
Video on Demand on AWS

Implementation Guide



Video on Demand on AWS: Implementation Guide

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

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

Table of Contents

Home	1
Overview	2
Cost	2
Example cost for a 60-minute source video	2
Architecture	3
Components	5
Encoding options	5
Quality-defined variable bitrate mode	5
Accelerated transcoding	5
Frame capture	5
MediaPackage	5
Error handling	6
Ingest step functions	6
Video-only workflow	6
Metadata and video workflow	6
Processing step functions	7
Publishing step functions	7
Considerations	8
Customization	8
Solution updates	8
Regional deployments	8
Template	9
Deployment	10
Launch the stack	10
Update the stack	12
Security	13
Amazon S3 bucket policy	13
IAM roles	13
Resources	14
Appendix A: Ingest workflow	15
Appendix B: Processing workflow	16
Appendix C: Publishing workflow	17
Appendix D: Workflow configuration	18
Appendix E: Metadata file	19
Appendix F: MediaConvert templates	21
Appendix G: Uninstall the solution	22
Using the AWS Management Console	22
Using AWS Command Line Interface	22
Deleting the Amazon S3 Buckets	22
Deleting the Amazon DynamoDB tables	23
Deleting the CloudWatch Logs	23
Appendix H: Operational metrics	24
Source code	25
Contributors	26
Revisions	27
Notices	28

Video on Demand on AWS

AWS Implementation Guide

AWS Solutions Builder Team

April 2017 (last update (p. 27): December 2020)

This implementation guide discusses architectural considerations and configuration steps for deploying Video on Demand on Amazon Web Services (AWS). It includes links to an [AWS CloudFormation](#) template that launches and configures the AWS services required to deploy this solution using AWS best practices for security and availability.

The guide is intended for IT infrastructure architects, administrators, and DevOps professionals who have practical experience with video-on-demand workflows and architecting in the AWS Cloud.

Overview

Amazon Web Services (AWS) enables broadcasters and content owners to automate media supply chains, streamline content distribution, and deliver media content to a global audience. With the AWS Cloud, you can seamlessly scale your infrastructure to meet demand, and pay only for what you use, helping you build highly available and cost-effective solutions for video-on-demand workflows.

AWS offers its customers several methods to cost-effectively deliver video content to a global audience on the AWS Cloud. These methods enable you to dynamically scale any combination of video processing, delivery, and storage services. [AWS Elemental MediaConvert](#) transcodes media files from their source format into versions that will playback on devices like smartphones, tablets, and PCs; [AWS Elemental MediaPackage](#) creates video streams formatted to play on several devices from a single video input; and [Amazon CloudFront](#) accelerates delivery of your video content to end users. Customers can leverage these AWS services to build a highly available, resilient architectures that ingest, store, process, and deliver video content on demand.

Video on Demand on AWS is a reference implementation that automatically provisions the AWS services necessary to build a scalable, distributed video-on-demand workflow. The video-on-demand solution combines AWS Elemental MediaConvert, AWS Elemental MediaPackage (optional), and CloudFront with AWS Step Functions, a service that allows you to build applications from individual components that each perform a discrete function, AWS Lambda, a service that allows you to run code without provisioning or managing servers, and Amazon Simple Storage Service (Amazon S3), a highly-scalable, reliable, and low-latency object store. The solution also leverages Amazon DynamoDB to track source and destination file metadata and progress through the workflow, Amazon CloudWatch to track encoding jobs, Amazon Simple Queue Service (Amazon SQS) to capture the workflow output, and Amazon Simple Notification Service (Amazon SNS) to send publishing, encoding, and error notifications.

The Video on Demand on AWS solution ingests metadata files and source videos, processes the videos for playback on a wide range of devices, stores the transcoded media files, and delivers the videos to end users through CloudFront. This reference implementation provides an example architecture to build a global consumer video workflow on AWS. You can customize the architecture to meet your specific needs. For more information, refer to [Customization \(p. 8\)](#).

Cost

You are responsible for the cost of the AWS services used while running this solution. The total cost for running this solution depends on the size of your videos, the number of outputs created, and the number of views the published content receives through CloudFront. MediaConvert composes the majority of the cost. For more information about MediaConvert pricing, refer to [MediaConvert Pricing](#).

Example cost for a 60-minute source video

At the time of publication, the estimated cost for using this solution with the default encoding settings to process a 60-minute source video in the US East (N. Virginia) Region is approximately **\$11.82**. This estimate may vary depending on the source video size and format.

The video outputs in this example use the following MediaConvert settings: AVC codec, 1 pass quality, 30 fps, with the DeInterlacer enabled.

The following table shows the professional tier costs for the MediaConvert settings used in this example:

Output	Cost
SD resolution	\$0.012/min
HD resolution	\$0.024/min
Audio only	\$0.005/min

Outputs with calculations for each option:

1. File group: 1 HD

$$(1 \times 60 \times \$0.024) = \$1.44$$

2. HLS profile: 2 HD and 3 SD

$$(2 \times 60 \times \$0.024) + (3 \times 60 \times \$0.012) = \$5.04$$

3. DASH profile: 2 HD and 3 SD

$$(2 \times 60 \times \$0.024) + (3 \times 60 \times \$0.012) = \$5.04$$

4. 1 audio only output for the DASH profile

$$(1 \times 60 \times \$0.005) = \$0.30$$

$$\text{Total cost} = \$1.44 + \$5.04 + \$5.04 + \$0.30 = \$11.82$$

The costs for CloudFront and Amazon S3 storage varies depending on the number and format of outputs created and the number of requests to view the content delivered through CloudFront. Prices are subject to change. For full details, refer to the pricing webpage for each AWS service used in this solution.

