Blackout applies to all outputs. You cannot choose to black out for some outputs (for example, the HLS output) and not black out for others (for example, the Microsoft Smooth output). It is an all-or-nothing decision.

Manifest decoration and passthrough have a smaller scope: they apply only to outputs that support these features.

Take important note of this fact, because if you do not do passthrough and do not do manifest decoration in a specific output (because they are not supported or because you choose not to) but you do implement blanking, there will be no “markers” for where the blanked content occurs. The only way of identifying where this blanking is occurring will be to look for the IDR i-frames that identify where the SCTE-35 message used to be.

Topics
• Enabling Blackout (p. 217)
• Triggers for Blackout (p. 218)
• Blackout Restriction Flags (p. 219)

Enabling Blackout

Follow this procedure if you want to enable the blackout feature.

To enable blackout

1. In the channel that you are creating, in the navigation pane, choose General settings.
2. For Avail configuration, set Avail settings, if you have not already done so:
   • SCTE-35 splice insert (default): Select this mode if the input uses splice inserts to indicate ad avails. The input might also contain messages for others events such as chapters or programs.
   • SCTE-35 time signal APOS: Select this mode if the input contains time signals of segmentation type Placement opportunity. The input might also contain messages for other events such as chapters or programs.

   The mode identifies which of all possible events are treated as triggers for “ad avails” and as triggers for “blackouts.” In turn, these triggers affect how manifests are decorated (p. 210), when video is blanked (p. 213), and when video is blacked out (p. 218).
3. For Ad avail offset, set a value, if desired. See the help for this field.
4. For web_delivery_allowed_flag and no_regional_blackout_flag, choose appropriate values. For information about these fields, see the section called “Triggers for Blackout” (p. 218).
   • Follow (default): Observe the restriction and blank the content for the ad avail event.
   • Ignore: Ignore the restriction and do not blank the content for the ad avail event.

   Warning
   Never set both fields to Ignore.
5. In Blackout slate, in State, choose Enabled.
6. For Blackout slate image, choose the appropriate value:
   • Disable: To use a plain black image for blackout.
• **Avail blanking image:** To use a special image for blackout. In the **URL** field, enter the path to a file in an Amazon S3 bucket. The file must be of type .bmp or .png. Also enter the user name and Systems Manager password parameter for accessing the S3 bucket. For information about this key, see the section called “About the Feature for Creating Password Parameters” (p. 29).

7. If you want to enable network end blackout (in other words, black out content when network transmission has ended and remove blackout only when network transmission resumes), continue reading. If you don’t want to enable it, you have now finished setting up.

8. For **Network end blackout**, choose **Enabled**.

9. For **Network end blackout image**, choose the appropriate value:
   - **Disable:** To use a plain black image for blackout.
   - **Network end blackout image:** To use a special image for network end blackout. In the **URL** field, enter the path to a file in an Amazon S3 bucket. The file must be of type .bmp or .png. Also enter the user name and Systems Manager password for accessing the S3 bucket. See the section called “About the Feature for Creating Password Parameters” (p. 29).

10. For **Additional settings**, in **Network ID**, type the EIDR ID of the network in the format 10.nnnn/xxxx-xxxx-xxxx-xxxx-xxxx-c (case insensitive). Only network end events with this ID will trigger blackout.

## Triggers for Blackout

The blackout feature is triggered only by time_signal messages of segmentation type **Other**. It is not triggered by splice_insert messages of any segmentation type, and is not triggered by time_signal messages of any type except **Other**.

SCTE-35 messages of type ID "splice insert" and messages of type ID "time signal" can both include "Other" time_signal messages. Therefore, when enabling blackout, the **ad avail mode** (p. 208) is not relevant. Blackout works the same with either mode.

The segmentation ID triggers blackout based on “events,” as shown in the following table.

<table>
<thead>
<tr>
<th>SCTE-35 Segmentation Type</th>
<th>Blacked out</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chapter Start</td>
<td>Start blacking out</td>
</tr>
<tr>
<td>Chapter End</td>
<td>End blacking out</td>
</tr>
<tr>
<td>Network Start</td>
<td></td>
</tr>
<tr>
<td>Network End</td>
<td><strong>End blacking out</strong></td>
</tr>
<tr>
<td>Program Start</td>
<td>Start blacking out</td>
</tr>
<tr>
<td>Program End</td>
<td>End blacking out</td>
</tr>
<tr>
<td>Unscheduled Event Start</td>
<td>Start blacking out</td>
</tr>
<tr>
<td>Unscheduled Event End</td>
<td>End blacking out</td>
</tr>
</tbody>
</table>

For example, if the blackout feature is enabled, then blanking always occurs when a Program Start message is encountered and always ends when a Program End message is encountered.

Note that the triggers for blackout on a Network event are different from the other events:

- With Network, blanking starts when the **Network End** instruction is encountered.
- With other events, blanking starts when the "Event Start" instruction is encountered.
End Event Trigger Hierarchy

Events have the following “strength hierarchy.”

<table>
<thead>
<tr>
<th>SCTE-35 Segmentation Type</th>
<th>Strength</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network</td>
<td>1 (Strongest)</td>
</tr>
<tr>
<td>Unscheduled Event</td>
<td>2</td>
</tr>
<tr>
<td>Program</td>
<td>3</td>
</tr>
<tr>
<td>Chapter</td>
<td>4 (Weakest)</td>
</tr>
</tbody>
</table>

A blackout can be ended only by an event of equal or greater strength than the event that started it.

For example, if the blackout is started by a Program Start, it can be ended by a Network Start, an Unscheduled Event End or a Program End. It cannot be ended by a Chapter End. MediaLive ignores the “end blackout” instruction implied by the Chapter End.

Blackout Restriction Flags

Restrictions in the Input

The segmentation descriptors in messages that are blackout triggers always include two types of flags. These flags provide additional information as guidance for blackout in specific situations:

- **web_delivery_allowed_flag**
  - True means that there is no restriction on including the event’s content in a stream that is intended for web delivery. There is no need to black out content in streams intended for web delivery.
  - False means that there is a restriction. The content should be blacked out.

- **no_regional_blackout_flag**
  - True means that there is no restriction on including the event’s video in a stream intended for regional markets. There is no need to black out content in streams intended for regional markets.
  - False means that there is a restriction. The content should be blacked out.

If both flags are present (which is usually the case; it is unusual to have only one flag present), then a “false” for one flag takes precedence over a “true” for the other flag. Blackout should occur.

Typically, in any message in the input only one of these flags is ever set to false, so only one restriction is ever in place. There would typically never be both a regional delivery restriction and a web delivery restriction. This is because if content is considered restricted for regional delivery, then it would not also be considered restricted for web delivery (where the concept of a region makes no sense).

To summarize, this is the blackout logic that applies to each event that is encountered.

<table>
<thead>
<tr>
<th>Content of corresponding SCTE-35 message: Web delivery allowed?</th>
<th>Content of corresponding SCTE-35 message: Regional delivery allowed?</th>
<th>Result</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1</td>
<td>Flag is not present</td>
<td>Flag is not present</td>
<td>Blackout occurs</td>
</tr>
</tbody>
</table>
Enabling SCTE-35 Passthrough or Removal

<table>
<thead>
<tr>
<th></th>
<th>Content of corresponding SCTE-35 message: Web delivery allowed?</th>
<th>Content of corresponding SCTE-35 message: Regional delivery allowed?</th>
<th>Result</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>S2</td>
<td>Flag is set to &quot;true&quot;</td>
<td>Flag is set to &quot;true&quot;</td>
<td></td>
<td>Blackout doesn't occur</td>
</tr>
<tr>
<td>S3</td>
<td>Flag is set to &quot;true&quot;</td>
<td>Flag is set to &quot;false&quot;</td>
<td></td>
<td>Blackout occurs</td>
</tr>
<tr>
<td>S4</td>
<td>Flag is set to &quot;false&quot;</td>
<td>Flag is set to &quot;true&quot;</td>
<td></td>
<td>Blackout occurs</td>
</tr>
</tbody>
</table>

**MediaLive Handling of Restrictions**

You can modify this default blackout behavior by instructing MediaLive to ignore a restriction flag that is set to false, so that blackout will *not* occur for this event. In other words, to use this logic, "Even if the message indicates to black out content because a regional blackout is in place, do not follow this instruction. Ignore the fact that a regional blackout is in place and do not black out content."

