Building a Live Video Channel with MediaLive, MediaPackage and CloudFront

**SPL-207 - Version 1.1.11**

© 2024 Amazon Web Services, Inc. or its affiliates. All rights reserved. This work may not be reproduced or redistributed, in whole or in part, without prior written permission from Amazon Web Services, Inc. Commercial copying, lending, or selling is prohibited. All trademarks are the property of their owners.

Note: Do not include any personal, identifying, or confidential information into the lab environment. Information entered may be visible to others.

Corrections, feedback, or other questions? Contact us at [*AWS Training and Certification*](https://support.aws.amazon.com/#/contacts/aws-training).

**Lab Overview**

Video is consumed over the internet in two main modes: on-demand (VOD) or as a real-time (live) stream.

* Video-on-demand streaming could be compared to renting or owning a DVD. The viewer may subscribe to a service, or pay a one-time fee, or purchase a movie. For the period of time that access is granted, the viewer can start, stop, pause, rewind, fast forward across the entire duration of the VOD asset.
* Live streaming is exactly what it says it is: live. The stream may be a mix of pre-recorded content (e.g., interviews) and live action (e.g., sports), but it has a defined start and stop time. The online viewer can join the stream and perhaps go “back in time” but eventually the stream will end and will no longer be accessible.

Live streaming is growing rapidly. More live streaming content is being made available by studios, as well as by users sharing on social networks. In conjunction with this, more devices capable of streaming live video being deployed to the market (for example, smart phones, tablets, PC, smart TV, set-top box, etc.).

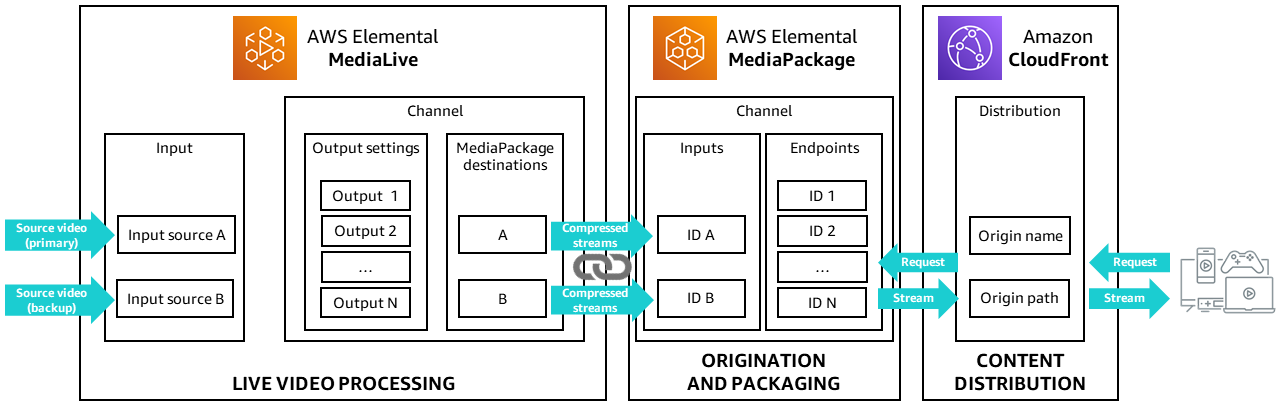
In this lab, you will create a live streaming video workflow using AWS Elemental MediaLive, AWS Elemental MediaPackage, and Amazon CloudFront.

LIVE STREAMING WORKFLOW

Live streaming involves three main stages:

* **Encoding/compression**: In this stage a high-resolution, high-bitrate source stream is compressed into several lower-resolution/bitrate versions, suitable for streaming in real time over the internet.
* **Packaging**: The compressed video steams are packaged into segments, usually 2-10 seconds each. This allows video players to adapt to fluctuating network conditions by requesting successive segments from the list of available resolutions/bitrates. This is called adaptive bitrate streaming, or ABR.
* **Distribution**: In order to serve hundreds, thousands, or millions of simultaneous viewers, the packaged video streams need to be fanned out through large distribution networks. This is referred to as the Content Delivery Network or Content Distribution Network (CDN).

