DEV309

CI/CD for Serverless and Containerized Applications

Clare Liguori
Principal Engineer
AWS Container Services

aws re:Invent

aws

Agenda

CI/CD for modern applications

Continuous integration

Continuous deployment

Infrastructure as code

Demo

What is a modern application?

Approaches to modern application development

- Simplify environment management with serverless technologies
- Reduce the impact of code changes with microservice architectures
- Automate operations by modeling applications & infrastructure as code
- Accelerate the delivery of new, high-quality services with CI/CD
- Gain insight across resources and applications by enabling observability
- Protect customers and the business with end-to-end security & compliance

Approaches to modern application development

AWS Lambda

Serverless functions

Event-driven
Many language runtimes
Data source integrations
No server management

AWS Fargate

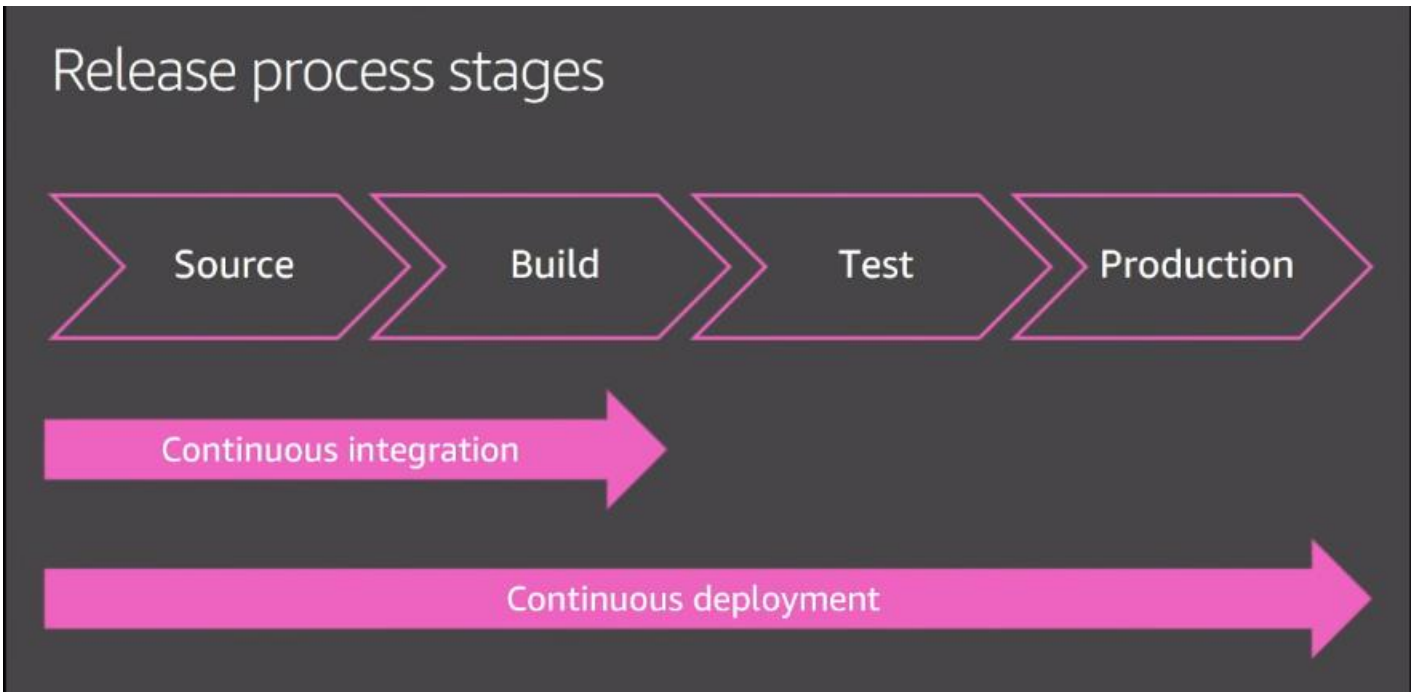
Serverless containers

Long-running
Abstracts the OS
Fully managed orchestration
Fully managed cluster scaling

Approaches to modern application development

- Simplify environment management with serverless technologies
- Reduce the impact of code changes with microservice architectures
- Automate operations by modeling applications & infrastructure as code
- Accelerate the delivery of new, high-quality services with CI/CD
- Gain insight across resources and applications by enabling observability
- Protect customers and the business with end-to-end security & compliance

Release process stages



Release process stages

Source

- Check-in source code such as .java files
- Peer review new code

Build

- Compile code
- Unit tests
- Style checkers
- Create container images and function deployment packages

Test

- Integration tests with other systems
- Load testing
- UI tests
- Security testing

Production

- Deployment to production environments
- Monitor code in production to quickly detect errors

Effects of CI/CD

Deployment frequency	Weekly – monthly	→	Hourly – daily
Change lead time	One – six months	→	One – seven days
Change failure rate	46 - 60%	→	0 - 15%

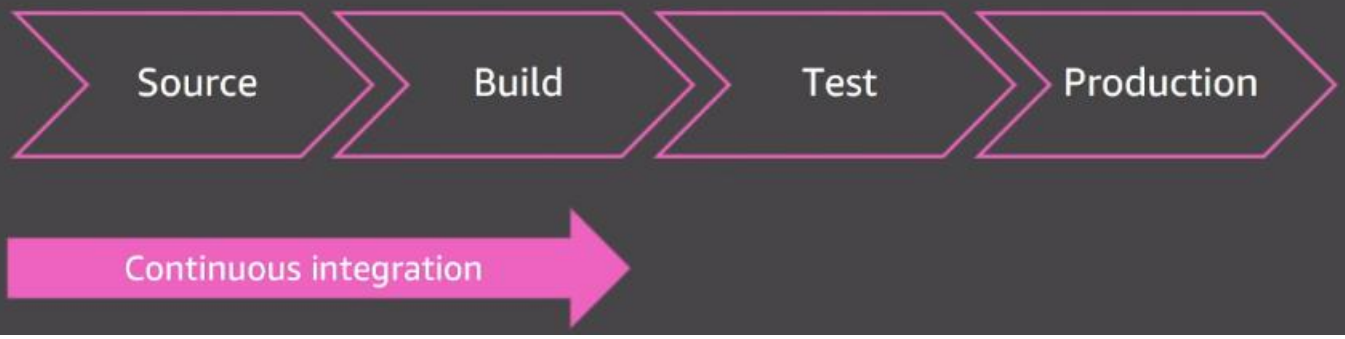
48% of software teams

Source: 2018 DORA State of DevOps Report

Pillars of releasing modern applications

Continuous integration | Continuous deployment | Infrastructure as code

Continuous integration goals



Continuous integration goals

1. Automatically kick off a new release when new code is checked in
2. Build and test code in a consistent, repeatable environment
3. Continually have an artifact ready for deployment
4. Continually close feedback loop when build fails

AWS CodePipeline

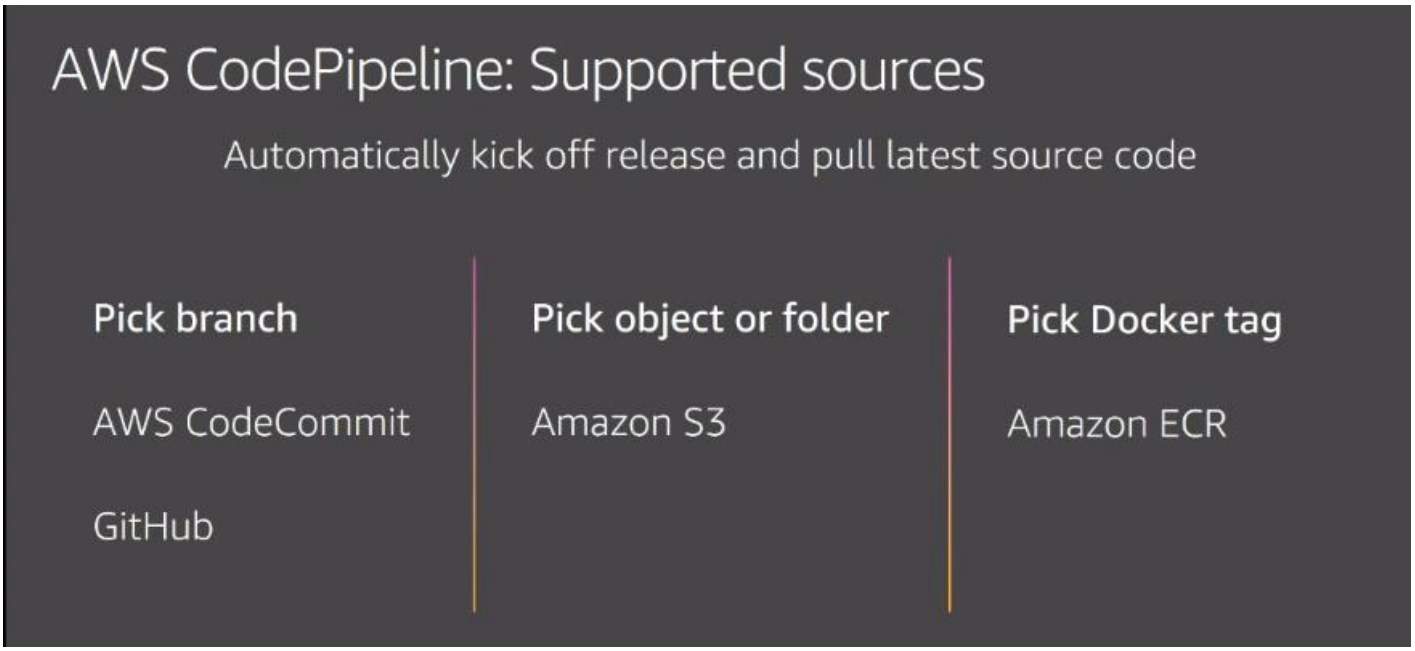
- Continuous delivery service for fast and reliable application updates
- Model and visualize your software release process
- Builds, tests, and deploys your code every time there is a code change
- Integrates with third-party tools and AWS

AWS CodePipeline: Supported sources

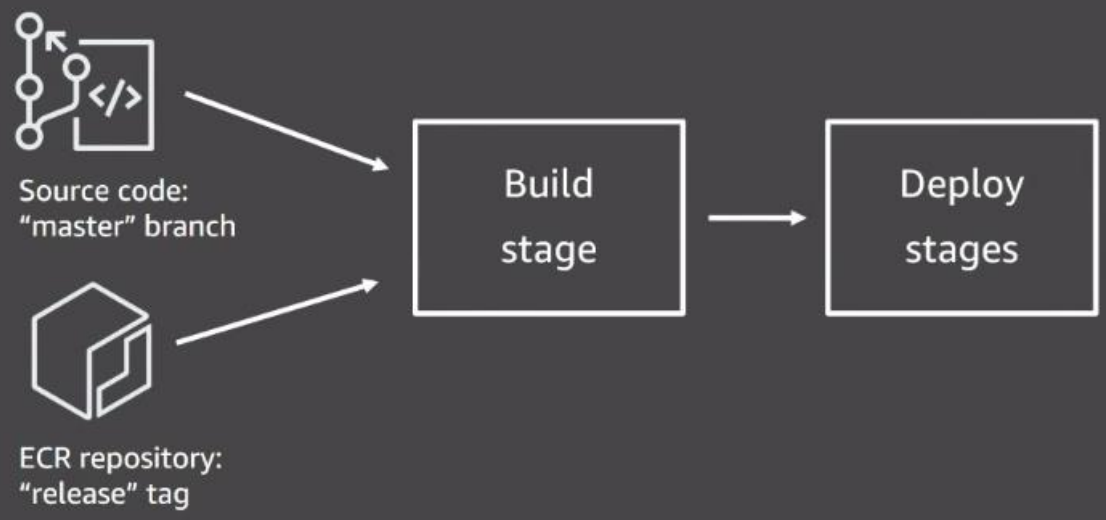
Automatically kick off release and pull latest source code

Pick branch	Pick object or folder
AWS CodeCommit	Amazon Simple Storage Service (Amazon S3)
GitHub	

We can now also trigger a build off a new Docker image in ECR as a pipeline source



AWS CodePipeline: ECR source action



AWS CodePipeline: Supported triggers

Automatically kick off release

Amazon CloudWatch Events

- Scheduled (nightly release)
- AWS Health events (Fargate platform retirement)

Available in CloudWatch Events console, API, SDK, CLI, and AWS CloudFormation

Webhooks

- DockerHub
- Quay
- Artifactory

Available in CodePipeline API, SDK, CLI, and CloudFormation

AWS CodeBuild

- Fully managed build service that compiles source code, runs tests, and produces software packages
- Scales continuously and processes multiple builds concurrently
- No build servers to manage
- Pay by the minute, only for the compute resources you use
- Monitor builds through CloudWatch Events

AWS CodeBuild

- Each build runs in a new Docker container for a consistent, immutable environment
- Docker and AWS CLI are installed in every official CodeBuild image
- Provide custom build environments suited to your needs through the use of Docker images

AWS CodeBuild: Lambda buildspec

```
version: 0.2

phases:
  build:
    commands:
      - npm ci
      - npm test
      - >
          aws cloudformation package
                   --template-file template.yml
                      --output-template template-output.yml
                      --s3_bucket $BUCKET

artifacts:
  type: zip
  files:
    - template-output.yml
```

AWS CodeBuild: Docker buildspec

```
version: 0.2

phases:
  build:
    commands:
      - $(aws ecr get-login --no-include-email)
      - docker build -t $IMAGE_REPO_NAME:$IMAGE_TAG .
      - docker tag $IMAGE_REPO_NAME:$IMAGE_TAG $ECR_REPO:$IMAGE_TAG
      - docker push $ECR_REPO:$IMAGE_TAG
```

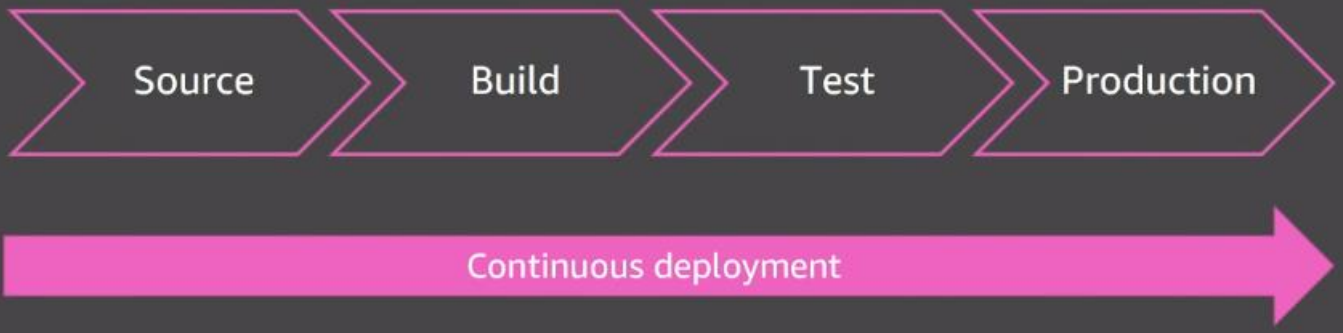
Continuous integration goals

1. Automatically kick off a new release when new code is checked in
2. Build and test code in a consistent, repeatable environment
3. Continually have an artifact ready for deployment
4. Continually close feedback loop when build fails

Pillars of releasing modern applications

Continuous integration | Continuous deployment | Infrastructure as code

Continuous deployment goals



Continuous deployment goals

1. Automatically deploy new changes to staging environments for testing
2. Deploy to production safely without impacting customers
3. Deliver to customers faster: Increase deployment frequency, and reduce change lead time and change failure rate

AWS CodeDeploy

- Automates code deployments to any instance and Lambda
- Handles the complexity of updating your applications
- Avoid downtime during application deployment
- Roll back automatically if failure detected
- Deploy to Amazon EC2, Lambda, or on-premises servers

CodeDeploy-Lambda deployments

- Shifts traffic using Lambda function weighted aliases
- Choose canary ("shift 10% of traffic for 10 minutes, then shift rest") or linear ("shift 10% more traffic every 10 minutes")
- Validation "hooks" enable testing at each stage of the deployment
- Fast rollback in seconds if case of hook failure or CloudWatch alarms
- Monitor deployment status and history via console, API, Amazon Simple Notification Service (Amazon SNS) notifications, and CloudWatch Events

CodeDeploy-Lambda deployments

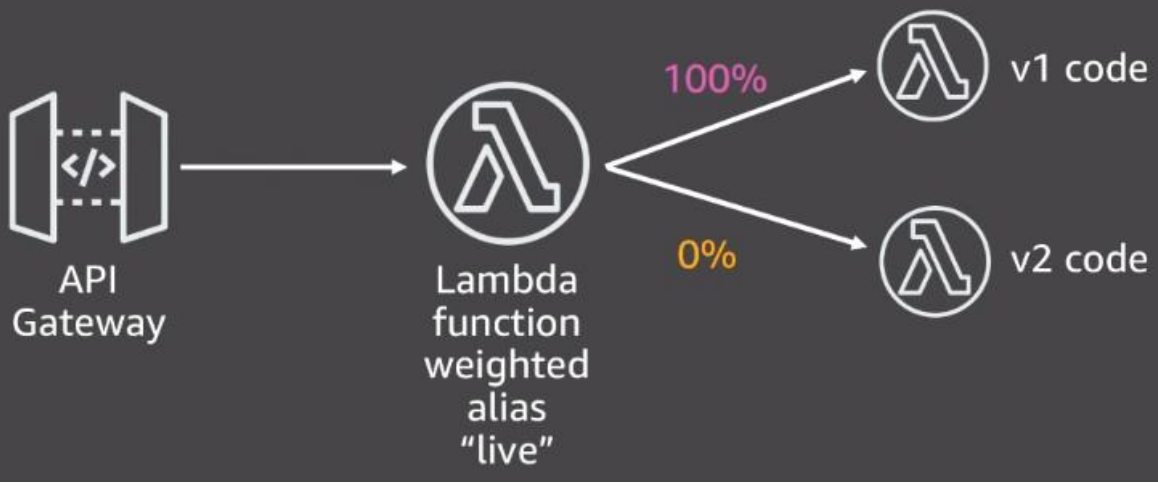
Enable in your serverless application template

```
Resources:
  GetFunction:
    Type: AWS::Serverless::Function
    Properties:
      DeploymentPreference:
        Type: Canary10Percent10Minutes
        Alarms:
          - !Ref ErrorsAlarm
        Hooks:
          PreTraffic: !Ref PreTrafficHook
```

CodeDeploy-Lambda canary deployment

CodeDeploy-Lambda canary deployment

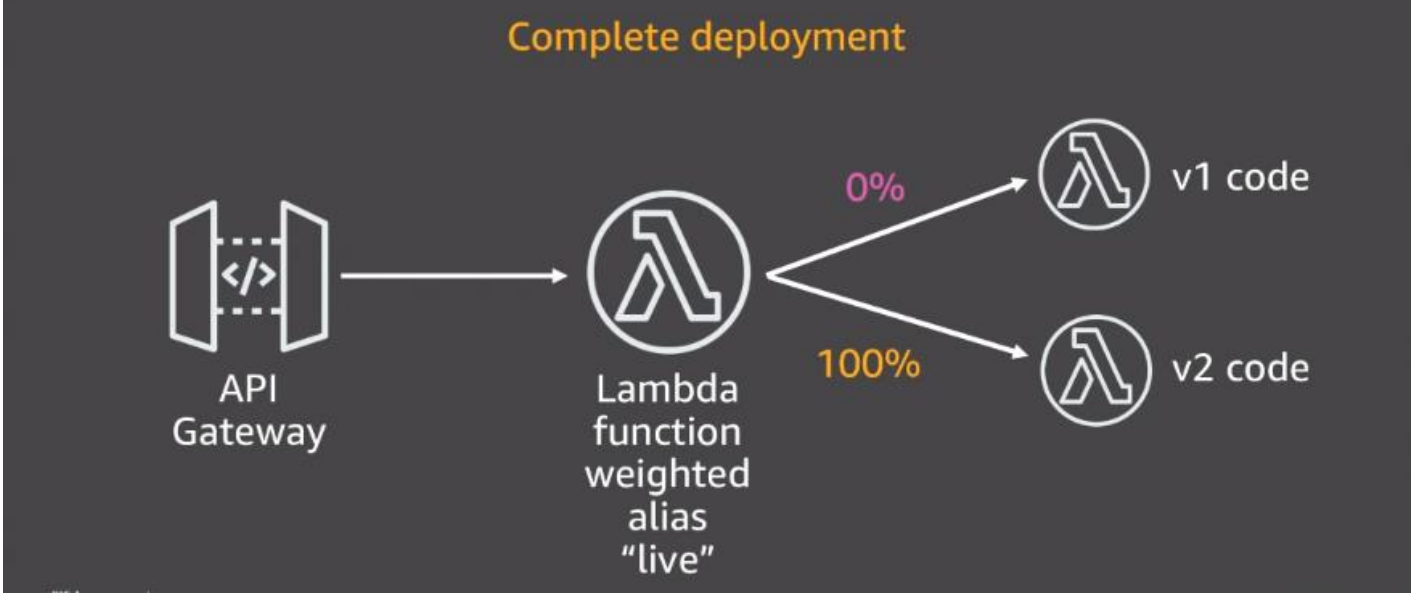
Run hook against v2 code before it receives traffic



CodeDeploy-Lambda canary deployment

Wait for 10 minutes, roll back in case of alarm

CodeDeploy-Lambda canary deployment



AWS CodeDeploy

New

AWS CodeDeploy now automates blue-green deployments to AWS Fargate and Amazon Elastic Container Service (ECS)

CodeDeploy-ECS blue-green deployments

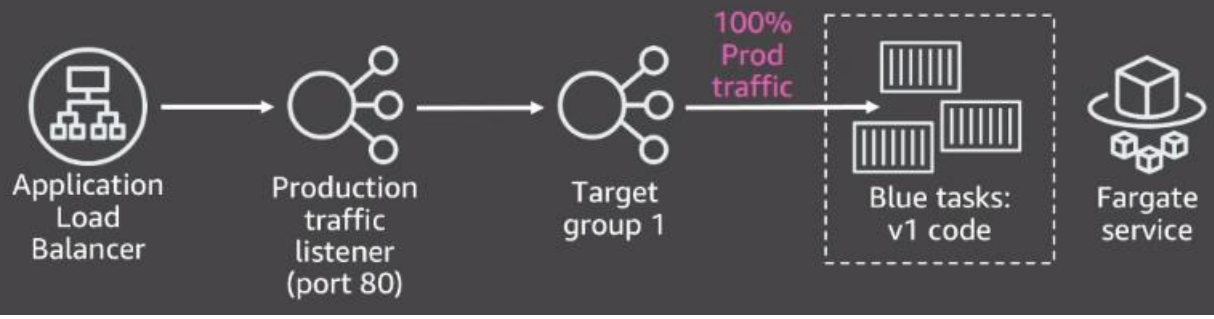
- Provisions "green" tasks, then flips traffic at the load balancer
- Validation "hooks" enable testing at each stage of the deployment
- Fast rollback to "blue" tasks in seconds if case of hook failure or CloudWatch alarms
- Monitor deployment status and history via console, API, Amazon SNS notifications, and CloudWatch Events
- Use "CodeDeploy-ECS" deploy action in CodePipeline or "aws ecs deploy" command in Jenkins

CodeDeploy-ECS appspec

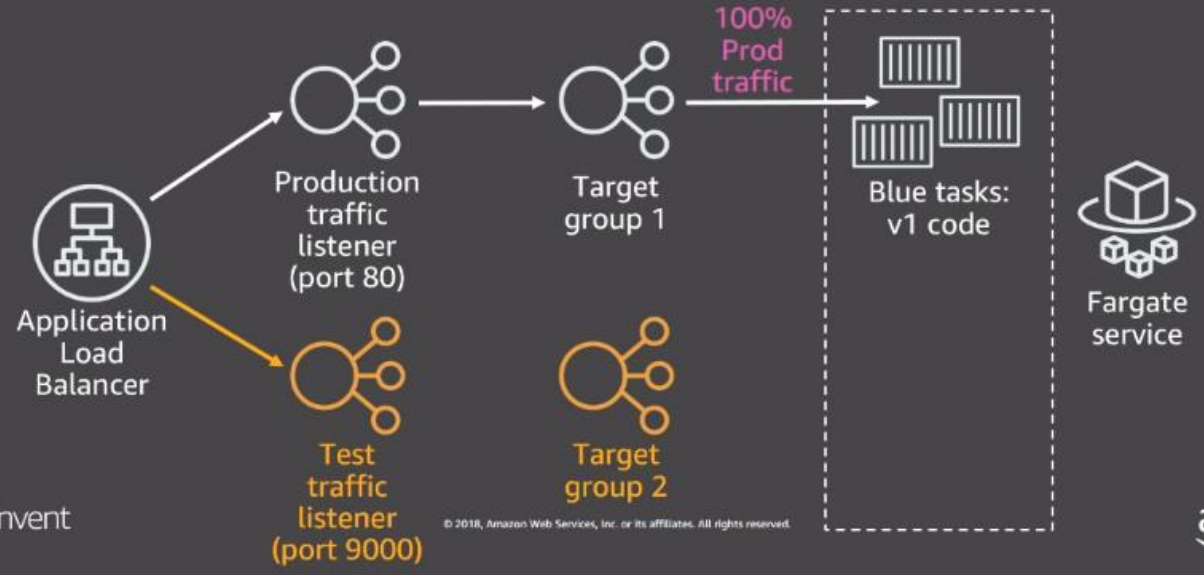
```
version: 1.0
Resources:
    - TargetService:
        Type: AWS::ECS::Service
        Properties:
            - TaskDefinition: "my_task_definition:8"
              LoadBalancerInfos:
                    - ContainerName: "SampleApp"
                      ContainerPort: 80

Hooks:
    - BeforeInstall: "LambdaFunctionToExecuteAnythingBeforeNewRevisionInstalltion"
    - AfterInstall: "LambdaFunctionToExecuteAnythingAfterNewRevisionInstallation"
    - AfterAllowTestTraffic: "LambdaFunctionToValidateAfterTestTrafficShift"
    - BeforeAllowTraffic: "LambdaFunctionToValidateBeforeTrafficShift"
    - AfterAllowTraffic: "LambdaFunctionToValidateAfterTrafficShift"
```

