Media Sharing Website

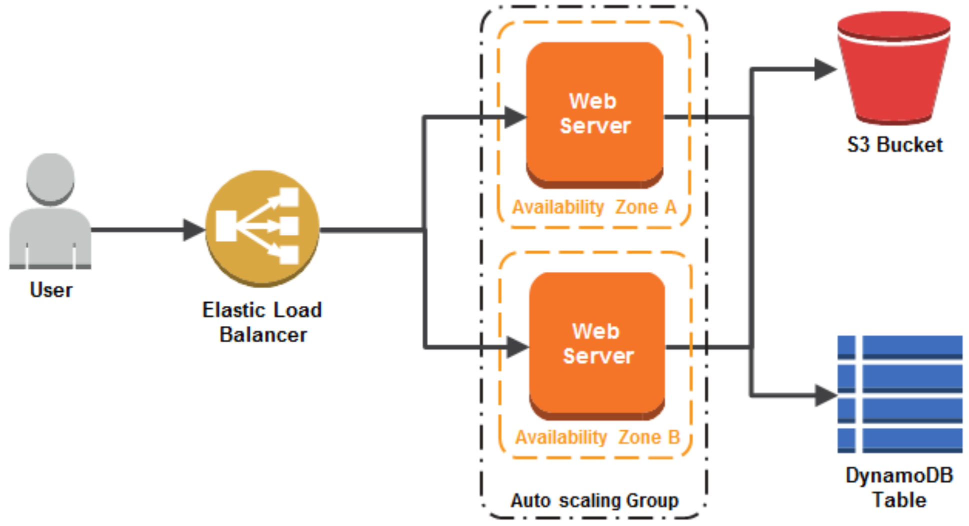
Part 2: Video Transcoding

Table of Contents

Overview

Note: It is not required to complete Lab 11 (part 1) prior to this lab, but the full value of the labs will be realized if done in that order.

In part 1 of this lab, you created simple yet fault-tolerant and scalable system to publish images online. The following diagram represents the architecture of the system we built so far:



In this second part, you will extend the system to support videos. Video files present different challenges compared to images. The architecture you deployed in the first part is able to receive video file uploads but only offers the ability to download them back. Here, we would like to offer video streaming features for the users to view the videos online without having to download the complete video file.

However, in order to offer video streaming capability, we must be able to transform the various video files we receive from the users into a common “streamable” format. This operation, known as video transcoding, is very CPU intensive and can take a significant amount of time (depending on the video file size) compared to the latency users experience in typical web applications. In order to keep our web application responsive, we are going to implement an asynchronous video transcoding system.

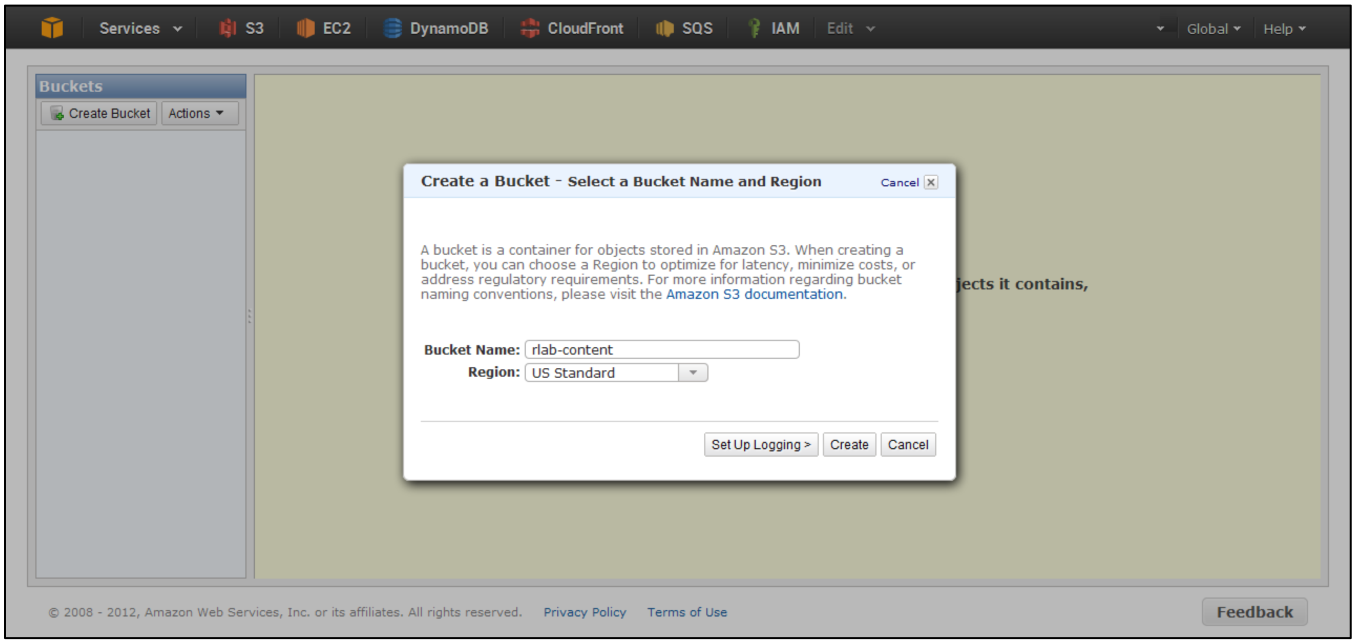
Step 1 – Media storage

Note: this step will walk you through the creation of an Amazon S3 bucket used in Part 1. Details have been removed.

Creating an Amazon S3 Bucket

Go to the S3 console (via Services / S3), click "**Create Bucket**" and chose a bucket name.

Important: Don't forget to write down the name of your bucket as it will be required to complete future steps.



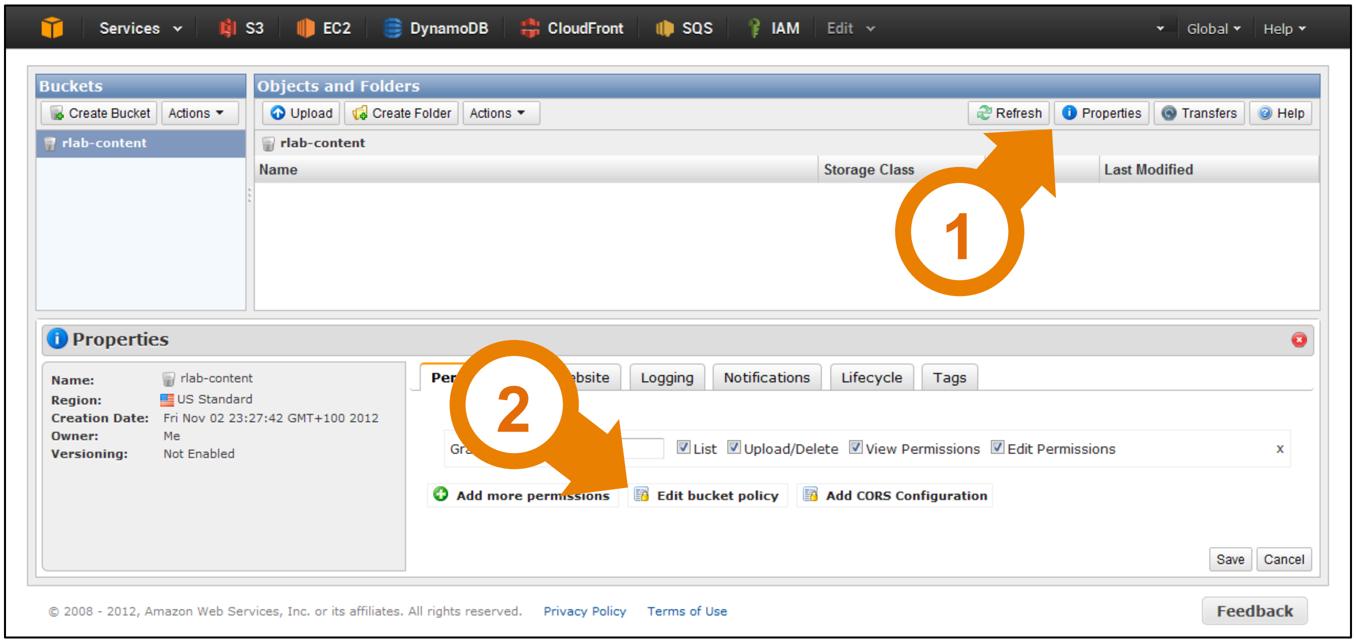
A bucket is a container for objects stored in Amazon S3. When creating a bucket, you can choose a Region to optimize for latency, minimize costs, or address regulatory requirements. For more information regarding bucket naming conventions, please visit the Amazon S3 documentation.

Bucket Name: rlab-content
Region: US Standard

© 2008 - 2012, Amazon Web Services, Inc. or its affiliates. All rights reserved.

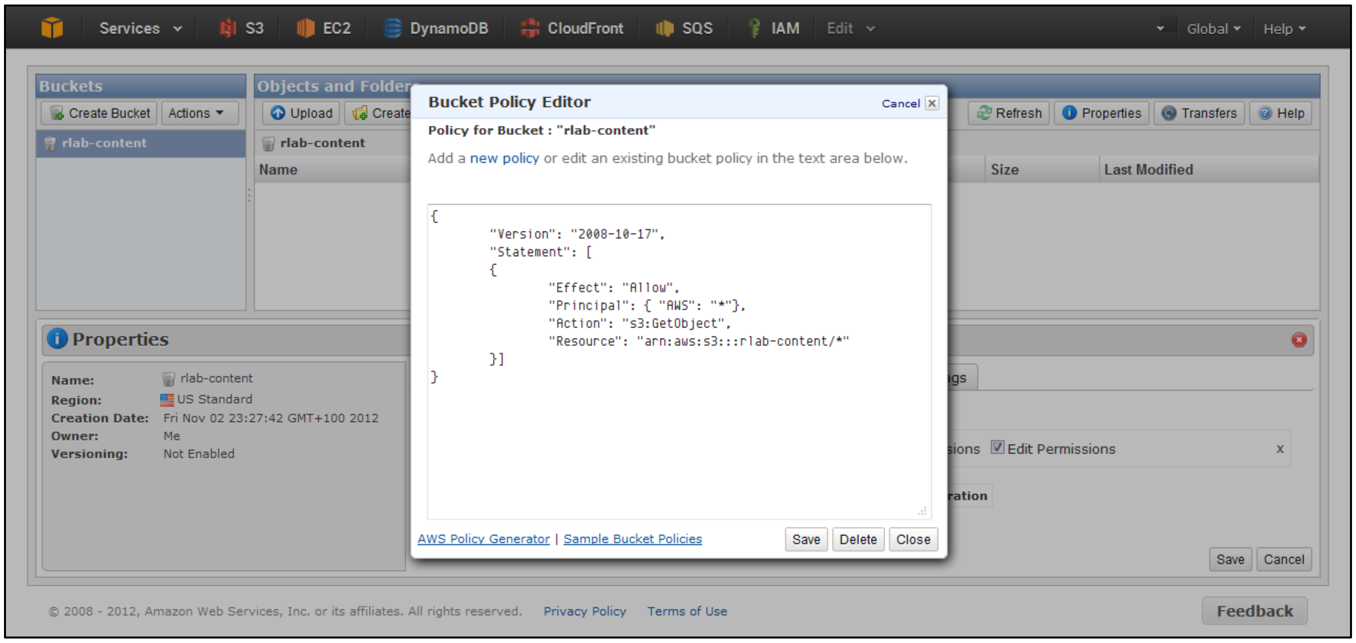
Assign a Bucket Policy

To add a policy to your bucket, go to the Amazon S3 section of the AWS web console. Select your newly created bucket, click "**Properties**" (1) and then click the "**Edit bucket policy**" (2) button from the permissions tab:



In the "**Bucket Policy Editor**" dialog box, add the following policy to your bucket, and change the highlighted text with your bucket name
(the policy file can be downloaded here: http://rlab-content.s3.amazonaws.com/lab12-bucket_policy.json) :

```
{
        "Version": "2008-10-17",
        "Statement": [
                {
                        "Sid": "AddPerm",
                        "Effect": "Allow",
                        "Principal": { "AWS": "*" },
                        "Action": "s3:GetObject",
                        "Resource": "arn:aws:s3:::YOUR_BUCKET_NAME/*"
                }
        ]
}
```



Bucket Policy Editor

Policy for Bucket : "rlab-content"

Add a new policy or edit an existing bucket policy in the text area below.

```
{
        "Version": "2008-10-17",
        "Statement": [
        {
                "Effect": "Allow",
                "Principal": { "AWS": "*"},
                "Action": "s3:GetObject",
                "Resource": "arn:aws:s3:::rlab-content/*"
        }]
}
```

Properties

Name:	rlab-content
Region:	US Standard
Creation Date:	Fri Nov 02 23:27:42 GMT+100 2012
Owner:	Me
Versioning:	Not Enabled

© 2008 - 2012, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

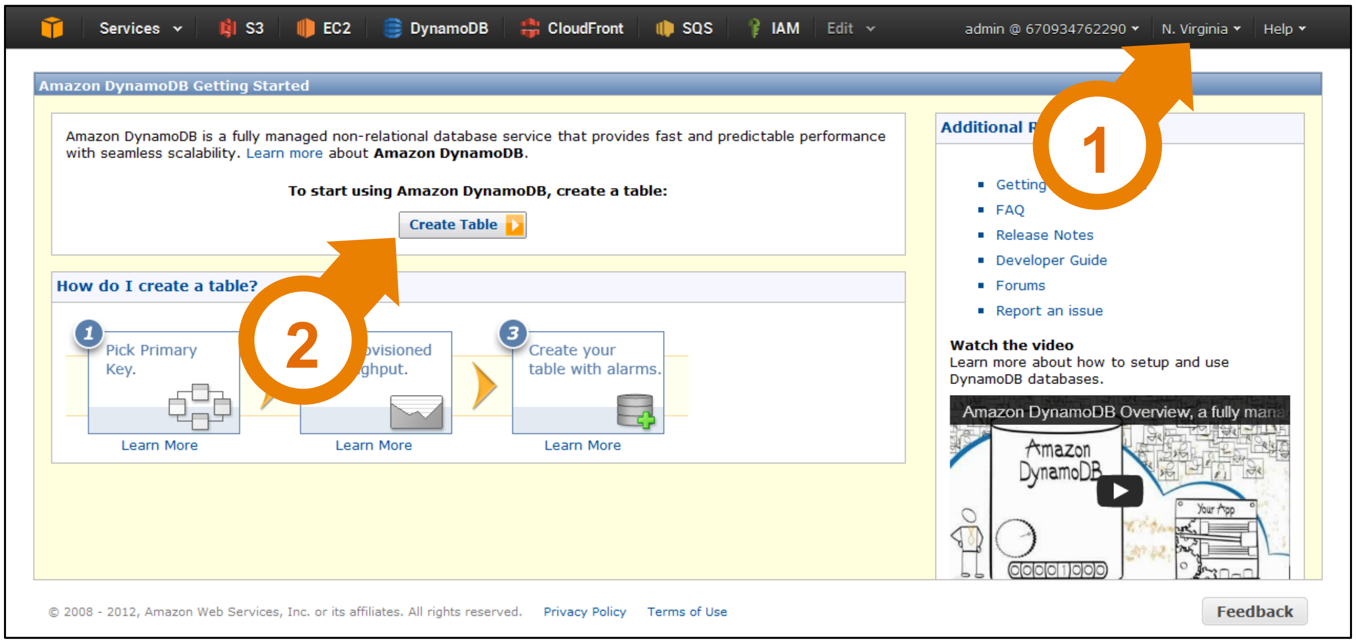
Click "**Save**", to apply the policy to your bucket.

Step 2 – Media Database

Note: this step will walk you through the creation of a DynamoDB used in Part1. Details have been removed.

Creating an Amazon DynamoDB table

Open the DynamoDB section of the AWS Web Console. Select the **US-East (N. Virginia)** region (1) and then click the "**Create Table**" button (2):



Amazon DynamoDB is a fully managed non-relational database service that provides fast and predictable performance with seamless scalability. Learn more about **Amazon DynamoDB**.

To start using Amazon DynamoDB, create a table:

Watch the video
Learn more about how to setup and use DynamoDB databases.

© 2008 - 2012, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Choose a name for the table (1). Primary Key Type has to be "**Hash / String**" and Hash Attribute Name is "**eib**" (2):

Important: Don't forget to write down the name of your table as it will be required to complete future steps.

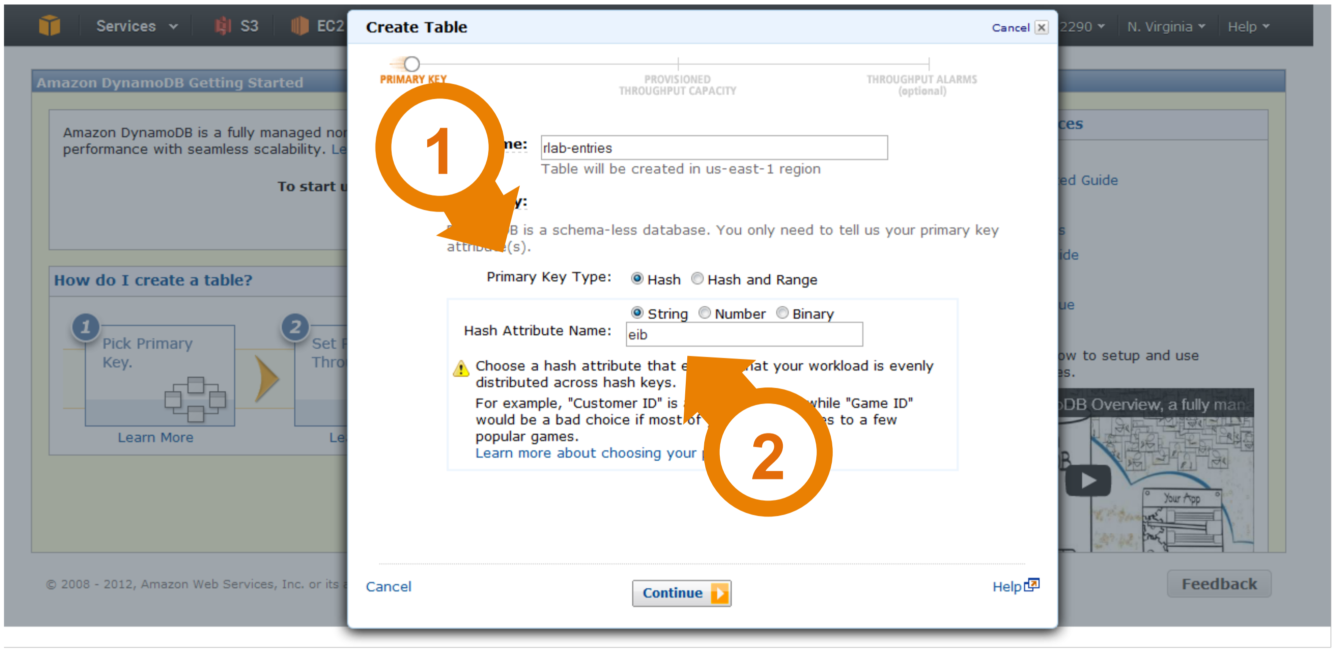


Table name: rlab-entries

Table will be created in us-east-1 region

…B is a schema-less database. You only need to tell us your primary key attri…(s).

Primary Key Type: [✓ Hash] Hash and Range

[✓ String] Number Binary

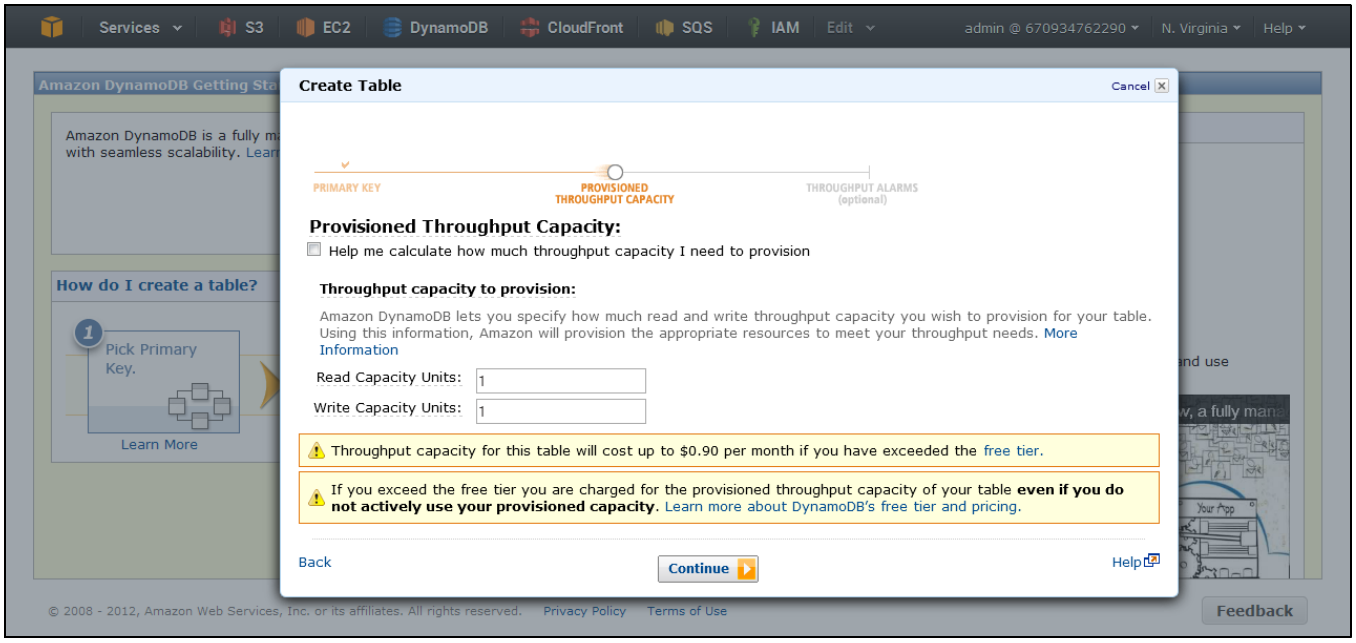
Hash Attribute Name: eib

Choose a hash attribute that e… …at your workload is evenly distributed across hash keys.

For example, "Customer ID" is … while "Game ID" would be a bad choice if most of … …s to a few popular games.