You modify the behavior by setting fields in the channel.

### Enabling SCTE-35 Passthrough or Removal in the Output

You can set up the MediaLive channel so that SCTE-35 messages from the input are passed through (included) in the data stream for the following outputs:

- Outputs in an Archive output group.
- Outputs in an HLS output group.
- Outputs in a MediaPackage output group. For these types of output groups, passthrough is always enabled. You can't disable it.
- Outputs in a UDP output group.

#### Alignment with Video

The PTS of the SCTE-35 message is adjusted to match the PTS of the corresponding video frame.

#### Passthrough Is at the Output Level

SCTE-35 passthrough or removal applies at the output level. The messages are passed through or removed only in a specific output. For most outputs, the default behavior (if you do not change the configuration fields) is to remove the messages. For MediaPackage outputs, the default behavior is to pass through the messages; you can't change this behavior.

### Enabling Passthrough for Archive Outputs

Follow this procedure if you want to enable or disable passthrough for Archive outputs.
To enable passthrough

1. In the channel that you are creating, find the Archive output group that contains the output that you want to set up.
2. Choose that output.
3. In PID settings, complete the following fields:
   - SCTE-35 control: Set to Passthrough.
   - SCTE-35 PID: Leave the default PID or enter the PID where you want the SCTE-35 messages to go.
4. If appropriate, repeat for other outputs in this or other Archive output groups.

All SCTE-35 messages from the input are included in the data stream of the outputs that you have set up.

Enabling Passthrough for HLS Outputs

Follow this procedure if you want to enable or disable passthrough for HLS outputs.

To enable passthrough

1. In the channel that you are creating, find the HLS output group that contains the output that you want to set up.
2. Choose that output.
3. In PID settings, complete the following fields:
   - SCTE-35 behavior: Set to Passthrough.
   - SCTE-35 PID: Leave the default PID or enter the PID where you want the SCTE-35 messages to go.
4. If appropriate, repeat for other outputs in this or other HLS output groups.

All SCTE-35 messages from the input will be included in the data stream of the outputs that you have set up.

Enabling Passthrough for UDP Outputs

Follow this procedure if you want to enable or disable passthrough for UDP outputs.

To enable passthrough

1. In the channel that you are creating, find the UDP output group that contains the output that you want to set up.
2. Choose that output.
3. In PID settings, complete the following fields:
   - SCTE-35 control: Set to Passthrough.
   - SCTE-35 PID: Leave the default PID or enter the PID where you want the SCTE-35 messages to go.
4. If appropriate, repeat for other outputs in this or other UDP output groups.

All SCTE-35 messages from the input will be included in the data stream of the outputs that you have set up.
Sample Manifests - HLS

MediaLive supports the following HLS manifest styles for outputs:

- Adobe
- Elemental
- SCTE-35 Enhanced

This section describes the ad marker tagging for each style of output manifest.

**Note**
MediaLive doesn't interpret the ad avail decoration information in the manifest attached to the input source.

### Ad Marker: Adobe

Inserts a CUE: DURATION for each ad avail. Does not insert any CUE-OUT CONT (continuation tags) to indicate to a player joining midbreak that there is a current avail. This does not insert a CUE-IN tag at the end of the avail.

**Structure**

<table>
<thead>
<tr>
<th>Segment</th>
<th>Tag</th>
<th>Tag Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Each succeeding segment.</td>
<td>CUE-OUT</td>
<td>1</td>
</tr>
<tr>
<td>Segment in which the ad avail starts.</td>
<td>CUE-DURATION tag</td>
<td>1</td>
</tr>
</tbody>
</table>

**Tag Contents**

- CUE:DURATION contains the following:
  - duration – Duration in fractional seconds
  - id – An identifier, unique among all ad avails CUE tags
  - type – SpliceOut
  - time – The PTS time for the ad avail, in fractional seconds

**Example**

This is the tag for an ad avail lasting 414.171 PTS:

```bash
#EXT-X-CUE:DURATION="201.467",ID="0",TYPE="SpliceOut",TIME="414.171"
```

### Ad Marker: Elemental

**Structure**

<table>
<thead>
<tr>
<th>Segment</th>
<th>Tag</th>
<th>Tag Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Each succeeding segment.</td>
<td>CUE-OUT</td>
<td>1</td>
</tr>
<tr>
<td>Segment in which the ad avail starts.</td>
<td>CUE-OUT-CONT</td>
<td>0-n</td>
</tr>
</tbody>
</table>
### Tag Contents
- CUE-OUT contains DURATION
- CUE-OUT-CONT contains Elapsed time and Duration
- CUE-IN has no content

### Example

```
#EXT-X-CUE-OUT:30.000
.
.
.
# EXT-X-CUE-OUT-CONT: 8.308/30
.
.
.
# EXT-X-CUE-OUT-CONT: 20.391/30
.
.
.
# EXT-X-CUE-IN
```

### Ad Marker: SCTE-35 Enhanced

#### Structure

<table>
<thead>
<tr>
<th>Segment</th>
<th>Tag</th>
<th>Tag Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Segment in which ad avail</td>
<td>OATCLS-SCTE35</td>
<td>1</td>
</tr>
<tr>
<td>ends.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Segment in which the ad avail</td>
<td>ASSET</td>
<td>1</td>
</tr>
<tr>
<td>starts.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Segment in which the ad avail</td>
<td>CUE-OUT</td>
<td>1</td>
</tr>
<tr>
<td>starts.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Each succeeding segment.</td>
<td>CUE-OUT-CONT</td>
<td>0-n</td>
</tr>
<tr>
<td>Segment in which ad avail</td>
<td>CUE-IN</td>
<td>1</td>
</tr>
<tr>
<td>ends.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### Tag Contents
- OATCLS-SCTE35 containing the base64 encoded raw bytes of the original SCTE-35 ad avail message.
- ASSET containing the CAID or UPID as specified in the original SCTE35 message.
- 1 CUE-OUT per ad avail.
- CUE-OUT-CONT containing the following:
  - The elapsed time of the avail.
• The duration declared in the original SCTE35 message.
• SCTE35 containing the base64 encoded raw bytes of the original SCTE-35 ad avail message.

These lines repeat until the ad avail ends.
• CUE-IN to indicate the end of the avail.