WHAT YOU WILL BE DOING IN THE LAB



In this lab, you will:

* Set up AWS Elemental MediaLive to ingest two source input feeds and transcode content into two adaptive bitrate (ABR) streams, and output those two streams.
* Configure AWS Elemental MediaPackage to ingest the MediaLive outputs and package the live streams into formats suitable for end user viewing (Packaging).
* Configure an Amazon CloudFront distribution to use the MediaPackage packaged content as its origin, so that the CloudFront distribution can deliver your live stream to viewers for access via mobile devices and desktop browsers with low latency and high transfer speeds (Distribution).

TOPICS COVERED

By the end of this lab, you will be able to:

* Connect AWS Elemental MediaLive to an incoming source
* Use MediaLive to create multiple bitrate versions of a source media file
* Configure MediaPackage and CloudFront to deliver live streaming content to mobile and desktop devices
* Play your live video stream on your mobile device or computer

TECHNICAL KNOWLEDGE PREREQUISITES

To successfully complete this lab, you should be familiar with basic navigation of the AWS Management Console.

**Start lab**

1. To launch the lab, at the top of the page, choose **Start lab**.

**Caution:** You must wait for the provisioned AWS services to be ready before you can continue.

1. To open the lab, choose **Open Console**.

You are automatically signed in to the AWS Management Console in a new web browser tab.

**WARNING:** **Do not change the Region unless instructed.**

COMMON SIGN-IN ERRORS

**Error: You must first sign out**



If you see the message, **You must first log out before logging into a different AWS account:**

* Choose the **click here** link.
* Close your **Amazon Web Services Sign In** web browser tab and return to your initial lab page.
* Choose **Open Console** again.

**Error: Choosing Start Lab has no effect**

In some cases, certain pop-up or script blocker web browser extensions might prevent the **Start Lab** button from working as intended. If you experience an issue starting the lab:

* Add the lab domain name to your pop-up or script blocker’s allow list or turn it off.
* Refresh the page and try again.

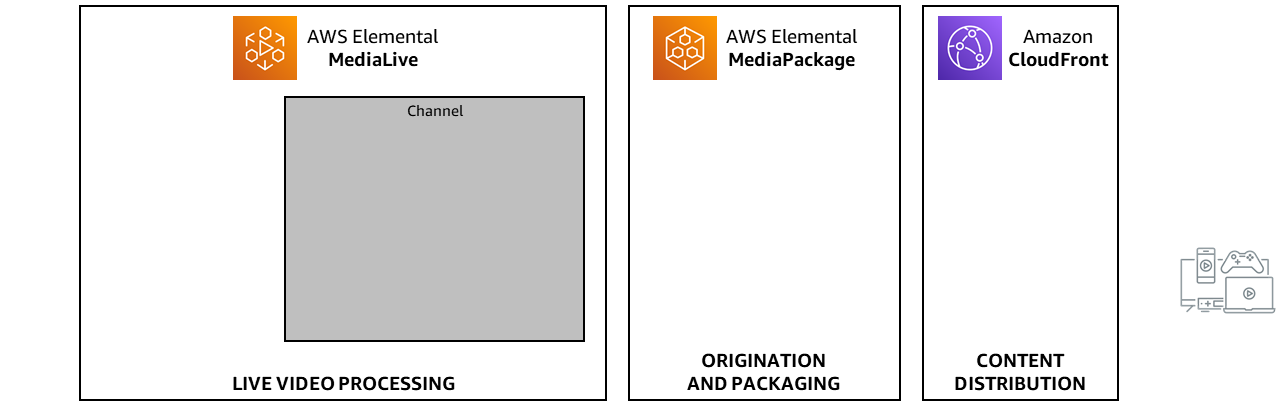
MediaLive has two main components: inputs and channels.

* An input is a video asset that is to be transcoded and packaged by MediaLive. The source of the video asset is the system in your end-to-end workflow whose activities occur before those of AWS Elemental MediaLive. (which connect to the sources)
* A channel is attached to an input and specifies the encoding instructions and output destinations.

In this task, you will configure an input in MediaLive, and attach that input to the MediaLive channel.

**Task 1: Create a Channel in MediaLive**

In this task you will begin configuring a MediaLive channel by naming your channel and specifying certain permissions.