Architecture overview

Deploying this solution builds the following environment in the AWS Cloud.

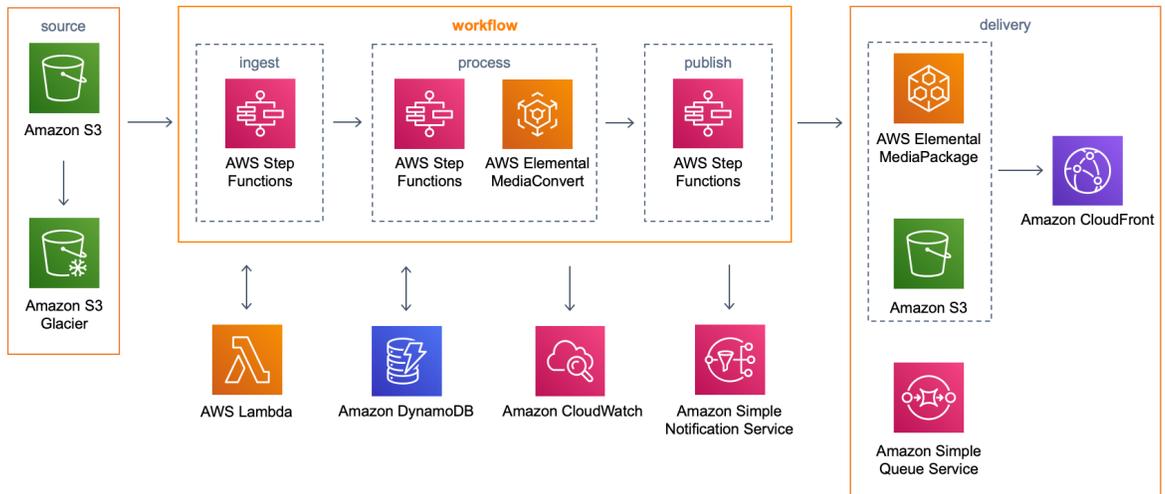


Figure 1: Video on Demand on AWS architecture

The AWS CloudFormation template deploys a workflow that ingests source videos, or source videos and metadata files. When you upload a source video only, encoding options are defined in the AWS

CloudFormation template at launch, and are applied to every video the solution encodes. When you upload a source video and metadata file, encoding parameters for each source video are defined in the metadata file, allowing customers to apply encoding options on a video-by-video basis.

The template deploys a workflow that includes AWS Elemental MediaConvert and AWS Step Functions, which creates ingest, processing, and publishing step functions. The template also launches AWS Lambda functions that perform the work of each step, and process error messages, Amazon S3 buckets for source and destination media files, Amazon CloudWatch for logging, Amazon CloudWatch Events rules for AWS Elemental MediaConvert notifications and an Amazon CloudFront distribution. An Amazon DynamoDB table stores data captured through the workflow, an Amazon SQS queue to capture the workflow outputs, and Amazon SNS topics send encoding, publishing, and error notifications.

Solution components

Encoding options

The Video on Demand on AWS solution leverages AWS Elemental MediaConvert job templates to define the solution's encoding options. By default, the solution creates three custom templates that produce UHD, HD, and SD output. Each template creates H.265 MP4; HTTP Live Streaming (HLS); and Dynamic Adaptive Streaming over HTTP (DASH). Additionally, the solution supports Microsoft Smooth Streaming (MSS) and Common Media Application Format (CMAF) outputs.

The workflow selects one of the three templates, based on the resolution of the source video. You can also customize the solution to work with any valid MediaConvert template. For more information, refer to [Appendix F \(p. 21\)](#).

Quality-defined variable bitrate mode

This solution leverages AWS Elemental MediaConvert Quality-Defined Variable Bitrate (QVBR) encoding mode which ensures consistent, high-quality video transcoding with the smallest file size for any type of source video content. With QVBR, the encoder determines the right number of bits to use for each part of the video to maintain the video quality that you specify. If enabled, the solution configures the encoding templates to enable QVBR mode with the recommended settings for each output.

We recommend you enable QVBR with this solution. For more information, refer to [Appendix F \(p. 21\)](#).

Accelerated transcoding

This solution includes an option to enable Accelerated Transcoding in AWS Elemental MediaConvert, which increases the processing speed of file-based video encoding jobs by up to 25 times. For more information, refer to the [MediaConvert documentation](#).

Important

There are 2 options to enable acceleration. The **Preferred** option enables acceleration, but will fallback to standard encoding if the source file isn't supported. The **Enabled** option applies acceleration to every encoding job, and the job will fail if the source file is not supported.

Frame capture

This solution can also create a set of thumbnails from your source videos. If this feature is enabled, the solution will create a set of thumbnails for each selected output. The thumbnails are stored in the Amazon S3 bucket with your video output.