Learn more about choosing your …

Keep the default throughput settings and click "**Continue**":



Provisioned Throughput Capacity:

☐ Help me calculate how much throughput capacity I need to provision

Throughput capacity to provision:

Amazon DynamoDB lets you specify how much read and write throughput capacity you wish to provision for your table. Using this information, Amazon will provision the appropriate resources to meet your throughput needs. More Information

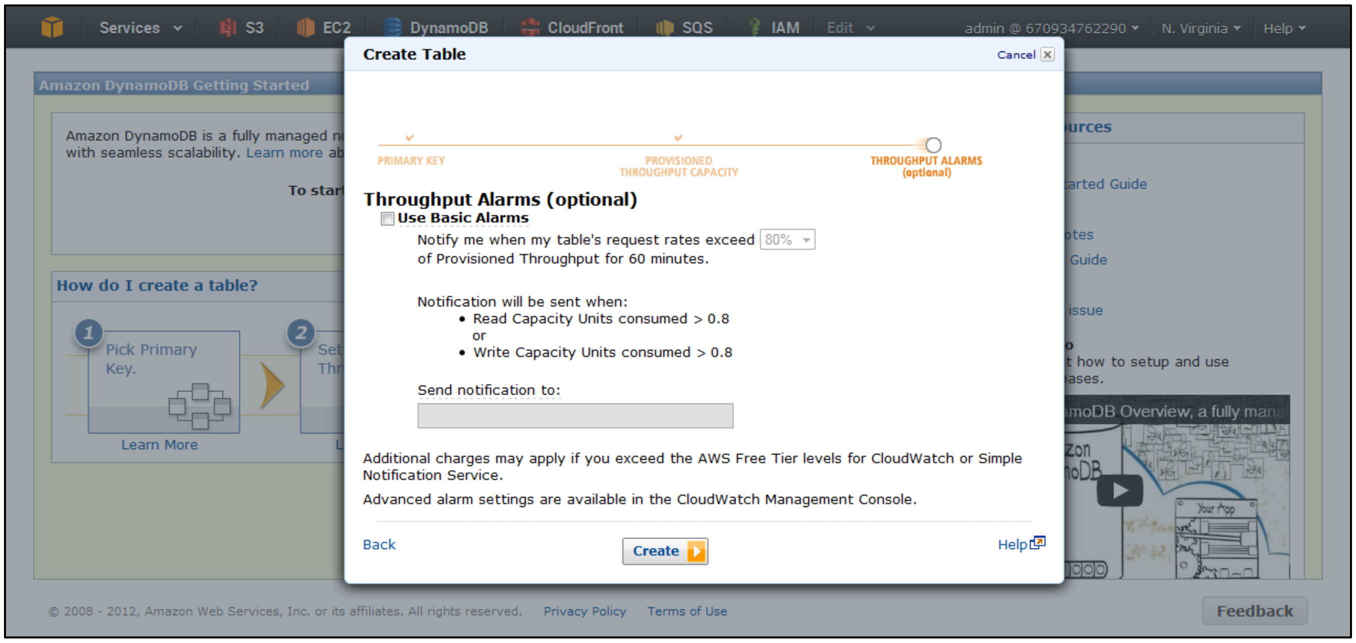
Read Capacity Units: 1

Write Capacity Units: 1

Throughput capacity for this table will cost up to $0.90 per month if you have exceeded the free tier.

If you exceed the free tier you are charged for the provisioned throughput capacity of your table **even if you do not actively use your provisioned capacity**. Learn more about DynamoDB's free tier and pricing.

For this lab, disable basic alarms, and click “**Create**”:



Create Table

Throughput Alarms (optional)

☐ Use Basic Alarms

Notify me when my table's request rates exceed 80% of Provisioned Throughput for 60 minutes.

Notification will be sent when:

- Read Capacity Units consumed > 0.8
  or
- Write Capacity Units consumed > 0.8

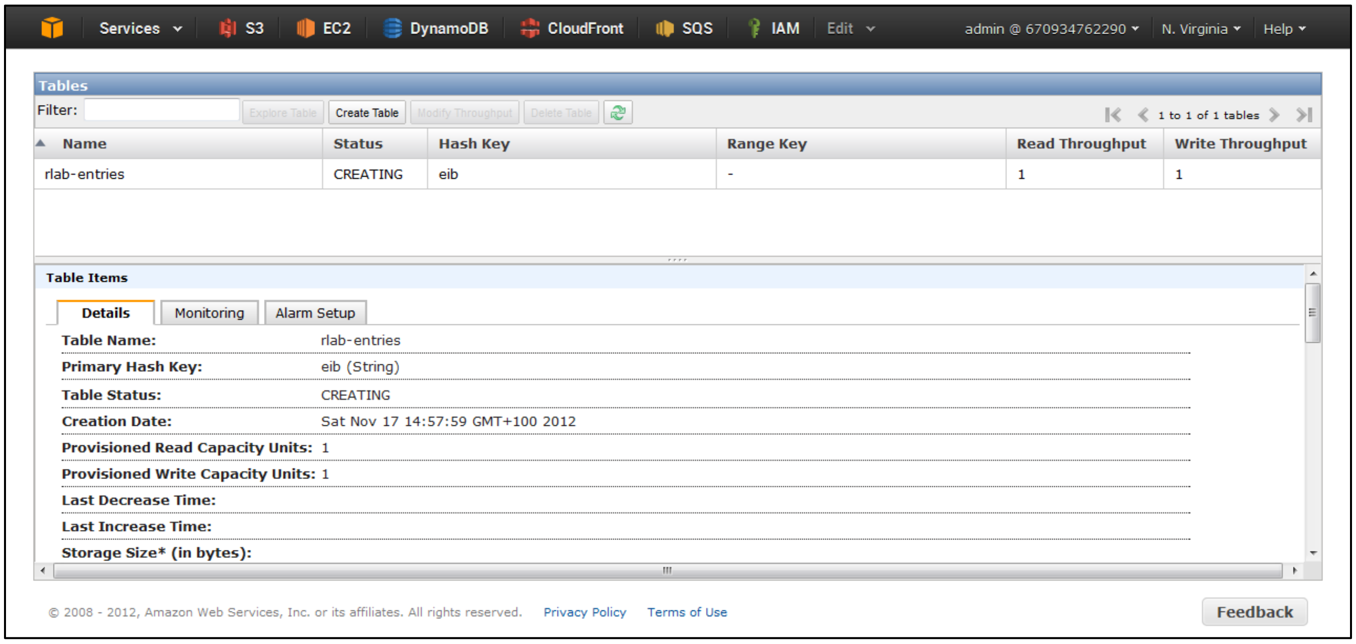
Send notification to:

Additional charges may apply if you exceed the AWS Free Tier levels for CloudWatch or Simple Notification Service.

Advanced alarm settings are available in the CloudWatch Management Console.

© 2008 - 2012, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

The table’s status will be noted “**CREATING**”. The table creation process might take a few minutes:



Name	Status	Hash Key	Range Key	Read Throughput	Write Throughput
rlab-entries	CREATING	eib	-	1	1

Table Name:	rlab-entries
Primary Hash Key:	eib (String)
Table Status:	CREATING
Creation Date:	Sat Nov 17 14:57:59 GMT+100 2012
Provisioned Read Capacity Units:	1
Provisioned Write Capacity Units:	1
Last Decrease Time:	
Last Increase Time:	
Storage Size* (in bytes):	

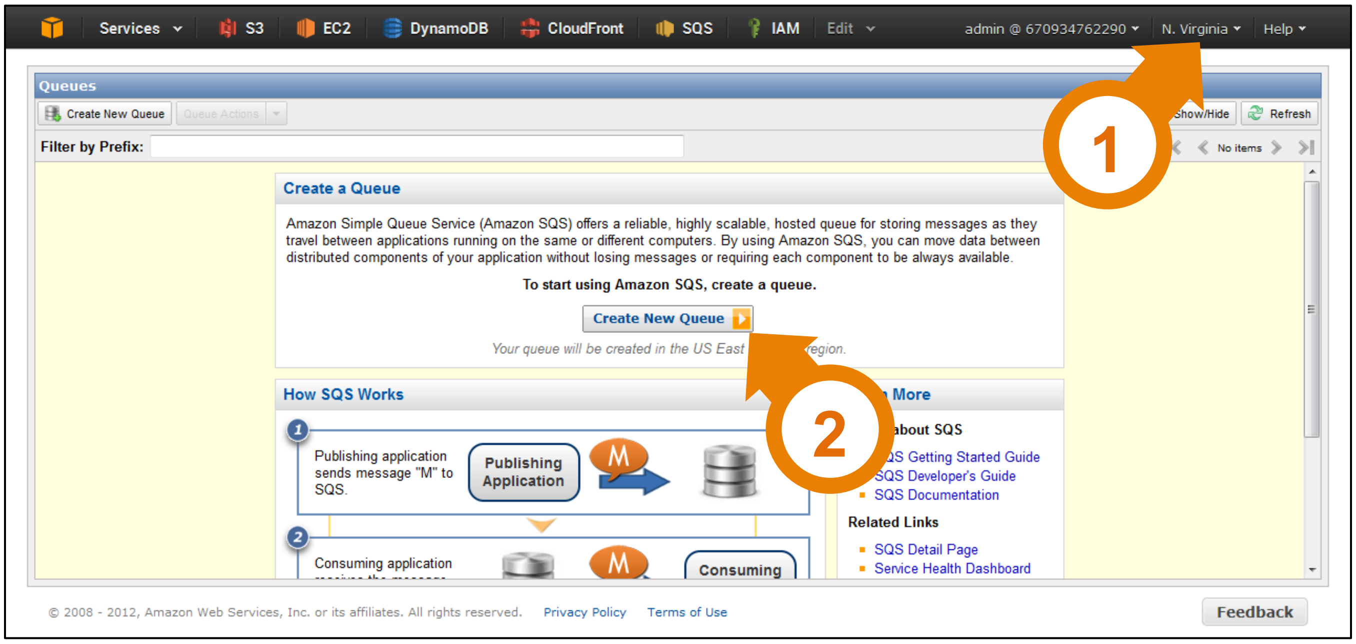
© 2008 - 2012, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Step 3 – Asynchronous Communication

Asynchronous communication between two subsystems is often implemented using message queues. Amazon Simple Queue Service (**Amazon SQS**) offers a reliable, highly scalable, hosted queue for storing messages. Amazon SQS is scalable, it was designed to enable an unlimited number of computers to read and write an unlimited number of messages at any time. To prevent messages from being lost or becoming unavailable, all messages are stored redundantly across multiple servers and availability zones. As such, Amazon SQS is a scalable and fault-tolerant system and doesn't introduce a weakness or a bottleneck in our current architecture.

Creating a SQS Queue

Open the Amazon SQS section of the AWS Web Console. Select the **US East (N.Virginia)** region (1), and then click the "**Create New Queue**" button (2):



Amazon Simple Queue Service (Amazon SQS) offers a reliable, highly scalable, hosted queue for storing messages as they travel between applications running on the same or different computers. By using Amazon SQS, you can move data between distributed components of your application without losing messages or requiring each component to be always available.

To start using Amazon SQS, create a queue.

Create New Queue

Your queue will be created in the US East region.

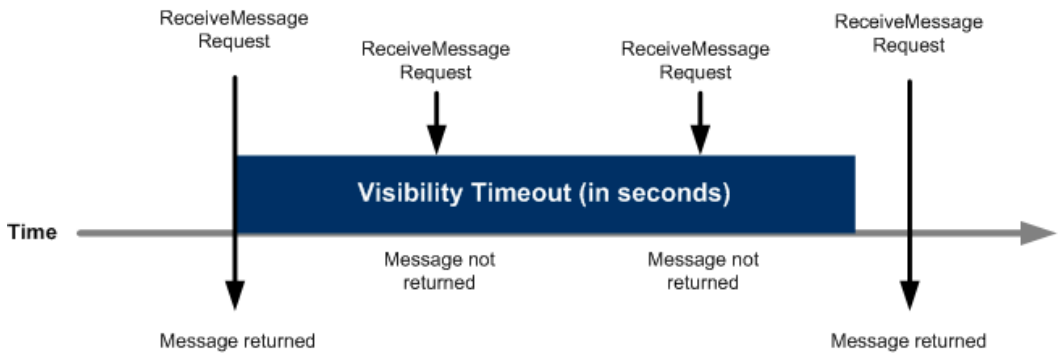
© 2008 - 2012, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Provide a **Name** for your queue (1).