1. In the **AWS Management Console**, on the **Services** menu, enter

MediaLive

 into the search field.

1. Click **MediaLive** in the results.

The MediaLive home page displays.

1. From the **Get started pane** select **Create channel**.

CONFIGURE AN IAM ROLE

MediaLive needs permissions to access other AWS services so it can perform its functions. The Identity and Access Management, or IAM service grants these permissions, via IAM roles. A MediaLiveAccessRole was created as part of the lab setup. MediaLive assumes this role in order to be able to access other AWS services.

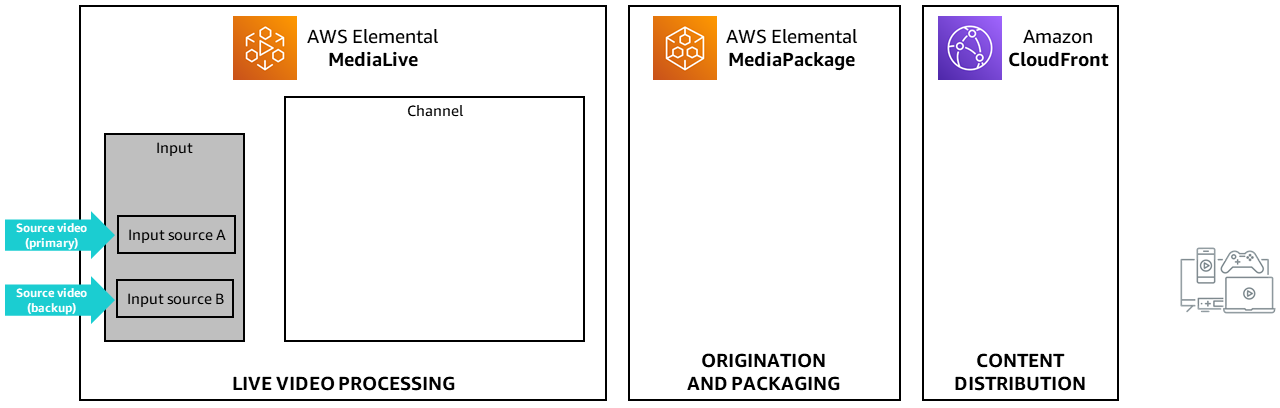
1. Under **IAM role** select  **Specify custom role ARN**.

The field may be prepopulated with a value but you will replace it with an updated one.

1. Next to **Specify custom role ARN** paste the value of **MediaLiveAccessRoleARN** located to the left of these instructions.

The role will look similar to: *arn:aws:iam::344121112345:role/MediaLiveAccessRole*

**Task 2: Create an Input and configure it in a MediaLive channel**



SET THE INPUT NAME AND INPUT TYPE

1. On the **Channel pane**, next to **Input attachments** select **Add**.
2. In the **Attach input pane** select **Create input**.

This opens a new tab.

1. In the **Input details pane** specify the following:

* **Input name - *required***:

HLS Input

* **Input type - *required***:  **HLS**

SET THE INPUT SOURCES

After specifying the type of input you will be ingesting (HLS), you need to provide the URLs of the source video. AWS Elemental MediaLive can work in a redundant mode, so you will provide two video streams that are configured as a redundant input source pair–meaning that the two streams should identical in terms of resolution and bitrate.

1. In the **Input source A** pane, for **URL**, enter:

(You can copy the text by clicking on the copy icon in the upper right of the text box. After pasting, delete any trailing spaces after “.m3u8”.)

https://d15an60oaeed9r.cloudfront.net/live\_stream\_v2/sports\_reel\_with\_markers.m3u8

1. In the **Input source B** pane, for **URL** enter:

https://d15an60oaeed9r.cloudfront.net/live\_stream\_v2/sports\_reel\_with\_markers.m3u8

You do not need to specify credentials for either input.

 For the purposes of this lab, you are using the same input for both source A and source B. However, in a real-life application, you would ideally have two sources that you connect to the cloud over two completely independent network routes.

1. At the bottom of the screen, click **Create input**.

MediaLive adds the input to the list of inputs and automatically creates two destinations (one primary and one redundant).