CodeDeploy-ECS blue-green deployment

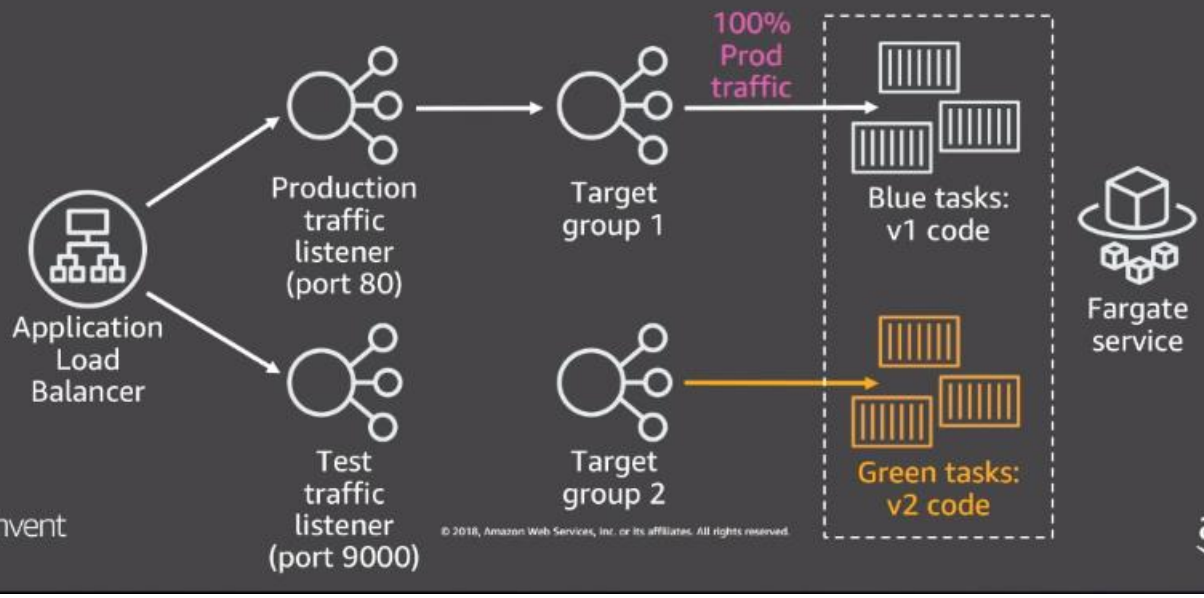


CodeDeploy-ECS blue-green deployment



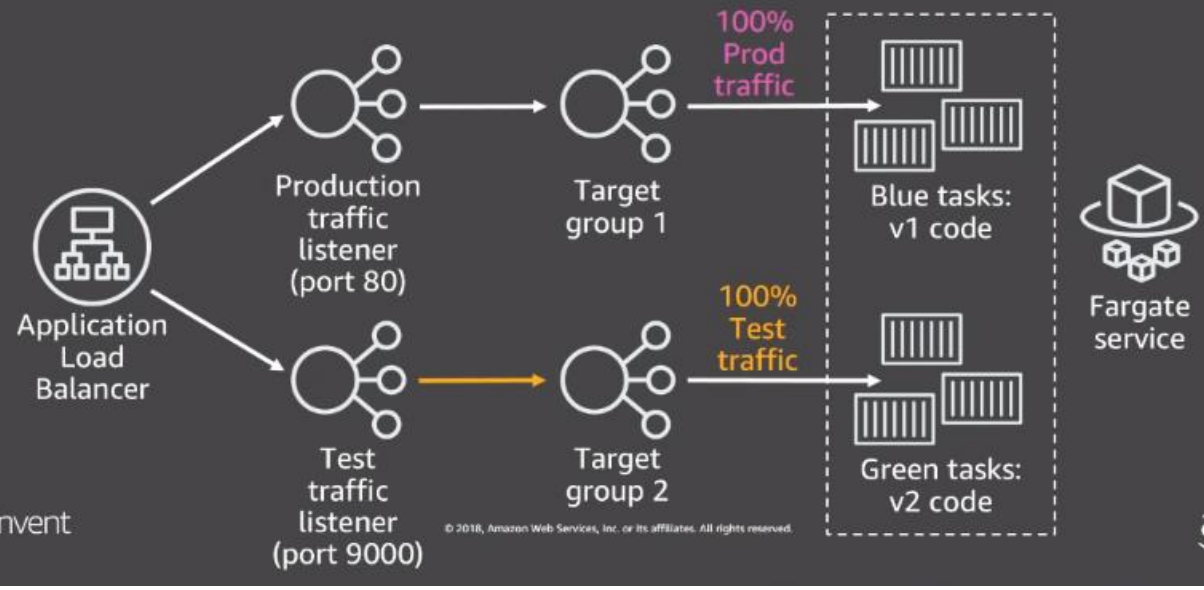
CodeDeploy-ECS blue-green deployment

Provision green tasks



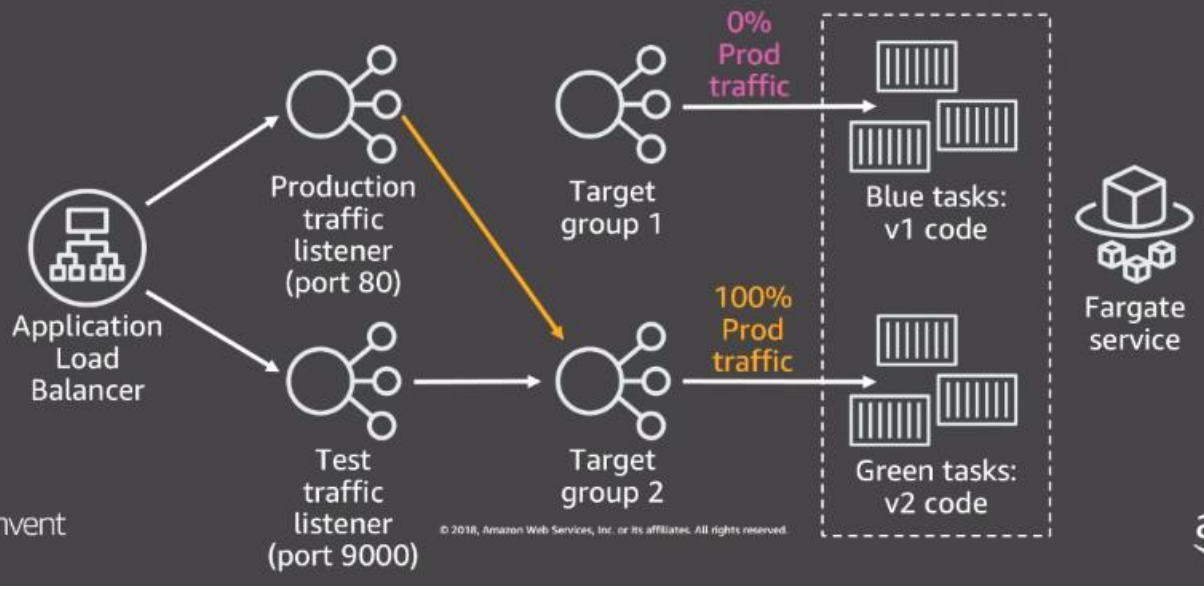
CodeDeploy-ECS blue-green deployment

Run hook against test endpoint before green tasks receive prod traffic



CodeDeploy-ECS blue-green deployment

Flip traffic to green tasks, rollback in case of alarm

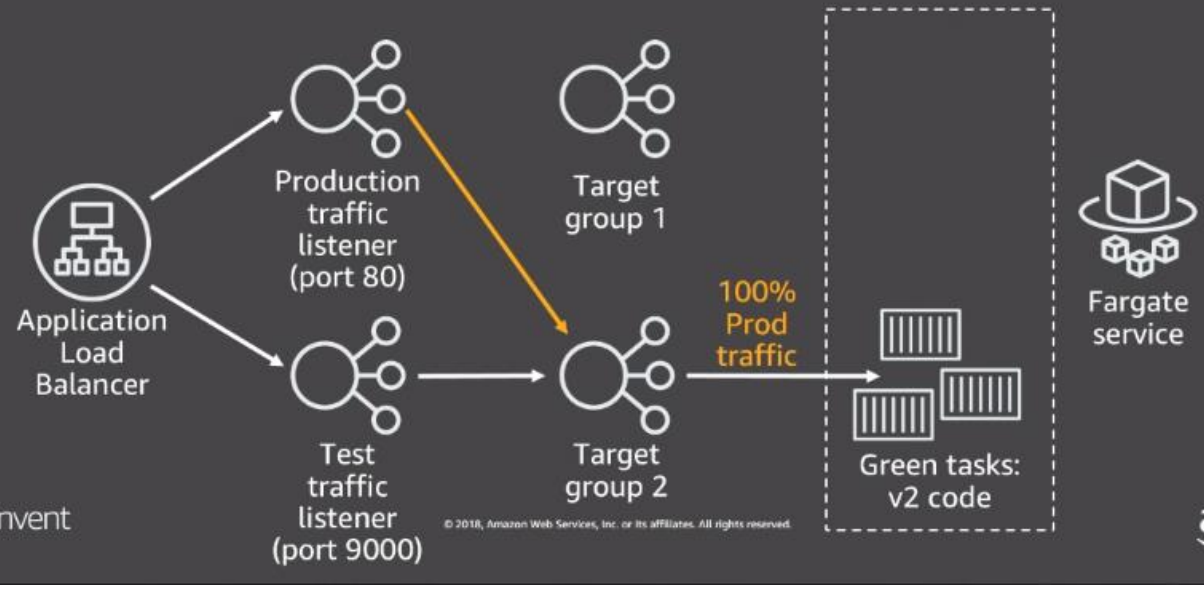


re:Invent

aws

CodeDeploy-ECS blue-green deployment

Drain blue tasks



re:Invent

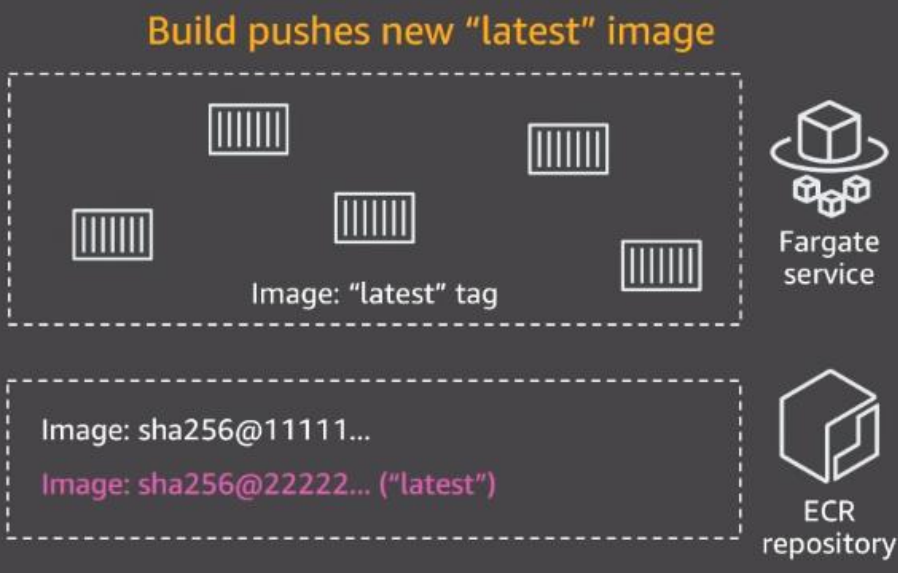
aws

Container image tagging for deployments

- Docker tags are resolved when each container starts, not just during deployments
- Deploying “latest” or “prod” can result in untested code in production after a scale-out event
- Use unique “immutable” tags for deployments

Container image tagging for deployments

Container image tagging for deployments



Container image tagging for deployments

Container image tagging for deployments

Deploy using immutable tags

```
{
    "name": "sample-app",
    "image": "amazon/amazon-ecs-
        sample@sha256:3e39d933b1d948c92309bb583b5a1f3d28f0119e1551ca1fe538ba414a41af48d"
}
```

SHA256 Digest

```
{
    "name": "sample-app",
    "image": "amazon/amazon-ecs-sample:build-b2085490-359f-4eaf-8970-6d1e26c354f0"
}
```

Build ID

Container image tagging for deployments

Compute immutable tags during build

SHA256 Digest

```
export IMAGE_URI=`docker inspect --format='{{index .RepoDigests 0}}' my_image:$IMAGE_TAG

Example Result:
amazon/amazon-ecs-sample@sha256:3e39d933b...
```

Build ID

```
export IMAGE_TAG=build-`echo $CODEBUILD_BUILD_ID | awk -F":" '{print $2}'`

Example Result:
build-b2085490-359f-4eaf-8970-6d1e26c354f0
```

Container image tagging for deployments

Container image tagging for deployments

Build pushes new image tagged with new build ID

Container image tagging for deployments

Service scales up, launching new tasks

Container image tagging for deployments

Deployment updates service's task definition, replacing tasks

Continuous deployment goals

1. Automatically deploy new changes to staging environments for testing
2. Deploy to production safely without impacting customers
3. Deliver to customers faster: Increase deployment frequency, and reduce change lead time and change failure rate

Pillars of releasing modern applications

Continuous integration | **Continuous deployment** | **Infrastructure as code**

Infrastructure as code goals

Infrastructure as code goals

Infrastructure as code

1. Make infrastructure changes repeatable and predictable
2. Release infrastructure changes using the same tools as code changes
3. Replicate production environment in a staging environment to enable continuous testing

Continuous testing with infrastructure as code

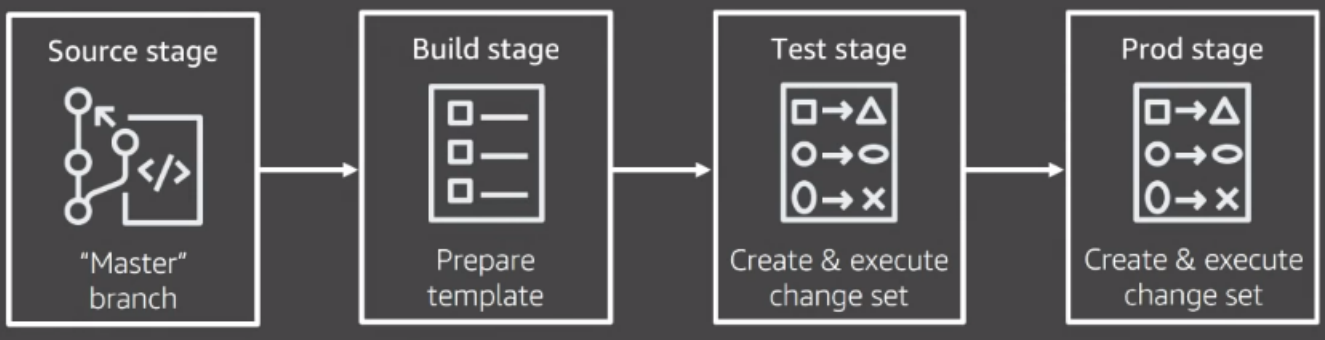
Validate an artifact
(Build stage)

- Unit tests
- Static analysis
- Mocked dependencies and environments
- Vulnerability image scans

Validate an environment
(Test stages)

- Integration tests against real dependencies and real environments
- Load testing
- Penetration testing
- Monitoring to test impact of deployments on environment

Release infrastructure-as-code



Model function environments with AWS Serverless Application Model (SAM)

- Open source framework for building serverless applications on AWS
- Shorthand syntax to express functions, APIs, databases, and event source mappings
- Transforms and expands SAM syntax into AWS CloudFormation syntax on deployment
- Supports all AWS CloudFormation resource types

https://aws.amazon.com/serverless/sam/

SAM template

```
AWSTemplateFormatVersion: '2010-09-09'
Transform: AWS::Serverless-2016-10-31
Resources:
  GetFunction:
    Type: AWS::Serverless::Function
    Properties:
      Handler: index.get
      Runtime: nodejs6.10
      CodeUri: src/
      Policies: AmazonDynamoDBReadOnlyAccess
      Events:
        GetResource:
          Type: Api
          Properties:
            Path: /resource/{resourceId}
            Method: get
  Table:
    Type: AWS::Serverless::SimpleTable
```

Shorthand syntax to express functions, tables, and events

SAM template

```
AWSTemplateFormatVersion: '2010-09-09'
Transform: AWS::Serverless-2016-10-31
Resources:
  GetFunction:
    Type: AWS::Serverless::Function
    Properties:
      Handler: index.get
      Runtime: nodejs6.10
      CodeUri: src/
      Policies: AmazonDynamoDBReadOnlyAccess
      Events:
        GetResource:
          Type: Api
          Properties:
            Path: /resource/{resourceId}
            Method: get
  Table:
    Type: AWS::Serverless::SimpleTable
```

Just 18 lines of SAM syntax creates many resources:

- AWS Lambda function
- Amazon API Gateway
- Amazon DynamoDB table
- AWS Identity and Access Management (IAM) roles

Use SAM CLI to package and deploy SAM templates

```
pip install --user aws-sam-cli

sam init

sam build

sam package

sam deploy
```

CodePipeline

Use CloudFormation deployment actions with any SAM application

Jenkins

Use SAM CLI plugin

Model container environments with AWS Cloud Development Kit (CDK)

AWS CDK

Developer Preview

- Open source framework to define cloud infrastructure in Typescript
- Provides library of higher-level resource types ("construct" classes) that have AWS best practices built in by default, packaged as npm modules
- Provisions resources with CloudFormation
- Supports all CloudFormation resource types

https://awslabs.github.io/aws-cdk

CDK template

```
import ec2 = require('@aws-cdk/aws-ec2');
import ecs = require('@aws-cdk/aws-ecs');
import cdk = require('@aws-cdk/cdk');

class BonjourFargate extends cdk.Stack {
  constructor(parent: cdk.App, name: string, props?: cdk.StackProps) {
    super(parent, name, props);

    const vpc = new ec2.VpcNetwork(this, 'MyVpc', { maxAZs: 2 });
    const cluster = new ecs.Cluster(this, 'Cluster', { vpc });

    new ecs.LoadBalancedFargateService(
      this, "FargateService", {
        cluster,
        image: ecs.DockerHub.image("amazon/amazon-ecs-sample"),
    });
  }
}

const app = new cdk.App();
new BonjourFargate(app, 'Bonjour');
app.run();
```

High-level VPC class includes VPC, subnets, security groups, internet gateway, NAT gateways, and route tables

aws re:Invent

aws

This is an example CDK template for a Fargate service with a LB in front of it.

CDK template

```
import ec2 = require('@aws-cdk/aws-ec2');
import ecs = require('@aws-cdk/aws-ecs');
import cdk = require('@aws-cdk/cdk');

class BonjourFargate extends cdk.Stack {
  constructor(parent: cdk.App, name: string, props?: cdk.StackProps) {
    super(parent, name, props);

    const vpc = new ec2.VpcNetwork(this, 'MyVpc', { maxAZs: 2 });
    const cluster = new ecs.Cluster(this, 'Cluster', { vpc });

    new ecs.LoadBalancedFargateService(
      this, "FargateService", {
        cluster,
        image: ecs.DockerHub.image("amazon/amazon-ecs-sample"),
    });
  }
}

const app = new cdk.App();
new BonjourFargate(app, 'Bonjour');
app.run();
```

High-level Fargate class includes ECS service, ECS task definition, Application Load Balancer, listener rule, target group, and optionally Amazon Route 53 alias record

CDK template

```
import ec2 = require('@aws-cdk/aws-ec2');
import ecs = require('@aws-cdk/aws-ecs');
import cdk = require('@aws-cdk/cdk');

class BonjourFargate extends cdk.Stack {
  constructor(parent: cdk.App, name: string, props?: cdk.StackProps) {
    super(parent, name, props);

    const vpc = new ec2.VpcNetwork(this, 'MyVpc', { maxAZs: 2 });
    const cluster = new ecs.Cluster(this, 'Cluster', { vpc });

    new ecs.LoadBalancedFargateService(
      this, "FargateService", {
        cluster,
        image: ecs.DockerHub.image("amazon/amazon-ecs-sample"),
    });
  }
}

const app = new cdk.App();
new BonjourFargate(app, 'Bonjour');
app.run();
```

22 Lines of Typescript code generate over 400 lines of CloudFormation syntax

CDK template

```
applets:
  MyHelloWorldService:
    type: @aws-cdk/aws-ecs:LoadBalancedFargateServiceApplet
    properties:
      image: 'amazon/amazon-ecs-sample'
```

Applets use yaml syntax to provide inputs to the same CDK abstractions

This will also generate all the CF resources for you

Model pipelines with AWS CDK

- Minimize copy-and-paste by using object-oriented language
- Define microservice pipeline “shape” in one class, then re-use it across many pipelines
- CDK includes many high-level constructs for modeling a CodePipeline pipeline, including automatically configuring IAM role policies

You can also use infrastructure as code for your microservice pipelines using the CDK by defining a higher-level class for your pipeline that you can reuse.

CDK pipelines: Construct

```
export class MyMicroservicePipeline extends cdk.Construct {
    constructor(parent: cdk.Construct, name: string, props: MyMicroservicePipelineProps) {
        super(parent, name);

        const pipeline = new codepipeline.Pipeline(this, 'Pipeline', {
            pipelineName: props.serviceName,
        });

        const githubAccessToken = new cdk.SecretParameter(this, 'GitHubToken',
            { ssmParameter: 'GitHubToken' });
        new codepipeline.GitHubSourceAction(this, 'GitHubSource', {
            stage: pipeline.addStage('Source'),
            owner: 'myorg',
            repo: props.serviceName,
            oauthToken: githubAccessToken.value
        });
…
```

This is a truncated example of a pipeline class that we can use as code in our repo.

CDK pipelines: Stack

```
import cdk = require('@aws-cdk/cdk');
import { MyMicroservicePipeline } from './pipeline';

class MyMicroservicePipelinesStack extends cdk.Stack {
    constructor(parent: cdk.App, name: string, props?: cdk.StackProps) {
        super(parent, name, props);

        new MyMicroservicePipeline(this, 'Pipeline1', { 'serviceName': 'Microservice1' });
        new MyMicroservicePipeline(this, 'Pipeline2', { 'serviceName': 'Microservice2' });
        new MyMicroservicePipeline(this, 'Pipeline3', { 'serviceName': 'Microservice3' });
        new MyMicroservicePipeline(this, 'Pipeline4', { 'serviceName': 'Microservice4' });
    }
}

const app = new cdk.App();
new MyMicroservicePipelinesStack(app, 'MyMicroservicePipelines');
app.run();
```

You can then easily create new pipelines as above

Use CDK CLI to synthesize and deploy CDK templates

```
npm install -g aws-cdk

cdk init app --language typescript

cdk synth

cdk deploy
```