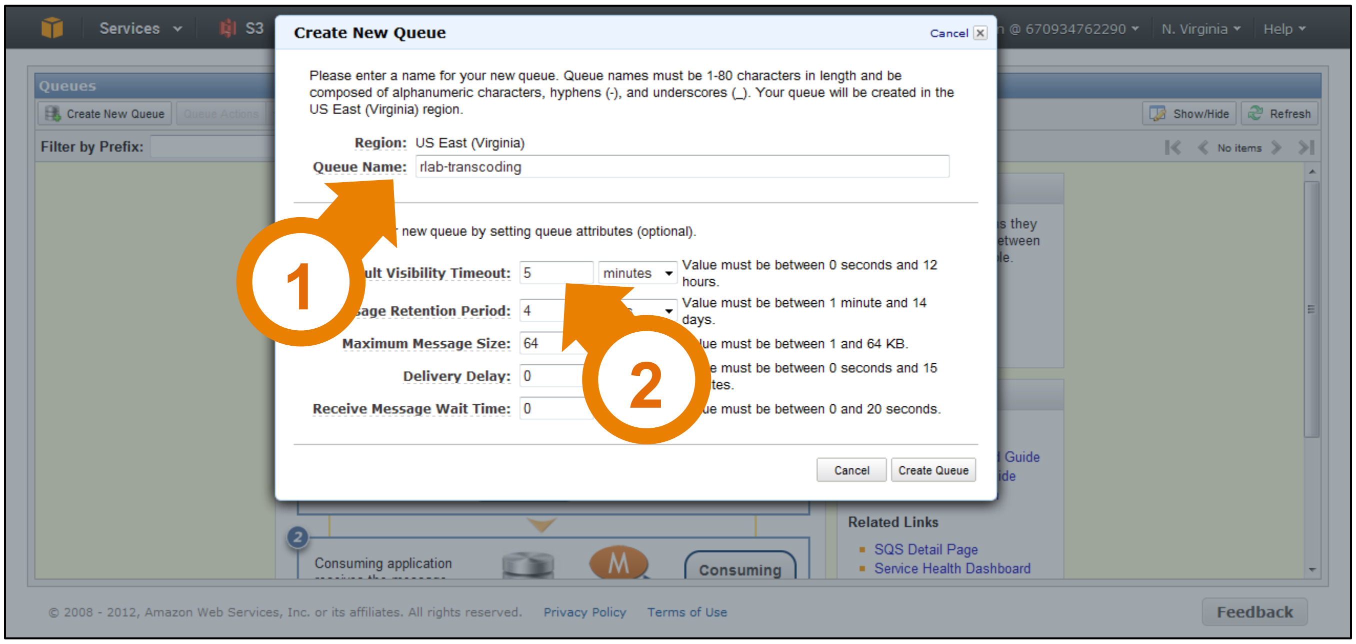
Important: Don't forget to write down the name of your queue as it will be required to complete future steps.

When a consuming component in your system receives and processes a message from the queue, the message remains in the queue. Why doesn't Amazon SQS automatically delete it? Because your system is distributed, there's no guarantee that the component will actually receive the message (it's possible the connection could break or the component could fail before receiving the message). Therefore, Amazon SQS does not delete the message, and instead, the transcoder must delete the message from the queue after receiving and processing it.

Immediately after the component receives the message, the message is still in the queue. However, you don't want other components in the system receiving and processing the message again. Therefore, Amazon SQS blocks them with a visibility timeout, which is a period of time during which Amazon SQS prevents other consuming components from receiving and processing that message. The following figure and discussion illustrates the concept.



For this lab, you can set the visibility timeout to **5 minutes** (2). It means that transcoders will have maximum 5 minutes to complete their task. Click “**Create Queue**”:



Create New Queue

Please enter a name for your new queue. Queue names must be 1-80 characters in length and be composed of alphanumeric characters, hyphens (-), and underscores (_). Your queue will be created in the US East (Virginia) region.

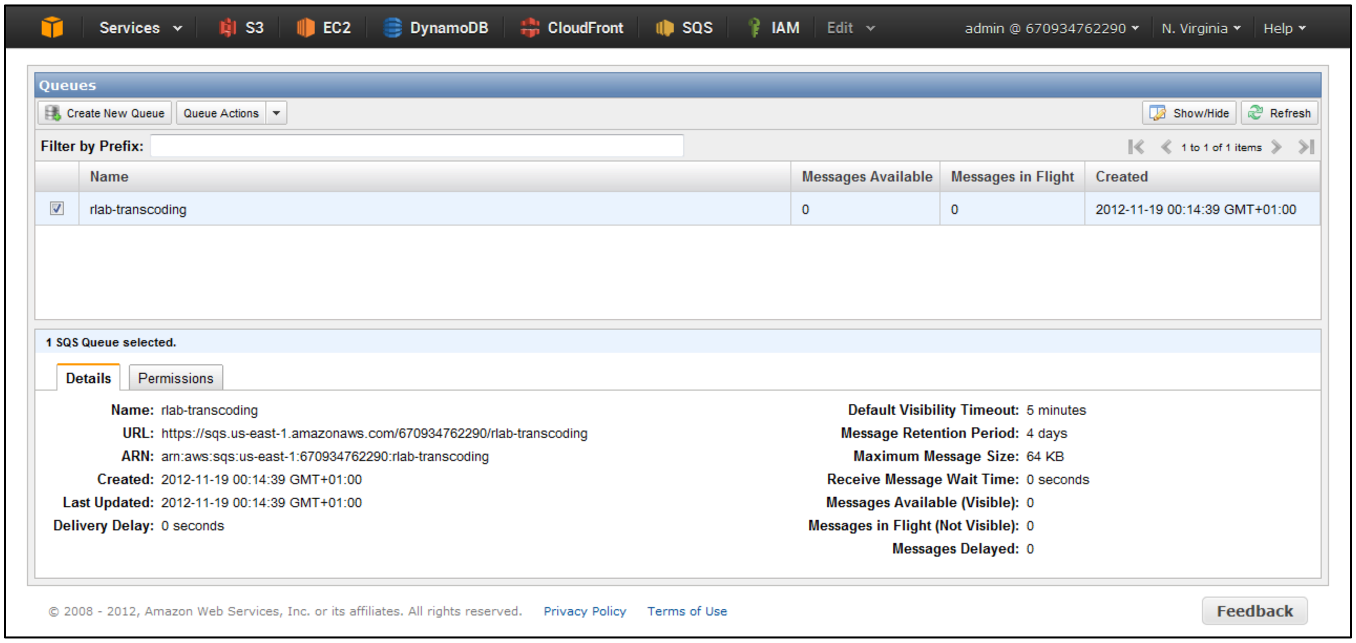
Region:	US East (Virginia)
Queue Name:	rlab-transcoding

new queue by setting queue attributes (optional).

ult Visibility Timeout:	5 minutes	Value must be between 0 seconds and 12 hours.
sage Retention Period:	4	Value must be between 1 minute and 14 days.
Maximum Message Size:	64	ue must be between 1 and 64 KB.
Delivery Delay:	0	e must be between 0 seconds and 15 tes.
Receive Message Wait Time:	0	ue must be between 0 and 20 seconds.

© 2008 - 2012, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

After a few seconds, the queue will be ready to receive messages. Note that the AWS Web Console shows the number of messages in the queue ("Messages Available") and the number of messages currently being processed ("Messages in Flight"):



Name	Messages Available	Messages in Flight	Created
rlab-transcoding	0	0	2012-11-19 00:14:39 GMT+01:00

Name:	rlab-transcoding
URL:	https://sqs.us-east-1.amazonaws.com/670934762290/rlab-transcoding
ARN:	arn:aws:sqs:us-east-1:670934762290:rlab-transcoding
Created:	2012-11-19 00:14:39 GMT+01:00
Last Updated:	2012-11-19 00:14:39 GMT+01:00
Delivery Delay:	0 seconds
Default Visibility Timeout:	5 minutes
Message Retention Period:	4 days
Maximum Message Size:	64 KB
Receive Message Wait Time:	0 seconds
Messages Available (Visible):	0
Messages in Flight (Not Visible):	0
Messages Delayed:	0

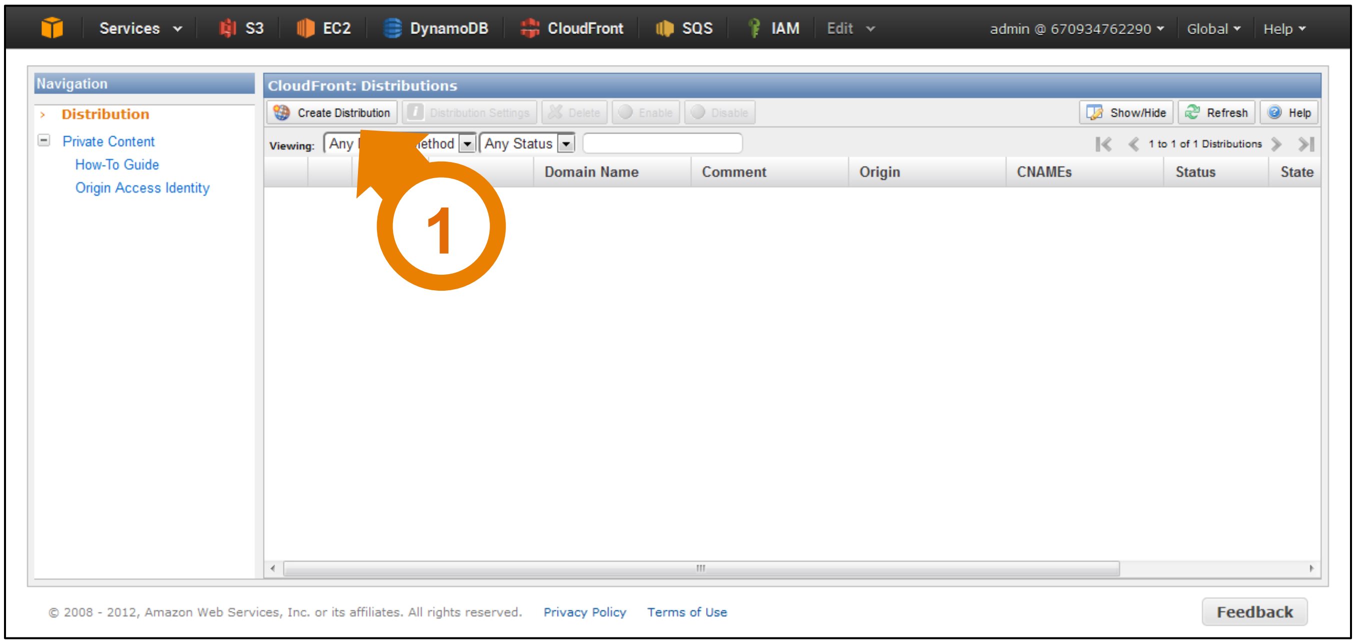
© 2008 - 2012, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Step 4 – Content Delivery

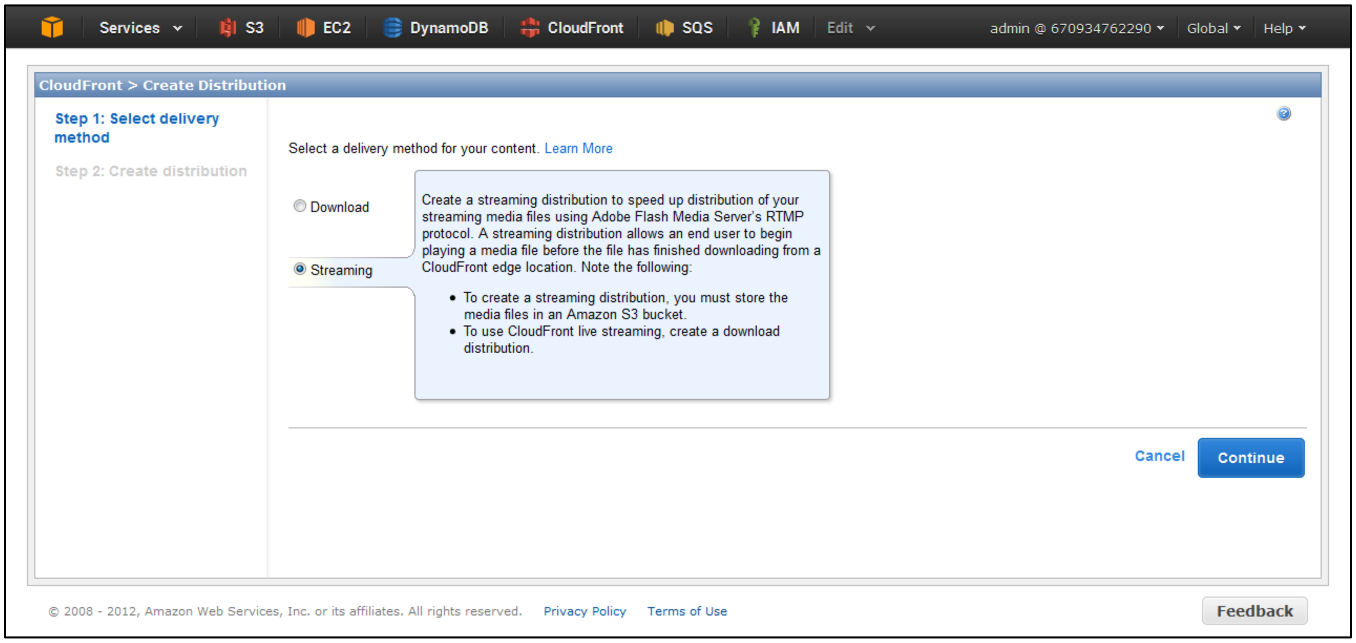
To deliver video streaming using a Flash player embedded in the web pages of the application, we need a streaming web server. Fortunately, this feature is supported by Amazon CloudFront, a web service that speeds up distribution of your static and dynamic web content to end users. Amazon CloudFront provides streaming distributions that deliver digital media using Adobe Flash Media Server and the Real-Time Messaging Protocol.