MediaPackage

This solution includes the option to use AWS Elemental MediaPackage as part of the workflow. When enabled, the solution creates a separate set of MediaConvert custom templates that include H.265 MP4 and HLS. The solution also creates a packaging group in MediaPackage that is configured to ingest the MediaConvert HLS output stored in Amazon S3. MediaPackage packages the content, formatting it in

response to playback requests from downstream devices. By default, this solution creates packaging configurations for HLS, DASH, MSS, and CMAF.

Important

Customers who ingest large quantities of files may exceed MediaPackage limits for video-on-demand content. For more information and instructions on how to request a limit increase, refer to [VOD Content Limits](#) in the *AWS Elemental MediaPackage User Guide*.

Error handling

The ingest, processing, and publishing workflow AWS Lambda functions, and Amazon CloudWatch Events are configured to invoke an *error handler* Lambda function that updates the Amazon DynamoDB table with error message details, and sends an Amazon Simple Notification Service (Amazon SNS) notification to a subscribed email address.

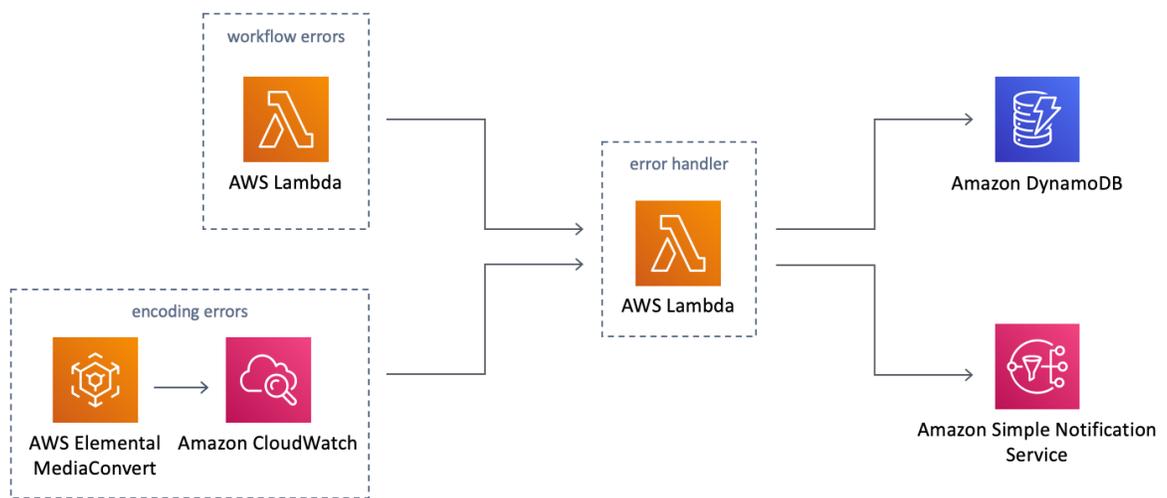


Figure 2: Video on Demand error handling

Ingest step functions

Video-only workflow

When a new MP4, MPG, M4V, M2TS, or MOV video is added to the source Amazon S3 bucket, a Lambda function triggers the ingest workflow. During ingestion, source video details are added to Amazon DynamoDB, the content is validated using [MedialInfo](#), open-source software that displays technical information about media files, and details are stored in DynamoDB.

Important

Source video file extensions (.mp4, .mpg, .m4v, .m2ts or .mov) must be lowercase and file names cannot contain spaces.

Metadata and video workflow

When a new metadata file is added to the source Amazon S3 bucket, a Lambda function triggers the ingest workflow. During ingestion, the metadata file, source video, and encoding configuration details are added to Amazon DynamoDB, the source video is validated using [MedialInfo](#) and details are stored in DynamoDB.

Important

You must upload the source video file to the Amazon S3 bucket before you upload the metadata file. Note that the upload must complete before you upload the metadata file.

Processing step functions

The solution uses the height and width of the source video to determine which job template to use to submit encoding jobs to AWS Elemental MediaConvert. If you enable frame capture, the frame capture parameters are added to the job template. Then, the encoding job is created in MediaConvert and the details are stored in DynamoDB.

Publishing step functions

After the video is encoded, AWS Elemental MediaConvert sends a notification to Amazon CloudWatch. An Amazon CloudWatch Events rule invokes the publishing AWS Step Functions step function, which validates the outputs, and updates the DynamoDB table with the new content details.

When the workflow is finished, Amazon SNS and/or Amazon SQS sends a publish notification based on the configuration you choose. If you choose to archive your source content, the source files are tagged to allow the [Amazon S3 lifecycle policy](#) to move files to Amazon S3 Glacier or Amazon Deep Archive.

Design considerations

Customization

This solution leverages AWS Step Functions, which breaks the workflow into individual steps, making it easier to customize or extend the architecture for your specific video-on-demand needs. For example, you can modify or replace the encoding steps to produce different content sets. You can also add steps to extend support for more complex workflows, including image processing for poster artwork or additional custom data to the metadata file that will then be stored in Amazon DynamoDB. The solution originates AWS Elemental MediaConvert output content directly from Amazon Simple Storage Service (Amazon S3) through Amazon CloudFront. You can, however, customize the solution to leverage a dedicated origin server such as AWS Elemental MediaPackage.

Each time the workflow is initiated, the solution creates a unique identifier. The unique identifier is used as the primary key in Amazon DynamoDB and the execution ID in AWS Step Functions. The unique identifier is passed to each step in the workflow, allowing information to be stored and retrieved in DynamoDB. This makes it easier to add and remove steps from the workflow.