CodePipeline

Use CloudFormation deployment actions with any synthesized CDK application

Jenkins

Use CDK CLI

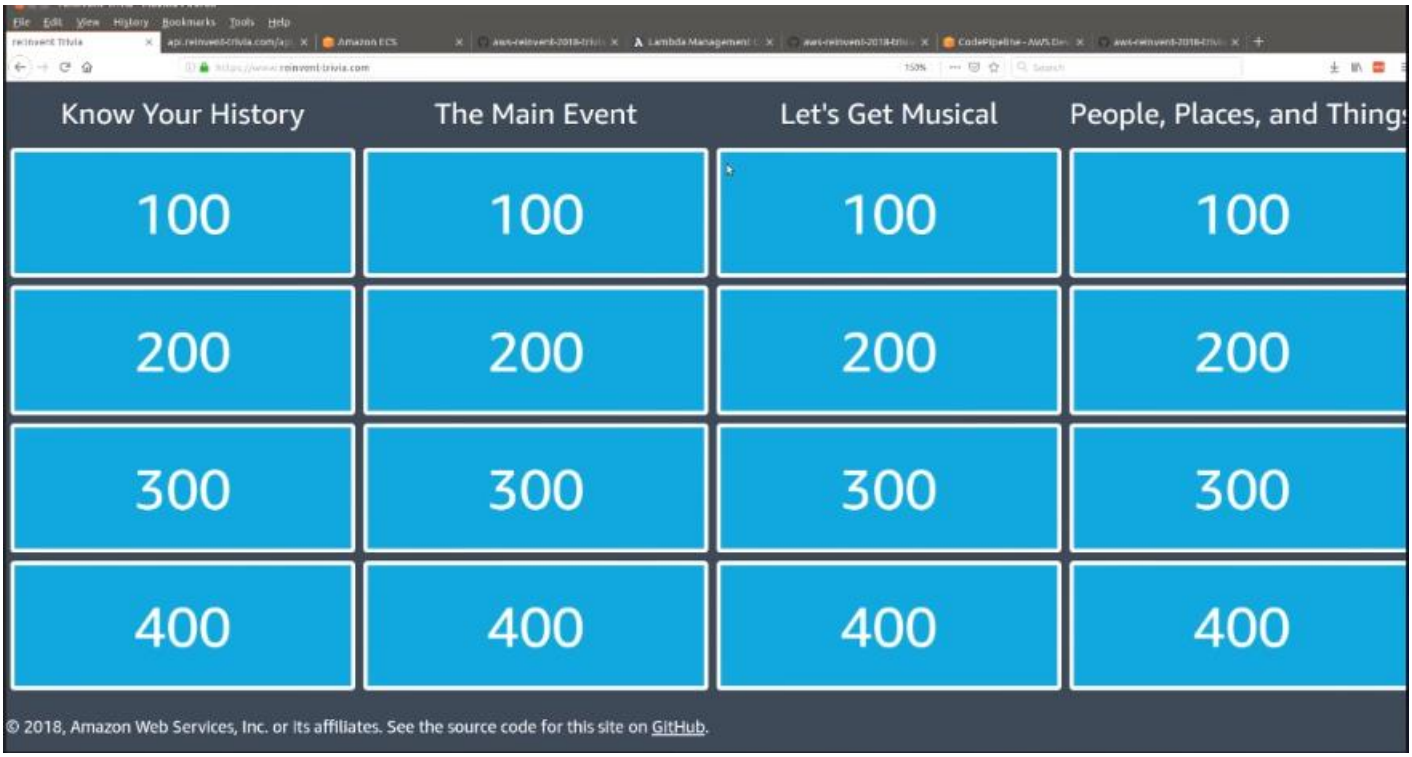
Infrastructure as code goals

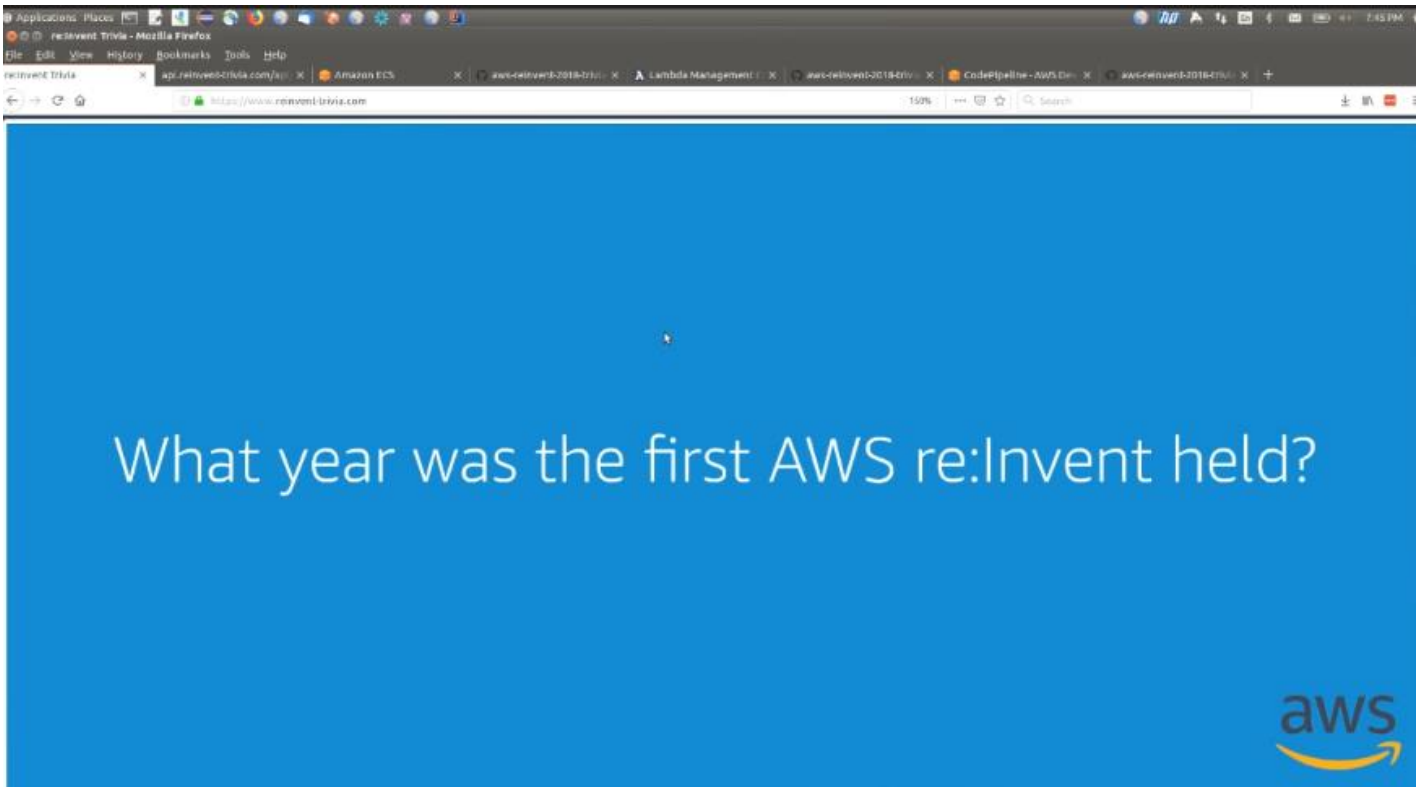
1. Make infrastructure changes repeatable and predictable
2. Release infrastructure changes using the same tools as code changes
3. Replicate production environment in a staging environment to enable continuous testing

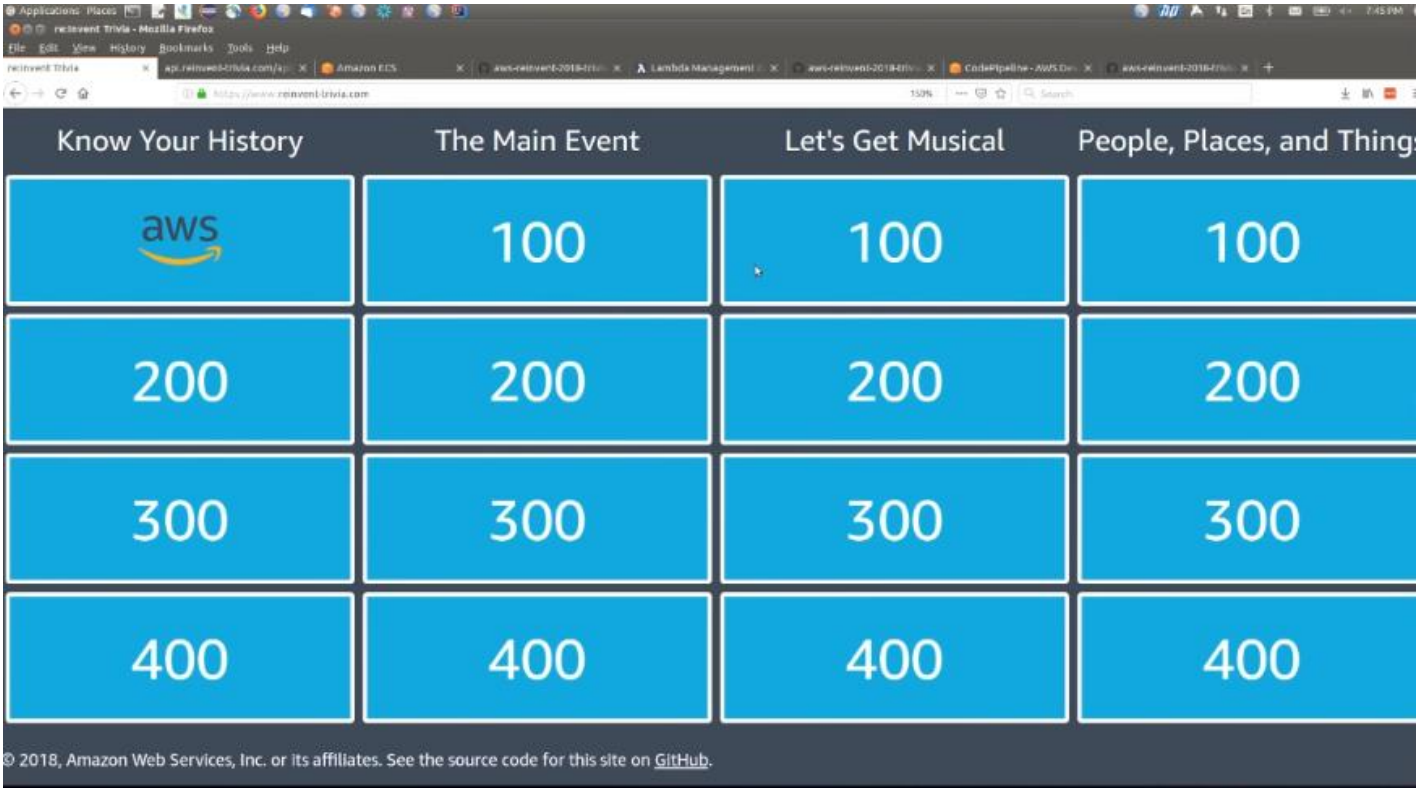
Pillars of releasing modern applications

Continuous integration	Continuous deployment	Infrastructure as code

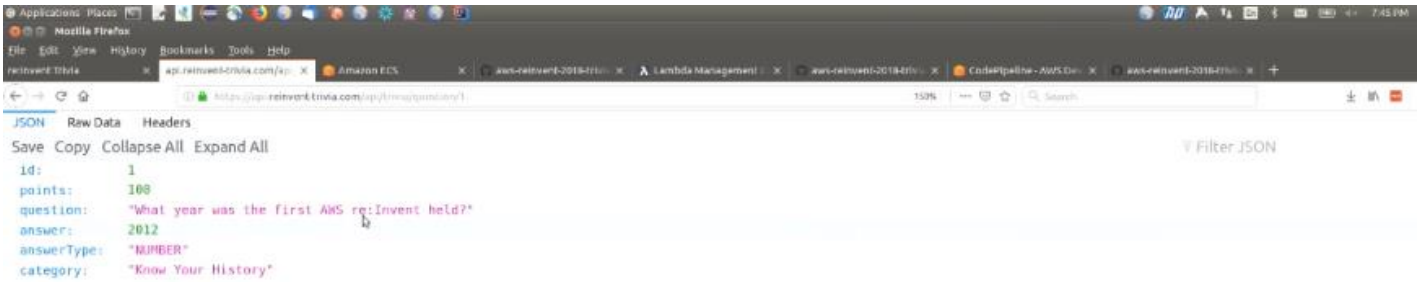
Demo



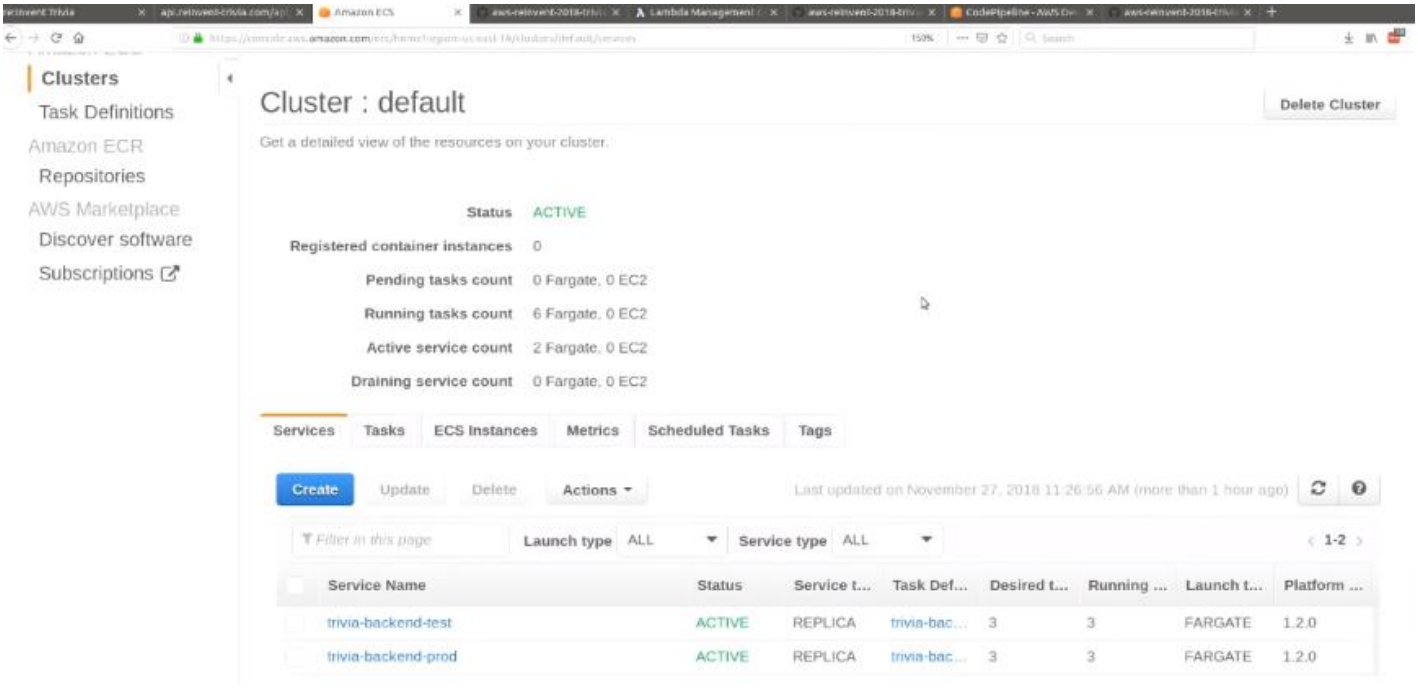


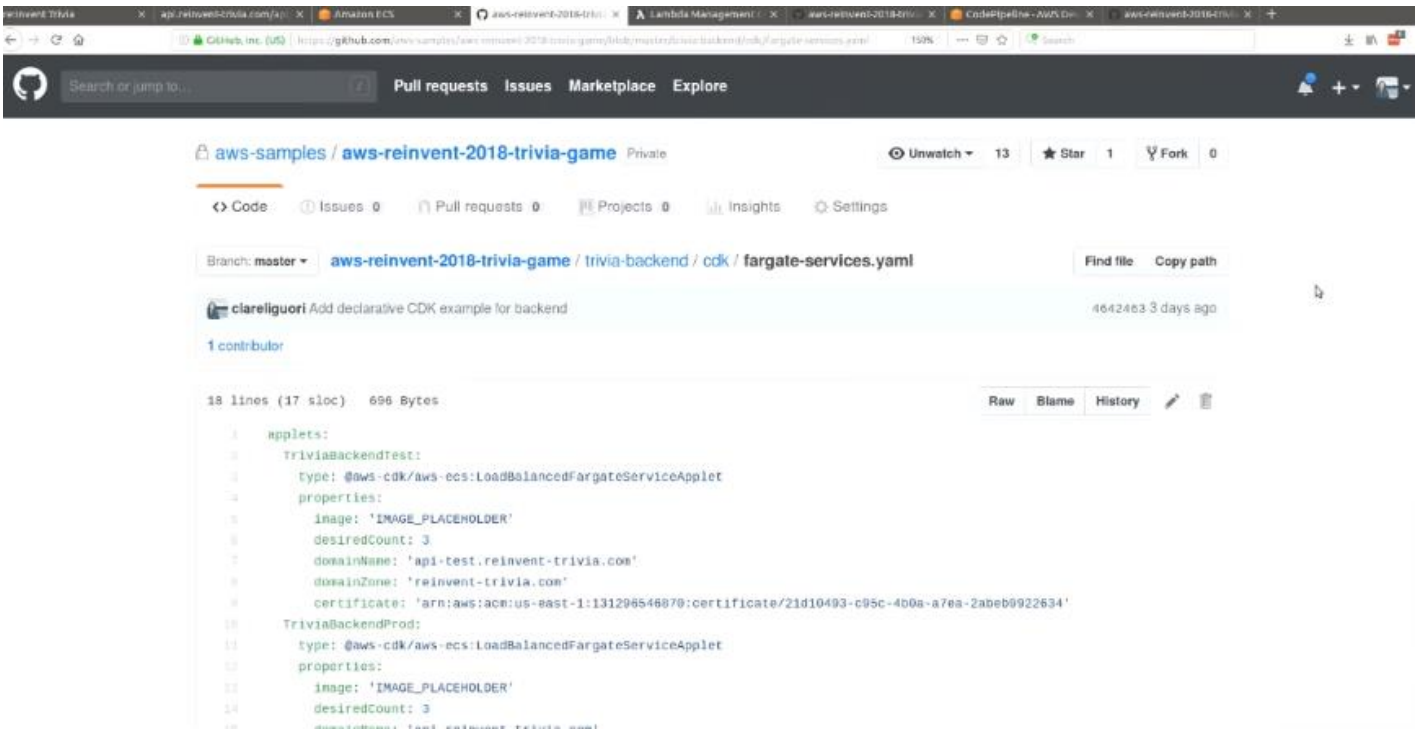


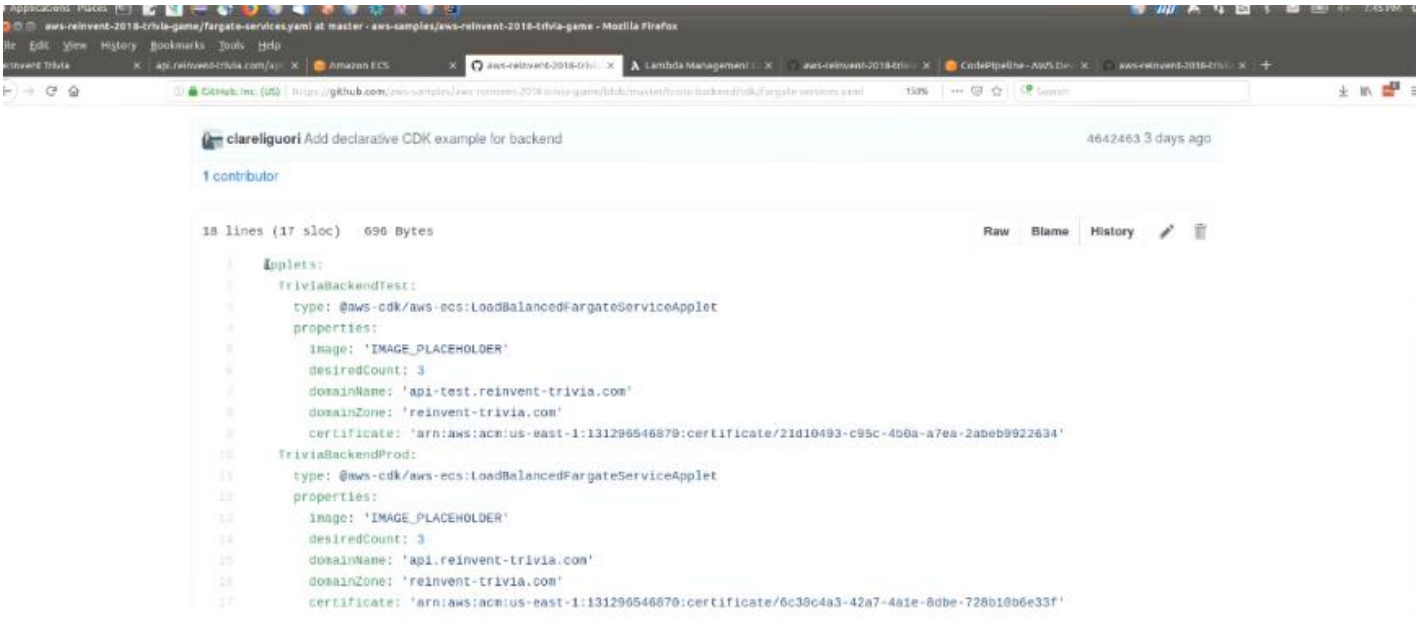
This is a static site that is backed by an API that serves up all of the trivia questions and answers. We have it packaged up as a container image being run in our Fargate service.



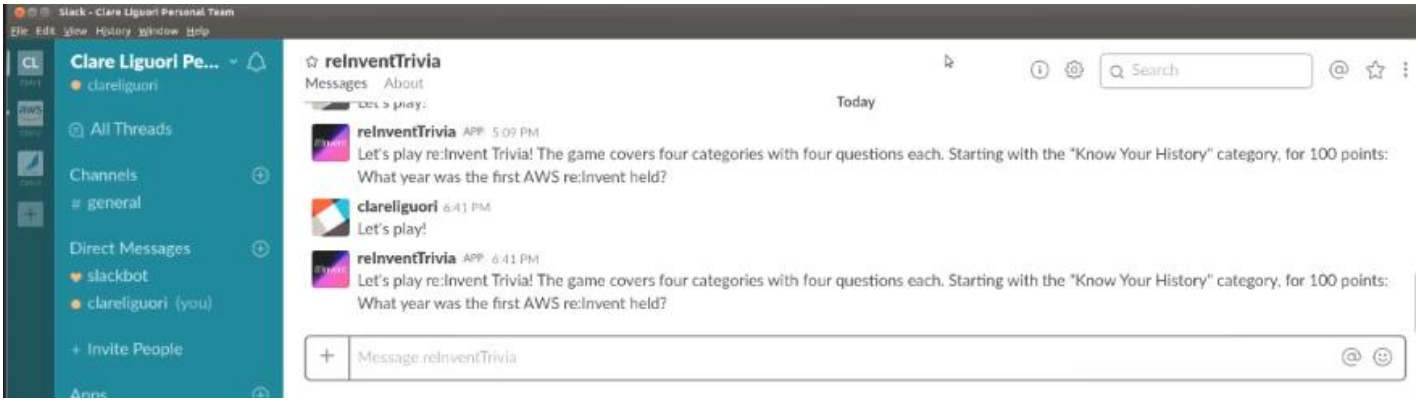
This is a sample data that we get from the Fargate API service to display in our app.



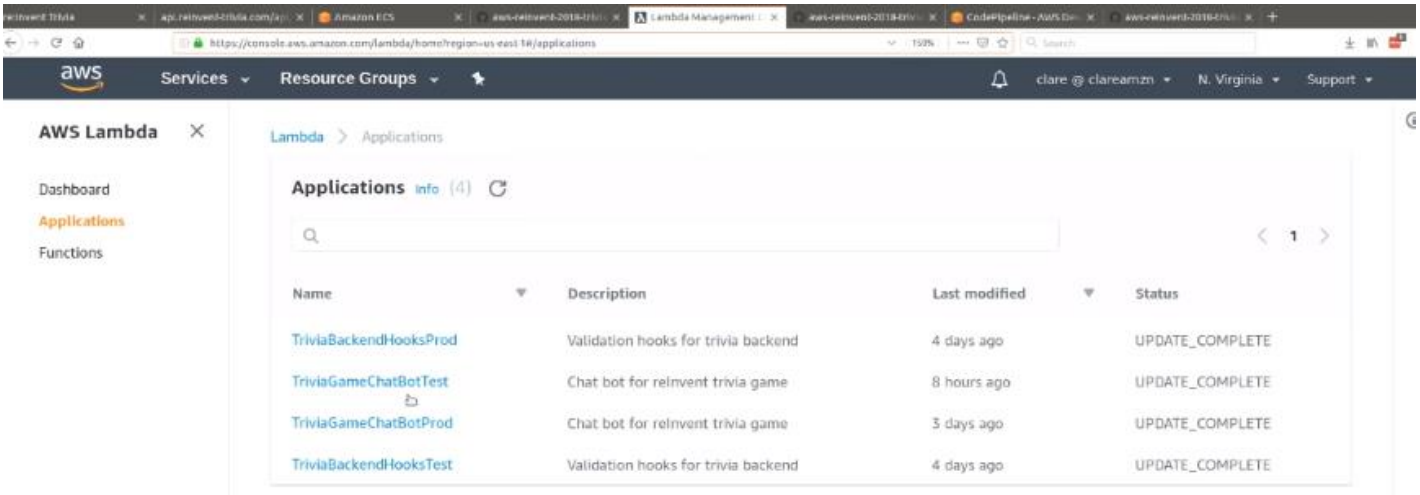


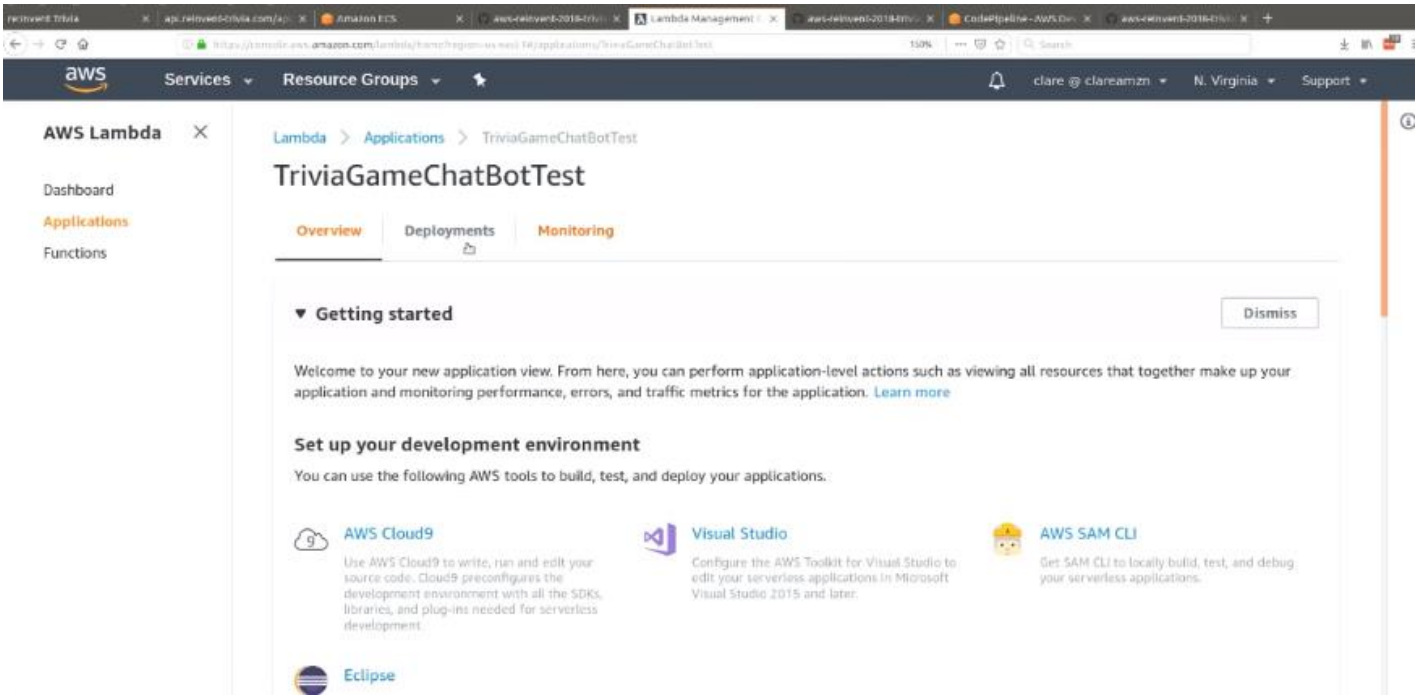


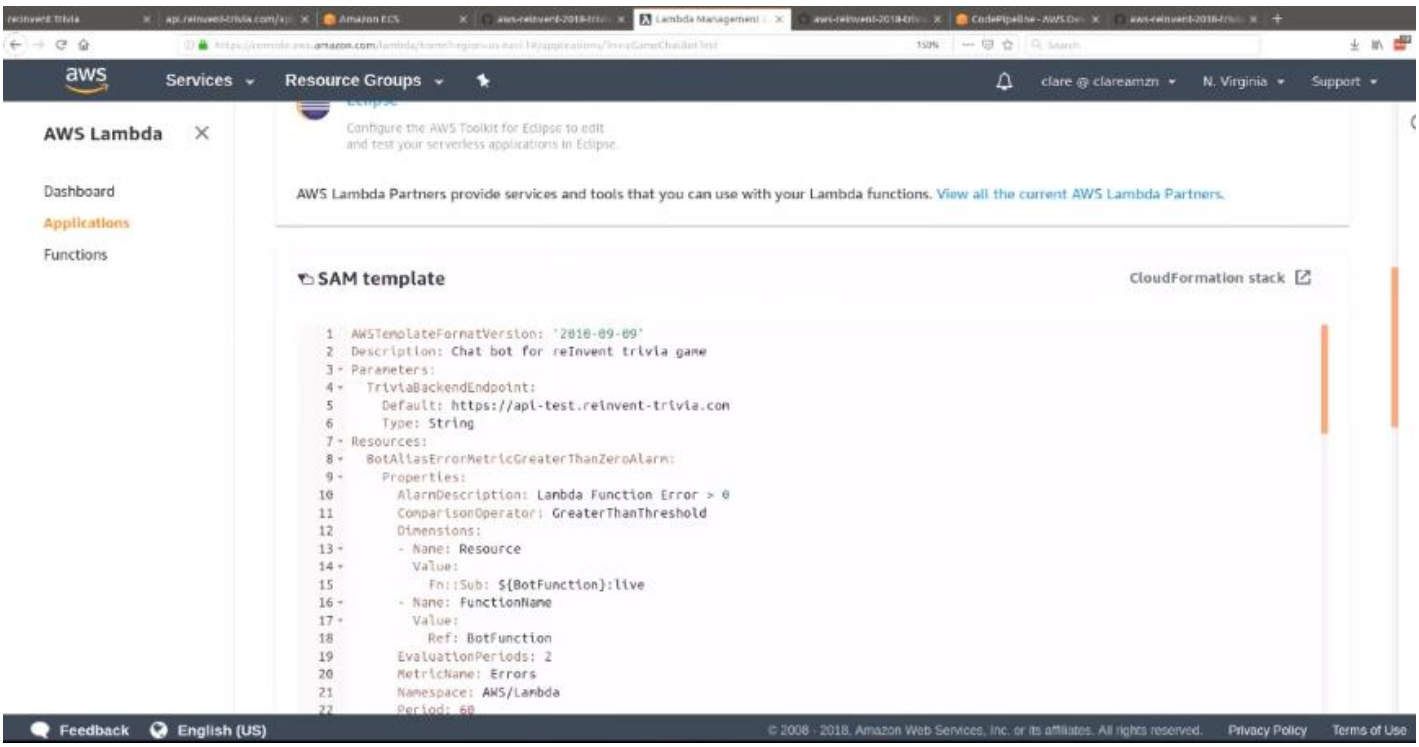
This is an example of the CDK applet that we used to define our build, test and production deployment stages