Creating an Amazon CloudFront Streaming Distribution

Open the Amazon CloudFront section of the AWS Web Console and click the "**Create Distribution**" button (1):



Choose "**Streaming**" distribution method and click "**Continue**":


Services
S3
EC2
DynamoDB
CloudFront
SQS
IAM
Edit
admin @ 670934762290
Global
Help
CloudFront > Create Distribution
Step 1: Select delivery method
Step 2: Create distribution
Cancel
Continue
© 2008 - 2012, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use
Feedback

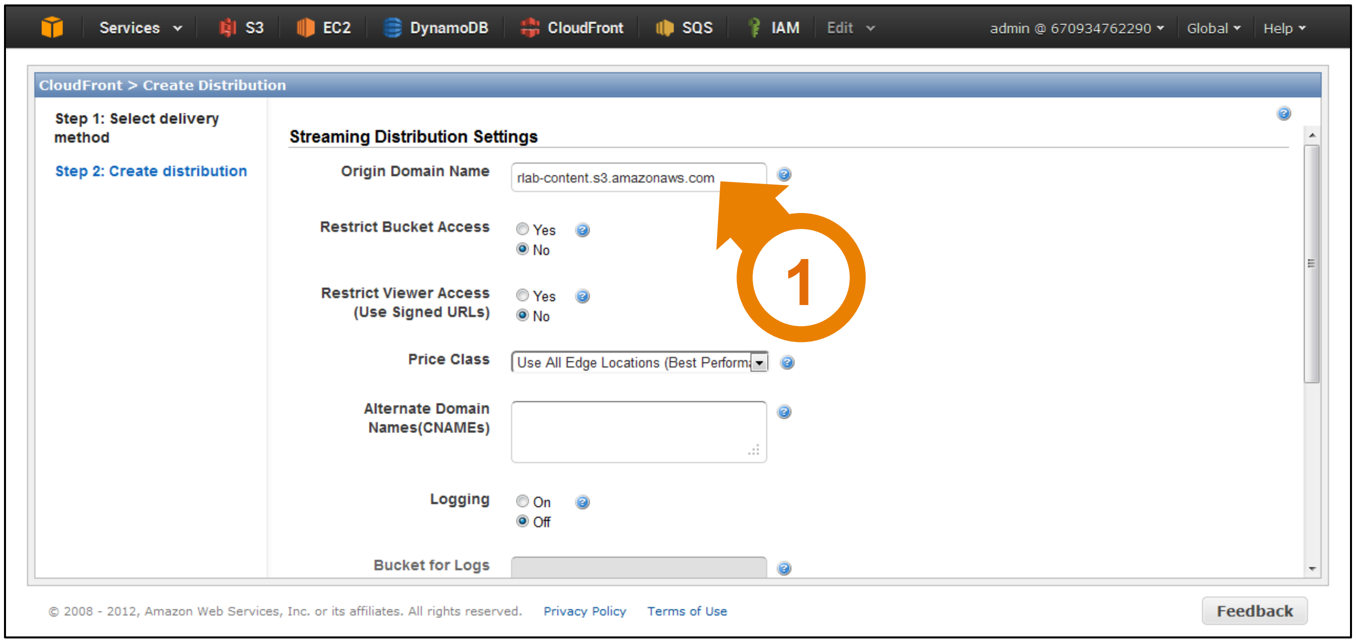
Select a delivery method for your content. Learn More

Download [✓ Streaming]

Create a streaming distribution to speed up distribution of your streaming media files using Adobe Flash Media Server's RTMP protocol. A streaming distribution allows an end user to begin playing a media file before the file has finished downloading from a CloudFront edge location. Note the following:

- To create a streaming distribution, you must store the media files in an Amazon S3 bucket.
- To use CloudFront live streaming, create a download distribution.

Provide your S3 bucket as "**Origin Domain Name**" (1) and click "**Create Distribution**":

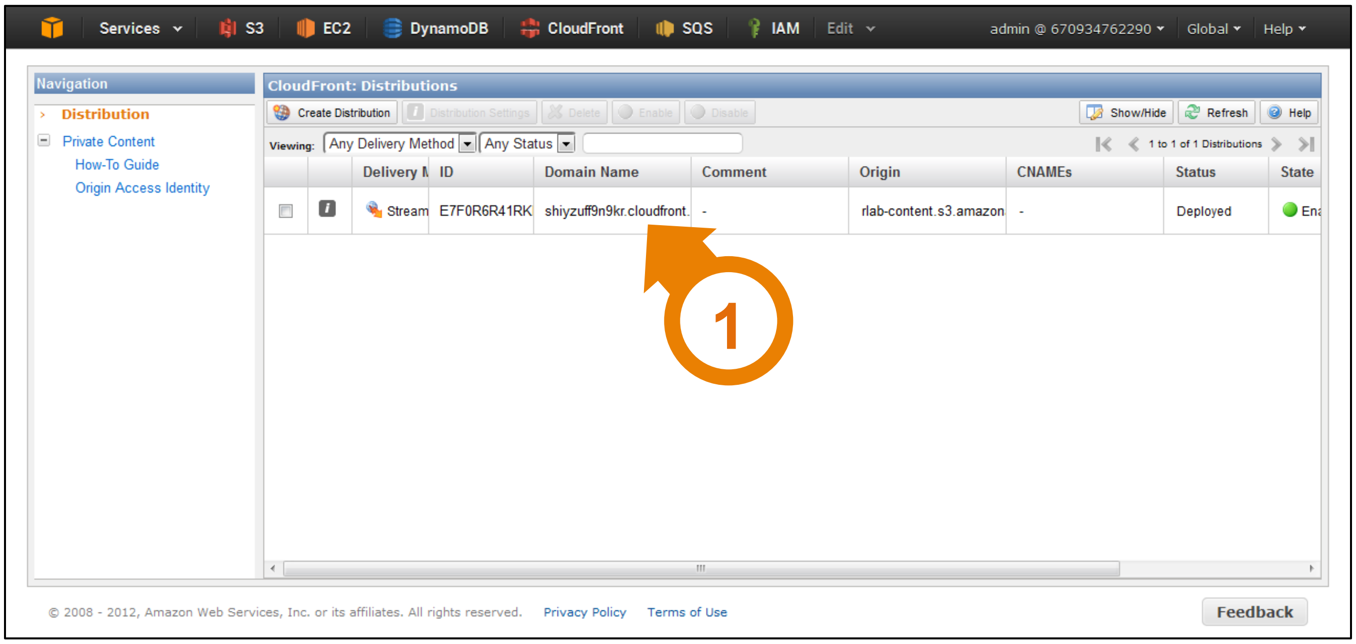

Services
S3
EC2
DynamoDB
CloudFront
SQS
IAM
Edit
admin @ 670934762290
Global
Help
CloudFront > Create Distribution
Step 1: Select delivery method
Step 2: Create distribution
1
Feedback

Streaming Distribution Settings

Origin Domain Name	rlab-content.s3.amazonaws.com
Restrict Bucket Access	Yes [✓ No]
Restrict Viewer Access (Use Signed URLs)	Yes [✓ No]
Price Class	Use All Edge Locations (Best Perform
Alternate Domain Names(CNAMEs)	
Logging	On [✓ Off]
Bucket for Logs	

© 2008 - 2012, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Your CloudFront distribution will be created in a couple of minutes. Note the Domain Name of your distribution (1), as it will be needed in the next steps:



		Delivery M	ID	Domain Name	Comment	Origin	CNAMEs	Status	State
		Stream	E7F0R6R41RK	shiyzuff9n9kr.cloudfront.	-	rlab-content.s3.amazon	-	Deployed	Ena

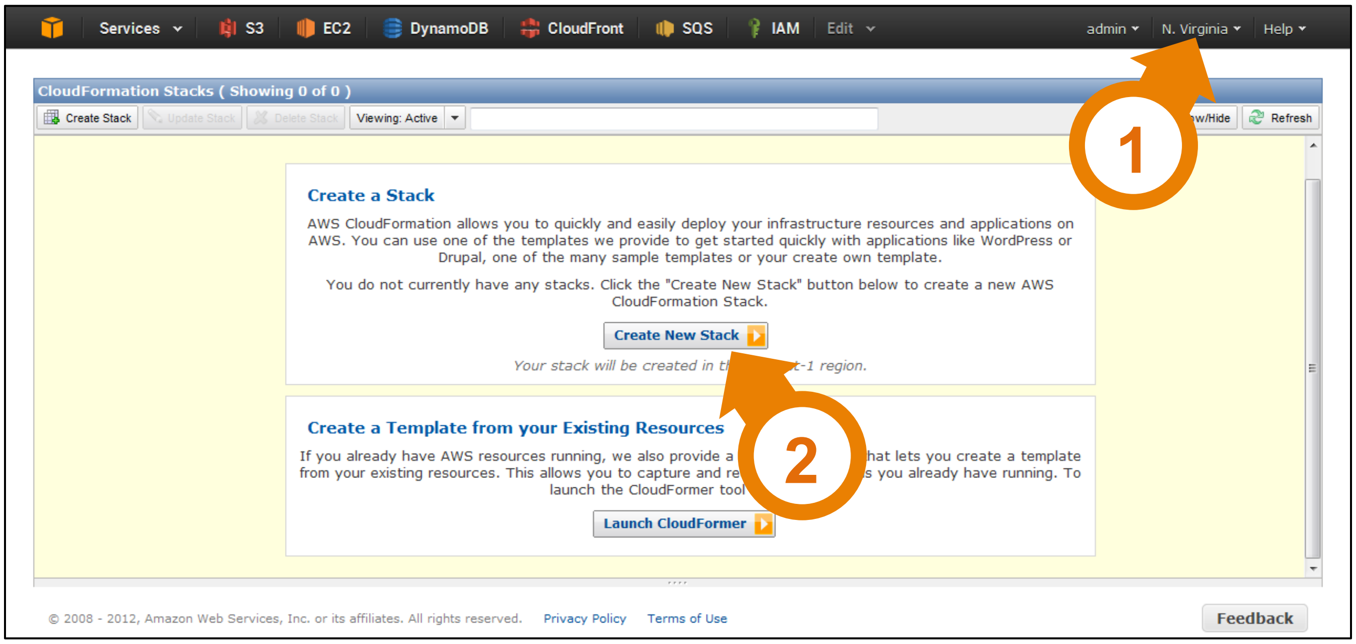
© 2008 - 2012, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Step 5 – Web Servers and Transcoders

As you did in Part 1, you will use a CloudFormation script to build the front end, including an Elastic Load Balancer, a Security Group, an Auto Scaling Group and EC2 instances for web servers. This script will also build an Auto Scaling group and transcoder instances polling the SQS queue.