ATTACH INPUT TO THE MEDIALIVE CHANNEL

1. Navigate back to the **Create channel** tab.
2. Choose the refresh button to refresh the attachments. The new input should be displayed.
3. In the **Attach input pane** for **Input** use the dropdown selector to choose the input you just created: *HLS Input*
4. To attach the input to the MediaLive channel, select **Confirm**.

CONFIGURE THE ATTACHED INPUT

Now that the Input *HLS Input* is attached to the MediaLive channel, there is one more thing you need to configure.

1. Under **General input settings**, scroll down to **Source End Behavior**.
2. For **Source End Behavior**, select **LOOP**.

This setting will cause the input source (a file) to play continuously–simulating a live channel.

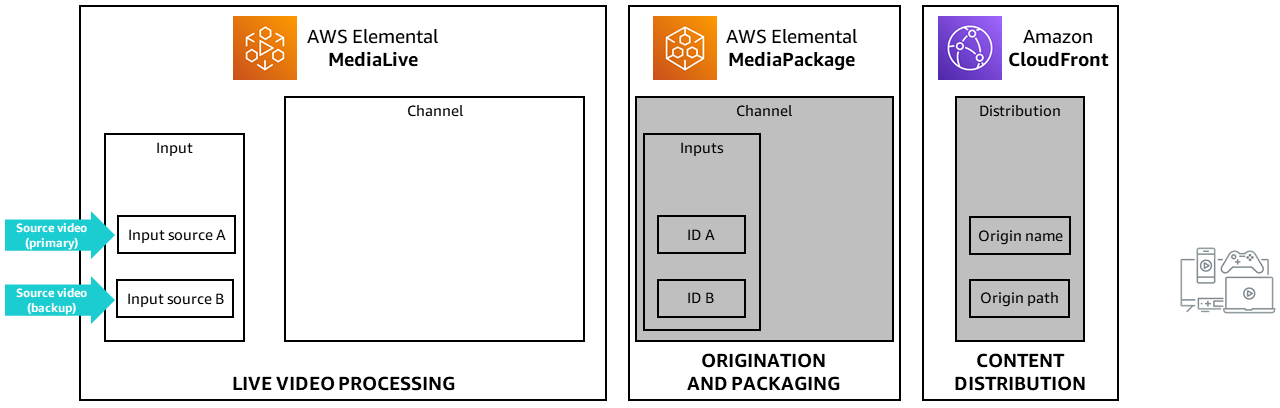
**Congratulations!** You’ve created an Input attachment with redundant sources in MediaLive. You’ve begun the process of creating a channel and attaching inputs in MediaLive. Next, you will provide this channel its output destinations, which for this this lab will be AWS Elemental MediaPackage. To get these values, you will navigate to MediaPackage.

You will return to MediaLive later to complete the process of configuring this channel.

**Note** *Do not close* the MediaLive tab you are currently working on in your browser, or navigate from MediaLive to another web page in this tab. You haven’t saved the MediaLive channel yet. If you close or navigate away from the MediaLive page, you will lose all of your work to this point.

**Task 3: Create a Channel in MediaPackage**

In this task, you will create a MediaPackage channel, which will receive the output from MediaLive and produce an Apple HLS media stream that browsers and mobile devices can consume and play for viewers.



1. In a new browser tab, open **MediaPackage**. It is very important that you do not navigate away from the MediaLive page, or close it, otherwise you will lose all the work you have done so far.

**Note** depending on your browser, there are several ways to open the MediaPackage page in a new browser tab. One method is listed below:

* In the **AWS Management Console**, click the **Services** menu, and enter

MediaPackage

 into the search field.

* Right-click on **MediaPackage** in the results and choose **Open link in new tab**.
* Go to the new **MediaPackage** tab.

1. Select **Create a new channel**.
2. In the **Channel details pane** for the **ID** name it:

MP-channel-1

1. Choose **Create**.

As part of creating the channel, MediaPackage created a pair of ingest URLs, along with their associated usernames and passwords. AWS Elemental MediaPackage will receive content on both of the ingest URLs, but only one of the streams will be used for source content at a time. If the active stream is missing any segments, then MediaPackage will automatically fail over to the other stream and continues to use this stream until failover is needed again.