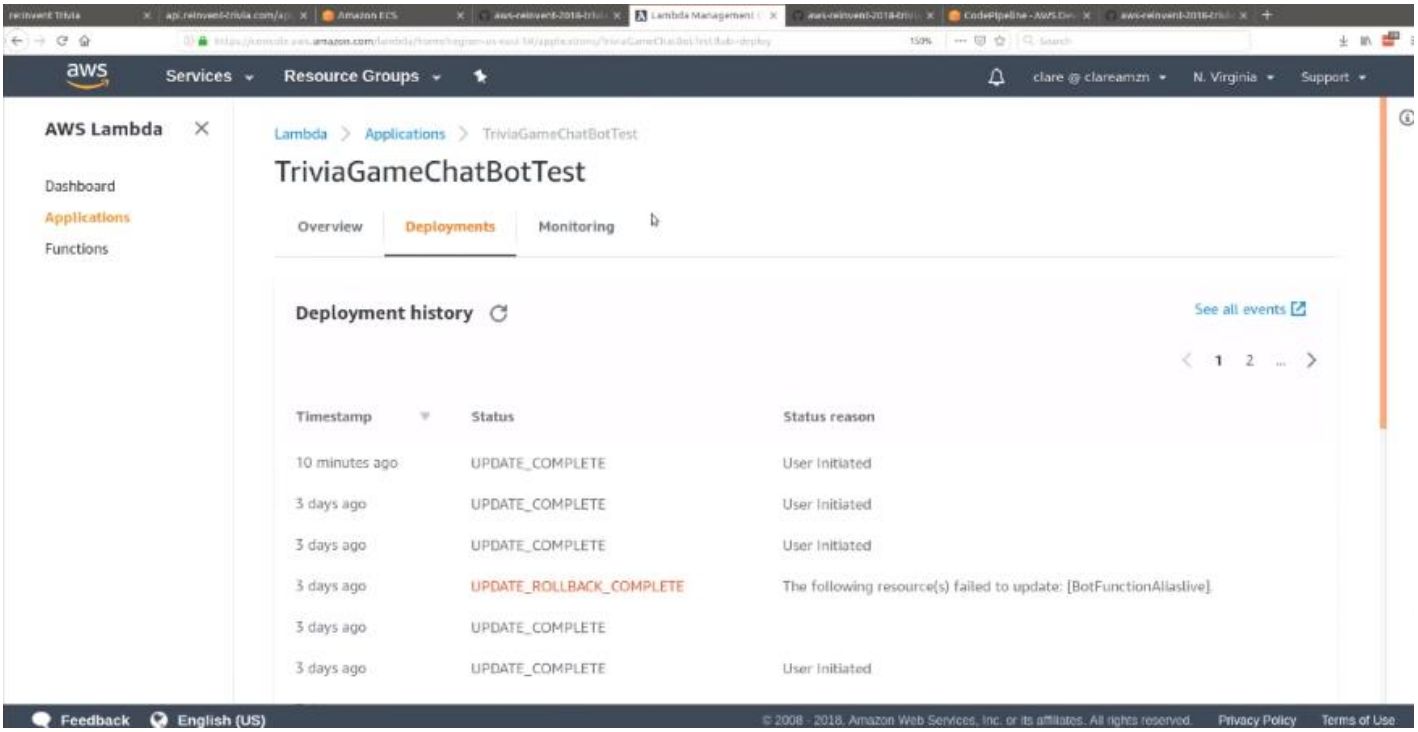


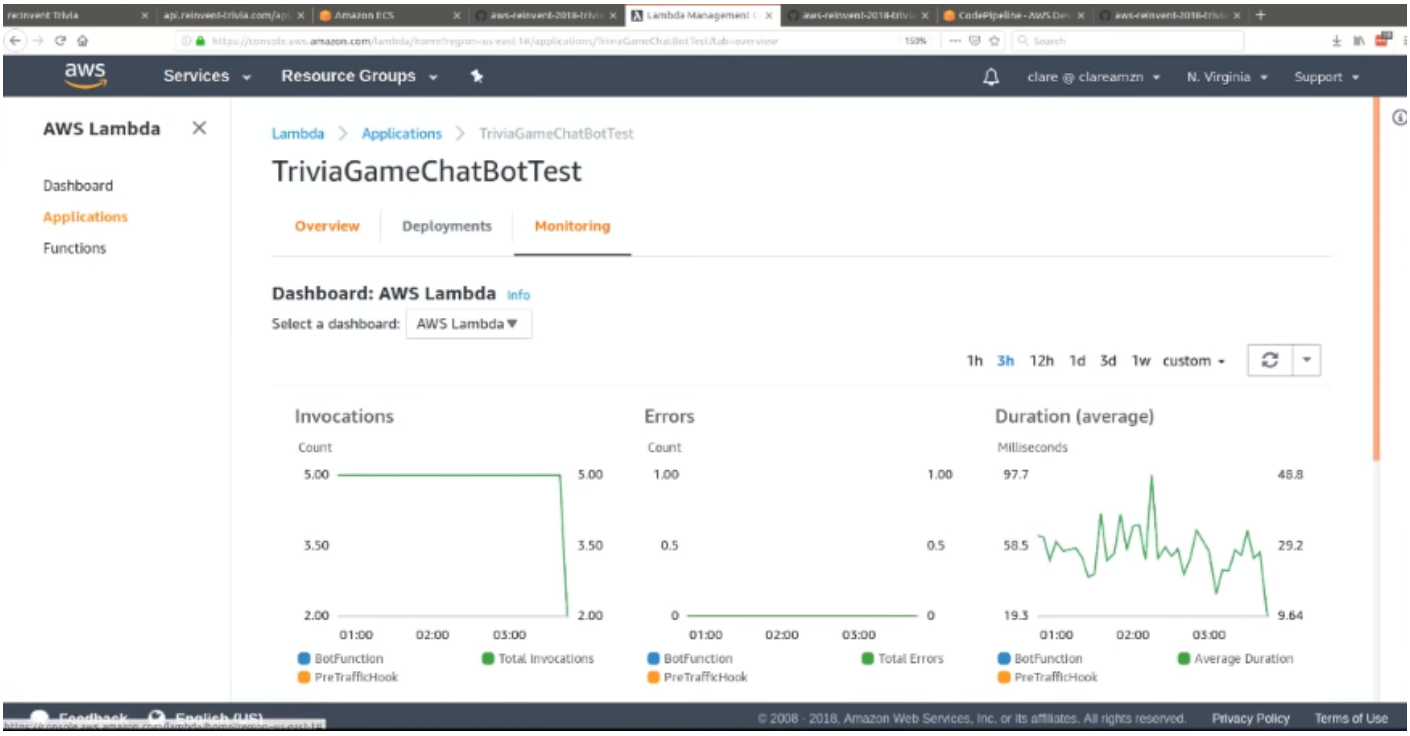
We also integrated with Amazon Lex to use in our Slack channel

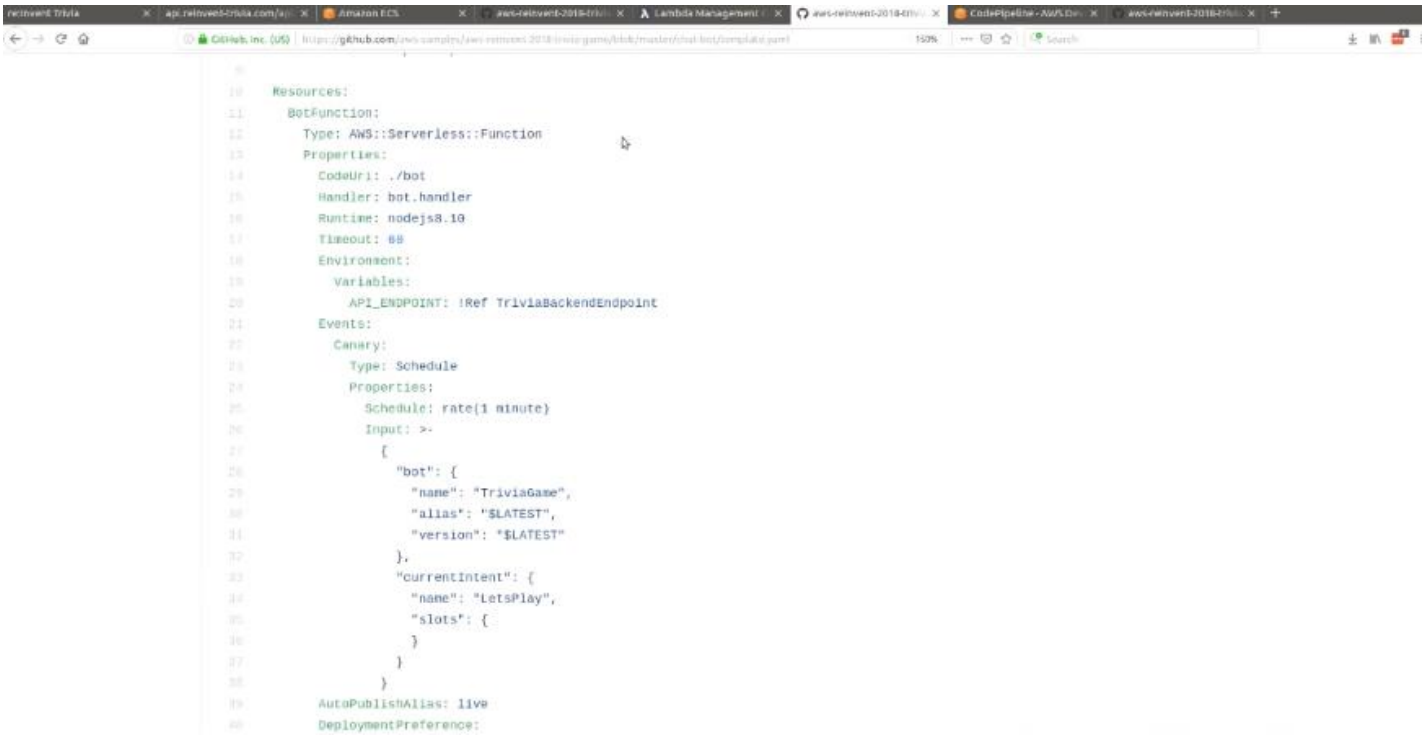



Services
Resource Groups
clare @ clareamzn
N. Virginia
Support
AWS Lambda
Dashboard
Applications
Functions
Lambda > Applications > TriviaGameChatBotTest
TriviaGameChatBotTest
Overview
Deployments
Monitoring
Getting started
Dismiss
Welcome to your new application view. From here, you can perform application-level actions such as viewing all resources that together make up your application and monitoring performance, errors, and traffic metrics for the application. Learn more
Set up your development environment
You can use the following AWS tools to build, test, and deploy your applications.
AWS Cloud9
Use AWS Cloud9 to write, run and edit your source code. Cloud9 preconfigures the development environment with all the SDKs, libraries, and plug-ins needed for serverless development.
Visual Studio
Configure the AWS Toolkit for Visual Studio to edit your serverless applications in Microsoft Visual Studio 2015 and later.
AWS SAM CLI
Get SAM CLI to locally build, test, and debug your serverless applications.
Eclipse

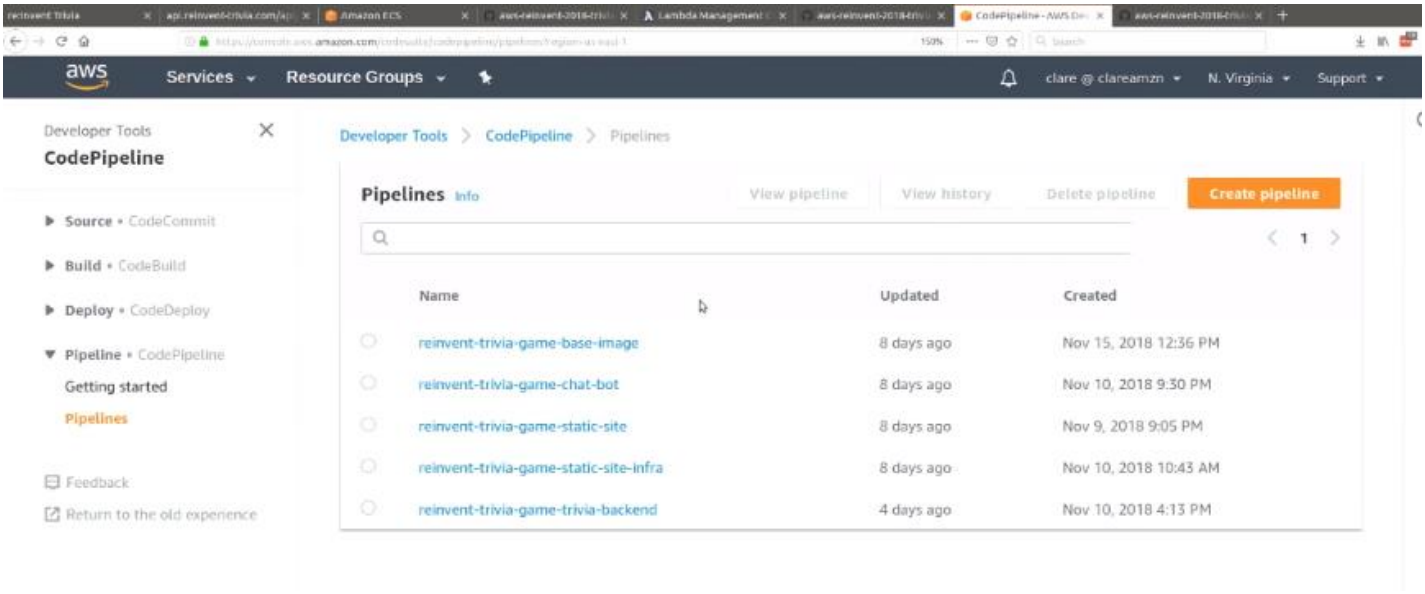

Services
Resource Groups
clare @ clareamzn
N. Virginia
Support
AWS Lambda
Dashboard
Applications
Functions
Configure the AWS Toolkit for Eclipse to edit and test your serverless applications in Eclipse.
AWS Lambda Partners provide services and tools that you can use with your Lambda functions. View all the current AWS Lambda Partners.
SAM template
CloudFormation stack
AWSTemplateFormatVersion: '2010-09-09'
Description: Chat bot for reInvent trivia game
Parameters:
TriviaBackendEndpoint:
Default: https://api-test.reinvent-trivia.com
Type: String
Resources:
BotAliasErrorMetricGreaterThanZeroAlarm:
Properties:
AlarmDescription: Lambda Function Error > 0
ComparisonOperator: GreaterThanThreshold
Dimensions:
- Name: Resource
Value:
Fn::Sub: ${BotFunction}:live
- Name: FunctionName
Value:
Ref: BotFunction
EvaluationPeriods: 2
MetricName: Errors
Namespace: AWS/Lambda
Period: 60
Feedback
English (US)
© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

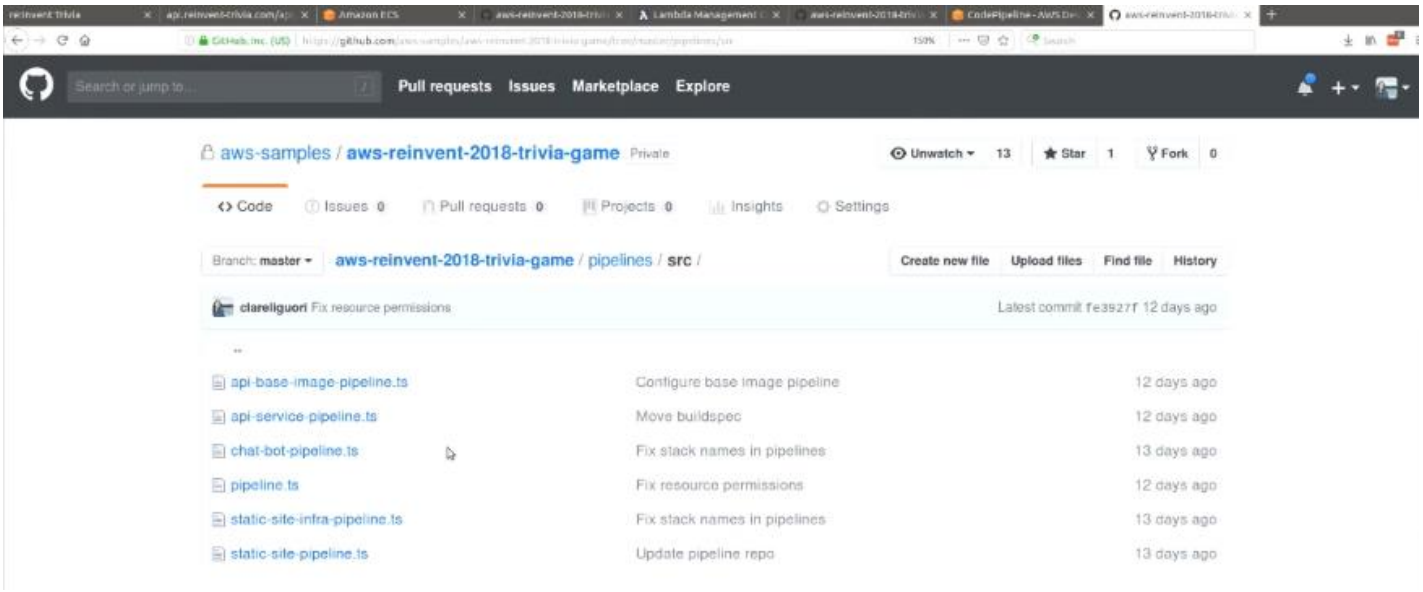

Services
Resource Groups
clare @ clareamzn
N. Virginia
Support
AWS Lambda
Dashboard
Applications
Functions
Lambda > Applications > TriviaGameChatBotTest
TriviaGameChatBotTest
Overview
Deployments
Monitoring
Deployment history
See all events
1 2 ...
Timestamp
Status
Status reason
10 minutes ago
UPDATE_COMPLETE
User Initiated
3 days ago
UPDATE_COMPLETE
User Initiated
3 days ago
UPDATE_COMPLETE
User Initiated
3 days ago
UPDATE_ROLLBACK_COMPLETE
The following resource(s) failed to update: [BotFunctionAliaslive].
3 days ago
UPDATE_COMPLETE
3 days ago
UPDATE_COMPLETE
User Initiated
Feedback
English (US)
© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

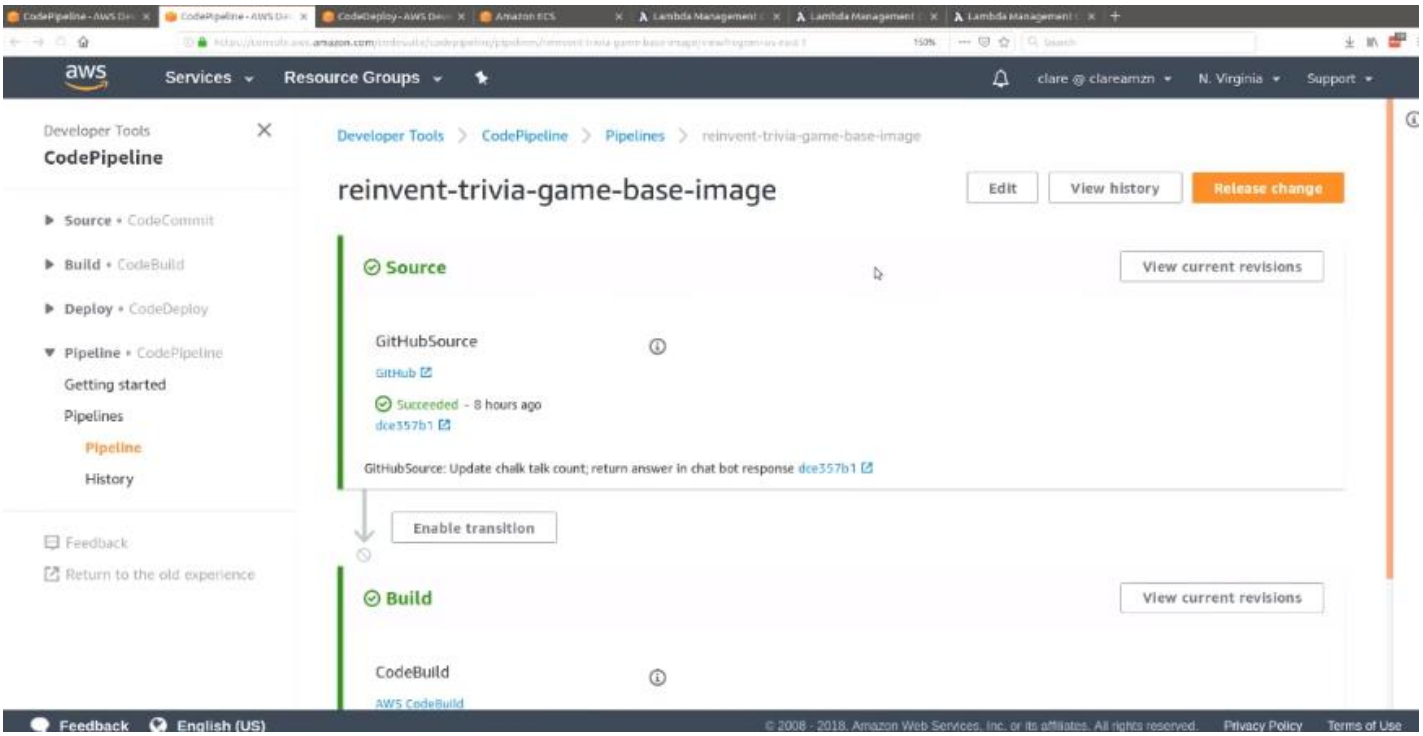

Services
Resource Groups
clare @ clareamzn
N. Virginia
Support
AWS Lambda
Dashboard
Applications
Functions
Lambda > Applications > TriviaGameChatBotTest
TriviaGameChatBotTest
Overview
Deployments
Monitoring
Dashboard: AWS Lambda
Info
Select a dashboard:
AWS Lambda
1h 3h 12h 1d 3d 1w custom
Invocations
Count
5.00
3.50
2.00
01:00
02:00
03:00
BotFunction
PreTrafficHook
Total Invocations
Errors
Count
1.00
0.5
0
01:00
02:00
03:00
BotFunction
PreTrafficHook
Total Errors
Duration (average)
Milliseconds
97.7
58.5
19.3
48.8
29.2
9.64
01:00
02:00
03:00
BotFunction
PreTrafficHook
Average Duration
Feedback
English (US)
© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

