CICD for PHP on AWS

SunshinePHP 2019

Michael Moussa
@michaelmoussa
https://joind.in/talk/05470

Objective

Objective

Set up CICD for a non-trivial PHP application running on AWS, and deploy it with zero downtime.

CICD?

Continuous **I**ntegration

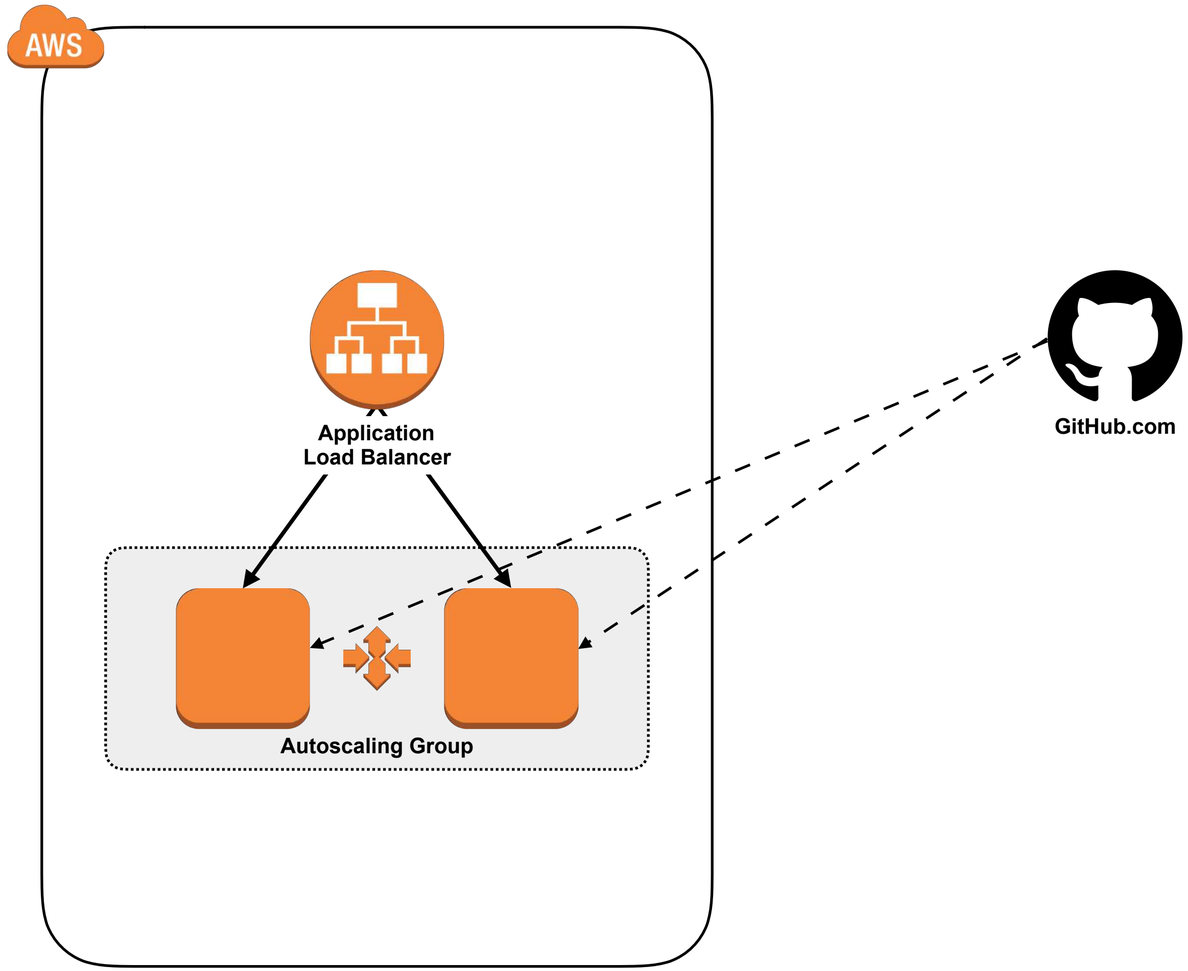
CICD?

Continuous ~~**D**eployment~~

CICD?