Solution updates

To continue using this solution with the latest features and improvements, you must deploy the latest version of this stack. For information about updating your stack, refer to [Update the stack \(p. 12\)](#).

Installing version 5.2.0 creates three new MediaConvert job templates, with fewer HLS and DASH renditions, and without the use of presets. For more information, refer to [Appendix F \(p. 21\)](#). Updating an existing solution deployment to version 5.2.0 creates these new templates without deleting the presets or templates created by the older versions. To use an older template with the latest version of the solution, specify the template using the **JobTemplate** field in your metadata file. For more information, refer to [Appendix E \(p. 19\)](#). Or, you can replace the default templates in the `InputValidate` AWS Lambda function by modifying the `MediaConvert_Template_<resolution>` environment variables.

Regional deployments

This solution uses AWS Elemental MediaConvert and AWS Elemental MediaPackage, which are available in specific AWS Regions only. Therefore, you must deploy this solution in a region that supports these services. For the most current service availability by region, refer to the [AWS Regional Services List](#).

AWS CloudFormation template

This solution uses AWS CloudFormation to automate the deployment of the Video on Demand on AWS solution in the AWS Cloud. It includes the following AWS CloudFormation template, which you can download before deployment:

[View
Template](#)

video-on-demand-on-aws.template: Use this template to launch the Video on Demand on AWS solution and all associated components. The default configuration deploys AWS Lambda functions, Amazon Simple Storage Service (Amazon S3) buckets, AWS Step Functions, AWS Elemental MediaConvert, AWS Elemental MediaPackage (optional), an Amazon DynamoDB table, Amazon CloudWatch Logs, Amazon CloudWatch Events rules, Amazon Simple Notification Service (Amazon SNS) topics, and an Amazon CloudFront distribution. You can also customize the template based on your specific needs.

Automated deployment

Before you launch the automated deployment, review the architecture and other considerations discussed in this guide. Follow the step-by-step instructions in this section to configure and deploy the Video on Demand on AWS solution into your account.

To update your existing stack to the latest version, refer to [Update the stack \(p. 12\)](#). For additional options, including using an older version of this solution, refer to [Solution updates \(p. 8\)](#).

Time to deploy: Approximately 20 minutes

Launch the stack

Use this AWS CloudFormation template to deploy the latest version of this solution.

Note

You are responsible for the cost of the AWS services used while running this solution. Refer to the [Cost \(p. 2\)](#) section for more details. For full details, refer to the pricing webpage for each AWS service you will be using in this solution.

1. Sign in to the AWS Management Console and click the button below to launch the `video-on-demand-on-aws` AWS CloudFormation template.



You can also [download the template](#) as a starting point for your own implementation.

2. The template is launched in the US East (N. Virginia) Region by default. To launch this solution in a different AWS Region, use the region selector in the console navigation bar.

Note

This solution uses AWS Elemental MediaConvert and AWS Elemental MediaPackage, which are available in specific AWS Regions only. Therefore, you must deploy this solution in a Region that supports these services. For the most current service availability by Region, refer to the [AWS Regional Services List](#).

3. On the **Select Template** page, verify that you selected the correct template and choose **Next**.
4. Under **Parameters**, review the parameters for the template, and modify them as necessary.

This solution uses the following default values.

Parameter	Default	Description
Notification Email Address	<i><Requires input></i>	A valid email address to receive Amazon SNS notifications.
Workflow Trigger	<i><Requires input></i>	Choose <code>VideoFile</code> to ingest source videos only; choose

Parameter	Default	Description
		MetadataFile to ingest metadata files and source videos.
Archive Source Content	false	Choose GLACIER to enable an Amazon S3 lifecycle policy on the source bucket to move applicable files to Amazon S3 Glacier after seven days, DEEP_ARCHIVE to move to Glacier Deep Archive.
Enable SNS	true	Choose true receive SNS notifications for the ingest and publish workflows. Choose false to only receive error messages.
Enable SQS	true	Choose true to deploy an SQS queue for publishing messages.
Enable Frame Capture	false	Choose true to create thumbnails for each AWS Elemental MediaConvert output.
Accelerated Transcoding	PREFERRED	Choose PREFERRED to enable Accelerated Transcoding for supported file types (recommended), ENABLE to apply to all encoding jobs.
Enable MediaPackage	false	Choose true to enable AWS Elemental MediaPackage as part of the workflow.

5. Choose **Next**.
6. On the **Options** page, choose **Next**.
7. On the **Review** page, review and confirm the settings. Be sure to check the box acknowledging that the template will create AWS Identity and Access Management (IAM) resources.
8. Choose **Create** to deploy the stack.

You can view the status of the stack in the AWS CloudFormation console in the **Status** column. You should receive a **CREATE_COMPLETE** status in approximately 20 minutes.

After the stack is created, Amazon SNS sends three subscription notifications to the admin email address with links to enable encoding, publishing, and error notification.

9. In the subscription notification emails, select each link to enable SNS notifications.

Note

In addition to the AWS Lambda functions that create solution resources and trigger the ingest and publishing processes, this solution includes the `custom-resource` Lambda function, which runs only during initial configuration or when resources are updated or deleted.

When running this solution, the `custom-resource` Lambda function is inactive. However, do not delete the function as it is necessary to manage associated resources.

Update the stack

Complete the following steps to update your AWS CloudFormation stack to the current version.