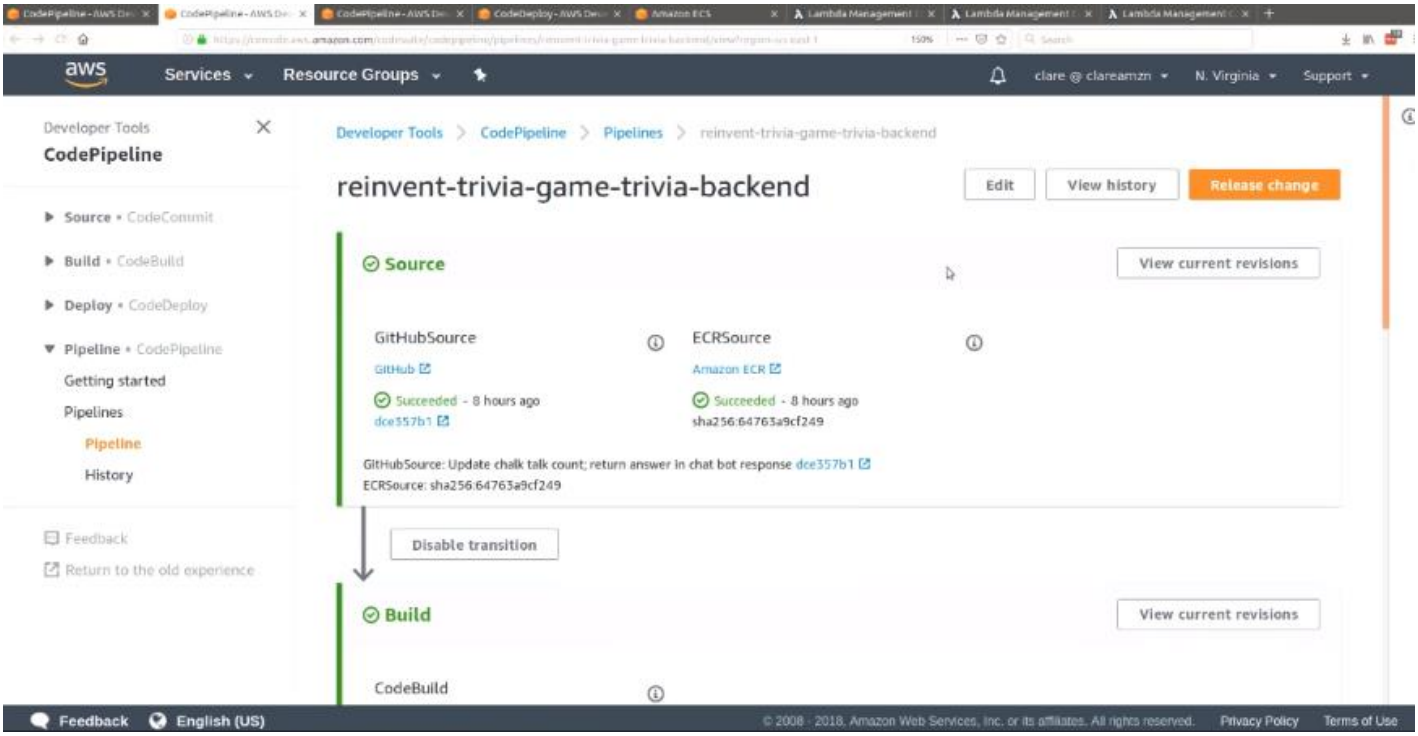


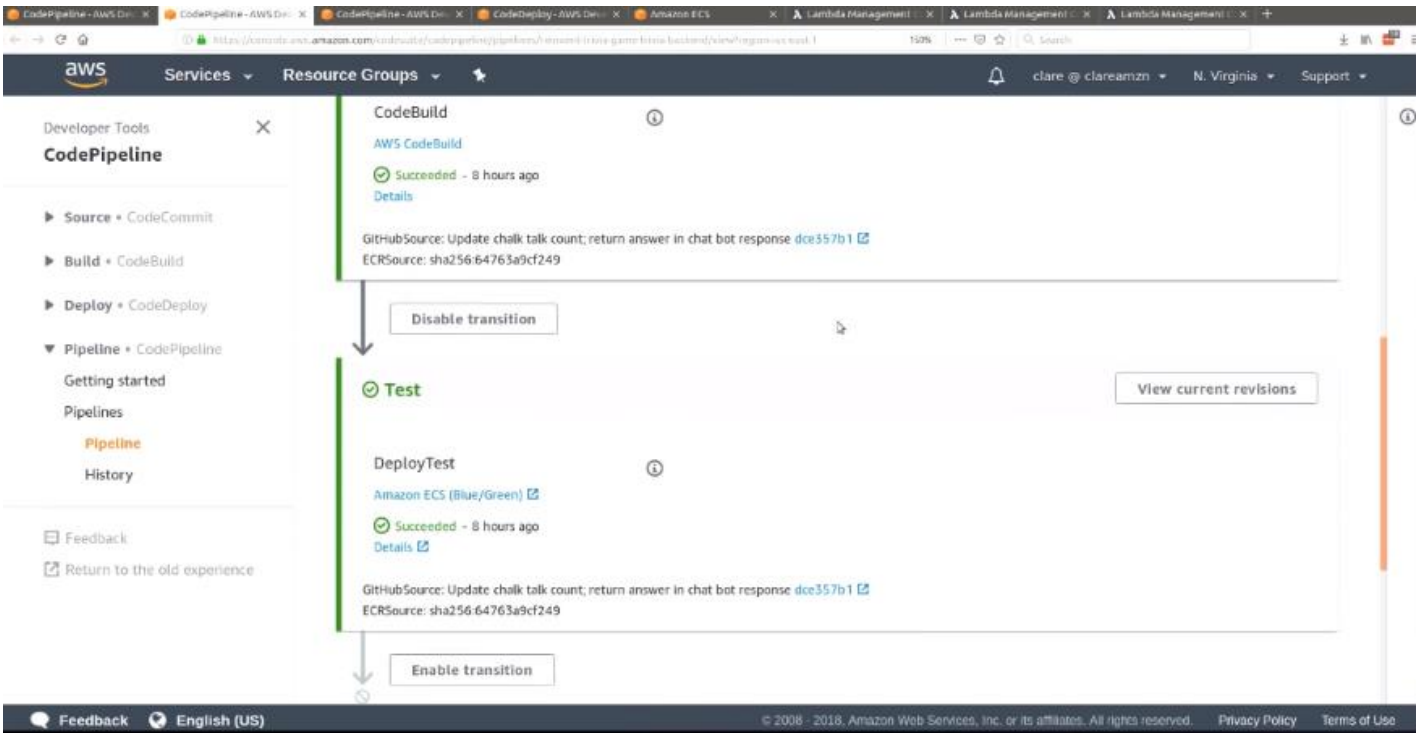
```
Resources:
  BotFunction:
    Type: AWS::Serverless::Function
    Properties:
      CodeUri: ./bot
      Handler: bot.handler
      Runtime: nodejs8.10
      Timeout: 60
      Environment:
        Variables:
          API_ENDPOINT: !Ref TriviaBackendEndpoint
      Events:
        Canary:
          Type: Schedule
          Properties:
            Schedule: rate(1 minute)
            Input: >-
              {
                "bot": {
                  "name": "TriviaGame",
                  "alias": "$LATEST",
                  "version": "$LATEST"
                },
                "currentIntent": {
                  "name": "LetsPlay",
                  "slots": {
                  }
                }
              }
      AutoPublishAlias: live
      DeploymentPreference:
```

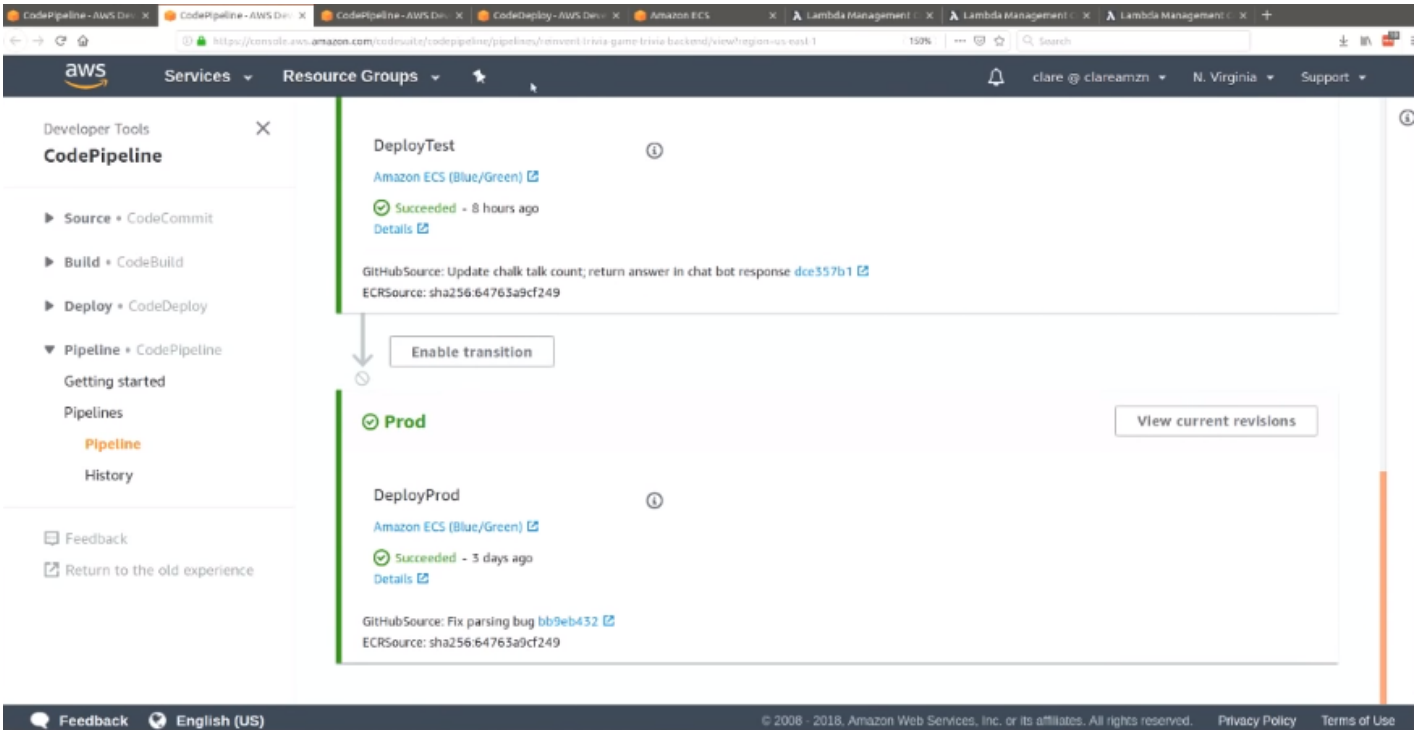


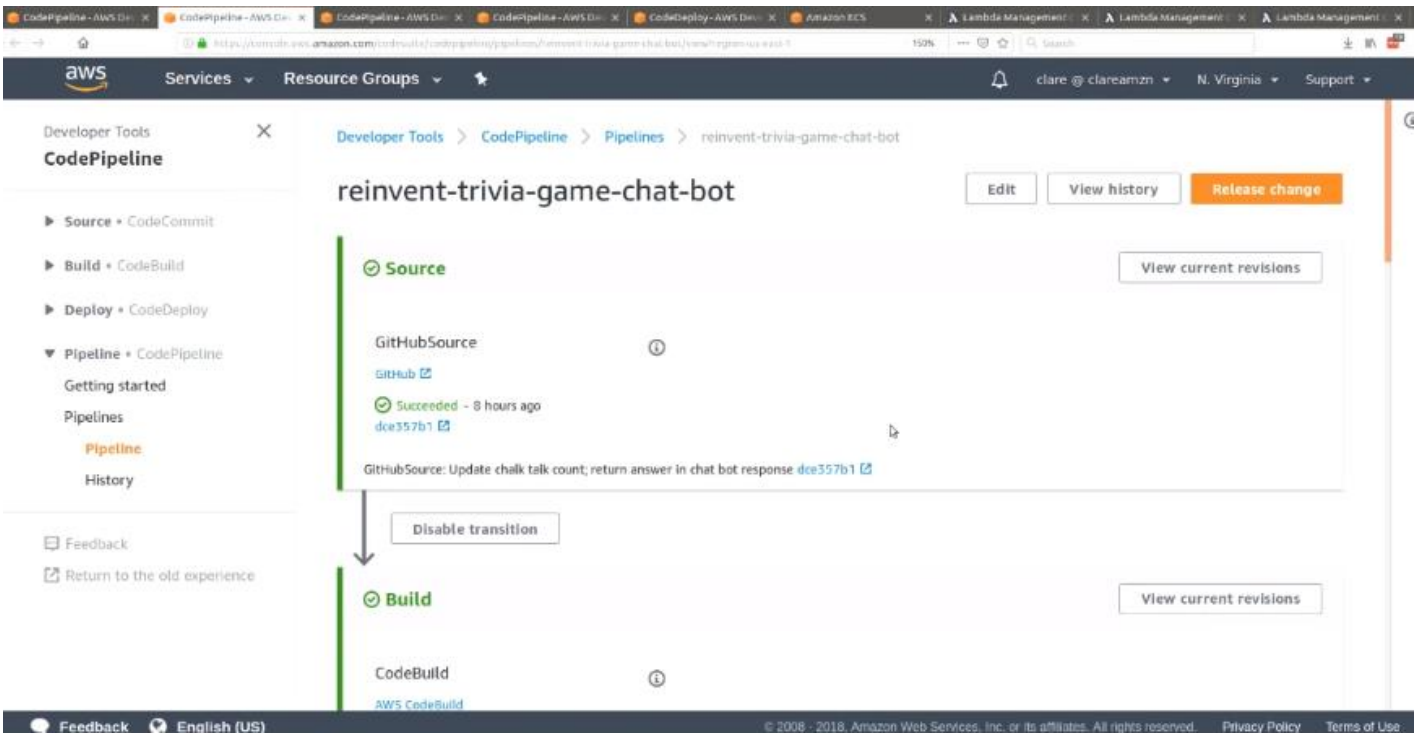

Search or jump to...
Pull requests
Issues
Marketplace
Explore
aws-samples / aws-reinvent-2018-trivia-game
Private
Unwatch 13
Star 1
Fork 0
Code
Issues 0
Pull requests 0
Projects 0
Insights
Settings
Branch: master
aws-reinvent-2018-trivia-game / pipelines / src /
Create new file
Upload files
Find file
History
clareliguori Fix resource permissions
Latest commit fe3927f 12 days ago
api-base-image-pipeline.ts
Configure base image pipeline
12 days ago
api-service-pipeline.ts
Move buildspec
12 days ago
chat-bot-pipeline.ts
Fix stack names in pipelines
13 days ago
pipeline.ts
Fix resource permissions
12 days ago
static-site-infra-pipeline.ts
Fix stack names in pipelines
13 days ago
static-site-pipeline.ts
Update pipeline repo
13 days ago

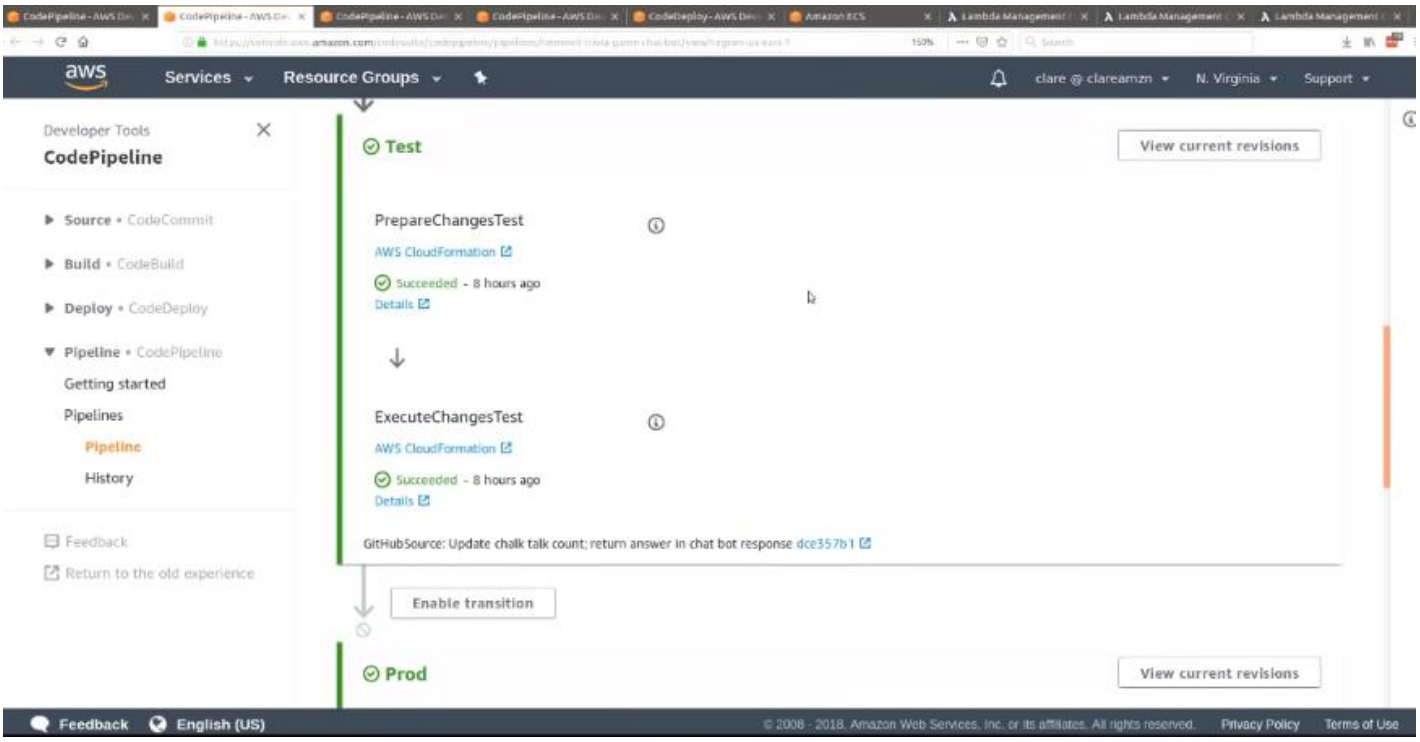

aws
Services
Resource Groups
clare @ clareamzn
N. Virginia
Support
Developer Tools
CodePipeline
Source • CodeCommit
Build • CodeBuild
Deploy • CodeDeploy
Pipeline • CodePipeline
Getting started
Pipelines
Pipeline
History
Feedback
Return to the old experience
Developer Tools > CodePipeline > Pipelines > reinvent-trivia-game-base-image
reinvent-trivia-game-base-image
Edit
View history
Release change
Source
View current revisions
GitHubSource
GitHub
Succeeded - 8 hours ago
dce357b1
GitHubSource: Update chalk talk count; return answer in chat bot response dce357b1
Enable transition
Build
View current revisions
CodeBuild
AWS CodeBuild
Feedback
English (US)
© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

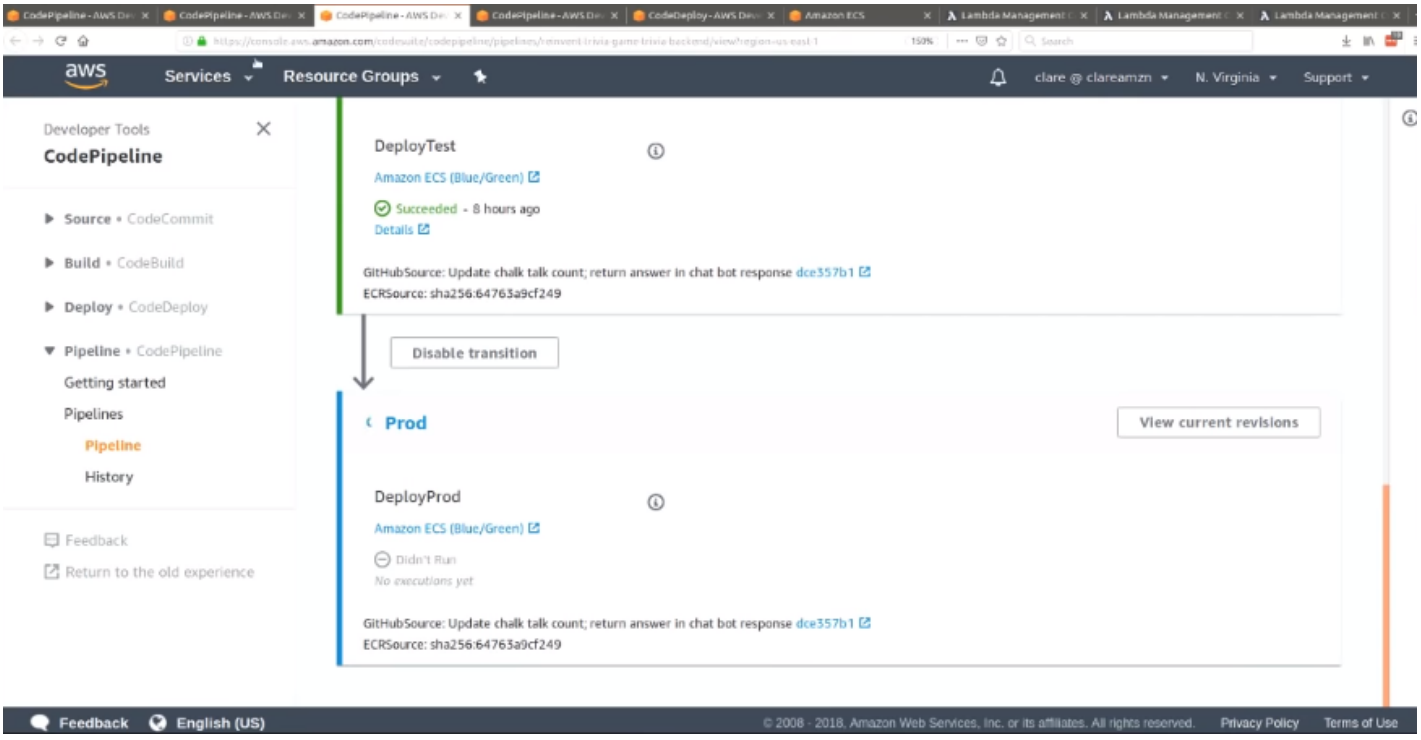

aws
Services
Resource Groups
clare @ clareamzn
N. Virginia
Support
Developer Tools
CodePipeline
Source • CodeCommit
Build • CodeBuild
Deploy • CodeDeploy
Pipeline • CodePipeline
Getting started
Pipelines
Pipeline
History
Feedback
Return to the old experience
Developer Tools > CodePipeline > Pipelines > reinvent-trivia-game-trivia-backend
reinvent-trivia-game-trivia-backend
Edit
View history
Release change
Source
View current revisions
GitHubSource
GitHub
Succeeded - 8 hours ago
dce357b1
ECRSource
Amazon ECR
Succeeded - 8 hours ago
sha256:64763a9cf249
GitHubSource: Update chalk talk count; return answer in chat bot response dce357b1
ECRSource: sha256:64763a9cf249
Disable transition
Build
View current revisions
CodeBuild
Feedback
English (US)
© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

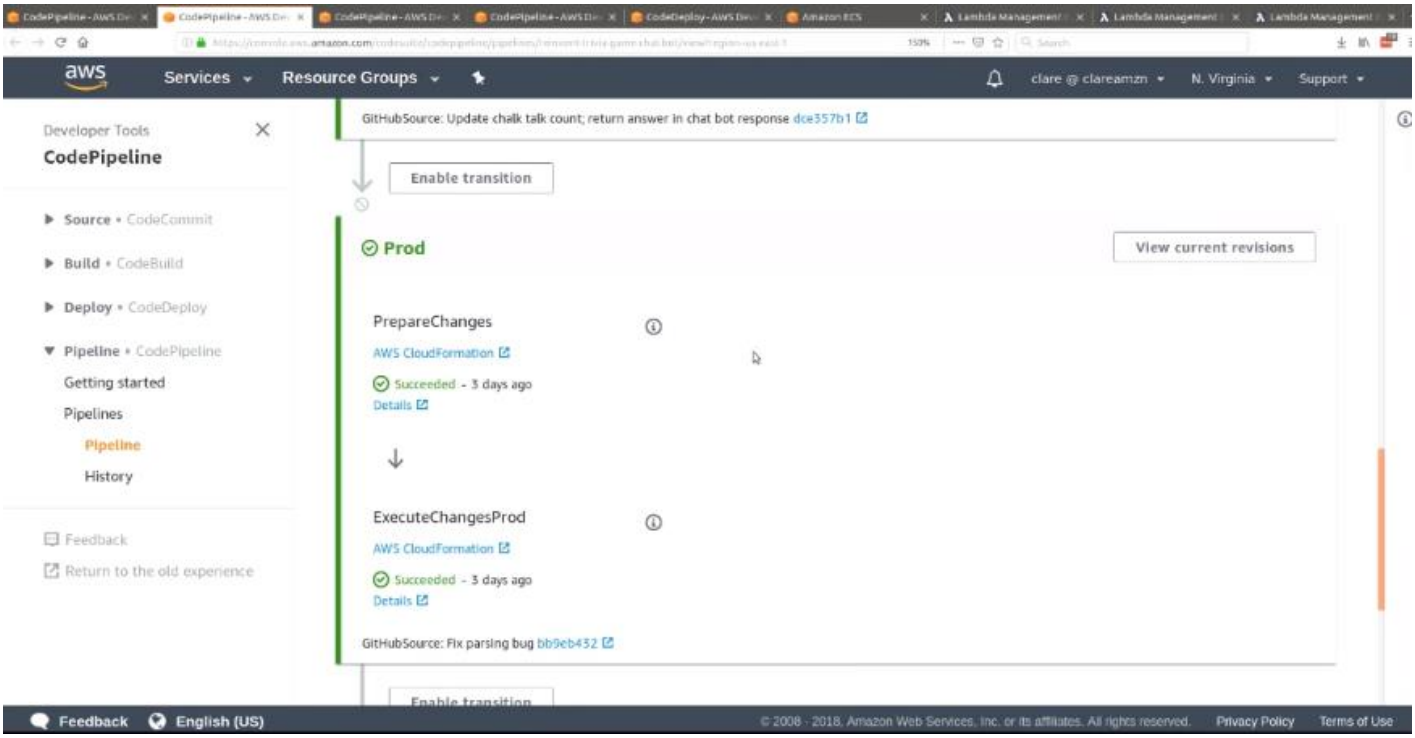

aws
Services
Resource Groups
clare @ clareamzn
N. Virginia
Support
Developer Tools
CodePipeline
Source • CodeCommit
Build • CodeBuild
Deploy • CodeDeploy
Pipeline • CodePipeline
Getting started
Pipelines
Pipeline
History
Feedback
Return to the old experience
CodeBuild
AWS CodeBuild
Succeeded - 8 hours ago
Details
GitHubSource: Update chalk talk count; return answer in chat bot response dce357b1
ECRSource: sha256:64763a9cf249
Disable transition
Test
View current revisions
DeployTest
Amazon ECS (Blue/Green)
Succeeded - 8 hours ago
Details
GitHubSource: Update chalk talk count; return answer in chat bot response dce357b1
ECRSource: sha256:64763a9cf249
Enable transition
Feedback
English (US)
© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