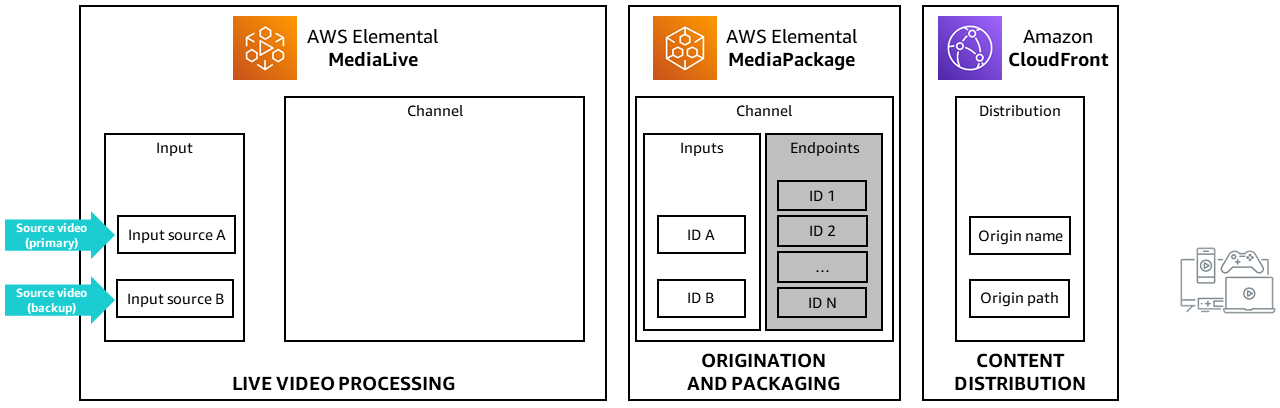
The outputs of MediaLive, will point to the two inputs of the MediaPackage channel.

**Task 4: Create Endpoints for the MediaPackage Channel**

When a user makes a request to view content, that content originates from the MediaPackage endpoint. MediaPackage endpoints also define how MediaPackage prepares content for delivery. Content can’t be served from a channel until it has an endpoint.

In this task, you will create two origin endpoints.

* The first endpoint will be for the the primary AWS Elemental MediaPackage channel. You will configure it to have a startover window that allows viewers to resume viewing from an earlier spot in the programming.
* The second endpoint is for the secondary (redundant) channel. You will configure this to play live output that is delayed by 30 seconds.



1. In the **Origin endpoints** section, click **Manage endpoints** , then configure:

* **ID:**

MP-channel-1-hls-startover

* **Startover window (sec):**

3600

**Note** For clarity, the endpoint ID should include the channel ID and the packaging format for the endpoint. This endpoint will use the startover feature, so that is noted in the endpoint name. This setting will allow a viewer to go back, or “rewind” up to 3600 seconds (1 hour) when they connect to the stream.

1. In the **Packaging type**, verify that **Apple HLS** is selected.

**Note** In this lab, MediaPackage is receiving an HLS input and is also packaging the output as Apple HLS, which is one the most commonly used packaging types. However, MediaPackage can also output to other important streaming formats, such as DASH (Dynamic Adaptive HTTP Streaming) and CMAF (Common Media Application Format). The format used is determined by the playback devices you are trying to reach.

At this point, you’ve entered one endpoint for your inputs. Next, you’ll add another endpoint for the other input.

1. Under **Endpoints**, click **Add** to add another endpoint.
2. In **New endpoint**, configure:

* **ID:**

MP-channel-1-hls-delay

* **Time delay (sec):**

30

This endpoint will use the time delay feature, so that is noted in the endpoint name. This setting will delay the live playback by 30 seconds, so that content that AWS Elemental MediaPackage receives at 12:20:00 isn’t available until 12:20:30. Requests for playback at 12:20:00 will be served with content from 12:19:30. Likewise, if you’re serving content across time zones, you can set a time delay equal to the time zone difference to make content available at, for example, 8:00 local time.

1. Leave all other settings at the default.
2. Click **Save** to save the endpoints.

**Congratulations!** You’ve successfully created a MediaPackage channel and two endpoints. Next, you will navigate back to MediaLive and use this information to finish configuring the MediaLive channel.