Continuous **D**elivery

Getting started



CodeDeploy

CodeDeploy

- Automates application deployments
  - EC2 instances
  - AWS Lambda
  - ECS (Elastic Container Service)
  - On-premise servers
- Deploy from GitHub or Amazon S3
- Highly customizable

CodeDeploy - Example

Assumptions

- "Hello, world!" application code hosted on GitHub
- Autoscaling group (3 instances) with application running
- First time setup steps completed:
  - https://docs.aws.amazon.com/codedeploy/latest/userguide/getting-started-codedeploy.html
- CodeDeploy Agent installed:
  - https://docs.aws.amazon.com/codedeploy/latest/userguide/codedeploy-agent-operations-install.html

CodeDeploy - Get started

Create AWS CodeDeploy deployment

Get started with AWS CodeDeploy by creating your first deployment application.

Create application

CodeDeploy - Create application

Application name

Enter an application name

DemoApplication

100 character limit

Compute platform

Choose a compute platform

EC2/On-premises

CodeDeploy - Create application

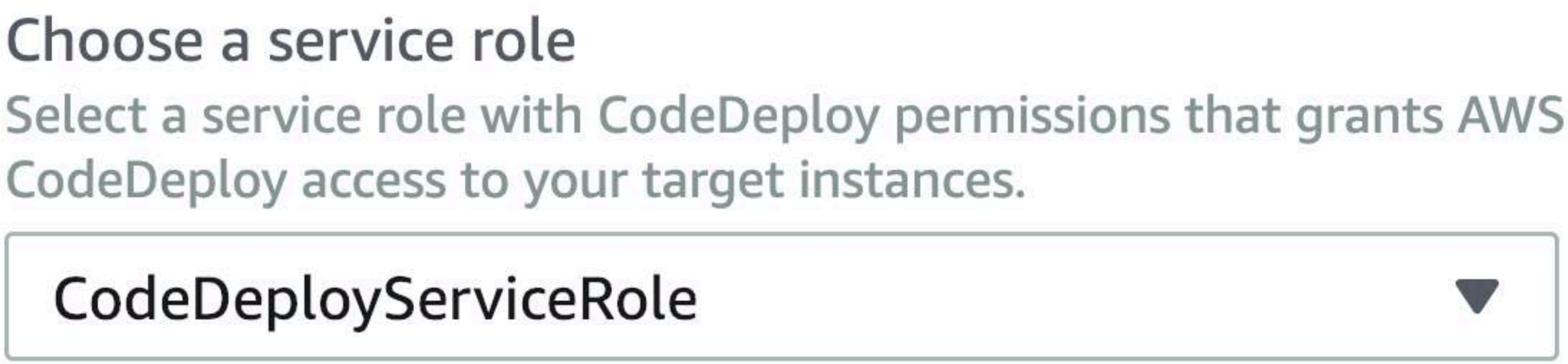
No deployment groups

Before you can deploy your application using CodeDeploy, you must create a deployment group.