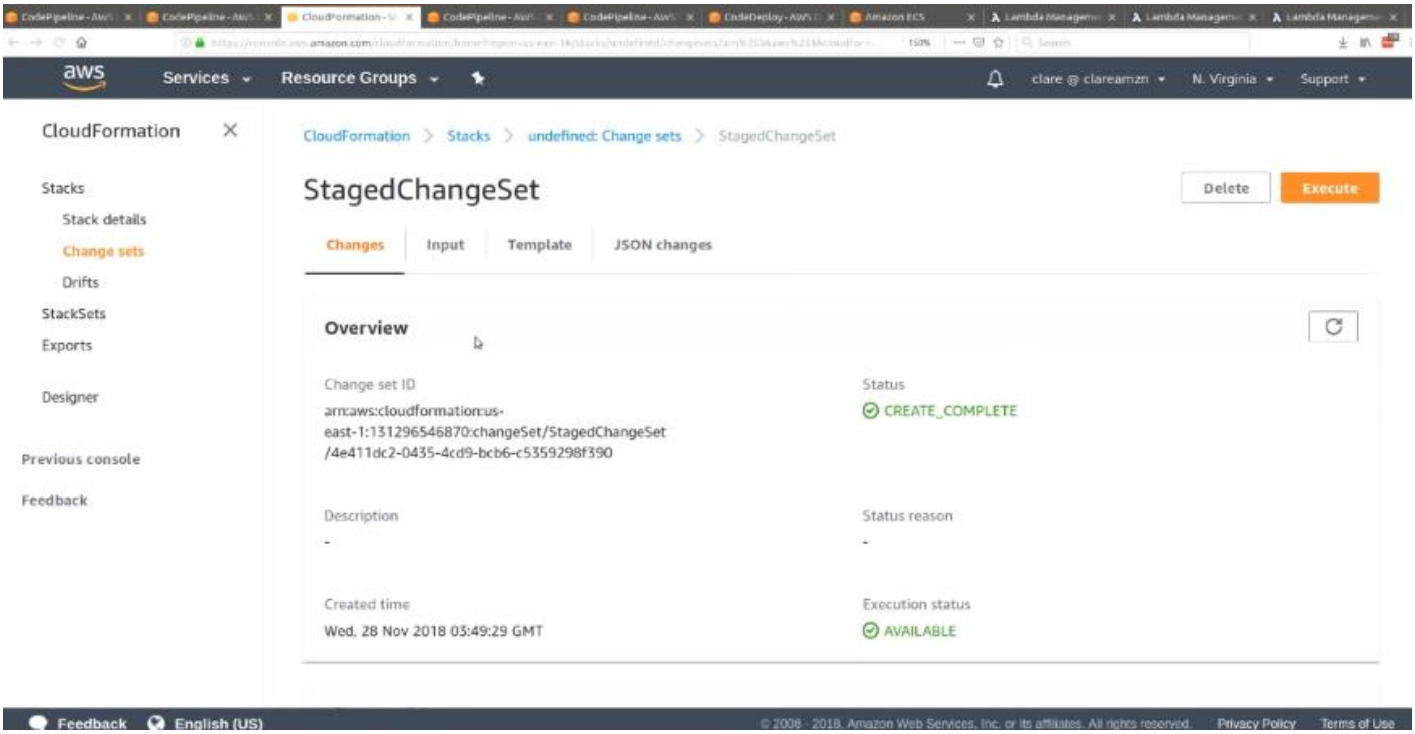

Services
Resource Groups
clare @ clareamzn
N. Virginia
Support
Developer Tools
CodePipeline
Source • CodeCommit
Build • CodeBuild
Deploy • CodeDeploy
Pipeline • CodePipeline
Getting started
Pipelines
Pipeline
History
Feedback
Return to the old experience
DeployTest
Amazon ECS (Blue/Green)
Succeeded - 8 hours ago
Details
GitHubSource: Update chalk talk count; return answer in chat bot response dce357b1
ECRSource: sha256:64763a9cf249
Enable transition
Prod
View current revisions
DeployProd
Amazon ECS (Blue/Green)
Succeeded - 3 days ago
Details
GitHubSource: Fix parsing bug bb9eb432
ECRSource: sha256:64763a9cf249
Feedback
English (US)
© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

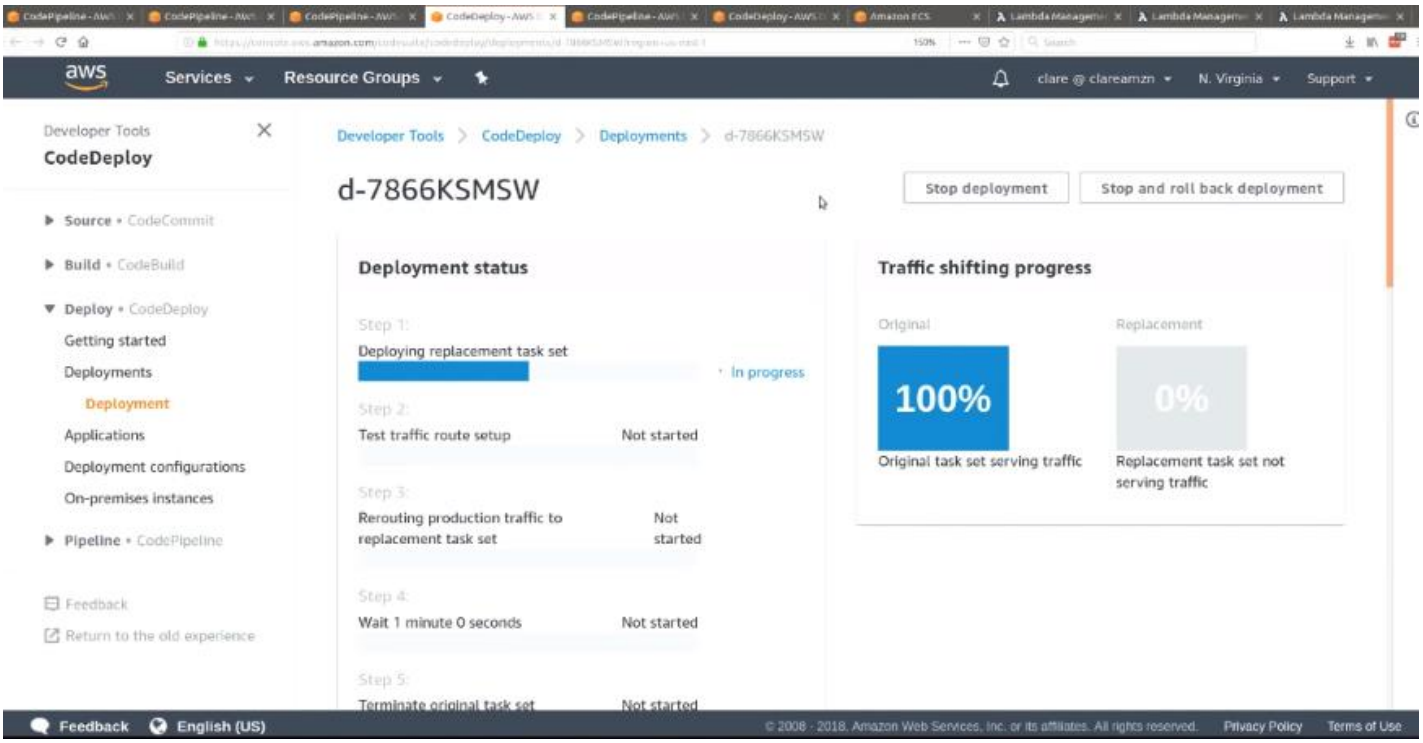

Services
Resource Groups
clare @ clareamzn
N. Virginia
Support
Developer Tools
CodePipeline
Source • CodeCommit
Build • CodeBuild
Deploy • CodeDeploy
Pipeline • CodePipeline
Getting started
Pipelines
Pipeline
History
Feedback
Return to the old experience
Developer Tools > CodePipeline > Pipelines > reinvent-trivia-game-chat-bot
reinvent-trivia-game-chat-bot
Edit
View history
Release change
Source
View current revisions
GitHubSource
GitHub
Succeeded - 8 hours ago
dce357b1
GitHubSource: Update chalk talk count; return answer in chat bot response dce357b1
Disable transition
Build
View current revisions
CodeBuild
AWS CodeBuild
Feedback
English (US)
© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

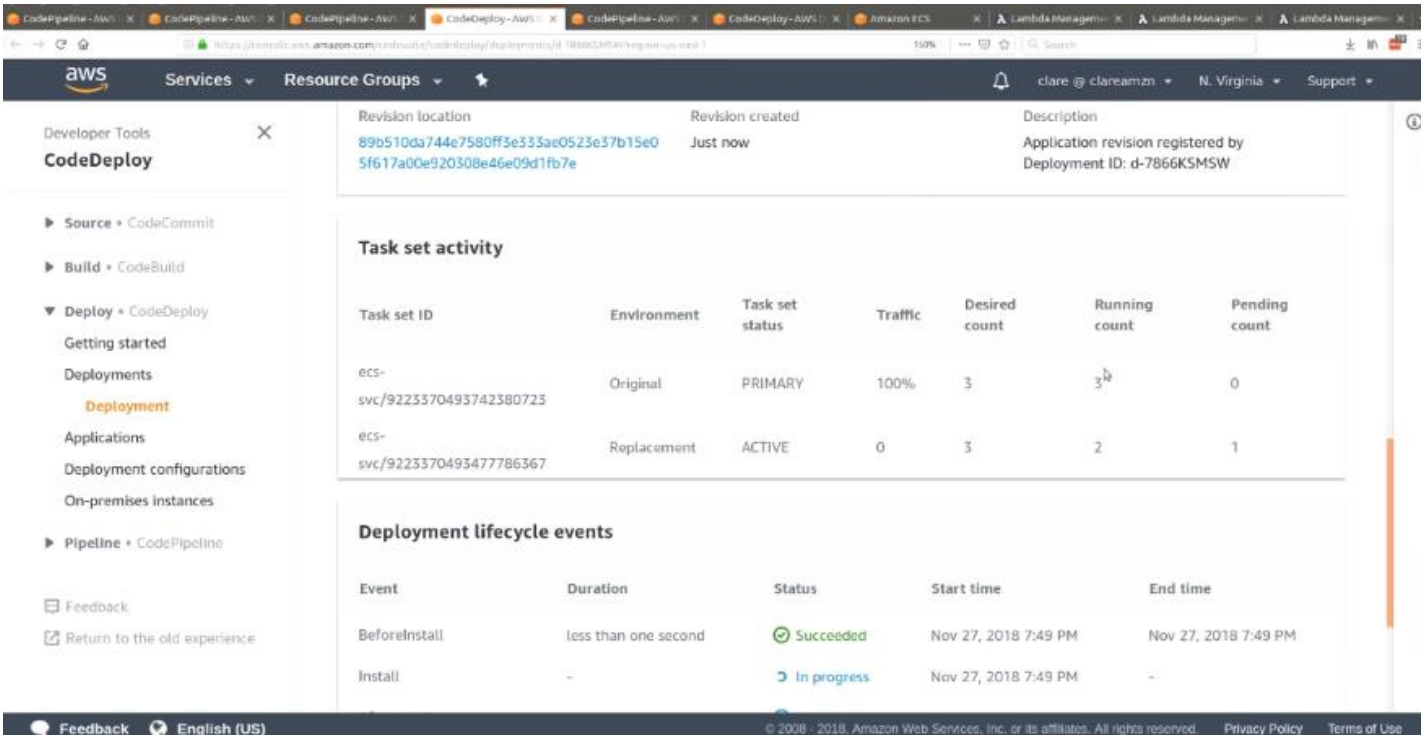

aws
Services
Resource Groups
clare @ clareamzn
N. Virginia
Support
Developer Tools
CodePipeline
Source • CodeCommit
Build • CodeBuild
Deploy • CodeDeploy
Pipeline • CodePipeline
Getting started
Pipelines
Pipeline
History
Feedback
Return to the old experience
Test
View current revisions
PrepareChangesTest
AWS CloudFormation
Succeeded - 8 hours ago
Details
ExecuteChangesTest
AWS CloudFormation
Succeeded - 8 hours ago
Details
GitHubSource: Update chalk talk count; return answer in chat bot response dce357b1
Enable transition
Prod
View current revisions
Feedback
English (US)
© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

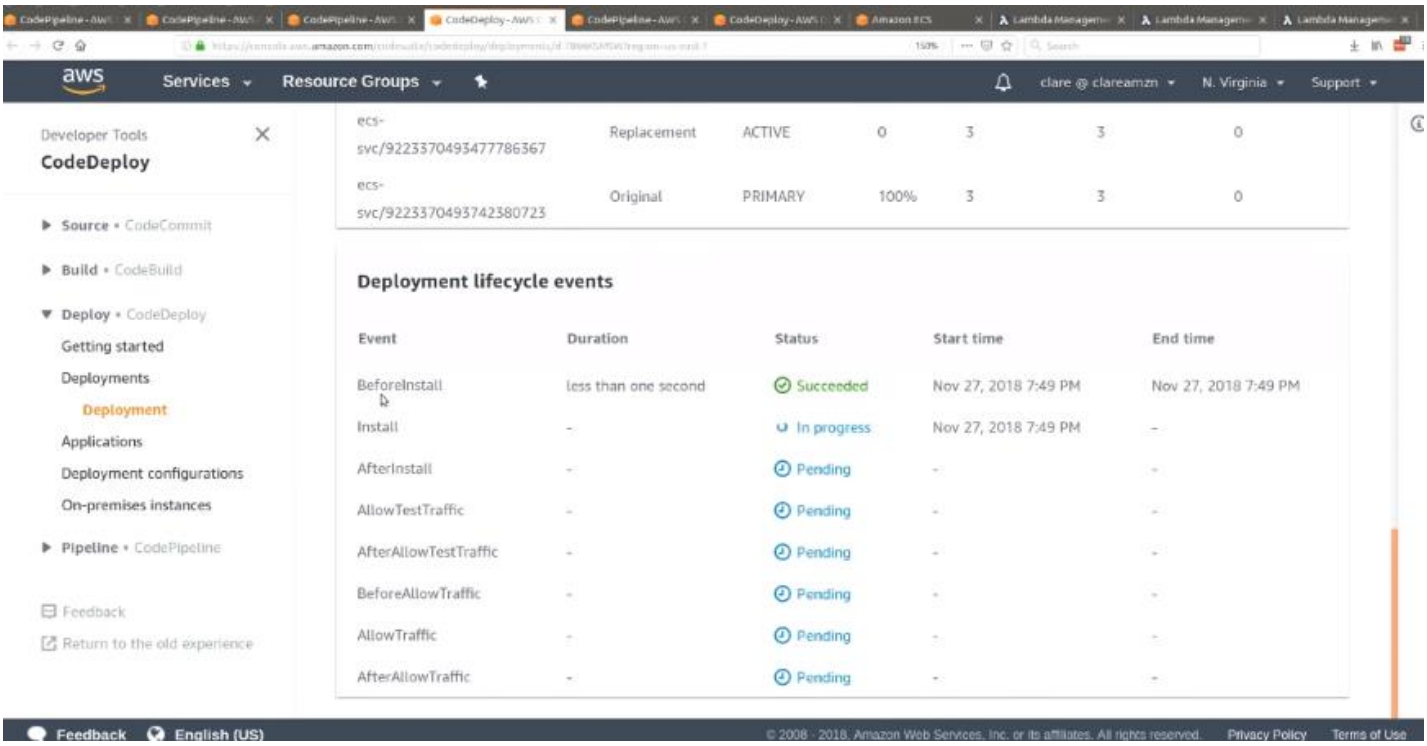

aws
Services
Resource Groups
clare @ clareamzn
N. Virginia
Support
Developer Tools
CodePipeline
Source • CodeCommit
Build • CodeBuild
Deploy • CodeDeploy
Pipeline • CodePipeline
Getting started
Pipelines
Pipeline
History
Feedback
Return to the old experience
DeployTest
Amazon ECS (Blue/Green)
Succeeded - 8 hours ago
Details
GitHubSource: Update chalk talk count; return answer in chat bot response dce357b1
ECRSource: sha256:64763a9cf249
Disable transition
Prod
View current revisions
DeployProd
Amazon ECS (Blue/Green)
Didn't Run
No executions yet
GitHubSource: Update chalk talk count; return answer in chat bot response dce357b1
ECRSource: sha256:64763a9cf249
Feedback
English (US)
© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

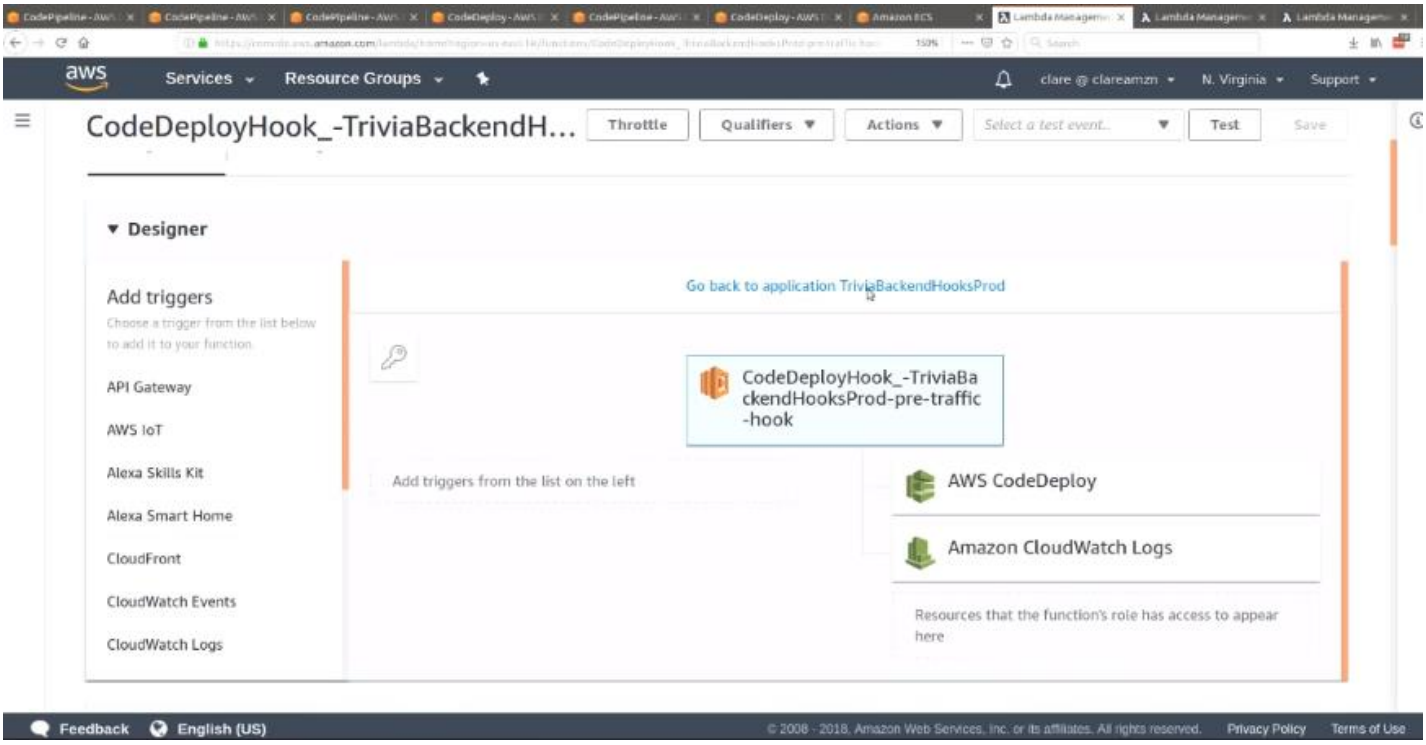

aws
Services
Resource Groups
clare @ clareamzn
N. Virginia
Support
Developer Tools
CodePipeline
Source • CodeCommit
Build • CodeBuild
Deploy • CodeDeploy
Pipeline • CodePipeline
Getting started
Pipelines
Pipeline
History
Feedback
Return to the old experience
GitHubSource: Update chalk talk count; return answer in chat bot response dce357b1
Enable transition
Prod
View current revisions
PrepareChanges
AWS CloudFormation
Succeeded - 3 days ago
Details
ExecuteChangesProd
AWS CloudFormation
Succeeded - 3 days ago
Details
GitHubSource: Fix parsing bug bb9eb432
Enable transition
Feedback
English (US)
© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

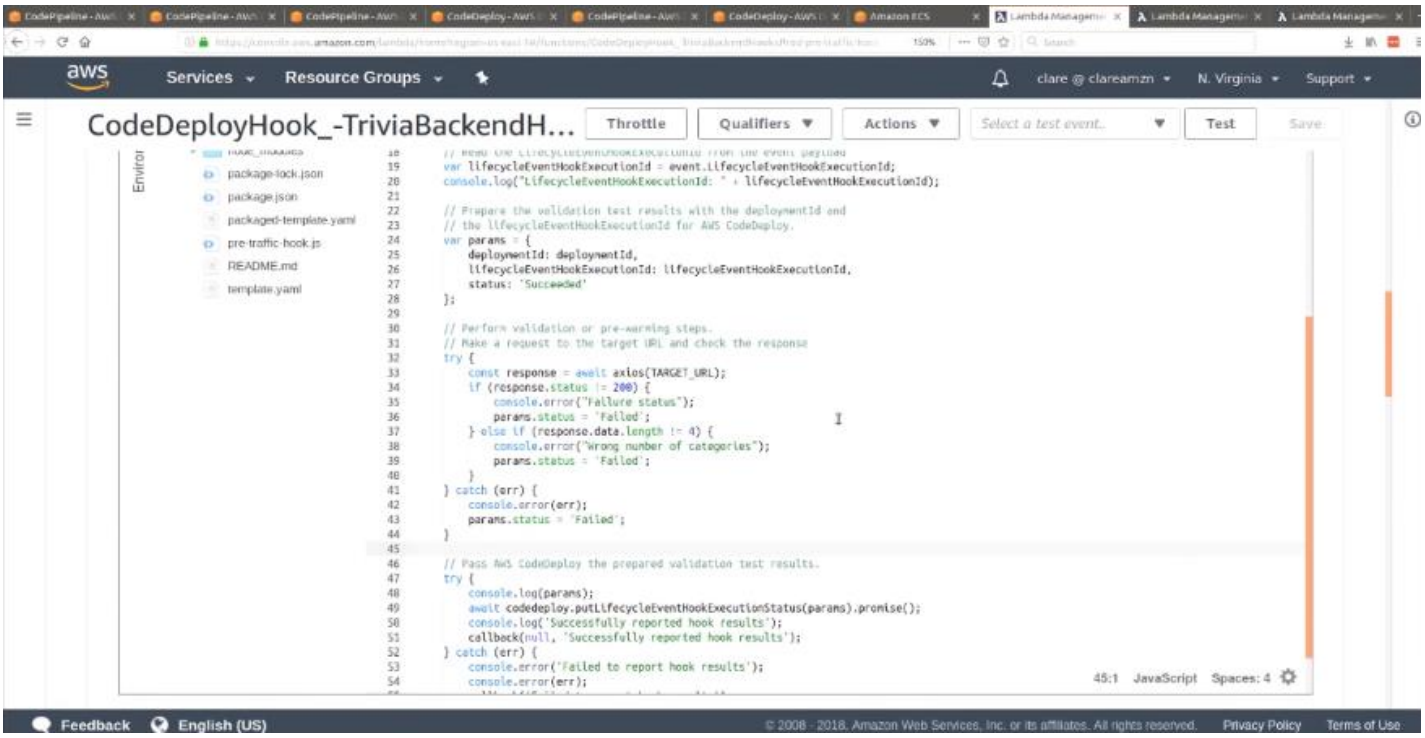

aws
Services
Resource Groups
clare @ clareamzn
N. Virginia
Support
CloudFormation
Stacks
Stack details
Change sets
Drifts
StackSets
Exports
Designer
Previous console
Feedback
CloudFormation > Stacks > undefined: Change sets > StagedChangeSet
StagedChangeSet
Delete
Execute
Changes
Input
Template
JSON changes
Overview
Change set ID
arn:aws:cloudformation:us-east-1:131296546870:changeSet/StagedChangeSet/4e411dc2-0435-4cd9-bcb6-c5359298f390
Status
CREATE_COMPLETE
Description
-
Status reason
-
Created time
Wed, 28 Nov 2018 03:49:29 GMT
Execution status
AVAILABLE
Feedback
English (US)
© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