Example

```ext-oatcls-scte35:/DA0AAAAAAAAAAAABq+ADAQ6QAeAhxDUVJQAAA3/PAAEUrEoICAAAAAg+2UNNAANvrtQ==
#EXT-X-ASSET:CAID=0x0000000020FB6501
#EXT-X-CUE-OUT:201.467 . .
#EXT-X-CUE-OUT-CONT:ElapsedTime=5.939,Duration=201.467,SCTE35=/DA0AAAA+…AAg+2UNNAANvrtQ==
 . .
#EXT-X-CUE-IN```

224
Working with Image Overlays

You can use the static image overlay feature to superimpose a static image onto a video in an MediaLive channel. A static image is a still image that doesn't have motion. You prepare the image and store it outside of MediaLive. You then use the schedule (p. 136) feature in MediaLive to set up a timetable that specifies when images (up to eight different images) will be shown in the running channel, and when each will be hidden.

Examples

Example 1

You want to insert a static image overlay at a specific time and run it for 10 seconds. You want the image overlay to appear in the lower-right corner of the video frame. You want the image overlay to be 50% opaque and to fade in from nothing to full 50% opacity over 2 seconds, then to fade out to nothing starting 2 seconds before the end of the insertion.

Example 2

You want to insert two static image overlays so that they both appear in the video frame either at the same time or with some overlap. You want the display of the image overlays to slightly overlap so that one image overlay appears in a location and, while that image overlay is still showing, another image overlay appears in another location. If the locations overlap either partially or completely, you want to specify which image overlay appears on top.

Features of the Static Image Overlay

The image that you overlay on a video can be a .bmp, .png, or .tga file.

You can insert up to eight images at one time. Each image is a separate "layer." You can set up the overlays to all appear on the underlying video at the same time (or not), and you can set them up to physically overlap each other (or not).

You can configure each image overlay with a start time and duration. You can insert the image overlay at any position on the video frame, as specified by x/y coordinates. You can configure with an opacity and with fade-in and fade-out.

The image is handled as follows:

- The image is overlaid on the underlying video pixel for pixel, without scaling.

  If the overlay is larger than the underlying video or overruns an edge of the underlying video, and if the system can identify this error at channel creation time, you will see an error message at that time.

  If the system can't identify the error in advance, an error message will appear while the channel is running. The channel won't stop, but the overlay request will fail.

- The image is overlaid before creation of individual output encodes (with their different resolutions and video quality). This means that if the underlying video is scaled for a particular output encode, then the image is similarly scaled.

- The image is inserted in all outputs.
Step 1: Prepare the Static Image Overlay File

You must prepare each image overlay that you want to use in your channels. The overlays are stored outside of MediaLive, for example, in an Amazon S3 bucket. An image overlay doesn't belong to MediaLive or to a specific channel in MediaLive. Rather, the image overlays are used by MediaLive.

Follow this procedure to prepare overlays when you need them.

To prepare the overlay file

1. Create a file with the following characteristics:
   - File type: A .bmp, .png, or .tga file.
   - Aspect ratio: The overlay can have any aspect ratio. It doesn't have to match the aspect ratio of the underlying video.
   - Size, in pixels: The overlay can be any resolution (size in pixels) up to the same size as the underlying video.

2. If you use a graphics program that outputs channels, set up to output the alpha channel. This ensures that the image overlay doesn't appear in a black or white box.

3. Place the prepared file in a location that is accessible to the MediaLive. You can specify the location in one of four ways:
   - Amazon S3 bucket, using SSL. For example:
     s3ssl://company.test/sample_bucket/overlay.png
   - Amazon S3 bucket, without SSL. For example:
     s3://company.test/sample_bucket/overlay.png
   - Accessible URL that requires SSL. For example:
     https://203.0.113.0/corporate_logos/large.bmp
   - Accessible URL without SSL. For example:
     http://203.0.113.254/corporate_logos/high_res.bmp

4. Make a note of the location. You will need it later.

Step 2: Insert the Overlay

You insert an overlay in the video by creating an insert action in the channel schedule. For detailed information, see Working with the Schedule (p. 136) and the section called “Creating Actions” (p. 138).

The schedule is a timetable that is attached to each channel. The schedule is designed to let you specify actions to perform on the channel at a specific time. So with an image overlay, for example, you create actions in the schedule to specify that a specific image will be overlaid on the underlying video at a specific time, for a specific duration.

When a channel is running, its configuration does not and cannot change. So the channel schedule lets you apply dynamically occurring actions to the channel without having to stop it and reconfigure.
Tagging AWS Elemental MediaLive Resources

A tag is a metadata label that you assign or that AWS assigns to an AWS resource. Each tag consists of a key and a value. For tags that you assign, you define the key and value. For example, you might define the key as stage and the value for one resource as test.

Tags help you do the following:

- Identify and organize your AWS resources. Many AWS services support tagging, so you can assign the same tag to resources from different services to indicate that the resources are related. For example, you could assign the same tag to an AWS Elemental MediaLive channel and an endpoint that you assign to an AWS Elemental MediaTailor configuration.
- Track your AWS costs. You activate these tags on the AWS Billing and Cost Management dashboard. AWS uses the tags to categorize your costs and deliver a monthly cost allocation report to you. For more information, see Use Cost Allocation Tags in the AWS Billing and Cost Management User Guide.

For tips on using tags, see the AWS Tagging Strategies post on the AWS Answers blog.

The following sections provide more information about tags for AWS Elemental MediaLive.

Supported Resources in AWS Elemental MediaLive

The following resources in AWS Elemental MediaLive support tagging:

- Channels
- Inputs
- Input security groups

For information about adding and managing tags, see Managing Tags (p. 228).

Tag Restrictions

The following basic restrictions apply to tags on AWS Elemental MediaLive resources:

- Maximum number of tags that you can assign to a resource – 50
- Maximum key length – 128 Unicode characters
- Maximum value length – 256 Unicode characters
- Valid characters for key and value – a-z, A-Z, 0-9, space, and the following characters: _ . / = + - and @
- Keys and values are case sensitive
- Don't use aws: as a prefix for keys; it's reserved for AWS use

Additionally, AWS Elemental MediaLive doesn't support the tag-based access control feature of AWS Identity and Access Management (IAM).
Managing Tags

Tags are made up of the Key and Value properties on a resource.

You can use the AWS Management Console to manage tags. You can also use the AWS Elemental MediaLive console, the AWS CLI, or the AWS Elemental MediaLive API to add, edit, or delete the values for these properties.

Tagging Using the AWS Management Console

We recommend that you manage tags by using the Tag Editor on the AWS Management Console. The Tag Editor provides a central, unified way to create and manage your tags. The Tag Editor provides the best results, including consistency between tags within MediaLive and between MediaLive and other services.

For more information, see Working with Tag Editor in Getting Started with the AWS Management Console.

Tagging Using MediaLive

For information about managing tags using the MediaLive console, see the following:

- the section called “Step 1: Complete the Channel and Input Details” (p. 86) – for information about including tags when you create a channel
- Editing and Deleting a Channel (p. 133) – for information about modifying tags in an existing channel
- Working with Inputs (p. 70) – for information about including tags in an input
- Working with Input Security Groups (p. 67) – for information about including tags in an input security group

For information about managing tags using the AWS Elemental MediaLive API, see the following:

- Resources in the AWS Elemental MediaLive API Reference
Reference: Supported Captions

This section contains tables that specify the caption formats that are supported in inputs and the caption formats that are supported in outputs.

There are several factors that control your ability to output captions in a given format:

- The type of input container. A given input container can contain captions in some formats and not in others.
- The format of the input captions. A given format of captions can be converted to some formats and not to others.
- The type of output containers. A given output container supports some caption formats and not others.

Topics
- How to Read the Supported Captions Information (p. 229)
- General Information About Supported Formats (p. 229)
- Formats Supported in an HLS Output or a MediaPackage Output (p. 231)
- Formats Supported in an RTMP Output (p. 232)
- Formats Supported in an MPEG2-TS File Output or MPEG2-UDP Streaming Output (p. 233)
- Formats Supported in a Microsoft Smooth Output (p. 234)

How to Read the Supported Captions Information

To determine if the input container and input captions that you have received are capable of producing the desired output captions, consult the tables in the following sections and follow these steps:

1. Find the table for your output container.
2. In that table, find the container type of the input that you have been provided with, and then find the input captions that are in that container.
3. In the third column, look for the output caption format that you require.

   If the format is listed, then your input is suitable.

   If the format is not listed, you must ask the provider of that input to provide an input container that includes input captions that can be converted to the required output format.

General Information About Supported Formats

The following table shows the supported formats, specifies whether they are supported in inputs or outputs, and specifies the standard that defines each format.

<table>
<thead>
<tr>
<th>Caption</th>
<th>Supported in Input</th>
<th>Supported in Output</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ancillary data</td>
<td>Yes</td>
<td></td>
<td>From MXF input, data that is compliant with “SMPTE 291M: Ancillary Data Package and</td>
</tr>
<tr>
<td>Caption</td>
<td>Supported in Input</td>
<td>Supported in Output</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>--------------------</td>
<td>---------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Space Formatting</td>
<td></td>
<td></td>
<td>Space Formatting” and that is contained in ancillary data.</td>
</tr>
<tr>
<td>ARIB</td>
<td>Yes</td>
<td>Yes</td>
<td>Captions that are compliant with ARIB STD-B37 Version 2.4.</td>
</tr>
<tr>
<td>Burn-in</td>
<td>N/A</td>
<td>Yes</td>
<td>From input: It is technically impossible for the encoder to read burn-in captions. Therefore, from an input viewpoint, they can't be considered to be captions.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>For output: Burn-in captions are captions that are converted into text and then overlaid on top of the picture directly in the video stream.</td>
</tr>
<tr>
<td>DVB-Sub</td>
<td>Yes</td>
<td>Yes</td>
<td>Captions that are compliant with ETSI EN 300 743.</td>
</tr>
<tr>
<td>Embedded</td>
<td>Yes</td>
<td>Yes</td>
<td>Captions that are compliant with the EIA-608 standard (also known as CEA-608 or SMPTE-259M or “line 21 captions”) or the CEA-708 standard (also known as EIA-708).</td>
</tr>
<tr>
<td>Embedded+SCTE-20</td>
<td>Yes</td>
<td>Yes</td>
<td>Captions that have both embedded and SCTE-20 in the video. The embedded captions are inserted before the SCTE-20 captions.</td>
</tr>
<tr>
<td>RTMP CaptionInfo</td>
<td></td>
<td>Yes</td>
<td>Captions that are compliant with the Adobe onCaptionInfo format.</td>
</tr>
<tr>
<td>RTMP CuePoint</td>
<td></td>
<td>Yes</td>
<td>Captions that are in the cuePoint format.</td>
</tr>
</tbody>
</table>
## Formats Supported in an HLS Output or a MediaPackage Output

In this table, look up your input container and captions type. Then read across to find the caption formats that are supported for an HLS output or MediaPackage output, when you have this input container and captions type.

<table>
<thead>
<tr>
<th>Source Caption Container</th>
<th>Source Caption Input</th>
<th>Supported Output Captions</th>
</tr>
</thead>
<tbody>
<tr>
<td>HLS Container</td>
<td>Embedded</td>
<td>Burn-in, Embedded</td>
</tr>
</tbody>
</table>
## Formats Supported in an RTMP Output

In this table, look up your input container and captions type. Then read across to find the caption formats that are supported for an RTMP output, when you have this input container and captions type.

<table>
<thead>
<tr>
<th>Source Caption Container</th>
<th>Source Caption Input</th>
<th>Supported Output Captions</th>
</tr>
</thead>
<tbody>
<tr>
<td>HLS Container</td>
<td>Embedded</td>
<td>Burn-in</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Embedded</td>
</tr>
<tr>
<td></td>
<td></td>
<td>WebVTT</td>
</tr>
<tr>
<td>RTMP Container</td>
<td>Embedded</td>
<td>Burn-in</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Embedded</td>
</tr>
<tr>
<td></td>
<td></td>
<td>WebVTT</td>
</tr>
<tr>
<td>MPEG2-TS Container (through the RTP or MediaConnect protocol)</td>
<td>Embedded</td>
<td>Burn-in</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Embedded</td>
</tr>
<tr>
<td></td>
<td></td>
<td>WebVTT</td>
</tr>
<tr>
<td></td>
<td>SCTE-20</td>
<td>Burn-in</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Embedded</td>
</tr>
<tr>
<td></td>
<td></td>
<td>WebVTT</td>
</tr>
<tr>
<td></td>
<td>Teletext</td>
<td>Burn-in</td>
</tr>
<tr>
<td></td>
<td></td>
<td>WebVTT</td>
</tr>
<tr>
<td></td>
<td>ARIB</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>DVB-Sub</td>
<td>Burn-in</td>
</tr>
<tr>
<td></td>
<td>SCTE-27</td>
<td>Burn-in</td>
</tr>
</tbody>
</table>

- WebVTT
- SCTE-20
- Burn-in
- Embedded
Formats Supported in an MPEG2-TS File Output or MPEG2-UDP Streaming Output

In this table, look up your input container and captions type. Then read across to find the caption formats that are supported for an MPEG2-TS or MPEG2-UDP output, when you have this input container and captions type.

<table>
<thead>
<tr>
<th>Source Caption Container</th>
<th>Source Caption Input</th>
<th>Supported Output Captions</th>
</tr>
</thead>
<tbody>
<tr>
<td>HLS Container</td>
<td>Embedded</td>
<td>Burn-in, DVB-Sub, Embedded, Embedded+SCTE-20, SCTE-20+Embedded</td>
</tr>
<tr>
<td></td>
<td>SCTE-20</td>
<td>Burn-in</td>
</tr>
<tr>
<td></td>
<td>ARIB</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>DVB-Sub</td>
<td>Burn-in</td>
</tr>
<tr>
<td></td>
<td>SCTE-27</td>
<td>Burn-in</td>
</tr>
<tr>
<td>RTMP Container</td>
<td>Embedded</td>
<td>Burn-in, DVB-Sub, Embedded, Embedded+SCTE-20, SCTE-20+Embedded</td>
</tr>
<tr>
<td></td>
<td>SCTE-20</td>
<td>Burn-in</td>
</tr>
<tr>
<td></td>
<td>ARIB</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>DVB-Sub</td>
<td>Burn-in</td>
</tr>
<tr>
<td></td>
<td>SCTE-20</td>
<td>Burn-in</td>
</tr>
<tr>
<td></td>
<td>Embedded</td>
<td>Burn-in, DVB-Sub, Embedded, Embedded+SCTE-20, SCTE-20+Embedded</td>
</tr>
<tr>
<td></td>
<td>Embedded+SCTE-20</td>
<td>Burn-in</td>
</tr>
<tr>
<td></td>
<td>SCTE-20+Embedded</td>
<td>Burn-in</td>
</tr>
<tr>
<td>MPEG2-TS Container (through the RTP or MediaConnect protocol)</td>
<td>Embedded</td>
<td>Burn-in, DVB-Sub, Embedded, Embedded+SCTE-20</td>
</tr>
</tbody>
</table>

233
### Formats Supported in a Microsoft Smooth Output

In this table, look up your input container and captions type. Then read across to find the caption formats that are supported for a Microsoft Smooth output, when you have this input container and captions type.

<table>
<thead>
<tr>
<th>Source Caption Container</th>
<th>Source Caption Input</th>
<th>Supported Output Captions</th>
</tr>
</thead>
<tbody>
<tr>
<td>HLS Container</td>
<td>Embedded</td>
<td>Burn-in, TTML</td>
</tr>
<tr>
<td></td>
<td>SCTE-20</td>
<td>Burn-in, TTML</td>
</tr>
<tr>
<td>RTMP Container</td>
<td>Embedded</td>
<td>Burn-in, TTML</td>
</tr>
<tr>
<td></td>
<td>SCTE-20</td>
<td>Burn-in, TTML</td>
</tr>
<tr>
<td>MPEG2-TS Container</td>
<td>Embedded</td>
<td>Burn-in, TTML</td>
</tr>
<tr>
<td>(through the RTP or</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MediaConnect protocol)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SCTE-20</td>
<td>Burn-in, TTML</td>
</tr>
</tbody>
</table>

---

**Source Caption Container**

- HLS Container
- RTMP Container
- MPEG2-TS Container (through the RTP or MediaConnect protocol)

**Source Caption Input**

- Embedded
- SCTE-20

**Supported Output Captions**

- Burn-in
- TTML
- SCTE-20+Embedded
- SCTE-20
- DVB-Sub
- Embedded
- Embedded+SCTE-20
- SCTE-20+Embedded
- Burn-in
- DVB-Sub
- Teletext
- ARIB
- DVB-Sub
- SCTE-27
- Burn-in
- DVB-Sub
- Teletext
- ARIB
- DVB-Sub
- SCTE-27
## Formats Supported in a Microsoft Smooth Output

<table>
<thead>
<tr>
<th>Source Caption Container</th>
<th>Source Caption Input</th>
<th>Supported Output Captions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Teletext</td>
<td></td>
<td>Burn-in, TTML</td>
</tr>
<tr>
<td>ARIB</td>
<td></td>
<td>None</td>
</tr>
<tr>