**Task 5: Create a CloudFront Distribution**

In this task you create an Amazon CloudFront distribution that will be used to deliver the ABR files packaged by MediaPackage to end-user devices. A distribution in Amazon CloudFront holds all information about content delivery, including where content is coming from and how it’s tracked and managed.

1. In the **AWS Management Console**, click the **Services** menu, and enter

CloudFront

 into the search field.

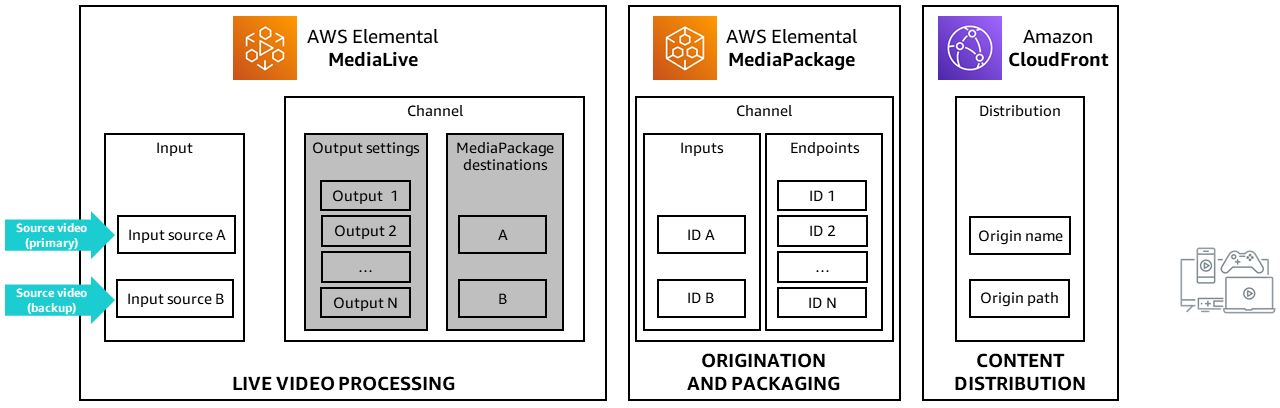
1. Right-click on **CloudFront** in the results and choose **Open link in new tab**.
2. Go to the new **CloudFront** tab.
3. Choose **Create a CloudFront distribution**.

A distribution in Amazon CloudFront holds all information about content delivery, including where content is coming from and how it’s tracked and managed. When you select this option, AWS Elemental MediaPackage communicates with Amazon CloudFront on your behalf to create a distribution for a channel and its endpoints. Because the creation process is automated and initiated from your actions in MediaPackage, there is no additional action required from you.

1. For **Origin domain**, choose, the first **MediaPackage container** link.
2. In the **Default cache behavior** section, for **Cache policy**, choose **Elemental-MediaPackage**.
3. In the **Web Application Firewall (WAF)** section, choose **Do not enable security protections**.
4. Choose **Create distribution**.

**Task 5: Create an Output Group for MediaLive**

In this task, you will create an output group that contains typical settings for outputs by using a Channel template. Then you will link the Output group to the destinations as specified by the MediaPackage channel.



1. Navigate back to your **MediaLive** browser tab that has the **Create channel** page displayed.

USE A CHANNEL TEMPLATE TO GENERATE AN OUTPUT GROUP

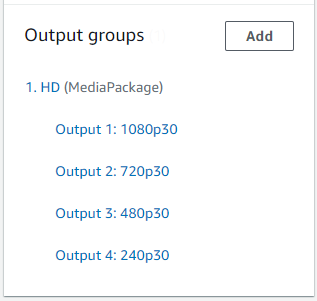
1. On the **Channel pane**, select **Channel and input details**
2. Scroll down to the **Channel template** section.
3. For **Template** search for

Live event

.

1. Choose **Live event MediaPackage**.

Your **Output groups** should look similar this:



MediaLive will take the Input attachments as specified and create an Output group that is an ABR Stack of 4 video+audio streams as specified per the selected Channel template. All the generated ABR stack needs now is a destination.

SET THE OUTPUT GROUP’S DESTINATION

1. Under **Output groups** section select the first Output group **1. HD (MediaPackage)**
2. In the **MediaPackage destination** section, enter or select

MP-channel-1

.