Create deployment group

CodeDeploy - Create application

CodeDeploy - Create application



CodeDeploy - Deployment type

Choose how to deploy your application

In-place

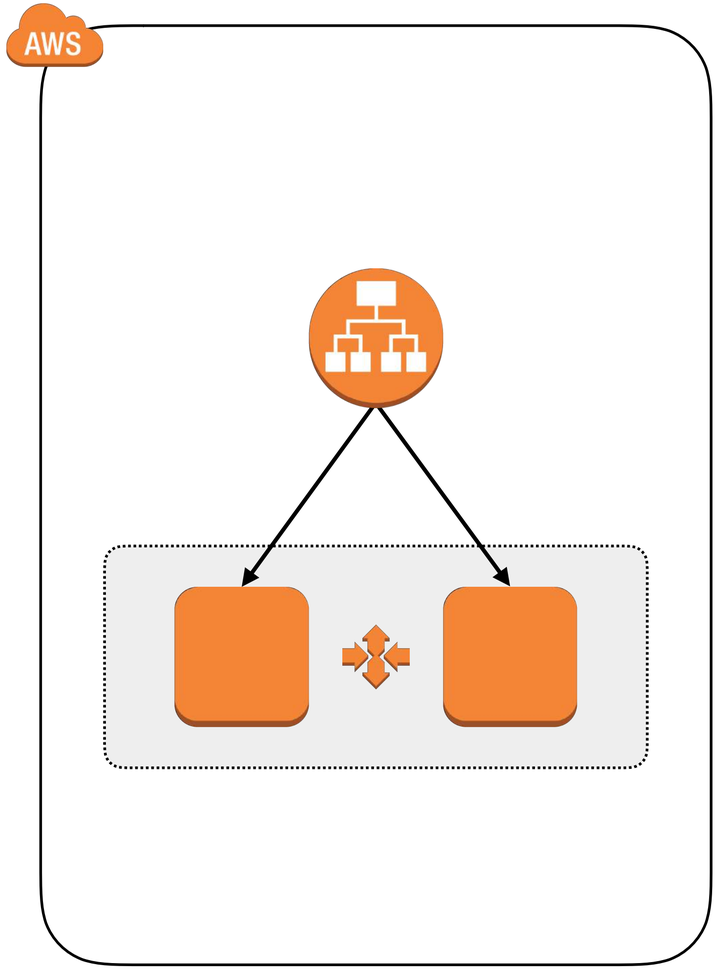
Updates the instances in the deployment group with the latest application revisions. During a deployment, each instance will be briefly taken offline for its update

Blue/green

Replaces the instances in the deployment group with new instances and deploys the latest application revision to them. After instances in the replacement environment are registered with a load balancer, instances from the original environment are deregistered and can be terminated.