1. From the main account where the Video on Demand on AWS template is deployed, sign in to the [AWS CloudFormation console](#).
2. From the **Stacks** page, select this solution's stack and choose **Update**.
3. On the **Update stack** page, verify that **Replace current template** is selected.
 - a. In the **Specify template** section, select **Amazon S3 URL**.
 - b. Copy the link of the [latest template](#).
 - c. Paste the link in the **Amazon S3 URL** box.
 - d. Verify that the correct template URL shows in the **Amazon S3 URL** text box and choose **Next**.
4. On the **Specify stack details** page, under **Parameters**, review the parameters for the template and modify them as necessary. Refer to [Launch the stack \(p. 10\)](#) for details about the parameters.
5. Choose **Next**.
6. On the **Configure stack options** page, choose **Next**.
7. On the **Review** page, review and confirm the settings. Check the box acknowledging that the template will create AWS Identity and Access Management (IAM) resources.
8. Choose **View change set** and verify the changes.
9. Choose **Update stack** to deploy the stack.

You can view the status of the stack in the AWS CloudFormation console in the **Status** column. You should receive an **UPDATE_COMPLETE** status in approximately 20 minutes.

Security

When you build systems on AWS infrastructure, security responsibilities are shared between you and AWS. This shared model can reduce your operational burden as AWS operates, manages, and controls the components from the host operating system and virtualization layer down to the physical security of the facilities in which the services operate. For more information about security on AWS, visit the [AWS Security Center](#).

Amazon S3 bucket policy

The Amazon Simple Storage Service (Amazon S3) buckets for AWS Elemental MediaConvert output includes a policy that allows access from Amazon CloudFront. Because the Amazon CloudFront endpoints are publicly accessible, the MediaConvert output bucket is also publicly accessible. For information on how to secure Amazon CloudFront, refer to [Serving Private Content through CloudFront](#) in the Amazon CloudFront Developer Guide.

IAM roles

AWS Identity and Access Management (IAM) roles enable customers to assign granular access policies and permissions to services and users on the AWS Cloud. Video on Demand on AWS creates several IAM roles, including a role that grants AWS Elemental MediaConvert access to Amazon API Gateway and Amazon Simple Storage Service. This role is necessary to allow the services to operate in your account.

Additional resources

AWS services

- [AWS Step Functions](#)
- [AWS Elemental MediaConvert](#)
- [Amazon CloudFront](#)
- [Amazon CloudWatch](#)
- [Amazon SNS](#)
- [AWS CloudFormation](#)
- [Amazon S3](#)
- [AWS Elemental MediaPackage](#)
- [AWS Lambda](#)
- [Amazon SQS](#)

Other resources

- [MedialInfo](#)

Appendix A: Ingest workflow

When a new video is added to the source Amazon Simple Storage Service (Amazon S3) bucket, an AWS Lambda function starts the ingest step function. The ingest step function includes:

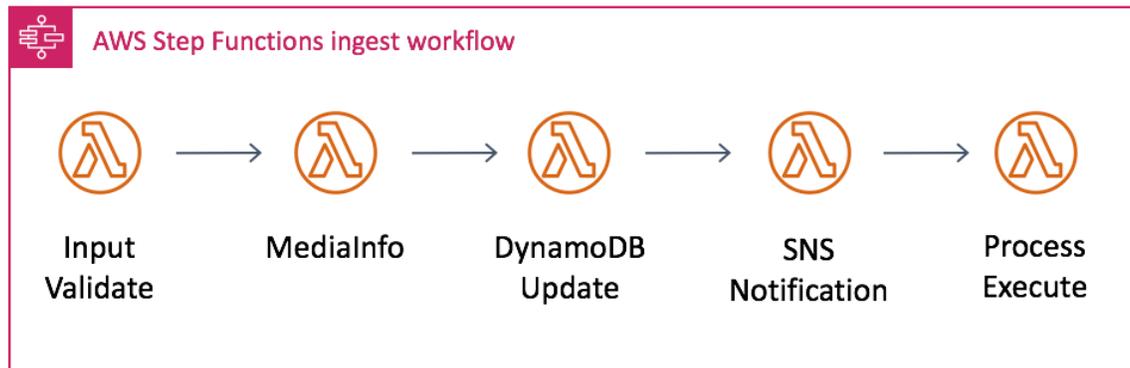


Figure 3: Ingest workflow

- **Input Validate** - Parses the input to the workflow, checks for the source video file, and defines the workflow configuration using the AWS Lambda function environment variables. If enabled, this step downloads the metadata file and overwrites the default environment variables with the variable definitions in the metadata file (metadata and video version only). For more information, refer to [Appendix E \(p. 19\)](#).
- **MediaInfo** - Generates a signed Amazon S3 URL for the source video and runs MediaInfo to extract metadata about the video.
- **DynamoDB Update** - Takes accumulated data from each step and stores it in Amazon DynamoDB.
- **SNS Notification** - Sends an Amazon Simple Notification Service (Amazon SNS) notification with a summary of the ingest process.
- **Process Execute** - Starts the processing workflow.