This is the MediaPackage channel ID that you created.

**Congratulations!** You’ve successfully created an output group and specified which MediaPackage channel ID will be the destination

1. Choose the **Channel and input details** link.
2. For **Channel name**, enter

HLS Stream Channel

.

1. Now that your configuration is complete, click **Create channel**.

After the channel has been created, **Channel state** will reflect **Idle**.

START THE MEDIALIVE CHANNEL

For the purposes of this lab, the source video content is already running. So, you can now start the MediaLive channel to begin encoding the content and sending it to AWS Elemental MediaPackage.

1. Click **Start**.

The **Channel state** changes to **Starting**, then updates to reflect **Running** in approximately 1 - 2 minutes.

 Once you start the channel in AWS Elemental MediaLive, it begins to send the streams to AWS Elemental MediaPackage. AWS Elemental MediaPackage will receive content on both of the ingest URLs you created, but only one of the streams will be used for source content at a time.

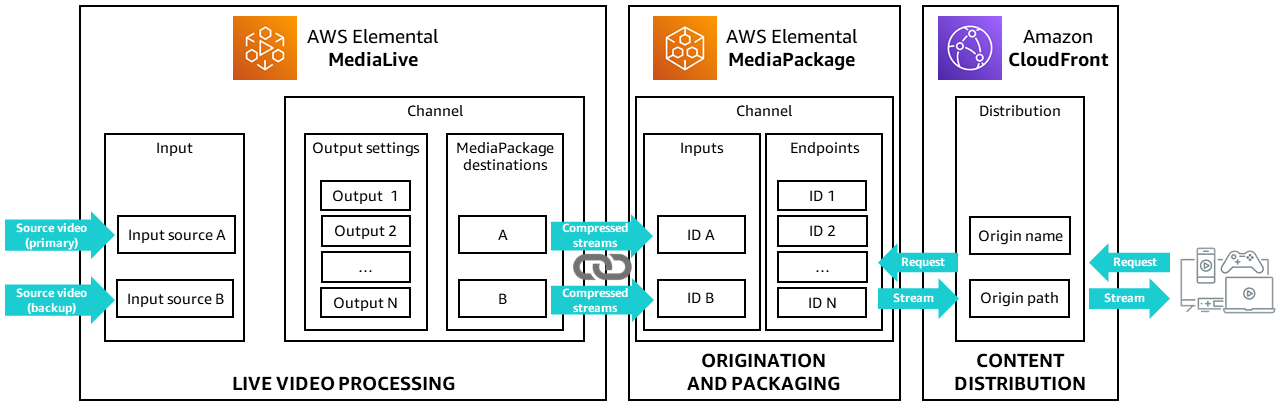
**Congratulations!** At this point, you’ve created a MediaLive channel, connected inputs and started the channel. Next, you will navigate to MediaPackage to view the content.

**Task 6: Checking Playout of the Stream**

You can use the AWS Elemental MediaPackage console to preview playback from the endpoint. This allows you to verify that MediaPackage is receiving the content stream and can package it, which is helpful for avoiding playback features after the endpoint is published, as well as for troubleshooting later if there are any playback issues.

When you view the video using the MediaPackage built in player, the content is coming directly from the MediaPackage origin, which may be hundreds or thousands of miles away. However in order to support hundreds, thousands or millions of simultaneous viewers, the content needs to be fanned through a Content Distribution Network (CDN) to get it closer to the consumer. In this lab, you are using CloudFront as the CDN.

In this task, you will check playout of the video using both MediaPackage and the Amazon CloudFront CDN distribution that was created when you created the MediaPackage channel.



PREVIEW PLAYBACK IN MEDIAPACKAGE

1. Navigate back to the **MediaPackage** console page you already have open in your browser.
2. In the **Endpoints** section, click **MP-channel-1-hls-startover**.
3. Under **Preview player**, click **Play**.
4. Scroll down, then choose the play icon.

The packaged HLS output begins playing.

**Note** There is no audio in the source video used in this lab.

PREVIEW THE MANIFEST

During HLS playback, the player receives a primary manifest, or playlist, that tells it what audio/video streams are available for playback.

1. From the **MP-channel-1-hls-startover** channel, choose the **Manifest preview** tab.

This shows the HLS manifest.