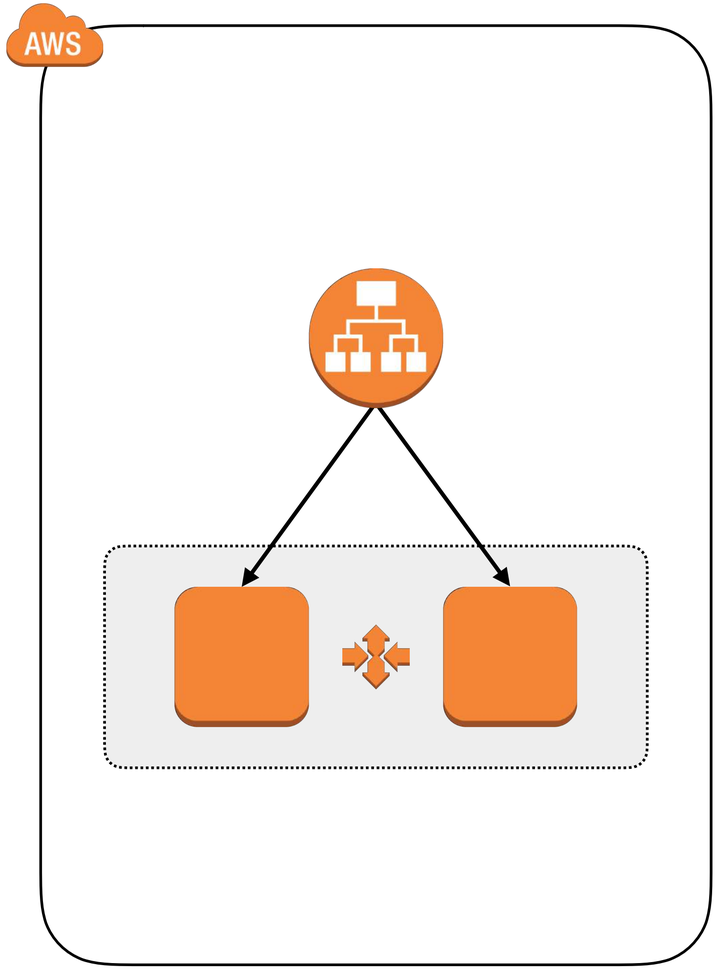
CodeDeploy - Deployment type

- **In-place deployment**
- Blue/green deployment



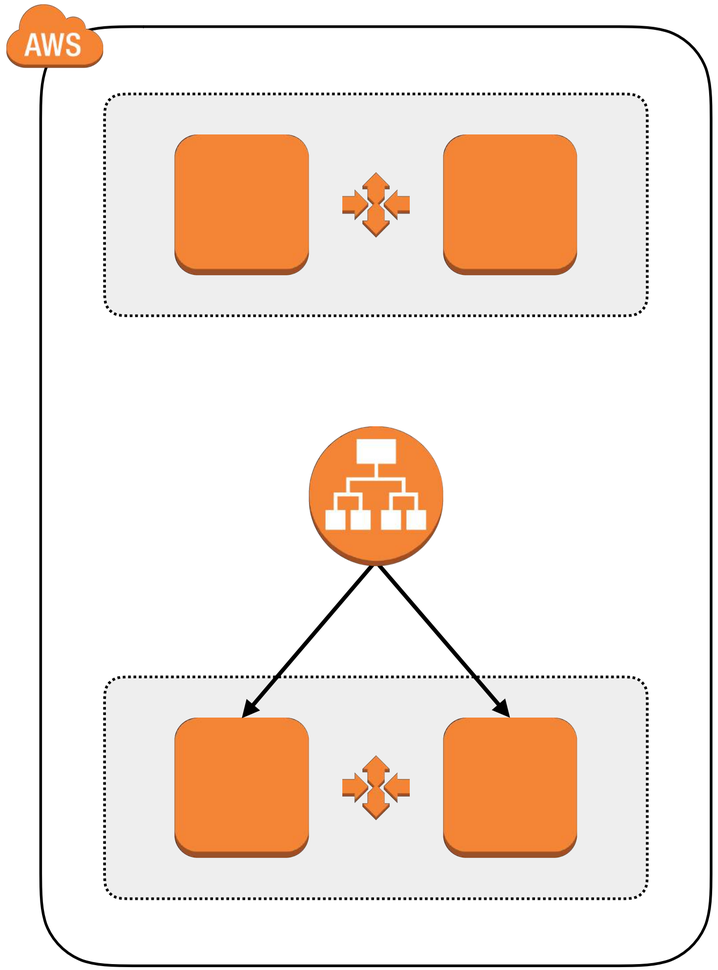
CodeDeploy - Deployment type

- In-place deployment
- **Blue/green deployment**



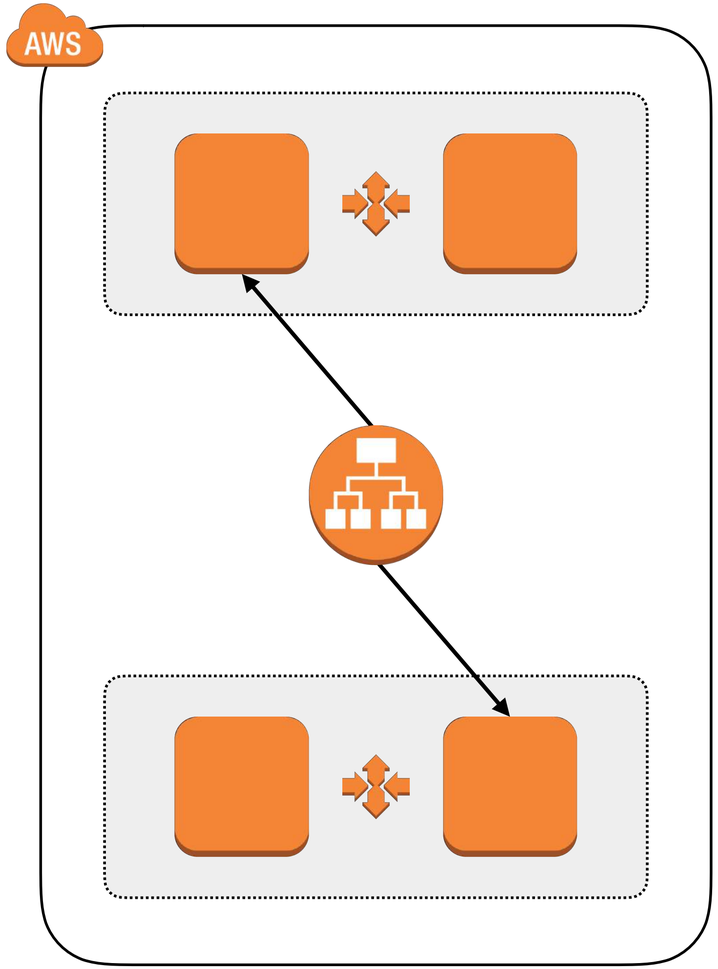
CodeDeploy - Deployment type

- In-place deployment
- **Blue/green deployment**



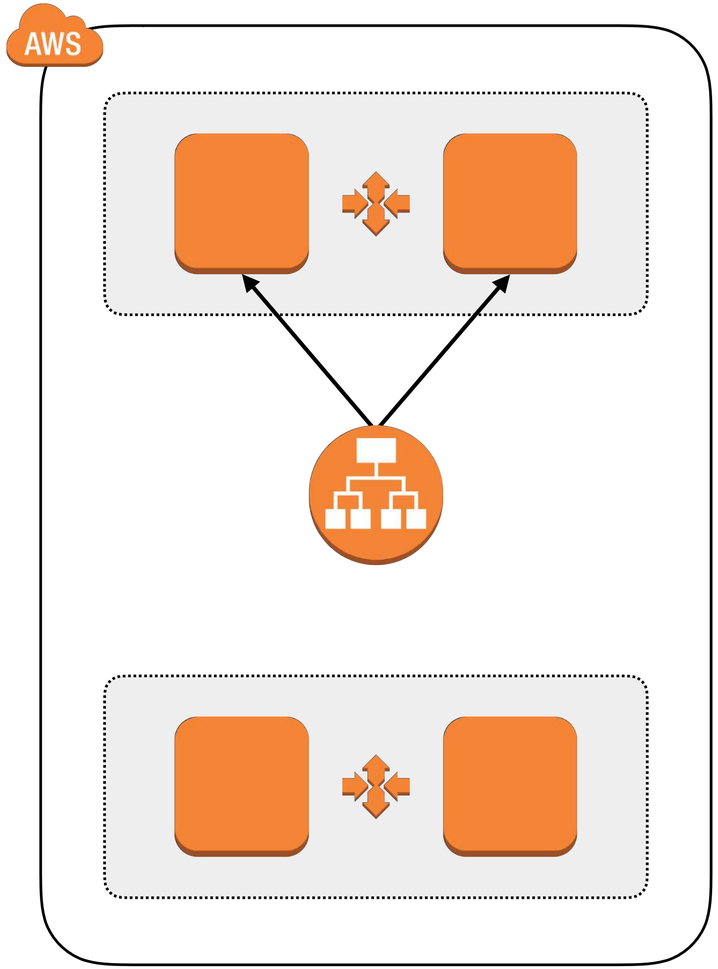
CodeDeploy - Deployment type

- In-place deployment
- **Blue/green deployment**



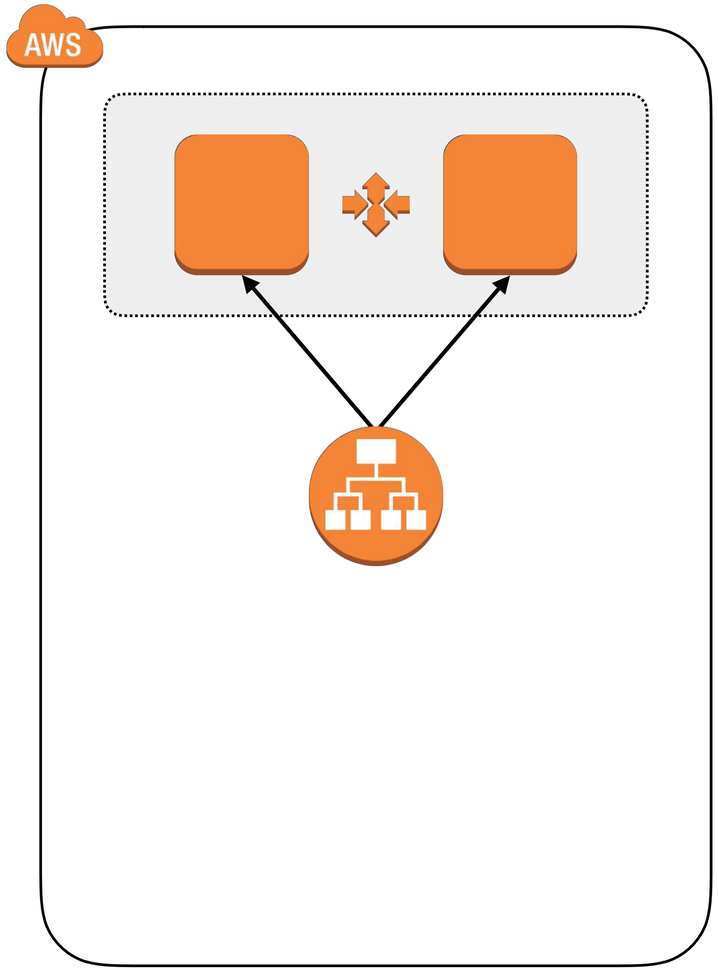
CodeDeploy - Deployment type

- In-place deployment
- **Blue/green deployment**



CodeDeploy - Deployment type

- In-place deployment
- **Blue/green deployment**



CodeDeploy - Deployment type

Choose how to deploy your application

○ **In-place**
Updates the instances in the deployment group with the latest application revisions. During a deployment, each instance will be briefly taken offline for its update

◉ **Blue/green**
Replaces the instances in the deployment group with new instances and deploys the latest application revision to them. After instances in the replacement environment are registered with a load balancer, instances from the original environment are deregistered and can be terminated.