Appendix B: Processing workflow

When the ingest workflow is complete, it starts the processing workflow. The processing workflow includes:

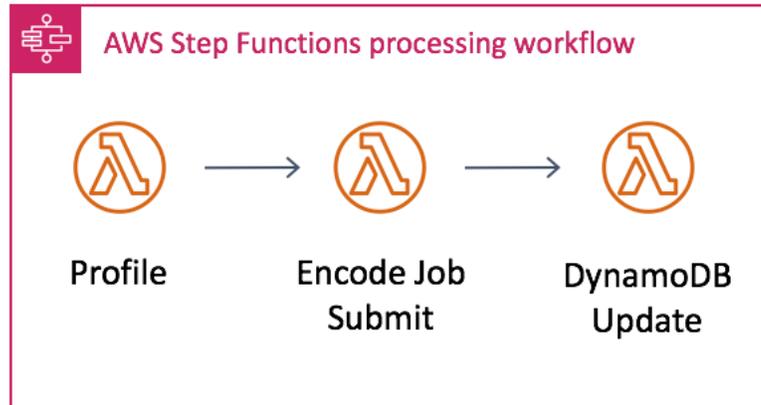


Figure 4: Processing workflow

- **Profiler** – Gets the source video's height and width from the metadata file, defines the settings for frame capture (if enabled), and chooses which template to use for encoding based on the source video's height. For example, if the source video is greater than or equal to 1080p, the 1080p job template will be used.
- **Encoding Profile Check, Accelerated Transcoding Check, and Frame Capture Check** – Helps visualize which settings the profiler step applied.
- **Encode Job Submit** – Submits the encoding job with the template defined by the profiler to AWS Elemental MediaConvert.
- **DynamoDB Update** – Takes accumulated data from each step and stores it in Amazon DynamoDB.

Appendix C: Publishing workflow

When encoding is complete, Amazon Simple Notification Service (Amazon SNS) sends a notification that triggers an AWS Lambda function that starts the publishing process. The publishing process includes:

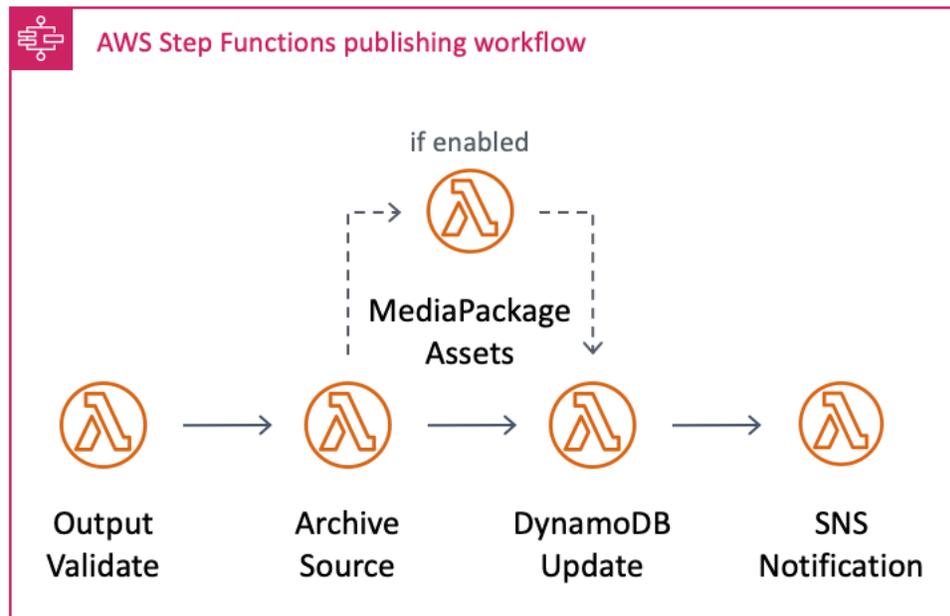


Figure 5: Publishing workflow

- **Output Validate** - Checks the event data for the completed encoding job, gets the GUID from the AWS Elemental MediaConvert notification, gets the asset details from Amazon DynamoDB, and generates the Amazon Simple Storage Service (Amazon S3) and Amazon CloudFront URLs for the MediaConvert outputs.
- **Archive Choice** - If Glacier or Glacier Deep Archive was enabled, this step tags the source video with a unique identifier and the archive to trigger the Amazon S3 Glacier lifecycle policy.
- **MediaPackage Choice** - If you configure the solution to use AWS Elemental MediaPackage, this step takes the output from MediaConvert and uses it as a source for a MediaPackage asset, which contains all the information MediaPackage requires to ingest file-based video content.
- **DynamoDB Update** - Updates Amazon DynamoDB table with the event data.
- **SQS Choice** – If enabled, this step sends all workflow outputs to an SQS queue that is ingested into upstream workflows or processes.
- **SNS Choice** – If enabled, this step sends an Amazon SNS notification with a summary of the workflow and the Amazon CloudFront URLs.

Appendix D: Workflow configuration

The `Input Validate` AWS Lambda function contains the following environment variables that define the workflow configuration.

Environment Variable	Description
Archive Source	Choose whether to archive source videos in Amazon Glacier or Glacier Deep Archive.
CloudFront	The Amazon CloudFront domain name. This is used to generate the playback URLs for the AWS Elemental MediaConvert outputs.
Destination	The name of the destination Amazon S3 bucket for all MediaConvert outputs.
FrameCapture	Choose whether to create thumbnails for each MediaConvert output.
MediaConvert_Template_2160p	The name of the UHD template for MediaConvert.
MediaConvert_Template_1080p	The name of the HD template for MediaConvert.
MediaConvert_Template_720p	The name of the SD template for MediaConvert.
Source	The name of the source Amazon S3 bucket.
WorkflowName	Used to tag MediaConvert encoding jobs. This is defined by the AWS CloudFormation stack name.
InputRotate	Specify how MediaConvert should rotate the source video.
AcceleratedTranscoding	The option to enable Accelerated Transcoding in MediaConvert.
EnableSQS	The option to enable SQS.
EnableSNS	The option to enable SNS.