Deployment using CloudFormation

Open the AWS CloudFormation section of the AWS Web Console. Select the "**US East (N.Virginia)**" region (1), and then click the "**Create New Stack**" button (2):



Create a Stack

AWS CloudFormation allows you to quickly and easily deploy your infrastructure resources and applications on AWS. You can use one of the templates we provide to get started quickly with applications like WordPress or Drupal, one of the many sample templates or your create own template.

You do not currently have any stacks. Click the "Create New Stack" button below to create a new AWS CloudFormation Stack.

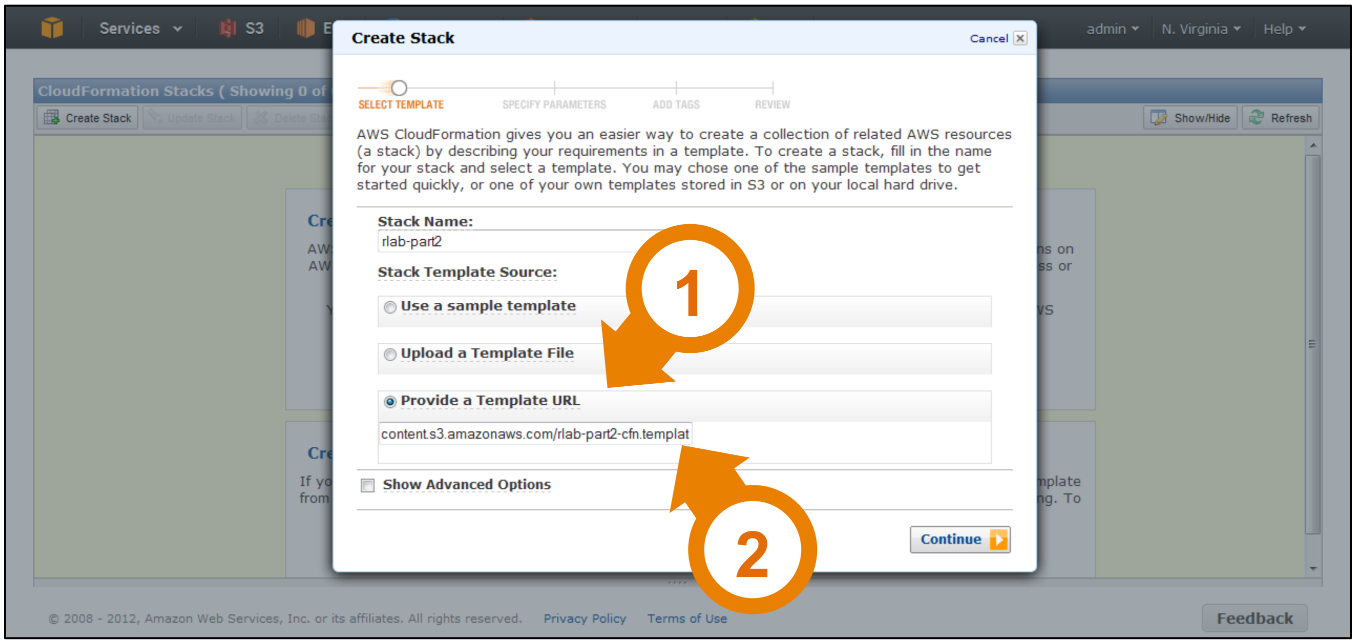
Your stack will be created in th t-1 region.

Create a Template from your Existing Resources

If you already have AWS resources running, we also provide a hat lets you create a template from your existing resources. This allows you to capture and re s you already have running. To launch the CloudFormer tool

© 2008 - 2012, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Provide a name to the CloudFormation stack (1) and choose "**Provide a Template URL**" for the origin of the stack (2). The URL containing the stack is http://rlab-content.s3.amazonaws.com/rlab-part2-cfn.template.



AWS CloudFormation gives you an easier way to create a collection of related AWS resources (a stack) by describing your requirements in a template. To create a stack, fill in the name for your stack and select a template. You may chose one of the sample templates to get started quickly, or one of your own templates stored in S3 or on your local hard drive.

Stack Name: rlab-part2

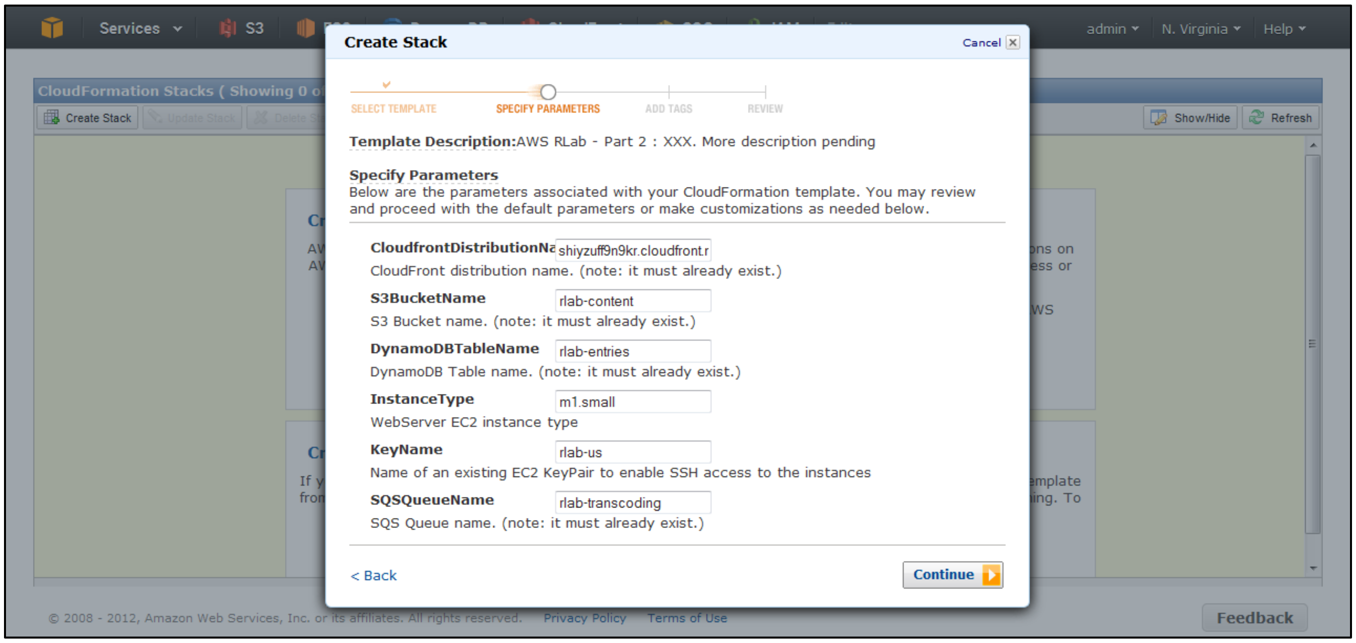
Stack Template Source:

- Use a sample template
- Upload a Template File
- [✓ Provide a Template URL] content.s3.amazonaws.com/rlab-part2-cfn.templat

Show Advanced Options

© 2008 - 2012, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

The stack contains parameter placeholders that allow you to specify your CloudFront distribution, DynamoDB table, your S3 bucket, customize the instance size and provide a Key Pair name for remote login:

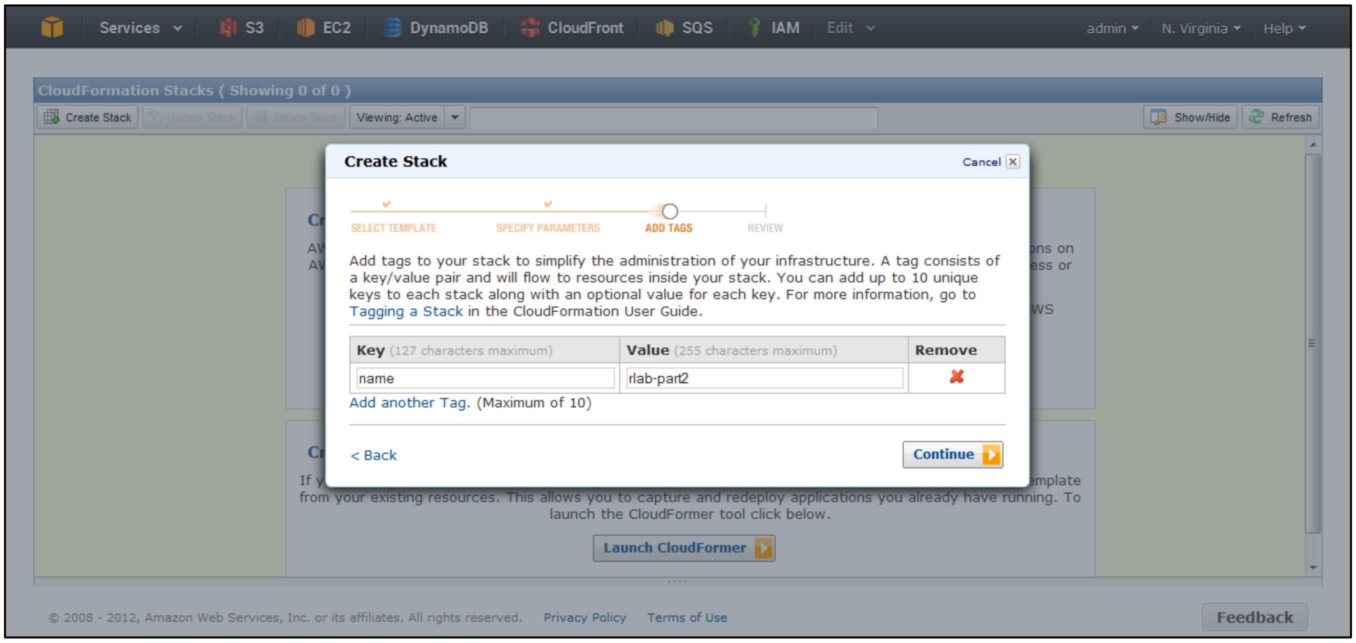


Template Description: AWS RLab - Part 2 : XXX. More description pending

Specify Parameters
Below are the parameters associated with your CloudFormation template. You may review and proceed with the default parameters or make customizations as needed below.

CloudfrontDistributionNa	shiyzuff9n9kr.cloudfront.r
CloudFront distribution name. (note: it must already exist.)	
S3BucketName	rlab-content
S3 Bucket name. (note: it must already exist.)	
DynamoDBTableName	rlab-entries
DynamoDB Table name. (note: it must already exist.)	
InstanceType	m1.small
WebServer EC2 instance type	
KeyName	rlab-us
Name of an existing EC2 KeyPair to enable SSH access to the instances	
SQSQueueName	rlab-transcoding
SQS Queue name. (note: it must already exist.)	

You can provide some tags which will be assigned to the resources created by this stack:



Add tags to your stack to simplify the administration of your infrastructure. A tag consists of a key/value pair and will flow to resources inside your stack. You can add up to 10 unique keys to each stack along with an optional value for each key. For more information, go to Tagging a Stack in the CloudFormation User Guide.