<td>DVB-Sub</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SCTE-27</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Reference: Supported Containers and Codecs

This chapter provides information about the input content types and input video and audio codecs that AWS Elemental MediaLive can ingest, and the output content types and output video and codecs that MediaLive can produce.

Topics
• Supported Input Types and Upstream Systems (p. 236)
• Supported Codecs for Inputs (p. 238)
• Supported Containers and Downstream Systems (p. 238)
• Supported Codecs for Outputs (p. 240)

Supported Input Types and Upstream Systems

The following table lists the input types and protocols that MediaLive supports. After the table are descriptions of the push and pull terms.

<table>
<thead>
<tr>
<th>MediaLive Input Type</th>
<th>Use Case</th>
<th>Upstream System and Supported Protocol</th>
<th>Live Stream Supported?</th>
<th>VOD Asset Supported?</th>
</tr>
</thead>
<tbody>
<tr>
<td>HLS</td>
<td>Pull an HLS stream or asset from an external endpoint using the HTTP protocol, with or without a secure connection.</td>
<td>HTTP server or HTTPS server</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>HLS</td>
<td>Pull an HLS stream or file from an AWS Elemental MediaStore container, using a secure connection.</td>
<td>AWS Elemental MediaStore with a custom protocol</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>HLS</td>
<td>Pull an HLS stream or file from an Amazon S3 bucket, with or without a secure connection.</td>
<td>Amazon S3 over a custom protocol</td>
<td>Yes. However, Amazon S3 isn't recommended as a source for a live stream.</td>
<td>Yes</td>
</tr>
<tr>
<td>MediaConnect</td>
<td>Push a transport stream (TS) from a flow in AWS Elemental MediaConnect.</td>
<td>AWS Elemental MediaConnect over an internal connection</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>MediaLive Input Type</td>
<td>Use Case</td>
<td>Upstream System and Supported Protocol</td>
<td>Live Stream Supported?</td>
<td>VOD Asset Supported?</td>
</tr>
<tr>
<td>----------------------</td>
<td>---------------------------------------------------------------------------</td>
<td>-----------------------------------------------</td>
<td>------------------------</td>
<td>----------------------</td>
</tr>
<tr>
<td></td>
<td>This input uses a MediaConnect flow ARN, not a URI.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MP4</td>
<td>Pull an MP4 file from an HTTP server, with or without a secure connection.</td>
<td>HTTP server or HTTPS server</td>
<td>No</td>
<td>Yes, with .mp4 file extension only</td>
</tr>
<tr>
<td>MP4</td>
<td>Pull an MP4 file from an Amazon Simple Storage Service bucket, with or without a secure connection.</td>
<td>Amazon S3 over a custom protocol</td>
<td>No</td>
<td>Yes, with .mp4 file extension only</td>
</tr>
<tr>
<td>RTP</td>
<td>Push a transport stream (TS) to a fixed endpoint on MediaLive, using the RTP protocol.</td>
<td>RTP server over RTP Push</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>RTP</td>
<td>Push a transport stream (TS) in your VPC to a fixed endpoint on MediaLive, using the RTP protocol.</td>
<td>Amazon VPC over RTP within a private cloud</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>RTMP Pull</td>
<td>Pull a stream from an external endpoint using the RTMP protocol.</td>
<td>RTMP server over RTMP Pull</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>RTMP Push</td>
<td>Push a stream to a fixed endpoint on MediaLive using the RTMP protocol.</td>
<td>RTMP server over RTMP Push</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>RTMP Push</td>
<td>Push a stream in your VPC to a fixed endpoint on MediaLive, using the RTMP protocol.</td>
<td>Amazon VPC over RTMP within a private cloud</td>
<td>Yes</td>
<td>No</td>
</tr>
</tbody>
</table>

**Ingesting with Push**

A push input works as follows: the source attempts to deliver to an endpoint that is specified in the MediaLive input. In the case of RTP protocols, the source is unaware of whether the content is being
Ingesting with Pull

A pull input works as follows: the source continually publishes to an endpoint that is outside of MediaLive. When the channel (that is connected to the input) is running, MediaLive connects to the input and ingests the content.

When the channel is not running, MediaLive does not connect to the input. (There might be other applications that do connect.)

A pull input works with a streaming input (where the source is continually being published) or a file input (where the source is made available on the endpoint and then does not change).

Supported Codecs for Inputs

The following table lists the codecs that MediaLive supports for source content.

<table>
<thead>
<tr>
<th>Container</th>
<th>Video Codecs</th>
<th>Audio Codecs</th>
</tr>
</thead>
<tbody>
<tr>
<td>RTP (MPEG TS)</td>
<td>H.264 (AVC), HEVC (H.265), MPEG-2</td>
<td>AAC, Dolby Digital, Dolby Digital Plus, MPEG Audio, PCM</td>
</tr>
<tr>
<td>HLS</td>
<td>H.264 (AVC)</td>
<td>AAC, Dolby Digital, Dolby Digital Plus</td>
</tr>
<tr>
<td>RTMP</td>
<td>H.264 (AVC)</td>
<td>AAC</td>
</tr>
</tbody>
</table>

Supported Containers and Downstream Systems

The following table lists the output formats and protocols that MediaLive supports.

<table>
<thead>
<tr>
<th>MediaLive Output Type (Output Group)</th>
<th>Use Case</th>
<th>Downstream System and Supported Protocol</th>
<th>Live Output Supported</th>
<th>VOD Output Supported</th>
</tr>
</thead>
<tbody>
<tr>
<td>Archive</td>
<td>Send transport stream (TS) files to an Amazon S3 bucket.</td>
<td>Amazon S3 over a custom protocol</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Frame Capture</td>
<td>Send a series of JPEG files to an Amazon S3 bucket.</td>
<td>Amazon S3 over a custom protocol</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>MediaLive Output Type (Output Group)</td>
<td>Use Case</td>
<td>Downstream System and Supported Protocol</td>
<td>Live Output Supported</td>
<td>VOD Output Supported</td>
</tr>
<tr>
<td>-------------------------------------</td>
<td>----------</td>
<td>--------------------------------------</td>
<td>----------------------</td>
<td>----------------------</td>
</tr>
<tr>
<td>HLS</td>
<td>Send an HLS stream to a server that supports HTTP PUT or WebDav.</td>
<td>HTTP server</td>
<td>Yes</td>
<td>Yes, when the output group is set up for VOD mode</td>
</tr>
<tr>
<td>HLS</td>
<td>Send an HLS stream to a server that supports HTTPS PUT or WebDav.</td>
<td>HTTPS server</td>
<td>Yes</td>
<td>Yes, when the output group is set up for VOD mode</td>
</tr>
<tr>
<td>HLS</td>
<td>Send an HLS stream to an Akamai CDN.</td>
<td>Akamai CDN over HTTP or HTTPS</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>HLS</td>
<td>Send an HLS stream to a MediaPackage channel using the HTTPS protocol.</td>
<td>AWS Elemental MediaPackage over HTTPS</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>HLS</td>
<td>Send an HLS stream to a container on MediaStore.</td>
<td>AWS Elemental MediaStore with a custom protocol</td>
<td>Yes</td>
<td>Yes, when the output group is set up for VOD mode</td>
</tr>
<tr>
<td>MediaPackage</td>
<td>Send an HLS stream to a MediaPackage channel.</td>
<td>AWS Elemental MediaPackage over an HTTPS WebDav</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Microsoft Smooth</td>
<td>Send a stream to an origin server or CDN that supports Microsoft Smooth Streaming.</td>
<td>A supported CDN over HTTP or HTTPS</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>RTMP</td>
<td>Send a stream to a server that supports the RTMP protocol.</td>
<td>RTMP server</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>RTMP</td>
<td>Send a stream to a server that supports the RTMPS protocol.</td>
<td>RTMPS server</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>UDP</td>
<td>Send a transport stream (TS) to a server that supports UDP.</td>
<td>UDP server</td>
<td>Yes</td>
<td>No</td>
</tr>
</tbody>
</table>
Supported Codecs for Outputs

Supported Codecs for File Outputs

The following table lists the codecs for file outputs that MediaLive supports.

<table>
<thead>
<tr>
<th>Container</th>
<th>Video Codecs</th>
<th>Audio Codecs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frame Capture</td>
<td>JPEG</td>
<td>None. A Frame Capture output doesn't include audio.</td>
</tr>
<tr>
<td>HLS</td>
<td>H.264 (AVC)</td>
<td>AAC, Dolby Digital, Dolby Digital Plus.</td>
</tr>
<tr>
<td>Microsoft Smooth</td>
<td>H.264 (AVC)</td>
<td>AAC, Dolby Digital, Dolby Digital Plus.</td>
</tr>
<tr>
<td>MPEG-2 TS</td>
<td>H.264 (AVC)</td>
<td>AAC, Dolby Digital, Dolby Digital Plus, MPEG-1 Layer II.</td>
</tr>
</tbody>
</table>

Supported Codecs for Streaming Outputs

The following table lists the codecs that MediaLive supports within streaming outputs.

<table>
<thead>
<tr>
<th>Container</th>
<th>Video Codecs</th>
<th>Audio Codecs</th>
</tr>
</thead>
<tbody>
<tr>
<td>HLS</td>
<td>H.264 (AVC)</td>
<td>AAC, Dolby Digital, Dolby Digital Plus.</td>