These variables are set when you deploy the AWS CloudFormation template and apply to all source videos uploaded to the solution's Amazon Simple Storage Service (Amazon S3) bucket.

If you set the solution to ingest source videos and metadata files, you can overwrite these files using a metadata file. For more information, refer to [Appendix F \(p. 21\)](#).

Appendix E: Metadata file

When you set the solution to ingest source videos and metadata files, the source Amazon Simple Storage Service (Amazon S3) bucket is configured with an event notification that triggers the workflow when you upload a JSON file.

Note

The JSON file does not need a specific name, but it must have the JSON file extension. We recommend naming the JSON file the same name as the video file for consistency and ease of reference.

To trigger the workflow, you must upload a JSON metadata file. If you only upload a source video file, the workflow will not start.

Important

You must upload the source video file to the Amazon S3 bucket before you upload the metadata file. Note that the upload must complete before you upload the metadata file.

The definitions in the metadata file overwrite the default settings you specified when you deployed the solution. This enables you to define different workflow configurations for each source video. If you do not specify a definition in the metadata file, the solution will use the default value you set during deployment. Note that the metadata file must include a definition for **srcVideo**.

The following example metadata files shows all available variable definitions.

```
{
  "srcVideo": "string",
  "archiveSource": "DISABLED|GLACIER|DEEP_ARCHIVE",
  "frameCapture": boolean,
  "srcBucket": "string",
  "destBucket": "string",
  "cloudFront": "string",
  "jobTemplate_2160p": "string",
  "jobTemplate_1080p": "string",
  "jobTemplate_720p": "string",
  "acceleratedTranscoding": "DISABLED|PREFERED|ENABLED",
  "enableSqs": boolean,
  "enableSns": boolean,
  "jobTemplate": "custom-job-template",
  "InputRotate": "DEGREE_0|DEGREES_90|DEGREES_180|DEGREES_270|AUTO"
}
```

Figure 6. List of metadata file definitions

The following sample JSON metadata file will overwrite the default settings for the **Archive Source Content** and **Enable Frame Capture** AWS CloudFormation template parameters for the `example.mpg` file. The file will also set the job template for AWS Elemental MediaConvert to `custom-job-template`.

```
{
  "srcVideo": "example.mpg",
  "archiveSource": "GLACIER",
  "frameCapture": false,
  "jobTemplate": "custom-job-template"
}
```

Figure 7. Sample JSON metadata file

The Video on Demand on AWS solution also supports adding additional metadata, such as title, genre, or any other information, you want to store in Amazon DynamoDB.

Appendix F: AWS Elemental MediaConvert templates

By default, the Video on Demand on AWS solution selects the job template for AWS Elemental MediaConvert based on the source video height. The solution includes three default job templates:

- MediaConvert_Template_2160p:
 - 1 mp4 2160p HEVC output
 - 5 HLS outputs AVC 1080p through 270p
 - 5 DASH outputs AVC 1080p through 270p
- MediaConvert_Template_1080p:
 - 1 mp4 1080p AVC output
 - 5 HLS outputs AVC 1080p through 270p
 - 5 DASH outputs AVC 1080p through 270p
- MediaConvert_Template_720p:
 - 1 mp4 720p AVC output
 - 4 HLS outputs AVC 720p through 270p
 - 4 DASH outputs AVC 720p through 270p

By default, the solution is configured to leverage Quality-Defined Variable Bitrate (QVBR) mode in MediaConvert. The QVBR settings are configured to the recommended values for each output, as shown in the following table.

Resolution	Maximum Bitrate	QVBR Quality Level
2160p	15,000 Kbps	9
1080p	8,500 Kbps	8
720p	6,000 Kbps	8
540p	3,500 Kbps	7
360p	1,500 Kbps	7
270p	400 Kbps	7

You can also modify the solution to use different QVBR settings, other system job templates, or your own custom job templates. For more information about working with job templates for MediaConvert, refer to [Working with MediaConvert Job Templates](#). For more information about QVBR Mode, refer to [Using the QVBR Rate Control Mode](#).

If you set the solution to ingest source videos and metadata files, you can specify the template using the **JobTemplate** field in your metadata file. For more information, refer to [Appendix E \(p. 19\)](#). Or, you can replace the default templates in the `Input Validate` AWS Lambda function by modifying the `MediaConvert_Template_<resolution>` environment variables.

Appendix G: Uninstall the solution

You can uninstall the Video on Demand on AWS solution from the AWS Management Console, or by using the AWS Command Line Interface (AWS CLI). You must manually delete the Amazon Simple Storage Service (Amazon S3) buckets, an Amazon DynamoDB table, and CloudWatch Logs created by this solution. AWS Solutions Implementations do not automatically delete these resources in case you have stored data that you wish to retain.

Note

AWS CloudFormation StackSets are automatically deleted when you uninstall the solution's stack.