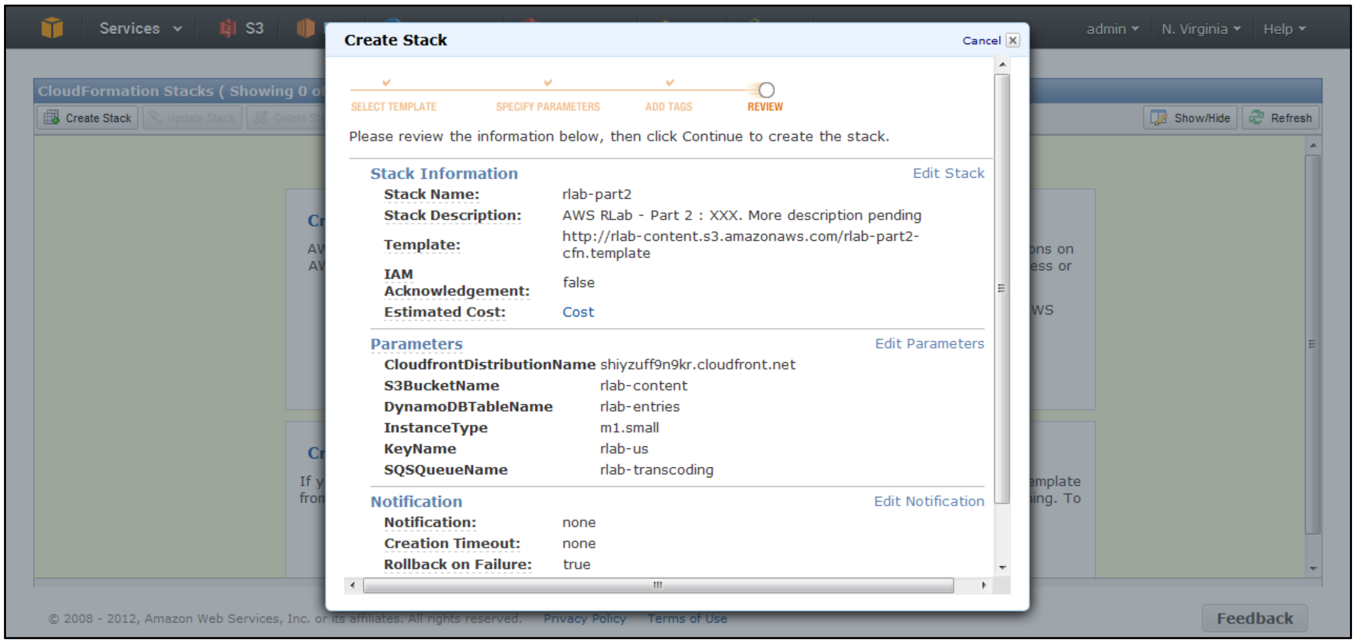
Key (127 characters maximum)	Value (255 characters maximum)	Remove
name	rlab-part2	✗

Add another Tag. (Maximum of 10)

from your existing resources. This allows you to capture and redeploy applications you already have running. To launch the CloudFormer tool click below.

© 2008 - 2012, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Finally, review your stack parameters and click “**Create**”:



Please review the information below, then click Continue to create the stack.

Stack Information

Stack Name:	rlab-part2
Stack Description:	AWS RLab - Part 2 : XXX. More description pending
Template:	http://rlab-content.s3.amazonaws.com/rlab-part2-cfn.template
IAM Acknowledgement:	false
Estimated Cost:	Cost

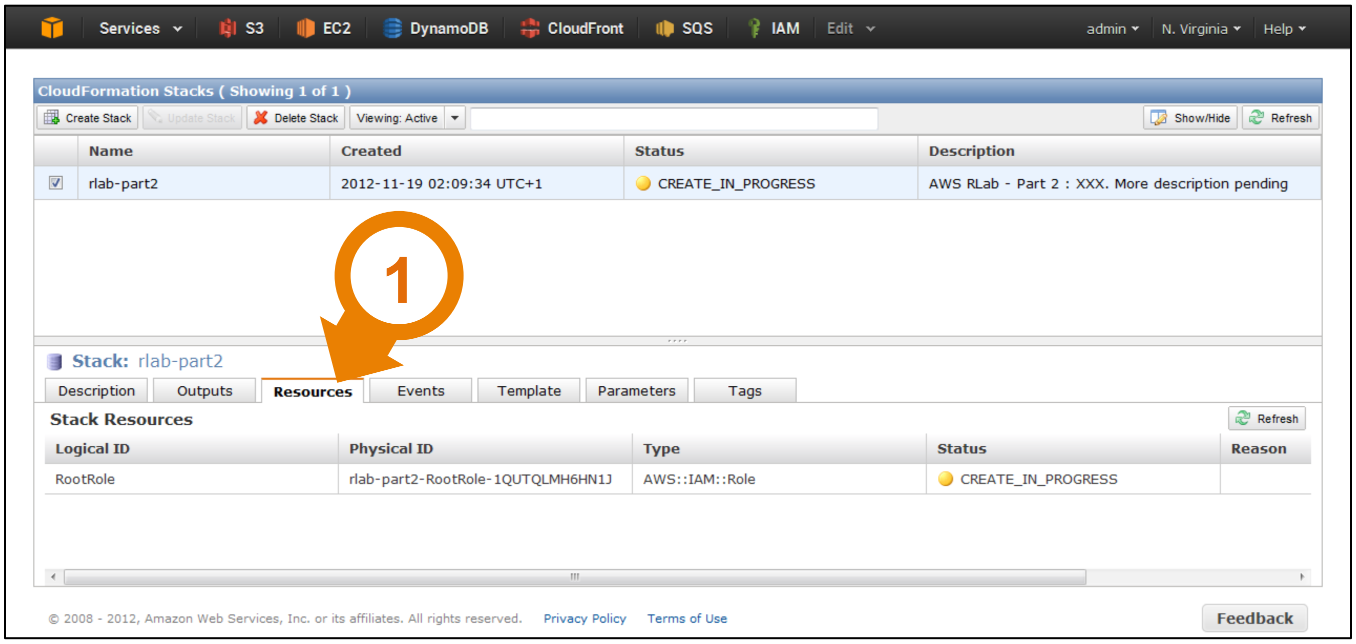
Parameters

CloudfrontDistributionName	shiyzuff9n9kr.cloudfront.net
S3BucketName	rlab-content
DynamoDBTableName	rlab-entries
InstanceType	m1.small
KeyName	rlab-us
SQSQueueName	rlab-transcoding

Notification

Notification:	none
Creation Timeout:	none
Rollback on Failure:	true

The stack creation process will take a couple of minutes. You can follow the stack creation process by checking the "**Resources**" tab (1):



CloudFormation Stacks (Showing 1 of 1)

Name	Created	Status	Description
rlab-part2	2012-11-19 02:09:34 UTC+1	CREATE_IN_PROGRESS	AWS RLab - Part 2 : XXX. More description pending

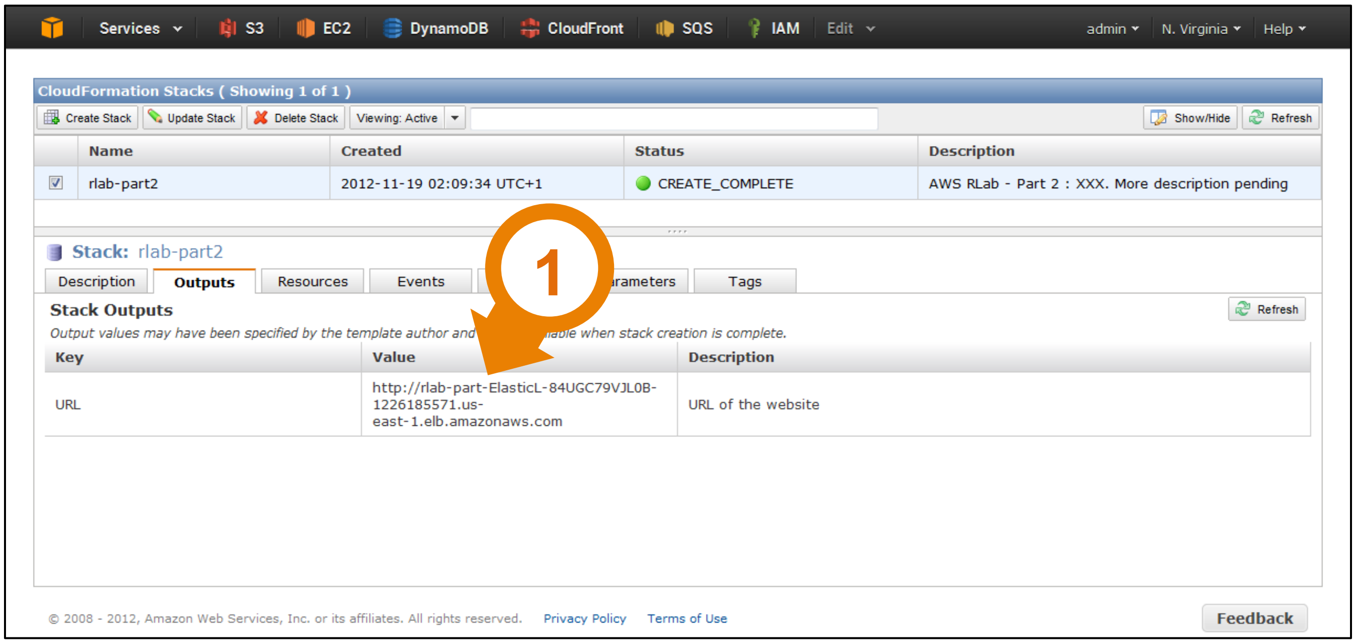
Stack: rlab-part2

Stack Resources

Logical ID	Physical ID	Type	Status	Reason
RootRole	rlab-part2-RootRole-1QUTQLMH6HN1J	AWS::IAM::Role	CREATE_IN_PROGRESS	

© 2008 - 2012, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Once the stack creation is completed, the "**Outputs**" tab will show the URL endpoint of the Elastic Load Balancer (1):



CloudFormation Stacks (Showing 1 of 1)

Name	Created	Status	Description
rlab-part2	2012-11-19 02:09:34 UTC+1	CREATE_COMPLETE	AWS RLab - Part 2 : XXX. More description pending

Stack: rlab-part2

Stack Outputs

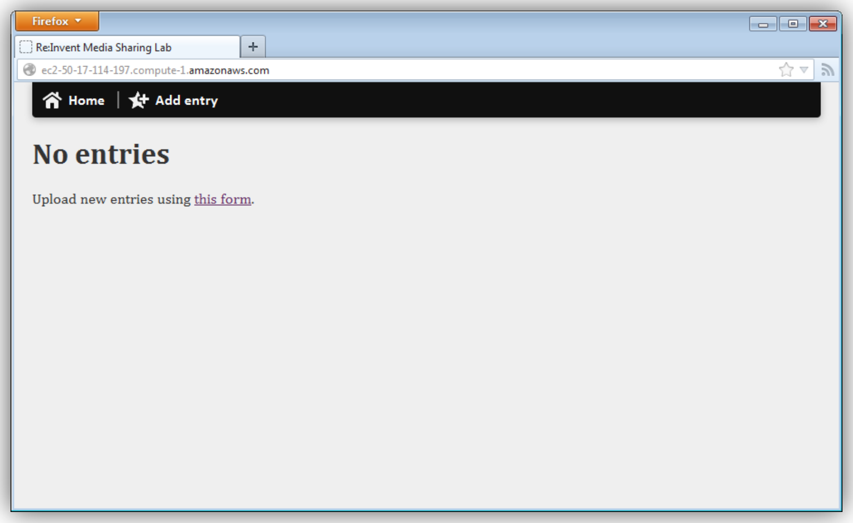
Output values may have been specified by the template author and ...lable when stack creation is complete.

Key	Value	Description
URL	http://rlab-part-ElasticL-84UGC79VJL0B-1226185571.us-east-1.elb.amazonaws.com	URL of the website

© 2008 - 2012, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Testing the deployment

To test the deployment, copy the **Elastic Load Balancer Domain Name** (1) and paste it in a new tab of your web browser. You should see the home page of the web application:



Firefox
Re:Invent Media Sharing Lab
ec2-50-17-114-197.compute-1.amazonaws.com
Home
Add entry

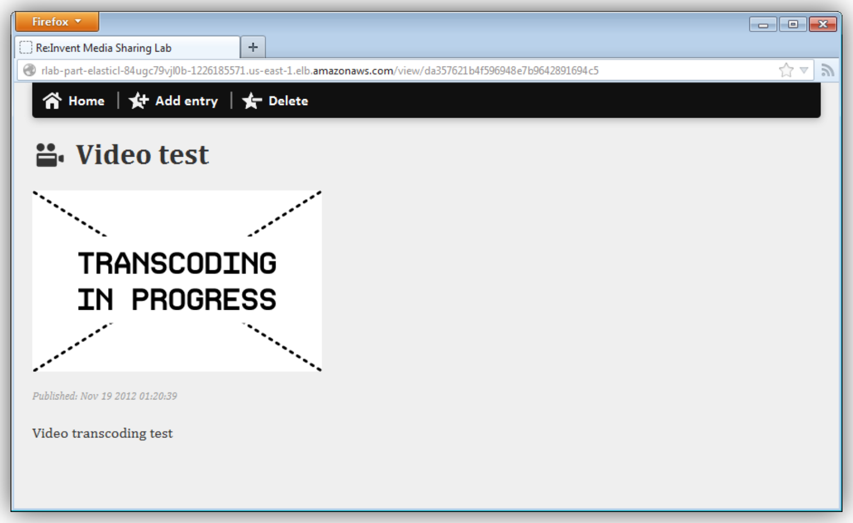
No entries

Upload new entries using this form.