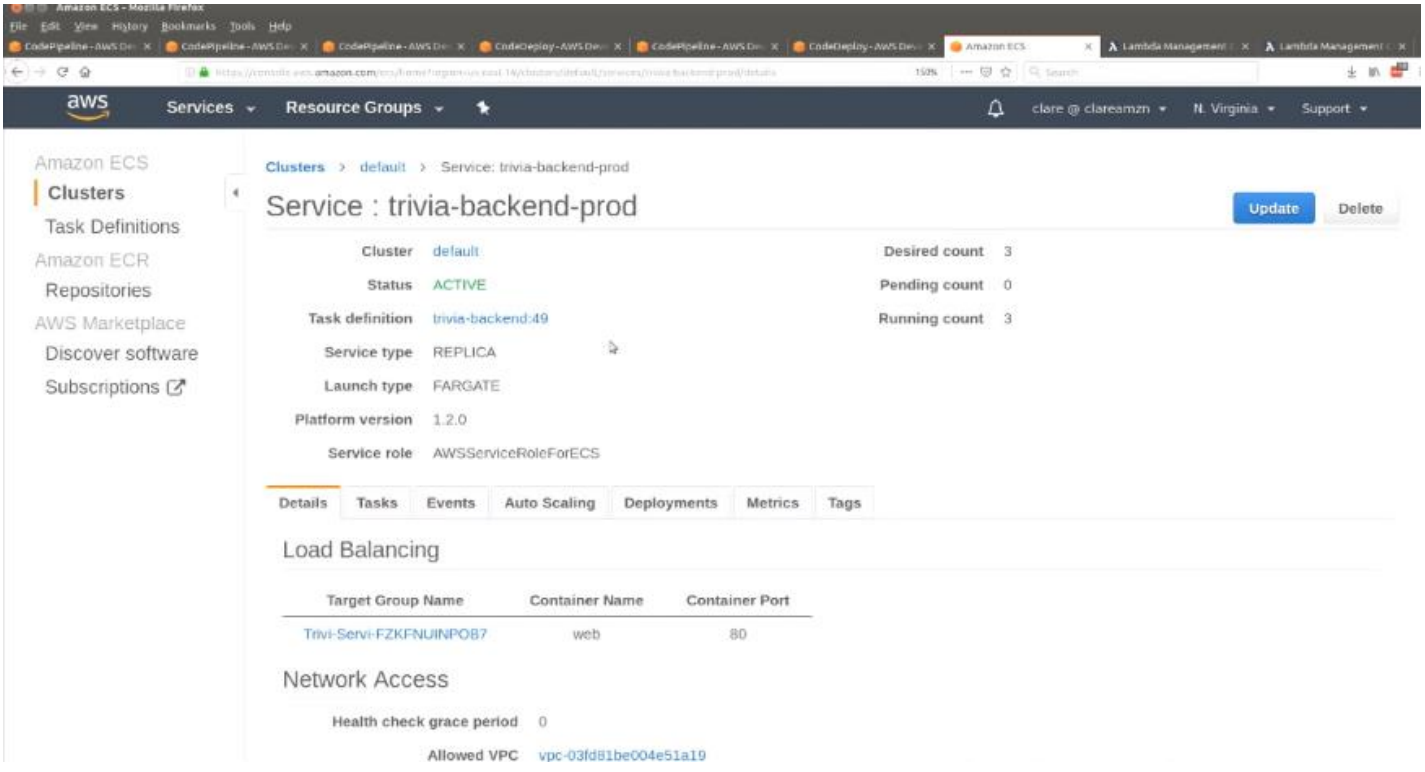

aws
Services
Resource Groups
clare @ clareamzn
N. Virginia
Support
Developer Tools
CodeDeploy
Source • CodeCommit
Build • CodeBuild
Deploy • CodeDeploy
Getting started
Deployments
Deployment
Applications
Deployment configurations
On-premises instances
Pipeline • CodePipeline
Feedback
Return to the old experience
Developer Tools > CodeDeploy > Deployments > d-7866KSMSW
d-7866KSMSW
Stop deployment
Stop and roll back deployment
Deployment status
Step 1:
Deploying replacement task set
In progress
Step 2:
Test traffic route setup
Not started
Step 3:
Rerouting production traffic to replacement task set
Not started
Step 4:
Wait 1 minute 0 seconds
Not started
Step 5:
Terminate original task set
Not started
Traffic shifting progress
Original
100%
Original task set serving traffic
Replacement
0%
Replacement task set not serving traffic
Feedback
English (US)
© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

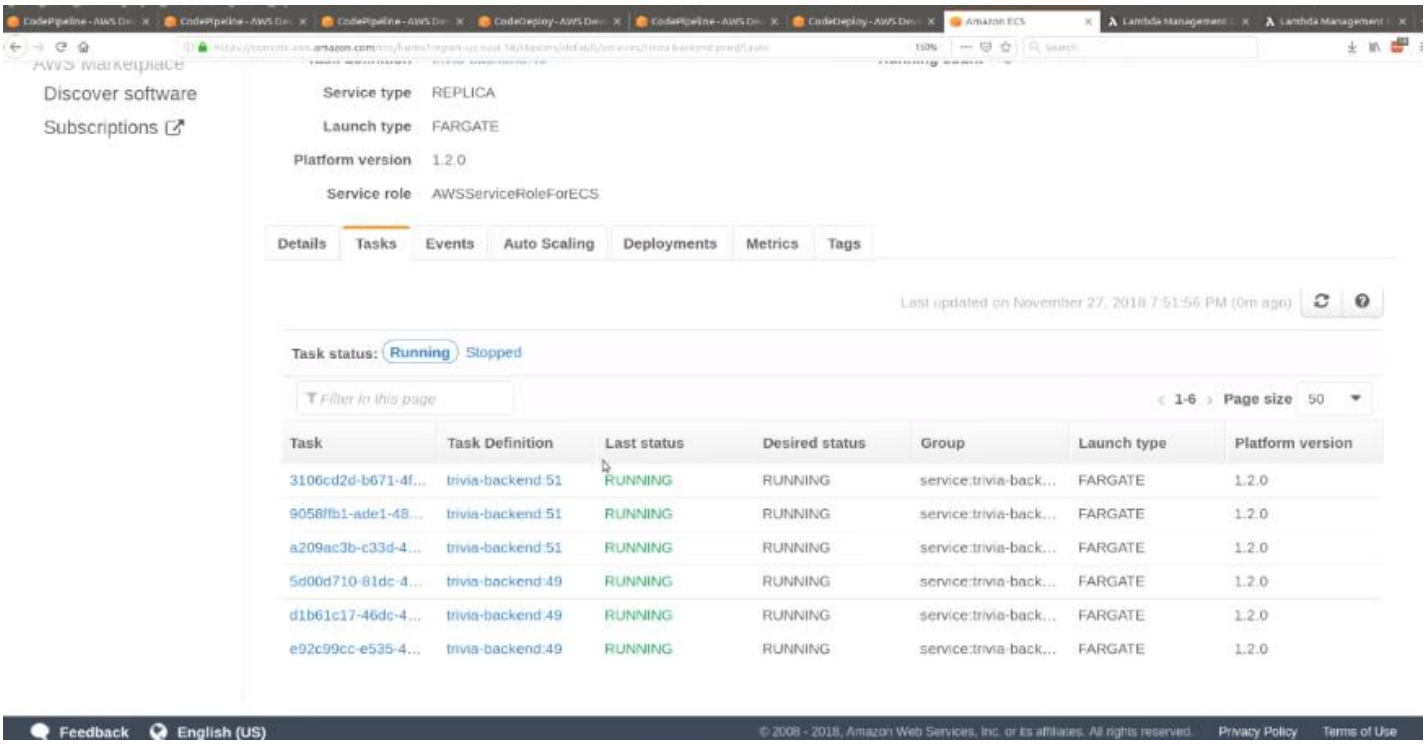

aws
Services
Resource Groups
clare @ clareamzn
N. Virginia
Support
Developer Tools
CodeDeploy
Source • CodeCommit
Build • CodeBuild
Deploy • CodeDeploy
Getting started
Deployments
Deployment
Applications
Deployment configurations
On-premises instances
Pipeline • CodePipeline
Feedback
Return to the old experience
Revision location
89b510da744e7580ff3e333ac0523e37b15e05f617a00e920308e46e09d1fb7e
Revision created
Just now
Description
Application revision registered by Deployment ID: d-7866KSMSW
Task set activity
Task set ID
Environment
Task set status
Traffic
Desired count
Running count
Pending count
ecs-svc/9223370493742380723
Original
PRIMARY
100%
3
3
0
ecs-svc/9223370493477786367
Replacement
ACTIVE
0
3
2
1
Deployment lifecycle events
Event
Duration
Status
Start time
End time
BeforeInstall
less than one second
Succeeded
Nov 27, 2018 7:49 PM
Nov 27, 2018 7:49 PM
Install
-
In progress
Nov 27, 2018 7:49 PM
-
Feedback
English (US)
© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

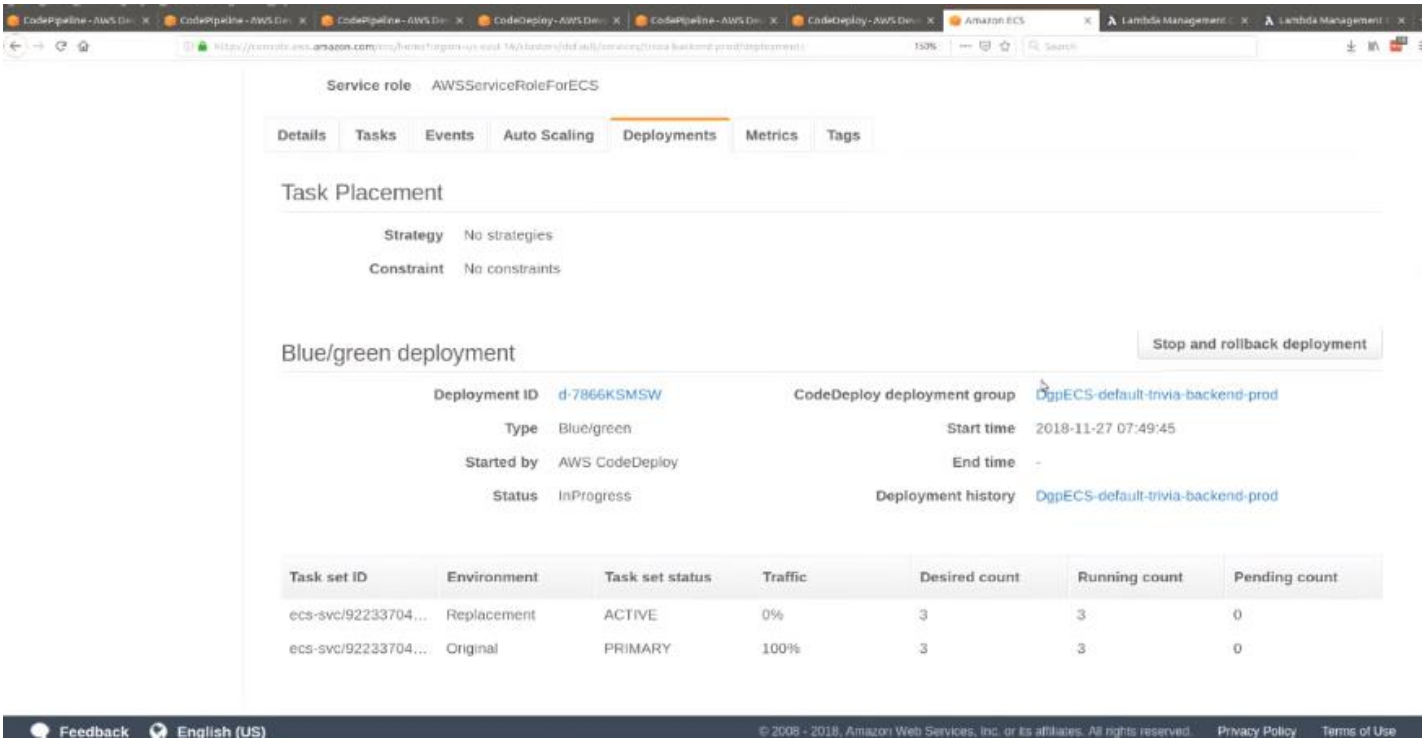

aws
Services
Resource Groups
clare @ clareamzn
N. Virginia
Support
Developer Tools
CodeDeploy
Source • CodeCommit
Build • CodeBuild
Deploy • CodeDeploy
Getting started
Deployments
Deployment
Applications
Deployment configurations
On-premises instances
Pipeline • CodePipeline
Feedback
Return to the old experience
ecs-svc/9223370493477786367 Replacement ACTIVE 0 3 3 0
ecs-svc/9223370493742380723 Original PRIMARY 100% 3 3 0
Deployment lifecycle events
Event Duration Status Start time End time
BeforeInstall less than one second Succeeded Nov 27, 2018 7:49 PM Nov 27, 2018 7:49 PM
Install - In progress Nov 27, 2018 7:49 PM -
AfterInstall - Pending - -
AllowTestTraffic - Pending - -
AfterAllowTestTraffic - Pending - -
BeforeAllowTraffic - Pending - -
AllowTraffic - Pending - -
AfterAllowTraffic - Pending - -
Feedback
English (US)
© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

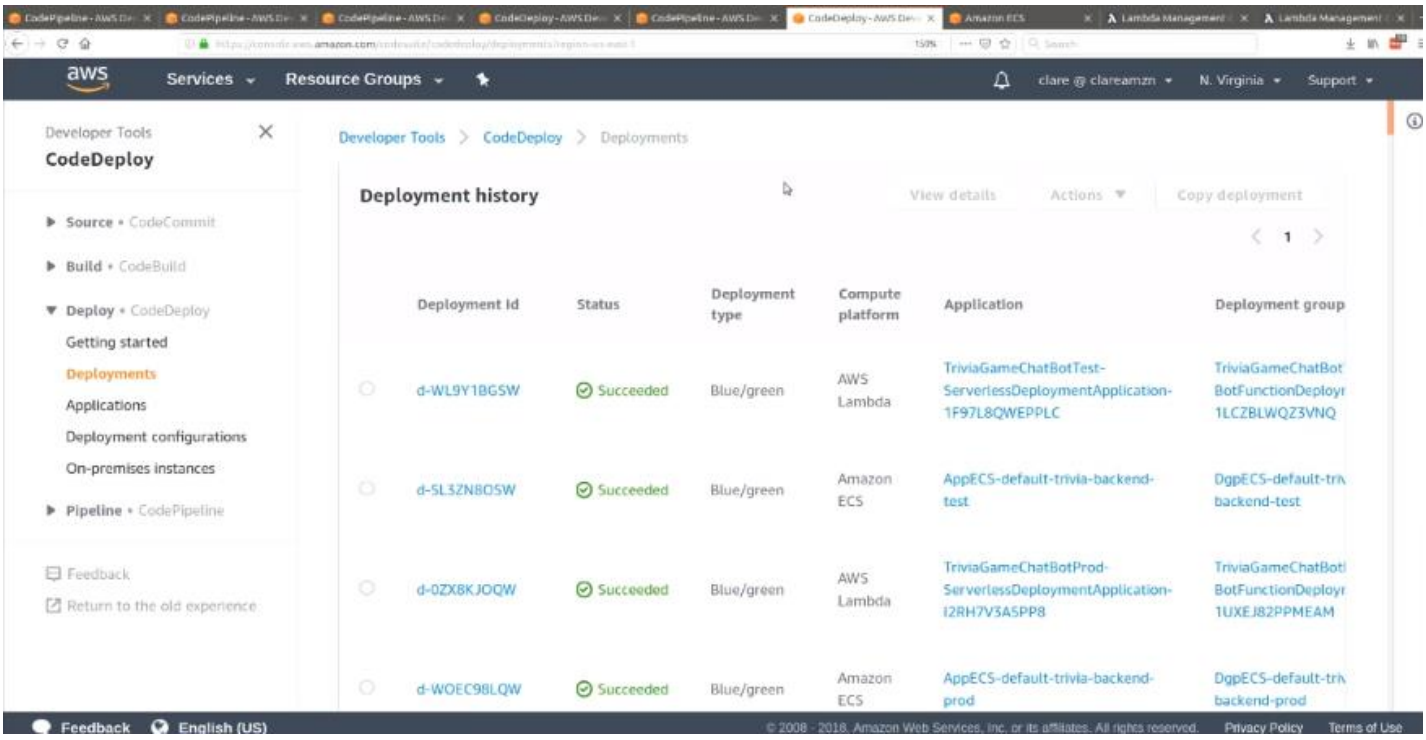

aws
Services
Resource Groups
clare @ clareamzn
N. Virginia
Support
CodeDeployHook_-TriviaBackendH...
Throttle
Qualifiers
Actions
Select a test event...
Test
Save
Designer
Add triggers
Choose a trigger from the list below to add it to your function.
API Gateway
AWS IoT
Alexa Skills Kit
Alexa Smart Home
CloudFront
CloudWatch Events
CloudWatch Logs
Go back to application TriviaBackendHooksProd
CodeDeployHook_-TriviaBackendHooksProd-pre-traffic-hook
Add triggers from the list on the left
AWS CodeDeploy
Amazon CloudWatch Logs
Resources that the function's role has access to appear here
Feedback
English (US)
© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

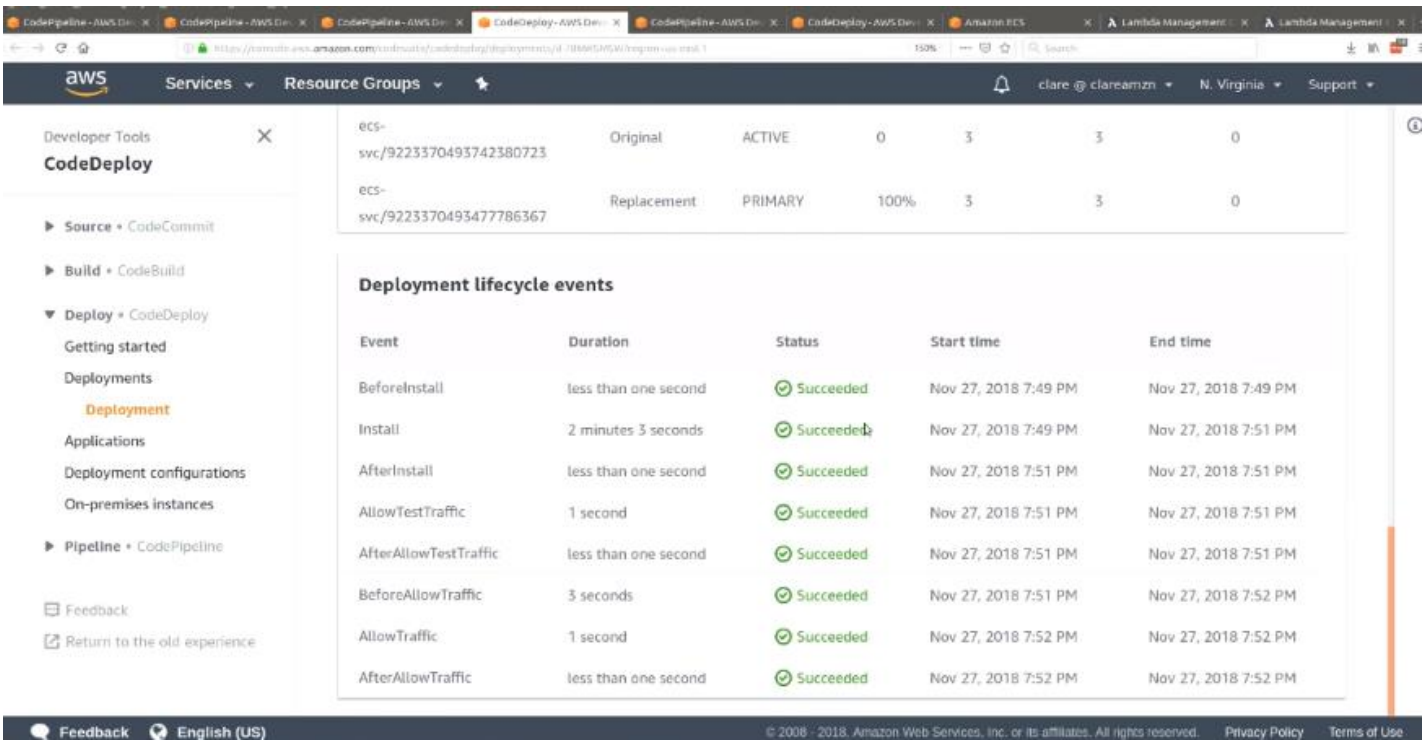

Services
Resource Groups
clare @ clareamzn
N. Virginia
Support
CodeDeployHook_-TriviaBackendH...
Throttle
Qualifiers
Actions
Select a test event.
Test
Save
Environment
package-lock.json
package.json
packaged-template.yaml
pre-traffic-hook.js
README.md
template.yaml
var lifecycleEventHookExecutionId = event.LifecycleEventHookExecutionId;
console.log('LifecycleEventHookExecutionId: ' + lifecycleEventHookExecutionId);
// Prepare the validation test results with the deploymentId and
// the lifecycleEventHookExecutionId for AWS CodeDeploy.
var params = {
deploymentId: deploymentId,
lifecycleEventHookExecutionId: lifecycleEventHookExecutionId,
status: 'Succeeded'
};
// Perform validation or pre-warming steps.
// Make a request to the target URL and check the response
try {
const response = await axios(TARGET_URL);
if (response.status != 200) {
console.error("Failure status");
params.status = 'Failed';
} else if (response.data.length != 4) {
console.error("Wrong number of categories");
params.status = 'Failed';
}
} catch (err) {
console.error(err);
params.status = 'Failed';
}
// Pass AWS CodeDeploy the prepared validation test results.
try {
console.log(params);
await codedeploy.putLifecycleEventHookExecutionStatus(params).promise();
console.log('Successfully reported hook results');
callback(null, 'Successfully reported hook results');
} catch (err) {
console.error('Failed to report hook results');
console.error(err);
45:1 JavaScript Spaces: 4
Feedback
English (US)
© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

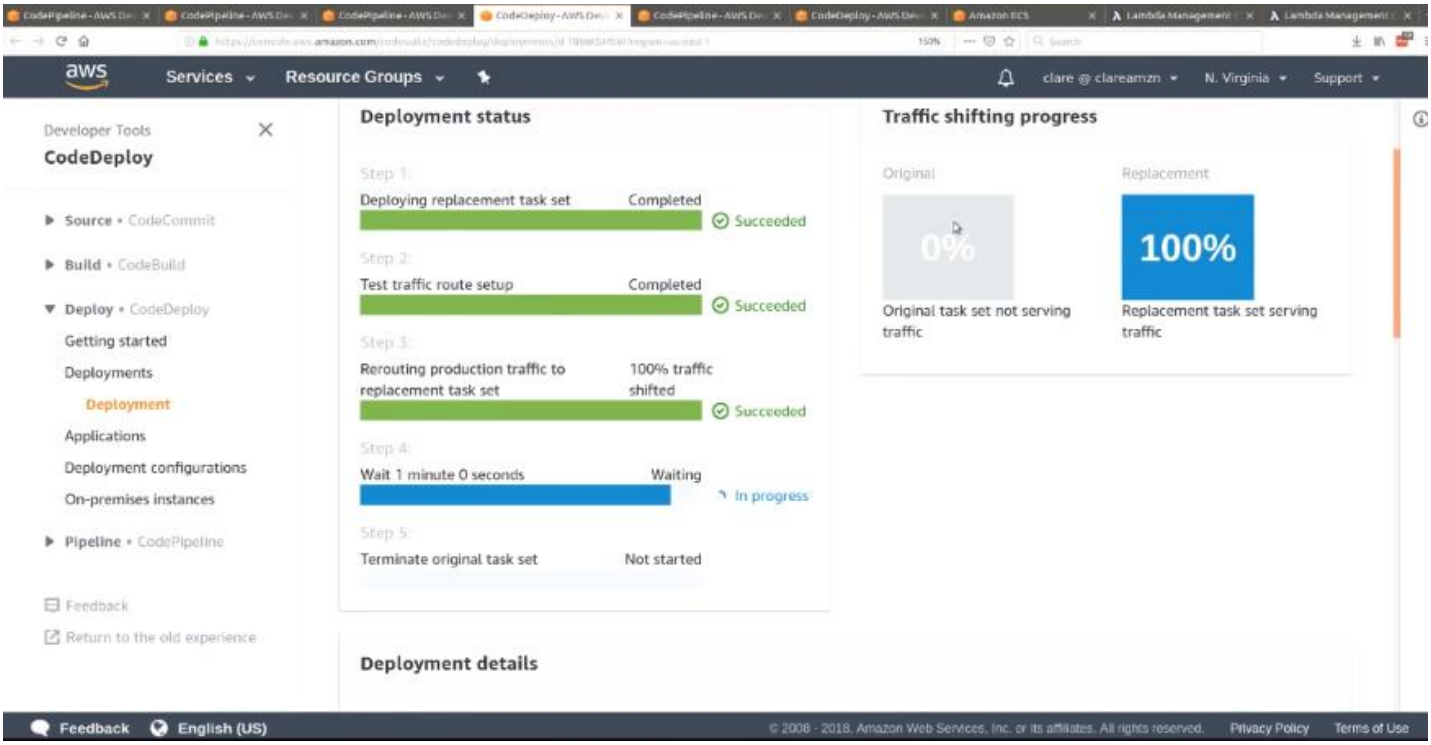

Services
Resource Groups
clare @ clareamzn
N. Virginia
Support
Amazon ECS
Clusters
Task Definitions
Amazon ECR
Repositories
AWS Marketplace
Discover software
Subscriptions
Clusters > default > Service: trivia-backend-prod
Service : trivia-backend-prod
Update
Delete
Cluster default
Status ACTIVE
Task definition trivia-backend:49
Service type REPLICA
Launch type FARGATE
Platform version 1.2.0
Service role AWSServiceRoleForECS
Desired count 3
Pending count 0
Running count 3
Details
Tasks
Events
Auto Scaling
Deployments
Metrics
Tags
Load Balancing
Target Group Name
Container Name
Container Port
Trivi-Servi-FZKFNUINPO87
web
80
Network Access
Health check grace period 0
Allowed VPC vpc-03fd81be004e51a19