CodeDeploy - Environment configuration

Specify the Amazon EC2 Auto Scaling groups or Amazon EC2 instances where the current application revision is deployed.

- (●) **Automatically copy Amazon EC2 Auto Scaling group**
  Provision an Amazon EC2 Auto Scaling group and deploy the new application revision to it. AWS CodeDeploy will create the Auto Scaling group by copying the one you specify here.

- ( ) **Manually provision instances**
  I will specify here the instances where the current application revision is running. I will specify the instances for the replacement environment when I create a deployment.

CodeDeploy - Environment configuration

Choose the Amazon EC2 Auto Scaling group where the current application revision is deployed.

demo-application

CodeDeploy - Deployment settings

Traffic rerouting

- (•) Reroute traffic immediately
- ( ) I will choose whether to reroute traffic

Choose whether instances in the original environment are terminated after the deployment is succeeds, and how long to wait before termination.

- (•) Terminate the original instances in the deployment group
- ( ) Keep the original instances in the deployment group running

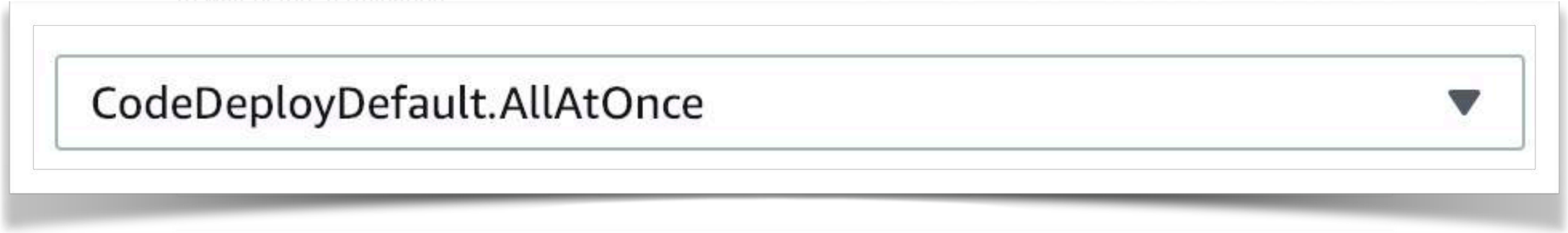
Days	Hours	Minutes
0 ▼	1 ▼	0 ▼

Deployment configuration

Choose from a list of default and custom deployment configurations. A deployment configuration is a set of rules that determines how fast

CodeDeployDefault.AllAtOnce ▼ or **Create deployment configuration**

CodeDeploy - Deployment settings



CodeDeploy - Deployment settings

CodeDeploy - Deployment settings

Traffic rerouting

Reroute traffic immediately

I will choose whether to reroute traffic

- (●) Terminate the original instances in the deployment group
- ( ) Keep the original instances in the deployment group running

Days	Hours	Minutes
0	1	0

Deployment configuration

Choose from a list of default and custom deployment configurations. A deployment configuration is a set of rules that determines how fast

CodeDeployDefault.AllAtOnce or Create deployment configuration

rackspace
the #1 managed cloud company

CodeDeploy - Environment configuration

Select a load balancer to manage incoming traffic during the deployment process. The load balancer blocks traffic from each instance while it's being deployed to and allows traffic to it again after the deployment succeeds.