Using the AWS Management Console

1. Sign in to the [AWS CloudFormation console](#).
2. On the **Stacks** page, select the solution stack.
3. Choose **Delete**.

Using AWS Command Line Interface

Verify that the AWS Command Line Interface (AWS CLI) is available in your environment. For installation instructions, refer to [What Is the AWS Command Line Interface](#) in the *AWS CLI User Guide*. After confirming the AWS CLI is available, run the following command.

```
$ aws cloudformation delete-stack --stack-name <your-stack-name>
```

Replace *<cloudformation-stack-name>* with the name of your CloudFormation stack.

Deleting the Amazon S3 buckets

This solution is configured to retain the Amazon S3 buckets if you decide to delete the AWS CloudFormation stack to prevent against accidental data loss. After uninstalling the solution, you can manually delete the S3 buckets if you do not need to retain the data. Follow these steps to delete the Amazon S3 buckets.

1. Sign in to the [Amazon S3 console](#).
2. Choose **Buckets** from the left navigation pane.
3. Locate the *<stack-name>* S3 buckets.
4. Select one of the S3 buckets and choose **Delete**.

Repeat the steps until you have deleted all the *<stack-name>* S3 buckets.

To delete the S3 buckets using AWS CLI, run the following command:

```
$ aws s3 rb s3://<bucket-name> --force
```

Alternatively, you can configure the AWS CloudFormation template to delete the Amazon S3 buckets automatically. Prior to deleting the stack, change the deletion behavior in the AWS CloudFormation [DeletionPolicy](#) attribute.

Deleting the Amazon DynamoDB tables

This solution is configured to retain the DynamoDB tables if you decide to delete the AWS CloudFormation stack to prevent accidental data loss. After uninstalling the solution, you can manually delete the DynamoDB tables if you do not need to retain the data. Follow these steps:

1. Sign in to the [Amazon DynamoDB console](#).
2. Choose **Tables** from the left navigation pane.
3. Select the `<stack-name>` table and choose **Delete**.

To delete the DynamoDB tables using AWS CLI, run the following command:

```
$ aws dynamodb delete-table <table-name>
```

Deleting the CloudWatch Logs

This solution retains the CloudWatch Logs if you decide to delete the AWS CloudFormation stack to prevent against accidental data loss. After uninstalling the solution, you can manually delete the logs if you do not need to retain the data. Follow these steps to delete the CloudWatch Logs.

1. Sign in to the [Amazon CloudWatch console](#).
2. Choose **Log Groups** from the left navigation pane.
3. Locate the log groups created by the solution.
4. Select one of the log groups.
5. Choose **Actions** and then choose **Delete**.

Repeat the steps until you have deleted all the solution log groups.

Appendix H: Collection of operational metrics

This solution includes an option to send anonymous operational metrics to AWS. We use this data to better understand how customers use this solution and related services and products. When enabled, the following information is collected and sent to AWS each time a video is processed:

- **Solution ID:** The AWS solution identifier
- **Unique ID (UUID):** Randomly generated, unique identifier for each live streaming solution deployment
- **Timestamp:** Data-collection timestamp
- **Use Glacier:** Whether Amazon Glacier is used
- **Workflow Trigger:** The workflow trigger selected
- **Frame Capture:** Whether thumbnails are created for AWS Elemental MediaConvert output
- **Enable MediaPackage:** Whether AWS Elemental MediaPackage is enabled

Note that AWS will own the data gathered via this survey. Data collection will be subject to the [AWS Privacy Policy](#). To opt out of this feature, modify the AWS CloudFormation template mapping section as follows:

```
AnonymousData:  
  SendAnonymousData:  
    Data: Yes
```

to

```
AnonymousData:  
  SendAnonymousData:  
    Data: No
```

Source code

Visit our [GitHub repository](#) to download the templates and scripts for this solution, and to share your customizations with others.

Contributors

The following individuals contributed to this document:

- Tom Nightingale
- Joan Morgan

Revisions

Date	Change
April 2017	Initial release
June 2017	Added information on new watermark functionality and processing workflow; updated information on the ingest and publishing workflows
October 2017	Added information on new metadata and video and video-only AWS CloudFormation templates and functionality; updated information on the ingest process and error handling
March 2018	Added AWS MediaConvert functionality
November 2018	Added information on new AWS Elemental MediaConvert job templates and functionality, and environment variables; removed Amazon Elastic Transcoder functionality
April 2019	Added information on support for AWS Elemental MediaConvert Quality-Defined Variable Bitrate (QVBR) Mode
November 2019	Added information on support for AWS Elemental MediaPackage, and Node.js and MedialInfo updates
April 2020	Added information about Accelerated Transcoding and notification options
December 2020	Added information about the new AWS Elemental MediaConvert templates, installation and upgrade behavior using latest solution version; for more information about changes for v5.2.0, refer to the CHANGELOG.md file in the Github repository

Notices

Customers are responsible for making their own independent assessment of the information in this document. This document: (a) is for informational purposes only, (b) represents AWS current product offerings and practices, which are subject to change without notice, and (c) does not create any commitments or assurances from AWS and its affiliates, suppliers or licensors. AWS products or services are provided “as is” without warranties, representations, or conditions of any kind, whether express or implied. AWS responsibilities and liabilities to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.

Video on Demand on AWS is licensed under the terms of the of the Apache License Version 2.0 available at [The Apache Software Foundation](#).