Add a video. There are some sample videos contained in the video library on your computer, and you can download additional videos here:

- http://rlab-content.s3.amazonaws.com/video_1_bigdata.mp4
- http://rlab-content.s3.amazonaws.com/video_2_spot.mp4
- http://rlab-content.s3.amazonaws.com/video_3_rds.mp4

Once the file upload is completed, the application shows "transcoding in progress":



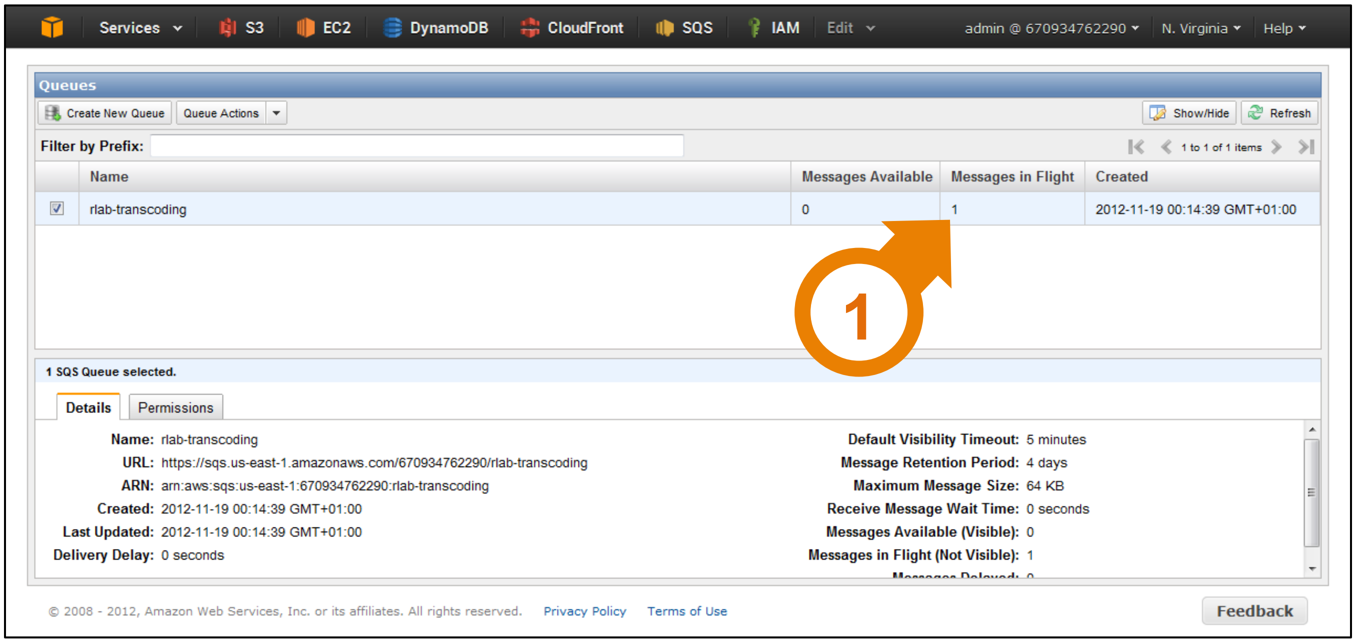
Firefox
Re:Invent Media Sharing Lab
rlab-part-elasticl-84ugc79vjl0b-1226185571.us-east-1.elb.amazonaws.com/view/da357621b4f596948e7b9642891694c5
Home
Add entry
Delete
TRANSCODING IN PROGRESS

Video test

Published: Nov 19 2012 01:20:39

Video transcoding test

You can check in the AWS Web Console that the SQS queue shows a message being processed (1):



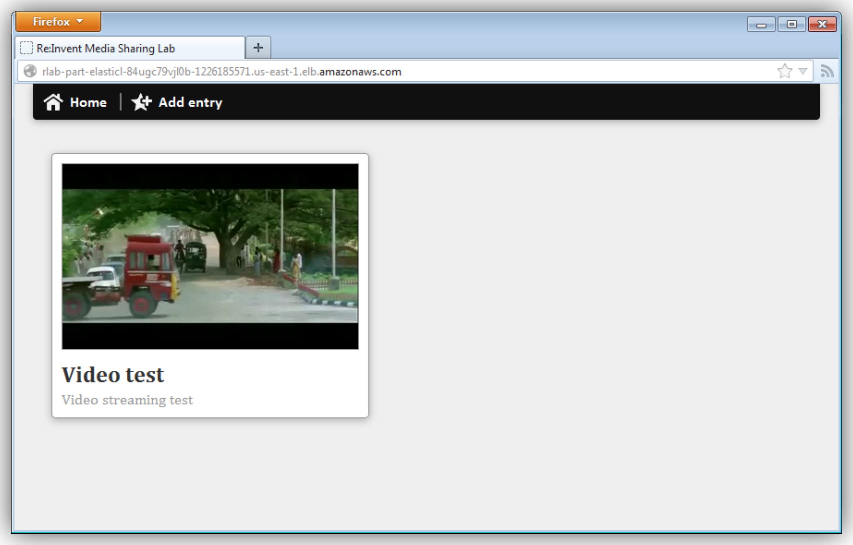
	Name	Messages Available	Messages in Flight	Created
☑	rlab-transcoding	0	1	2012-11-19 00:14:39 GMT+01:00

1 SQS Queue selected.

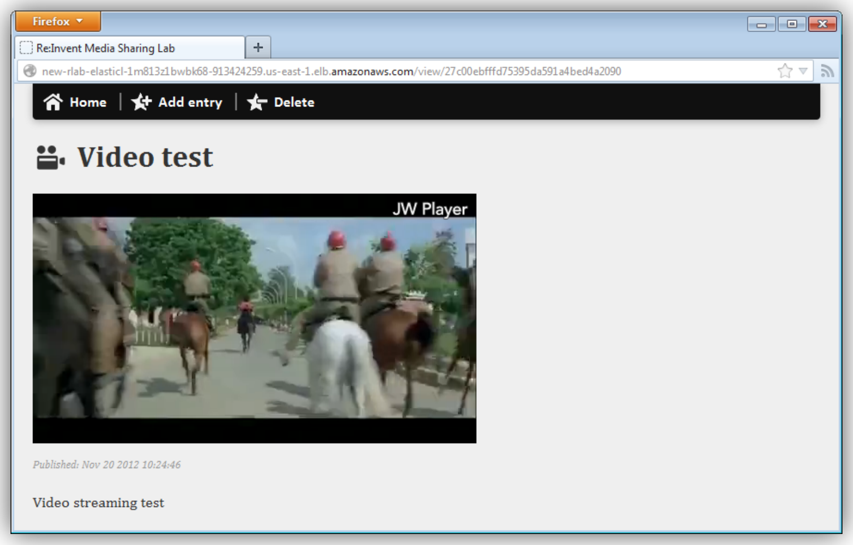
Name:	rlab-transcoding	Default Visibility Timeout:	5 minutes
URL:	https://sqs.us-east-1.amazonaws.com/670934762290/rlab-transcoding	Message Retention Period:	4 days
ARN:	arn:aws:sqs:us-east-1:670934762290:rlab-transcoding	Maximum Message Size:	64 KB
Created:	2012-11-19 00:14:39 GMT+01:00	Receive Message Wait Time:	0 seconds
Last Updated:	2012-11-19 00:14:39 GMT+01:00	Messages Available (Visible):	0
Delivery Delay:	0 seconds	Messages in Flight (Not Visible):	1

© 2008 - 2012, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

After a couple of seconds, refresh the web application home page. You should see a thumbnail of your video:


Firefox
Re:Invent Media Sharing Lab
rlab-part-elasticl-84ugc79vjl0b-1226185571.us-east-1.elb.amazonaws.com
Home
Add entry
Video test
Video streaming test

Click on your video thumbnail, you should be able to watch your video with a Video Streaming Flash viewer:


Firefox
Re:Invent Media Sharing Lab
new-rlab-elasticl-1m813z1bwbk68-913424259.us-east-1.elb.amazonaws.com/view/27c00ebfffd75395da591a4bed4a2090
Home
Add entry
Delete
JW Player

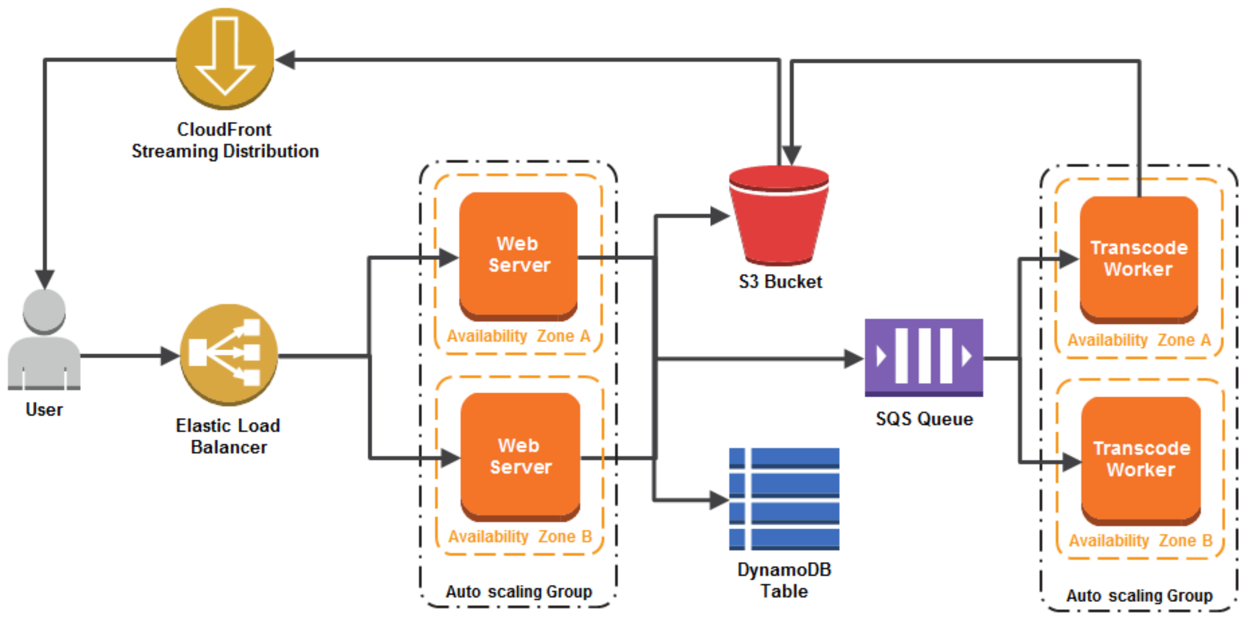
Video test

Published: Nov 20 2012 10:24:46

Video streaming test

Final Architecture Overview

The following diagram represents the final architecture you built in this lab:



The front end web servers and the back end transcoders are both fault-tolerant and scalable. In case of increased inbound traffic, the Auto Scaling group will automatically provision new instances for the front end, and independently provision transcoder instances if there are many videos to manage. Data storage systems and the queue are fault-tolerant and scalable by design.