</tr>
<tr>
<td>MediaPackage</td>
<td>H.264 (AVC)</td>
<td>AAC, Dolby Digital, Dolby Digital Plus.</td>
</tr>
<tr>
<td>Microsoft Smooth</td>
<td>H.264 (AVC)</td>
<td>AAC, Dolby Digital, Dolby Digital Plus.</td>
</tr>
<tr>
<td>RTMP or RTMPS</td>
<td>H.264 (AVC)</td>
<td>AAC, Dolby Digital, Dolby Digital Plus.</td>
</tr>
<tr>
<td>UDP</td>
<td>H.264 (AVC)</td>
<td>AAC, Dolby Digital, Dolby Digital Plus.</td>
</tr>
</tbody>
</table>
Reference: Identifiers for Variable Data

Identifiers for variable data are $ codes that you can include in a field value to represent variable data. Typically, the variable data (for example, $d$ for the date) is resolved when you run the channel. You can include them in any of the fields that make up part of the output destination:

- Destination in Output group
- Name modifier in Output
- Segment modifier in Output

At runtime, the identifier is resolved to the appropriate data. For example, $dt$ resolves to a date and time.

When you use these identifiers, make sure that the channel doesn't end up with two (or more) outputs with identical destinations. If that happens, the channel passes validation upon creation, but fails on start.

<table>
<thead>
<tr>
<th>Identifier</th>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$dt$</td>
<td>YYYYMMDDTHHMMSS</td>
<td>UTC date and time of the start time of the channel (for all outputs except HLS) or the date and time of each segment (for HLS outputs).</td>
</tr>
<tr>
<td>$d$</td>
<td>YYYYMMDD</td>
<td>UTC date of the start time of the channel (for all outputs except HLS) or the date and time of each segment (for HLS outputs).</td>
</tr>
<tr>
<td>$t$</td>
<td>HHMMSS</td>
<td>Start time of the channel (for all outputs except HLS) or the time of each segment (for HLS outputs).</td>
</tr>
<tr>
<td>$rv$</td>
<td>Kb</td>
<td>Video bit rate.</td>
</tr>
<tr>
<td>$ra$</td>
<td>Kb</td>
<td>Total of all audio bit rates in the output.</td>
</tr>
<tr>
<td>$rc$</td>
<td>Kb</td>
<td>Container bit rate for the output, or the sum of video and all audio bit rates for the output, if the container bit rate is not specified.</td>
</tr>
<tr>
<td>$w$</td>
<td>Pixels</td>
<td>Horizontal resolution.</td>
</tr>
<tr>
<td>$h$</td>
<td>Pixels</td>
<td>Vertical resolution.</td>
</tr>
<tr>
<td>Identifier</td>
<td>Format</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>$f$</td>
<td>Integer</td>
<td>FPS Framerate without decimal places. For example, “23.976” appears as “23”.</td>
</tr>
<tr>
<td>$d$</td>
<td>$</td>
<td>Escaped $ .</td>
</tr>
</tbody>
</table>
| %0n        | Padding modifier | Modifier for any data identifier. The modifier pads the resolved value with leading zeros. The format is %0n, where n is a number. For example, to ensure the resolved value in the $h$ identifier is 5 characters long, specify the identifier as $h%05$. If the vertical resolution is “720”, then the resolved, padded value is “00720”.

The rules for which identifiers can be used in a specified destination field depend on the output type.

<table>
<thead>
<tr>
<th>Field</th>
<th>Applicable Output Types</th>
<th>Acceptable Identifiers</th>
</tr>
</thead>
<tbody>
<tr>
<td>Destination in Output group</td>
<td>Archive, HLS, Microsoft Smooth</td>
<td>$dt$, $d$, $t$</td>
</tr>
<tr>
<td>Name modifier in Output</td>
<td>Archive, Microsoft Smooth</td>
<td>All except $ra$ and $rc$</td>
</tr>
<tr>
<td>Name modifier in Output</td>
<td>HLS</td>
<td>All</td>
</tr>
<tr>
<td>Segment modifier in Output</td>
<td>Archive, Microsoft Smooth</td>
<td>All except $ra$ and $rc$</td>
</tr>
<tr>
<td>Segment modifier in Output</td>
<td>HLS</td>
<td>All</td>
</tr>
</tbody>
</table>
Working with Reservations in AWS Elemental MediaLive

You can purchase a reservation for the processing of resources that you are billed for: input processing, output processing, and add-ons such as codec licenses. Each reservation gives you a reduced rate for the processing of the relevant resources.

A reservation is a one-year commitment to a particular input or output configuration. The reservation is allocated and billed on a monthly basis through that year.

You pay an hourly rate (which is lower than the per-minute rate) for a pool of minutes that your channels consume for a month.

For information on charges for reservations, see the MediaLive price list.

Topics
• Input and Output Reservations (p. 243)
• Add-on Reservations (p. 245)
• Purchasing a Reservation (p. 247)
• Viewing Purchased Reservations (p. 247)
• Deleting a Reservation (p. 248)

Input and Output Reservations

Reservation offerings are available for inputs and for outputs.

Input Reservation Attributes and Matching

An input reservation has these attributes:

• Codec
• Resolution (a range)
• Bit rate (a range)
• Region that the input runs in

An input reservation applies to the cost of processing input. For a reservation to apply to an input, the attributes of the input reservation must match the fields in the channel's Input specification, and the channel must run in the region that is specified in the reservation. For example, suppose that your input specification for a channel is AVC, HD, and Max 20 Mbps. A reservation that matches those attributes could apply to the input in that channel.

Output Reservation Attributes and Matching

An output reservation has these attributes:

• Codec
• Resolution (a range)
• Bit rate (a range)
• Framerate (a range)
• Region that the input runs in

An output reservation applies to the cost of processing output. For a reservation to apply to an input, the attributes of the output reservation must match the corresponding fields in the channel configuration, and the channel must run in the region that is specified in the reservation. You can find the fields on the AWS Elemental MediaLive console:

• For a regular video and audio output, the fields are in the Video output section of the channel configuration. To make most of the fields appear, you must choose a codec on the page.
• For an audio-only output, the fields are in the Audio output section of the channel configuration.

There is a match if the value of a field in the channel is equal to or falls within the range of the corresponding attribute. For example, a bit rate of 29.97 fps in the channel configuration falls within the range of a bit rate attribute of \(\leq30\text{fps}\) in the reservation.

If just one of the fields does not match its corresponding reservation attribute, then there is no match between the output and reservation.

How an Input or Output Reservation Is Applied

At the start of each monthly billing cycle, AWS replenishes each reservation with the pool of minutes for the month.

At the end of the cycle, AWS applies the minutes from a given reservation to reduce the cost for the processed items (inputs or outputs) whose attributes match this reservation. For each minute in the month, AWS determines if one or more matching items was running. It accumulates these “running minutes” within the hour, up to a maximum of 60 minutes in the hour.

After the reservation minutes are used up for the hour, AWS charges the regular rate-per-minute for the remainder of the items in that hour.

Running Minutes Can Be Allocated Over Items

The running minutes could come from more than one item. For example, you start Channel A with an input that matches a given reservation. You have purchased only one instance of this reservation. After 45 minutes you start Channel B that also has an input that matches a given reservation. After 15 more minutes you stop Channel A. The running minutes are accumulated as shown by the green shading in the following illustration.

Here is another example of how different items can consume the running minutes. Suppose that in one hour you run only outputs that match a given reservation. You have purchased only one instance of this reservation. You run these four matching outputs simultaneously for 15 minutes each. During that hour, you don’t run any other matching outputs. Those four outputs would all contribute to the 60 minutes.
Add-on Reservations

Reservations are available for those items in the MediaLive price list, such as codec licenses, that are considered to be add-ons.

An add-on reservation applies to the cost of the add-on for the entire channel. The reservation reduces the cost of the add-on regardless of how many times the add-on applies to the channel. For example, if three outputs in the same channel both use an advanced audio codec, you need only one reservation to reduce the cost of the add-on. You don’t need three reservations for this channel.

Reservation Attributes

The add-on reservations have these attributes:

- Add-on (Advanced Audio, or Audio Normalization)
- Region in which the channel is running

How an Add-on Reservation Is Applied

At the start of each monthly billing cycle, AWS replenishes each add-on reservation with the pool of minutes for the month.