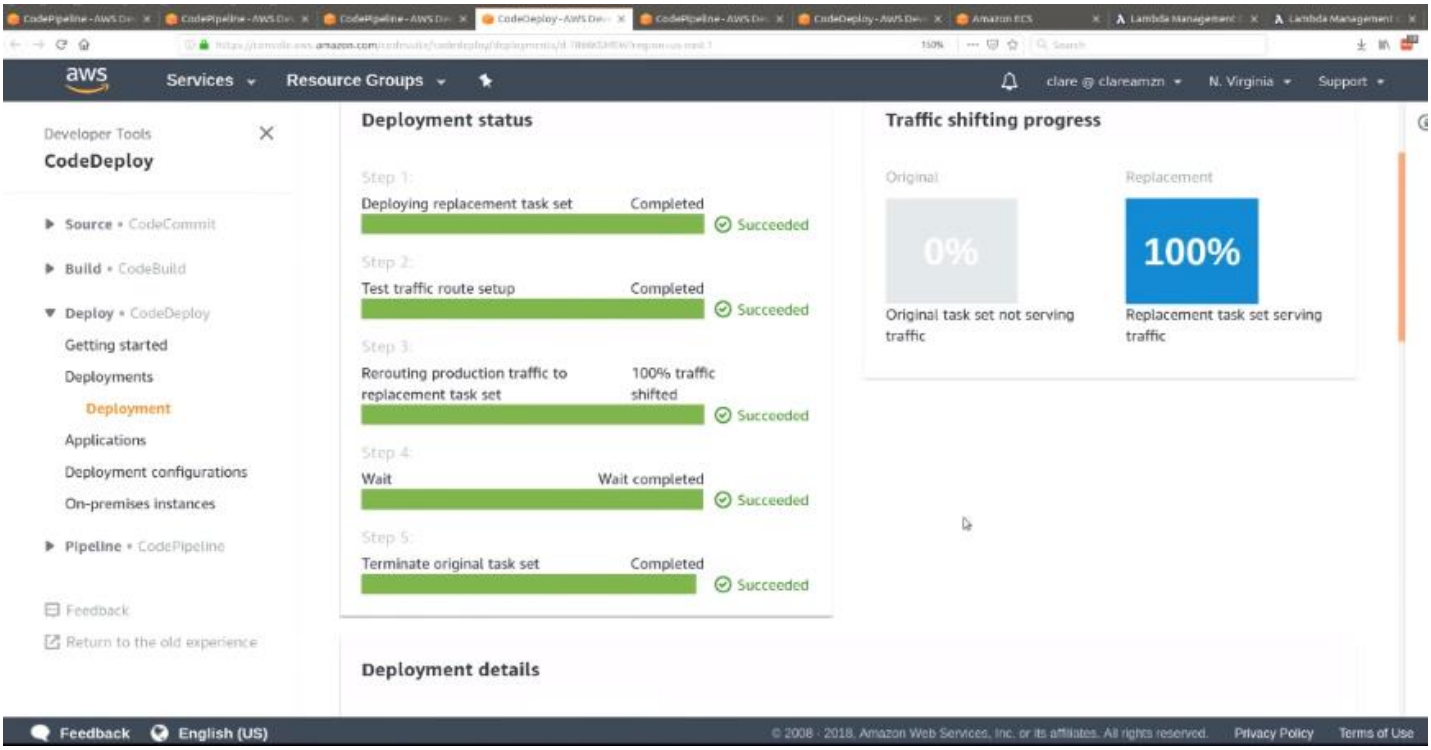

AWS Marketplace
Discover software
Subscriptions
Service type REPLICA
Launch type FARGATE
Platform version 1.2.0
Service role AWSServiceRoleForECS
Details Tasks Events Auto Scaling Deployments Metrics Tags
Last updated on November 27, 2018 7:51:56 PM (0m ago)
Task status: Running Stopped
Filter in this page
1-6 Page size 50
Task | Task Definition | Last status | Desired status | Group | Launch type | Platform version
3106cd2d-b671-4f... | trivia-backend:51 | RUNNING | RUNNING | service:trivia-back... | FARGATE | 1.2.0
9058ffb1-ade1-48... | trivia-backend:51 | RUNNING | RUNNING | service:trivia-back... | FARGATE | 1.2.0
a209ac3b-c33d-4... | trivia-backend:51 | RUNNING | RUNNING | service:trivia-back... | FARGATE | 1.2.0
5d00d710-81dc-4... | trivia-backend:49 | RUNNING | RUNNING | service:trivia-back... | FARGATE | 1.2.0
d1b61c17-46dc-4... | trivia-backend:49 | RUNNING | RUNNING | service:trivia-back... | FARGATE | 1.2.0
e92c99cc-e535-4... | trivia-backend:49 | RUNNING | RUNNING | service:trivia-back... | FARGATE | 1.2.0
Feedback English (US)
© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy Terms of Use

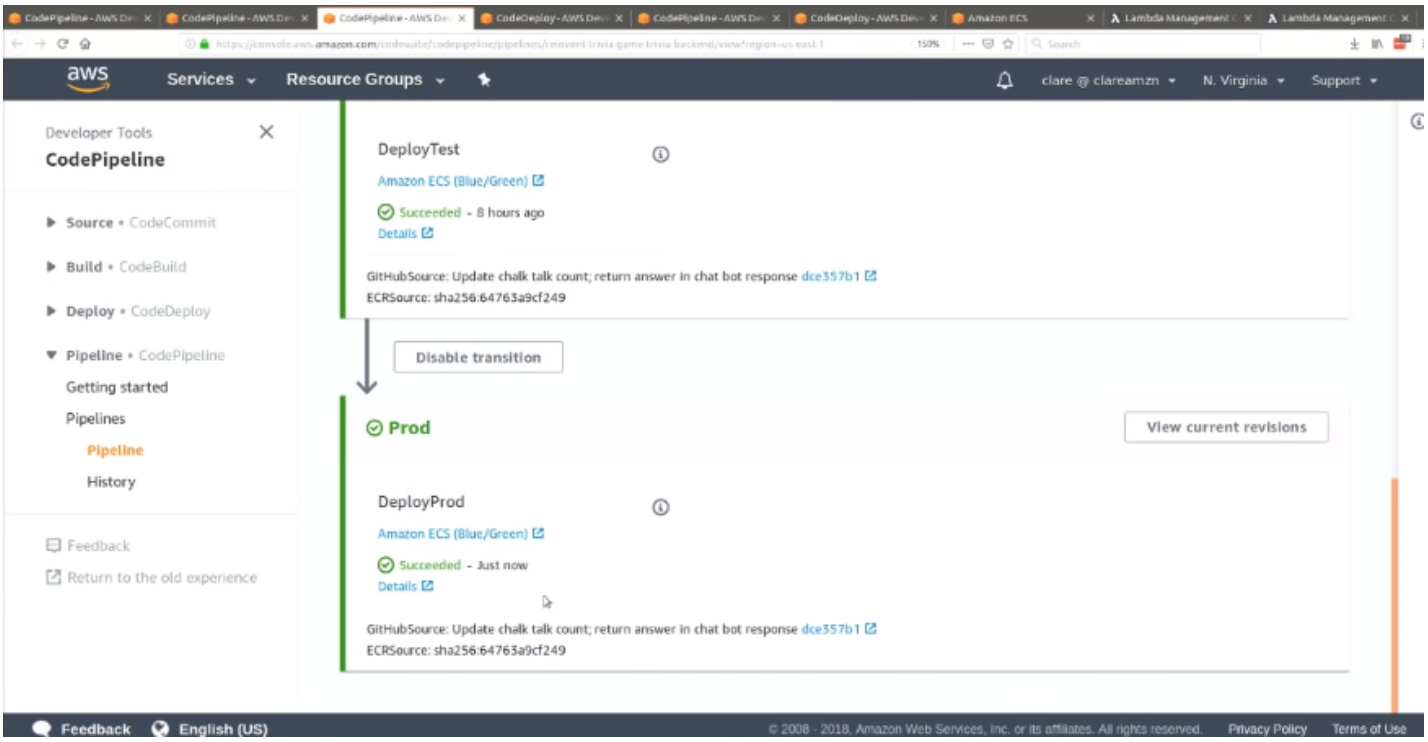

Service role AWSServiceRoleForECS
Details Tasks Events Auto Scaling Deployments Metrics Tags
Task Placement
Strategy No strategies
Constraint No constraints
Blue/green deployment
Stop and rollback deployment
Deployment ID d-7866KSMSW
Type Blue/green
Started by AWS CodeDeploy
Status InProgress
CodeDeploy deployment group DggECS-default-trivia-backend-prod
Start time 2018-11-27 07:49:45
End time -
Deployment history DggECS-default-trivia-backend-prod
Task set ID | Environment | Task set status | Traffic | Desired count | Running count | Pending count
ecs-svc/92233704... | Replacement | ACTIVE | 0% | 3 | 3 | 0
ecs-svc/92233704... | Original | PRIMARY | 100% | 3 | 3 | 0
Feedback English (US)
© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy Terms of Use

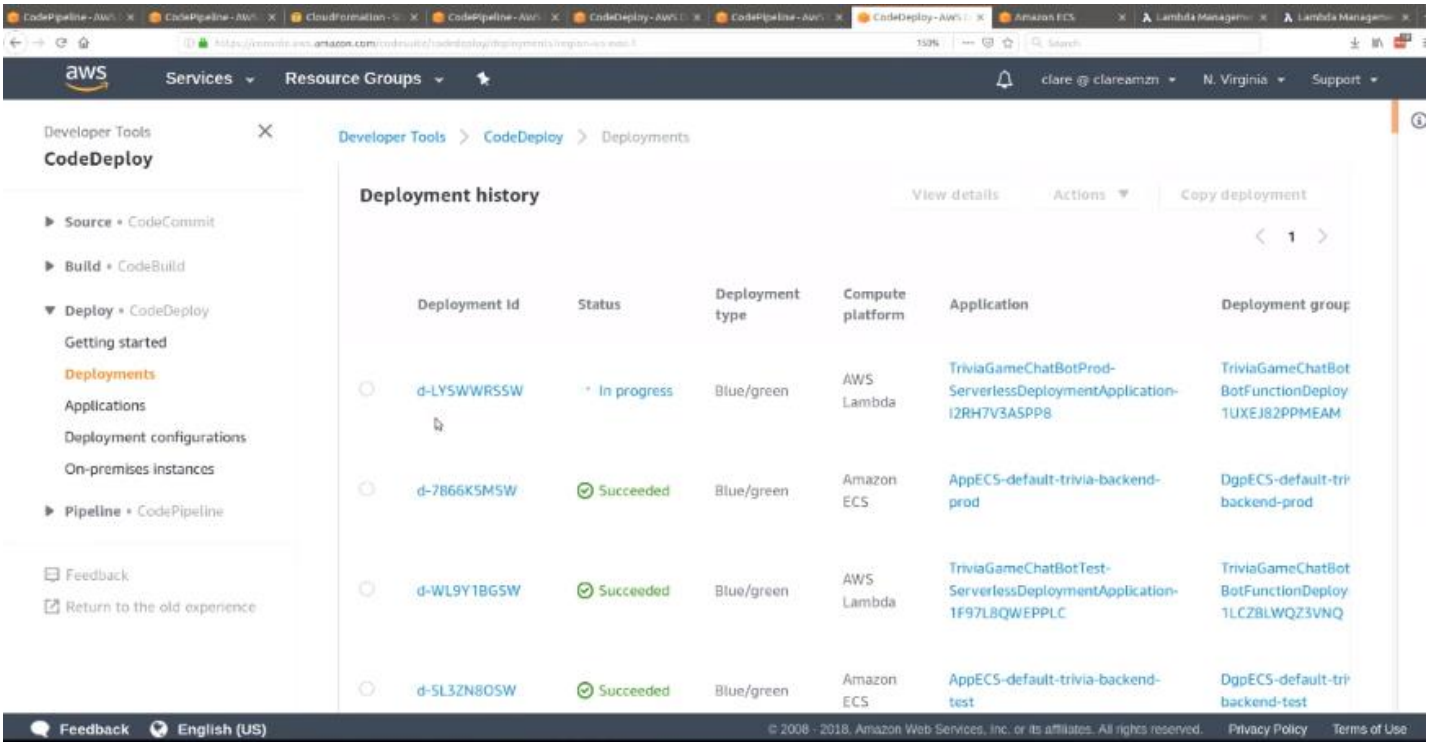

aws
Services
Resource Groups
clare @ clareamzn
N. Virginia
Support
Developer Tools
CodeDeploy
Source • CodeCommit
Build • CodeBuild
Deploy • CodeDeploy
Getting started
Deployments
Applications
Deployment configurations
On-premises instances
Pipeline • CodePipeline
Feedback
Return to the old experience
Developer Tools > CodeDeploy > Deployments
Deployment history
View details
Actions
Copy deployment
1
Deployment Id
Status
Deployment type
Compute platform
Application
Deployment group
d-WL9Y1BGSW
Succeeded
Blue/green
AWS Lambda
TriviaGameChatBotTest-ServerlessDeploymentApplication-1F97L8QWEPPLC
TriviaGameChatBotBotFunctionDeployr1LCZBLWQZ3VNQ
d-5L3ZN8O5W
Succeeded
Blue/green
Amazon ECS
AppECS-default-trivia-backend-test
DgpECS-default-trivbackend-test
d-0ZX8KJOQW
Succeeded
Blue/green
AWS Lambda
TriviaGameChatBotProd-ServerlessDeploymentApplication-I2RH7V3A5PP8
TriviaGameChatBotBotFunctionDeployr1UXEJ82PPMEAM
d-WOEC98LQW
Succeeded
Blue/green
Amazon ECS
AppECS-default-trivia-backend-prod
DgpECS-default-trivbackend-prod
Feedback
English (US)
© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

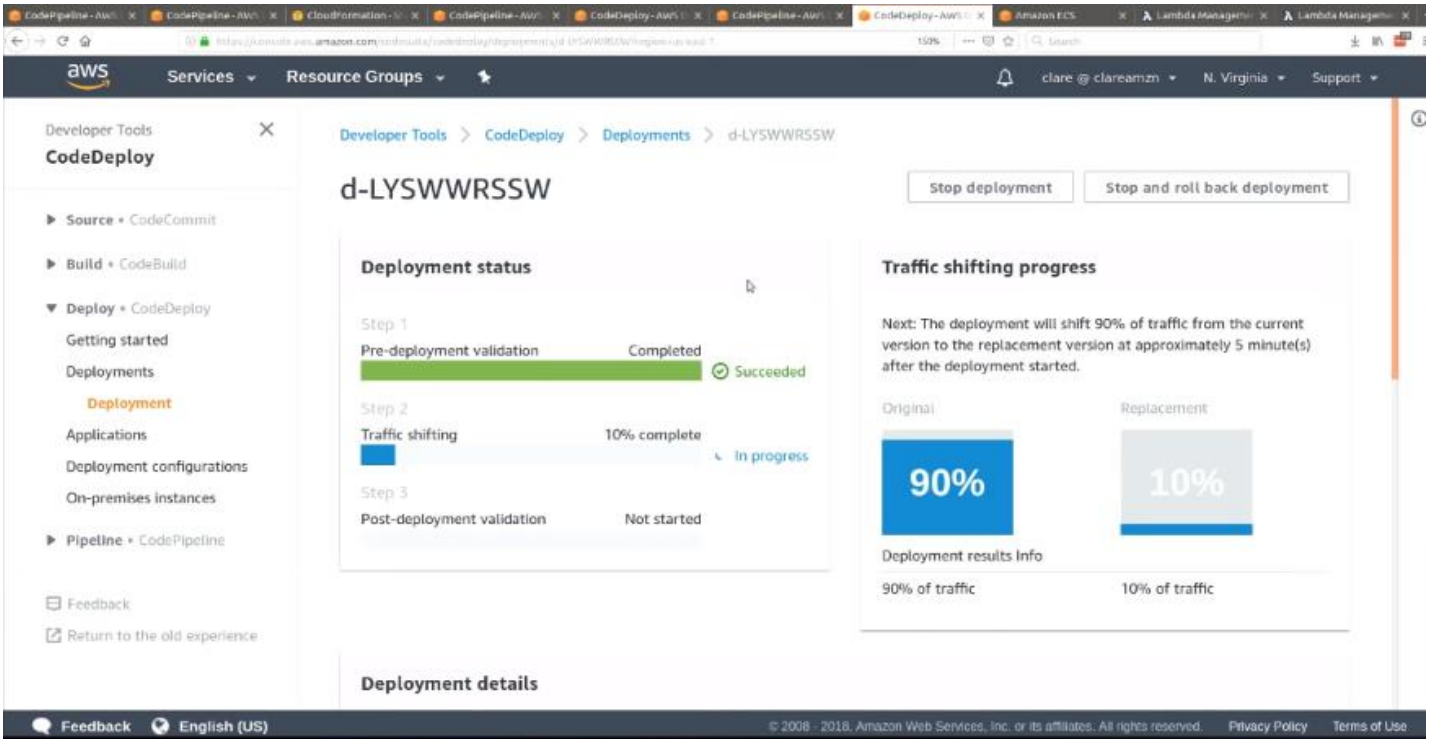

aws
Services
Resource Groups
clare @ clareamzn
N. Virginia
Support
Developer Tools
CodeDeploy
Source • CodeCommit
Build • CodeBuild
Deploy • CodeDeploy
Getting started
Deployments
Deployment
Applications
Deployment configurations
On-premises instances
Pipeline • CodePipeline
Feedback
Return to the old experience
ecs-svc/9223370493742380723
Original
ACTIVE
0
3
3
0
ecs-svc/9223370493477786367
Replacement
PRIMARY
100%
3
3
0
Deployment lifecycle events
Event
Duration
Status
Start time
End time
BeforeInstall
less than one second
Succeeded
Nov 27, 2018 7:49 PM
Nov 27, 2018 7:49 PM
Install
2 minutes 3 seconds
Succeeded
Nov 27, 2018 7:49 PM
Nov 27, 2018 7:51 PM
AfterInstall
less than one second
Succeeded
Nov 27, 2018 7:51 PM
Nov 27, 2018 7:51 PM
AllowTestTraffic
1 second
Succeeded
Nov 27, 2018 7:51 PM
Nov 27, 2018 7:51 PM
AfterAllowTestTraffic
less than one second
Succeeded
Nov 27, 2018 7:51 PM
Nov 27, 2018 7:51 PM
BeforeAllowTraffic
3 seconds
Succeeded
Nov 27, 2018 7:51 PM
Nov 27, 2018 7:52 PM
AllowTraffic
1 second
Succeeded
Nov 27, 2018 7:52 PM
Nov 27, 2018 7:52 PM
AfterAllowTraffic
less than one second
Succeeded
Nov 27, 2018 7:52 PM
Nov 27, 2018 7:52 PM
Feedback
English (US)
© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use

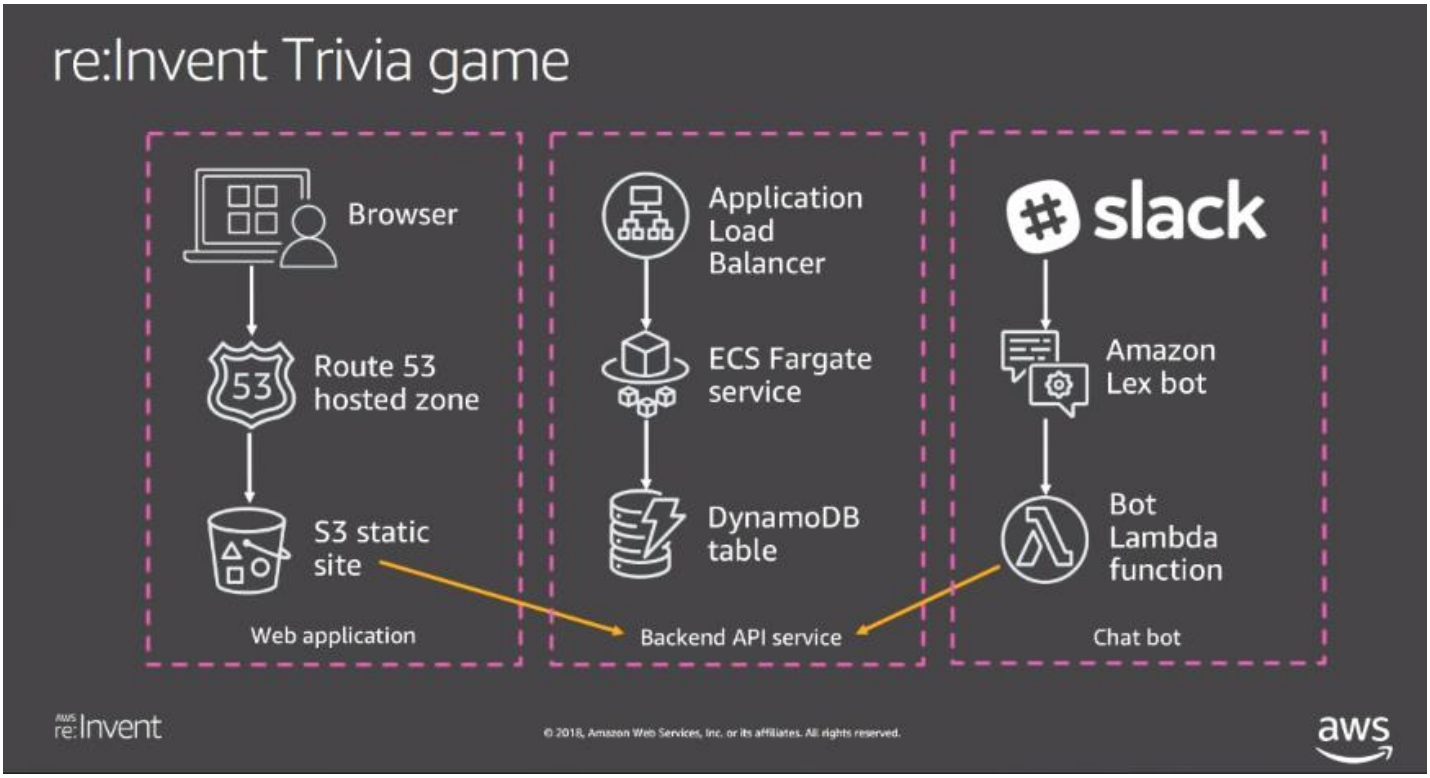

aws
Services
Resource Groups
clare @ clareamzn
N. Virginia
Support
Developer Tools
CodeDeploy
Source • CodeCommit
Build • CodeBuild
Deploy • CodeDeploy
Getting started
Deployments
Deployment
Applications
Deployment configurations
On-premises instances
Pipeline • CodePipeline
Feedback
Return to the old experience
Deployment status
Step 1:
Deploying replacement task set
Completed
Succeeded
Step 2:
Test traffic route setup
Completed
Succeeded
Step 3:
Rerouting production traffic to replacement task set
100% traffic shifted
Succeeded
Step 4:
Wait 1 minute 0 seconds
Waiting
In progress
Step 5:
Terminate original task set
Not started
Traffic shifting progress
Original
Replacement
0%
100%
Original task set not serving traffic
Replacement task set serving traffic
Deployment details
Feedback
English (US)
© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use


aws
Services
Resource Groups
clare @ clareamzn
N. Virginia
Support
Developer Tools
CodeDeploy
Source • CodeCommit
Build • CodeBuild
Deploy • CodeDeploy
Getting started
Deployments
Deployment
Applications
Deployment configurations
On-premises instances
Pipeline • CodePipeline
Feedback
Return to the old experience
Deployment status
Step 1:
Deploying replacement task set
Completed
Succeeded
Step 2:
Test traffic route setup
Completed
Succeeded
Step 3:
Rerouting production traffic to replacement task set
100% traffic shifted
Succeeded
Step 4:
Wait
Wait completed
Succeeded
Step 5:
Terminate original task set
Completed
Succeeded
Traffic shifting progress
Original
Replacement
0%
100%
Original task set not serving traffic
Replacement task set serving traffic
Deployment details
Feedback
English (US)
© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use


Services
Resource Groups
clare @ clareamzn
N. Virginia
Support
Developer Tools
CodePipeline
Source • CodeCommit
Build • CodeBuild
Deploy • CodeDeploy
Pipeline • CodePipeline
Getting started
Pipelines
Pipeline
History
Feedback
Return to the old experience
DeployTest
Amazon ECS (Blue/Green)
Succeeded - 8 hours ago
Details
GitHubSource: Update chalk talk count; return answer in chat bot response dce357b1
ECRSource: sha256:64763a9cf249
Disable transition
Prod
View current revisions
DeployProd
Amazon ECS (Blue/Green)
Succeeded - Just now
Details
GitHubSource: Update chalk talk count; return answer in chat bot response dce357b1
ECRSource: sha256:64763a9cf249
Feedback
English (US)
© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use


Services
Resource Groups
clare @ clareamzn
N. Virginia
Support
Developer Tools
CodeDeploy
Source • CodeCommit
Build • CodeBuild
Deploy • CodeDeploy
Getting started
Deployments
Applications
Deployment configurations
On-premises instances
Pipeline • CodePipeline
Feedback
Return to the old experience
Developer Tools > CodeDeploy > Deployments
Deployment history
View details
Actions
Copy deployment
1
Deployment Id
Status
Deployment type
Compute platform
Application
Deployment group
d-LY5WWRSSW
In progress
Blue/green
AWS Lambda
TriviaGameChatBotProd-ServerlessDeploymentApplication-I2RH7V3A5PP8
TriviaGameChatBot-BotFunctionDeploy-1UXEJ82PPMEAM
d-7866K5M5W
Succeeded
Blue/green
Amazon ECS
AppECS-default-trivia-backend-prod
DgpECS-default-trivia-backend-prod
d-WL9Y1BG5W
Succeeded
Blue/green
AWS Lambda
TriviaGameChatBotTest-ServerlessDeploymentApplication-1F97L8QWEPPLC
TriviaGameChatBot-BotFunctionDeploy-1LCZBLWQZ3VNQ
d-5L3ZN8OSW
Succeeded
Blue/green
Amazon ECS
AppECS-default-trivia-backend-test
DgpECS-default-trivia-backend-test
Feedback
English (US)
© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use


Services
Resource Groups
clare @ clareamzn
N. Virginia
Support
Developer Tools
CodeDeploy
Source • CodeCommit
Build • CodeBuild
Deploy • CodeDeploy
Getting started
Deployments
Deployment
Applications
Deployment configurations
On-premises instances
Pipeline • CodePipeline
Feedback
Return to the old experience
Developer Tools > CodeDeploy > Deployments > d-LYSWWRSSW
d-LYSWWRSSW
Stop deployment
Stop and roll back deployment
Deployment status
Step 1
Pre-deployment validation
Completed
Succeeded
Step 2
Traffic shifting
10% complete
In progress
Step 3
Post-deployment validation
Not started
Traffic shifting progress
Next: The deployment will shift 90% of traffic from the current version to the replacement version at approximately 5 minute(s) after the deployment started.
Original
Replacement
90%
10%
Deployment results Info
90% of traffic
10% of traffic
Deployment details
Feedback
English (US)
© 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use


re:Invent Trivia game
Browser
Route 53 hosted zone
S3 static site
Web application
Application Load Balancer
ECS Fargate service
DynamoDB table
Backend API service
slack
Amazon Lex bot
Bot Lambda function
Chat bot
re:Invent
© 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved.
aws

Related sessions

SRV325

Leadership Session: Using DevOps, Microservices, and Serverless to Accelerate Innovation
Thursday, Nov 29 | 12:15 – 1:15 pm | Venetian, Level 2, Venetian Theater

SRV305

Inside AWS: Technology Choices for Modern Applications
Wednesday, Nov 28 | 5:30 – 6:30 pm | MGM, Level 3, Premier Ballroom 316
Thursday, Nov 29 | 2:30 – 3:30 pm | Aria East, Level 2, Mariposa 5

DEV327

Beyond the Basics: Advanced Infrastructure as Code Programming on AWS
Thursday, Nov 29 | 4:00 – 5:00 pm | Aria West, Level 3, Ironwood 5

Thank you!

Clare Liguori