The primary manifest points to a number of variant manifests, one for each of the video quality outputs. In this lab, you created four different renditions of the stream, in a variety of resolutions and bitrates, which is why there are four variant manifests listed.

The variant manifest lists the locations of the actual video segments at each quality level.

The manifest lists the types and locations of the video segments in the stream. The player device refers to the manifest when it requests video for viewing. Next, you will play back the HLS output in a web browser.

1. Copy the Manifest preview link to a text editor. You will use this later to view the view in CloudFront.

The Manifest preview link should look similar to this: *https://10380e91fda5e303.mediapackage.us-west-2.amazonaws.com/out/v1/10b8d76eb1cd49b19d735e2621baa74c/index.m3u8*

1. In your text editor, remove the first part of the link unitl the word **/out**.
2. The edited text should look similar to: */out/v1/10b8d76eb1cd49b19d735e2621baa74c/index.m3u8*.

VIEW HLS OUTPUT IN A WEB BROWSER

HLS does not play back natively in all browsers, so depending on which browser you are using, there are several methods you can use.

1. Choose the **Origin endpoint details** tab.
2. In the **Overview** section, under **Endpoint URL**, copy the URL.

To use:

* **Safari**: paste the URL into a Safari browser and play.
* **Microsoft Edge**:
  + Open Microsoft Edge.
  + Paste the URL into the **Address bar** and press **Enter**.
* **Chrome or another browser**:
  + Use **JW Player Stream Tester**
  + Navigate to https://developer.jwplayer.com/tools/stream-tester/
  + Paste the link into the **FILE URL** field and click **Test Stream**.

You may notice that the resolution chosen by the player changes as the video plays over time. This is normal; in fact, it is the *adaptive* feature in ABR at work. The player adapts the bitrate of the video based on sampled bandwith over time.

VIEW PLAYBACK FROM CLOUDFRONT

1. In the **AWS Management Console**, on the **Services** menu, enter

CloudFront

 into the search field.

1. Click **CloudFront** in the results.
2. Choose the CloudFront distribution that you created earlier.
3. Copy the **Distribution domain name** to your text editor.

The **Distribution domain name** should look similar to: *https://d3cy1zwf07l3ex.cloudfront.net*.

1. Append the manifest string that is also in your text editor to the end of the CloudFront Distribution name.

The updated string should look similar to: **https://d3cy1zwf07l3ex.cloudfront.net/out/v1/10b8d76eb1cd49b19d735e2621baa74c/index.m3u8**

1. Use one of the method below to view your video in CloudFront.

* **Safari**: paste the URL into a Safari browser and play.
* **Microsoft Edge**:
  + Open Microsoft Edge.
  + Paste the URL into the **Address bar** and press **Enter**.
* **Chrome or another browser**:
  + Use **JW Player Stream Tester**
  + Navigate to https://developer.jwplayer.com/tools/stream-tester/
  + Paste the link into the **FILE URL** field and click **Test Stream**.

 Congratulations! You have successfully previewed your live streaming output using several methods.

**Conclusion**

 Congratulations! You now have successfully:

* Configured an input into MediaLive
* Configured a channel, with outputs and encoding instructions, in MediaLive
* Configured inputs and endpoints in MediaPackage
* Viewed playout from MediaPackage and CloudFront

While not addressed in this lab, a key consideration of any live streaming video application is monitoring the health of the system to ensure that it is working properly–beyond previewing playback in MediaPackage or CloudFront. You can perform this type of monitoring using Amazon CloudWatch, which is addressed in a separate lab.

**End lab**

Follow these steps to close the console and end your lab.

1. Return to the **AWS Management Console**.
2. At the upper-right corner of the page, choose **AWSLabsUser**, and then choose **Sign out**.
3. Choose **End lab** and then confirm that you want to end your lab.

**Additional Resources**

For more information about AWS Training and Certification, see [*https://aws.amazon.com/training/*](https://aws.amazon.com/training/).

*Your feedback is welcome and appreciated.*  
If you would like to share any feedback, suggestions, or corrections, please provide the details in our [*AWS Training and Certification Contact Form*](https://support.aws.amazon.com/#/contacts/aws-training).