At the end of the cycle, AWS applies the minutes from a given reservation to reduce the cost for channels that use the add-on. For each minute in the month, it determines if one or more matching channels was running. A channel matches the reservation if the add-on feature is enabled.
AWS accumulates these running minutes within the hour, up to a maximum of 60 minutes in the hour. After the reservation minutes are used up for the hour, AWS charges the regular rate-per-minute for the remainder for those channels for that hour.

**Add-ons Are Per Channel**

A channel matches the reservation if the add-on feature is enabled one or more times. Within one channel, the number of outputs that use the add-on isn't relevant; the reservation is consumed only once for the entire channel. For example, if there are two outputs in one channel that enable audio normalization, only one reservation is consumed.

**Running Minutes Can Be Allocated Over Channels**

The same rule applies to add-ons as to input and output reservations (p. 244), except that the item is always a channel. For example, you start Channel A with two outputs that match the Advanced Audio reservation. You have purchased only one instance of this reservation. After 45 minutes you start Channel B that has one output that matches the same reservation. After 15 more minutes you stop Channel A. The running minutes are accumulated as shown by the green shading in the following illustration.

Here is another example of how different channels can consume the running minutes. Suppose that in one hour you run only channels that match the Advanced Audio reservation. You have purchased only one instance of this reservation. You run these four matching outputs simultaneously for 15 minutes each. During that hour, you don't run any other matching outputs. Those four outputs would all contribute to the 60 minutes.

**Licensing Bursts Are Not Supported**

The same rule applies to add-ons as to input and output reservations (p. 244), except that the item is always a channel. For example, in one hour you run four channels that match the Advanced Audio reservation. You have purchased only one instance of this reservation. You run these four matching channels simultaneously for 60 minutes each. Only one of these channels is eligible for the reservation because one channel is enough to use up the 60 running minutes per hour.

**Unused Minutes**

At the end of the cycle, if some of the minutes in the reservation are not used, those minutes are lost; the minutes are not transferred to the next month.
Purchasing a Reservation

On the console, use the Reservations tab to purchase one or more reservations.

To purchase a reservation (console)

2. In the navigation pane, choose Reservations, and then choose Reserve offerings.
3. On the Offerings page, complete the Filter offerings section to filter for specific offerings. For more information, see Filtering on the Offering Page (p. 247).
4. Choose an offering, and then choose Add to cart. The Cart tab title in the upper-left pane increments to show the total offerings currently in the cart. (To remove an offering that you added to the cart, switch to the Cart tab.)
5. To view the cart contents, choose the Cart tab.
6. To purchase all the offerings that are displayed on the Cart tab, choose Purchase.

Filtering on the Offerings Page

The Offerings page shows the different reservations that you can purchase:

- Input and output offerings, which are described as follows:
  - Resolution – Codec – Input/output – Bitrate – Framerate (for outputs only) – Region
  - For example: UHD AVC input at 10-20 mbps in US West (Oregon)
- Channel (add-ons) offerings, which are described as follows:
  - Add-on – Region
  - For example: Advanced Audio reserved outputs in US West (Oregon)

You can filter the offerings using the filters in the left pane:

- You can filter for the reservation type: input, output, or channel (for add-ons).
- You can filter the offerings based on attributes, such as resolution or bit rate.
- Match existing channel filters the offerings to show only those offerings that match the inputs and outputs in the chosen channel.
- Special feature filters the offerings to show only add-on offerings.

Filtering does not affect the items in the cart.

Viewing Purchased Reservations

On the console, you can view the reservations that you have purchased.

To view your purchased reservations (console)

2. In the navigation pane, choose Reservations.

   The information displayed for each reservation includes its expiry date in the End column.
Deleting a Reservation

When a reservation has expired, you can delete the reservation from the list.

**To delete an expired reservation (console)**

2. In the navigation pane, choose Reservations.
3. Choose the item or items, and then choose Delete.
Limits in MediaLive

The following sections provide information about the limits in AWS Elemental MediaLive. You can request an increase to soft limits. See request a limit increase. You cannot change hard limits.

Soft Limits

Soft limits are limits for which you can request an increase.

<table>
<thead>
<tr>
<th>Resource or Operation</th>
<th>Default Limit</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Channels</td>
<td>5</td>
<td>The maximum number of channels that you can create in the current region.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>You can request a limit increase.</td>
</tr>
<tr>
<td>Push inputs (not including VPC push inputs)</td>
<td>5</td>
<td>The maximum number of push inputs (not including VPC push inputs) that you can create in the current region.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>You can request a limit increase.</td>
</tr>
<tr>
<td>Pull inputs</td>
<td>100</td>
<td>The maximum number of pull inputs that you can create in the current region.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>You can request a limit increase.</td>
</tr>
<tr>
<td>VPC push inputs</td>
<td>50</td>
<td>The maximum number of VPC push inputs that you can create in the current region.</td>
</tr>
<tr>
<td>Input security groups</td>
<td>5</td>
<td>The maximum number of input security groups that you can create in the current region.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>You can request a limit increase.</td>
</tr>
<tr>
<td>Reservations</td>
<td>50</td>
<td>The maximum number of reservations that you can create in the current region.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>You can request a limit increase.</td>
</tr>
</tbody>
</table>

Hard Limits

Hard limits are limits you can't increase.

<table>
<thead>
<tr>
<th>Resource or Operation</th>
<th>Default Limit</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Any MediaLive Request</td>
<td>5 steady-state TPS (transactions per second)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>30 burst TPS</td>
<td></td>
</tr>
</tbody>
</table>
# Document History for User Guide

The following table describes the documentation for this release of AWS Elemental MediaLive.

- **API version**: latest

<table>
<thead>
<tr>
<th>Update History Change</th>
<th>Update History Description</th>
<th>Update History Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Limits for VPC inputs (p. 249)</td>
<td>The table for MediaLive limits now includes the limits for VPC inputs</td>
<td>May 3, 2019</td>
</tr>
<tr>
<td>Changing the channel class (p. 133)</td>
<td>You can now change the channel class for an existing channel</td>
<td>May 3, 2019</td>
</tr>
<tr>
<td>Standard channels and Single-pipeline channels (p. 86)</td>
<td>You can now set up a channel with a single-pipeline. The Create channel page includes a Channel class field</td>
<td>April 5, 2019</td>
</tr>
<tr>
<td>Revision to documentation for setting up access (p. 16)</td>
<td>The procedures for setting up the user and the service with AWS IAM access have been revised and split into two chapters, Setting Up&gt; and Setting Up for Production.</td>
<td>March 22, 2019</td>
</tr>
<tr>
<td>MediaPackage output group (p. 105)</td>
<td>There is a new output group type: MediaPackage, which lets your send output to AWS Elemental MediaPackage. This new output group type creates an HLS output that is streamlined for delivery to MediaPackage.</td>
<td>March 13, 2019</td>
</tr>
<tr>
<td>Support for pausing and unpauing a channel pipeline (p. 138)</td>
<td>The channel schedule now supports the ability to enter actions to pause and unpause a pipeline in a channel.</td>
<td>March 8, 2019</td>
</tr>
<tr>
<td>Support for encrypted HLS inputs (p. 59)</td>
<td>MediaLive now supports ingest of encrypted HLS inputs.</td>
<td>March 5, 2019</td>
</tr>
<tr>
<td>Push inputs from your VPC (p. 70)</td>
<td>You can create an RTP push input or an RTMP push input to push content from an upstream system that is in your Amazon VPC to MediaLive. Several chapters have been updated, including the chapter about inputs and the chapter about creating a channel from scratch.</td>
<td>February 20, 2019</td>
</tr>
<tr>
<td>Feature</td>
<td>Description</td>
<td>Date</td>
</tr>
<tr>
<td>------------------------------------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-----------------------</td>
</tr>
<tr>
<td>Limits for reservations (p. 249)</td>
<td>The table for MediaLive limits now includes the limits for reservations.</td>
<td>February 11, 2019</td>
</tr>
<tr>
<td>Tagging (p. 227)</td>
<td>There is now support for attaching tags to channels, inputs, and input security groups.</td>
<td>February 8, 2019</td>
</tr>
<tr>
<td>Frame capture output group (p. 98)</td>
<td>There is a new output group type: frame capture output group, to send a series of frame capture files to Amazon Simple Storage Service.</td>
<td>January 25, 2019</td>
</tr>
<tr>
<td>Integration with AWS CloudTrail (p. 130)</td>
<td>There is now support for logging MediaLive API calls with CloudTrail.</td>
<td>January 18, 2019</td>
</tr>
<tr>
<td>Integration with AWS Elemental MediaConnect (p. 70)</td>
<td>You can set up to use a flow from AWS Elemental MediaConnect as an input for a channel. Information has been added to the chapter about inputs and the chapter about creating a channel from scratch.</td>
<td>December 7, 2018</td>
</tr>
<tr>
<td>Revision to documentation for schedules (p. 136)</td>
<td>The channel schedule now supports the ability to enter actions to switch the channel from ingesting one input to another input. This chapter has been revised.</td>
<td>November 8, 2018</td>
</tr>
<tr>
<td>Input switching (p. 189)</td>
<td>There is now support for input switching. The channel must be set up with multiple inputs, and the schedule must contain actions to switch from one input to another. A new chapter on input switching has been added. In addition, information about multiple inputs (which are required for input switching) has been added in the existing chapters on planning and creating a channel.</td>
<td>November 8, 2018</td>
</tr>
<tr>
<td>Change in input limits (p. 249)</td>
<td>Input limits are now split into two categories: push and pull.</td>
<td>November 8, 2018</td>
</tr>
<tr>
<td>QVBR rate control mode feature (p. 116)</td>
<td>There is a new option for the rate control field that is part of setting up the video encode in the outputs of a channel. The option is “quality-defined variable bitrate” (QVBR).</td>
<td>October 17, 2018</td>
</tr>
</tbody>
</table>
Doc-only update, Introduced RSS feed (p. 250) | You can now subscribe to RSS feeds for notifications of updates to this user guide. | June 21, 2018

<table>
<thead>
<tr>
<th>Change</th>
<th>Description</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requirement for redundant channels in AWS Elemental MediaPackage</td>
<td>If your downstream system is AWS Elemental MediaPackage, there is no longer a requirement to set up two channels in that service. The sections the section called “Step 2: Set Up the Downstream System” (p. 48) and the section called “Setting Up the Downstream System” (p. 63) have been revised to reflect this change.</td>
<td>August 31, 2018</td>
</tr>
<tr>
<td>Schedule feature</td>
<td>There is now support for adding SCTE-35 messages and static image overlays to the channel's schedule, for inclusion in a running channel. The sections Working with the Schedule (p. 156) and Working with Image Overlays (p. 225) have been added.</td>
<td>August 24, 2018</td>
</tr>
<tr>
<td>Reservations feature</td>
<td>There is now support for purchasing a reservation for processing. With a reservation, you pay a lower rate on specific processing. The section Working with Reservations (p. 243) has been added.</td>
<td>June 19, 2018</td>
</tr>
<tr>
<td>Channel logs feature</td>
<td>There is now support for sending log information to Amazon CloudWatch Logs. The sections the section called “Logging” (p. 95) and the section called “Monitoring Using Amazon CloudWatch Logs” (p. 128) have been added. The section Setting Up for Production (p. 16) has been revised to include setup for logs.</td>
<td>June 13, 2018</td>
</tr>
<tr>
<td>Console alerts feature</td>
<td>There is now support for viewing channel alerts on the console. See Viewing and Monitoring Channels (p. 124). The information on setting up for alerts has been moved to this chapter.</td>
<td>June 6, 2018</td>
</tr>
<tr>
<td>Change</td>
<td>Description</td>
<td>Date</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
<td>------------</td>
</tr>
<tr>
<td><strong>Support for RTMP outputs</strong></td>
<td>the section called “Setting Up the Downstream System” (p. 63) now includes RTMP. The sections the section called “Creating an RTMP Output Group” (p. 106) and the section called “Settings for an RTMP Output” (p. 113) have been added. the section called ” Step 3: Match Formats to Categories” (p. 177) now includes RTMP CaptionInfo. the section called “Formats Supported in an RTMP Output” (p. 232) has been added.</td>
<td>April 18, 2018</td>
</tr>
<tr>
<td><strong>Input edit feature, Input security group edit feature</strong></td>
<td>Editing of an existing input and an editing of an existing input security group are now supported. See the section called “Editing an Input” (p. 84) the section called “Editing an Input Security Group” (p. 68)</td>
<td>March 23, 2018</td>
</tr>
<tr>
<td><strong>Input delete feature, Input security group delete feature</strong></td>
<td>Deleting an input and deleting an input security group are now supported. See the section called “Editing an Input” (p. 84) the section called “Editing an Input Security Group” (p. 68)</td>
<td>March 23, 2018</td>
</tr>
<tr>
<td><strong>Creating a channel</strong></td>
<td>Creating a Channel from Scratch (p. 86) has been extensively revised, particularly with more information and examples on setting up destinations for output groups.</td>
<td>March 23, 2018</td>
</tr>
<tr>
<td><strong>Input specification feature</strong></td>
<td>Input specification fields ensure that the service allocates sufficient processing resources and correctly calculates processing charges. See the section called “Step 2: Attach Inputs to the Channel” (p. 89).</td>
<td>February 15, 2018</td>
</tr>
<tr>
<td><strong>Channel edit feature</strong></td>
<td>Editing of the fields in an existing (saved) channel is now supported. See the section called “Editing a Channel” (p. 133).</td>
<td>February 15, 2018</td>
</tr>
<tr>
<td>Change</td>
<td>Description</td>
<td>Date</td>
</tr>
<tr>
<td>------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>------------</td>
</tr>
<tr>
<td>Custom template feature</td>
<td>Users can create custom templates from existing channels, and can import those templates into new channels. See <em>Creating a Channel from a Template or by Cloning</em> (p. 120).</td>
<td>February 15, 2018</td>
</tr>
<tr>
<td>New service and guide</td>
<td>This is the initial release of <em>AWS Elemental MediaLive User Guide</em>.</td>
<td>November 27, 2017</td>
</tr>
</tbody>
</table>

**Note**

- The AWS Media Services are not designed or intended for use with applications or in situations requiring fail-safe performance, such as life safety operations, navigation or communication systems, air traffic control, or life support machines in which the unavailability, interruption or failure of the services could lead to death, personal injury, property damage or environmental damage.

- A component of AWS Elemental MediaLive is licensed under the AVC patent portfolio license for the personal and non-commercial use of a consumer to (i) encode video in compliance with the AVC standard ("AVC video") and/or (ii) decode AVC video that was encoded by a consumer engaged in a personal and non-commercial activity and/or was obtained from a video provider licensed to provide AVC video. No license is granted or shall be implied for any other use. A component of AWS Elemental MediaLive is licensed under the mpeg-4 patent portfolio license for the personal and non-commercial use of a consumer for (i) encoding video in compliance with the mpeg-4 visual standard ("mpeg-4 video") and/or (ii) decoding mpeg-4 video that was encoded by a consumer engaged in a personal and non-commercial activity and/or was obtained from a video provider licensed to provide AVC video. No license is granted or shall be implied for any other use. Additional information may be obtained from MPEG-LA, LLC. See [http://www.mpegla.com](http://www.mpegla.com).

- AWS Elemental MediaLive may contain Dolby Digital and Dolby Digital Plus, which are protected under international and U.S. copyright laws as unpublished works. Dolby Digital and Dolby Digital Plus are confidential and proprietary to Dolby Laboratories. Their reproduction or disclosure, in whole or in part, or the production of derivative works therefrom without the express permission of Dolby Laboratories is prohibited. © Copyright 2003-2015 Dolby Laboratories. All rights reserved.
AWS Glossary

For the latest AWS terminology, see the AWS Glossary in the AWS General Reference.