☑ Enable load balancing

◉ Application Load Balancer or Network Load Balancer

○ Classic Load Balancer

Choose a load balancer

demo-application ▼

appspec.yml

CodeDeploy - appspec.yml

```yaml
version: 0.0
os: linux
files:
  - source: /
    destination: /var/www/demo-application/
permissions:
  - object: /var/www/demo-application
    owner: root
    group: nginx
    mode: 750
    type:
      - directory
  - object: /var/www/demo-application
    owner: root
    group: nginx
    mode: 640
    type:
      - file
hooks:
  BeforeInstall:
    - location: .aws/codedeploy/hooks/before-install.sh
      timeout: 300
      runas: root
```

CodeDeploy - appspec.yml

```
version: 0.0
os: linux
files:
  - source: /
    destination: /var/www/demo-application/
permissions:
  - object: /var/www/demo-application
    owner: root
    group: nginx
    mode
    type
      -
  - obje
    owne
    group: nginx
    mode: 640
    type:
      - file
hooks:
  BeforeInstall:
    - location: .aws/codedeploy/hooks/before-install.sh
      timeout: 300
      runas: root
```

```
version: 0.0
```

CodeDeploy - appspec.yml

```
version: 0.0
os: linux
files:
  - source: /
    destination: /var/www/demo-application/
permissions:
  - object: /var/www/demo-application
    owner: root
    group: nginx
    mode: 75
    type:
      - dire
  - object:
    owner: r
    group: nginx
    mode: 640
    type:
      - file
hooks:
  BeforeInstall:
    - location: .aws/codedeploy/hooks/before-install.sh
      timeout: 300
      runas: root
```

os: linux

CodeDeploy - appspec.yml

```
version: 0.0
os: linux
files:
  - source: /
    destination: /var/www/demo-application/
permissions:
    object: /var/www/demo-application
    type:
      - file
hooks:
```

```
files:
  - source: /
    destination: /var/www/demo-application/
```

https://docs.aws.amazon.com/codedeploy/latest/userguide/reference-appspec-file-structure-files.html

CodeDeploy - appspec.yml

```yaml
permissions:
  - object: /var/www/demo-application
    owner: root
    group: nginx
    mode: 750
    type:
      - directory
  - object: /var/www/demo-application
    owner: root
    group: nginx
    mode: 640
    type:
      - file
```

https://docs.aws.amazon.com/codedeploy/latest/userguide/reference-appspec-file-structure-permissions.html

CodeDeploy - appspec.yml

```yaml
version: 0.0
os: linux
files:
  - source: /
    destination: /var/www/demo-application/
permissions:
  - object: /var/www/demo-application
    type:
      - file
hooks:
  BeforeInstall:
    - location: .aws/codedeploy/hooks/before-install.sh
      timeout: 300
      runas: root
```

rackspace
the #1 managed cloud company

CodeDeploy - appspec.yml

```
version: 0.0
os: linux
files:
  - source: /
    destination: /var/www/demo-application/
permissions:
  - object: /var/www/demo-application
```

```
hooks:
  BeforeInstall:
    - location: .aws/codedeploy/hooks/before-install.sh
      timeout: 300
      runas: root
```

```
#!/bin/bash

rm -rf /var/www/demo-application/*
```

CodeDeploy - appspec.yml

```yaml
version: 0.0
os: linux
files:
  - source: /
    destination: /var/www/demo-application/
permissions:
  - object: /var/www/demo-application
```

```yaml
hooks:
  BeforeInstall:
    - location: .aws/codedeploy/hooks/before-install.sh
      timeout: 300
      runas: root
```

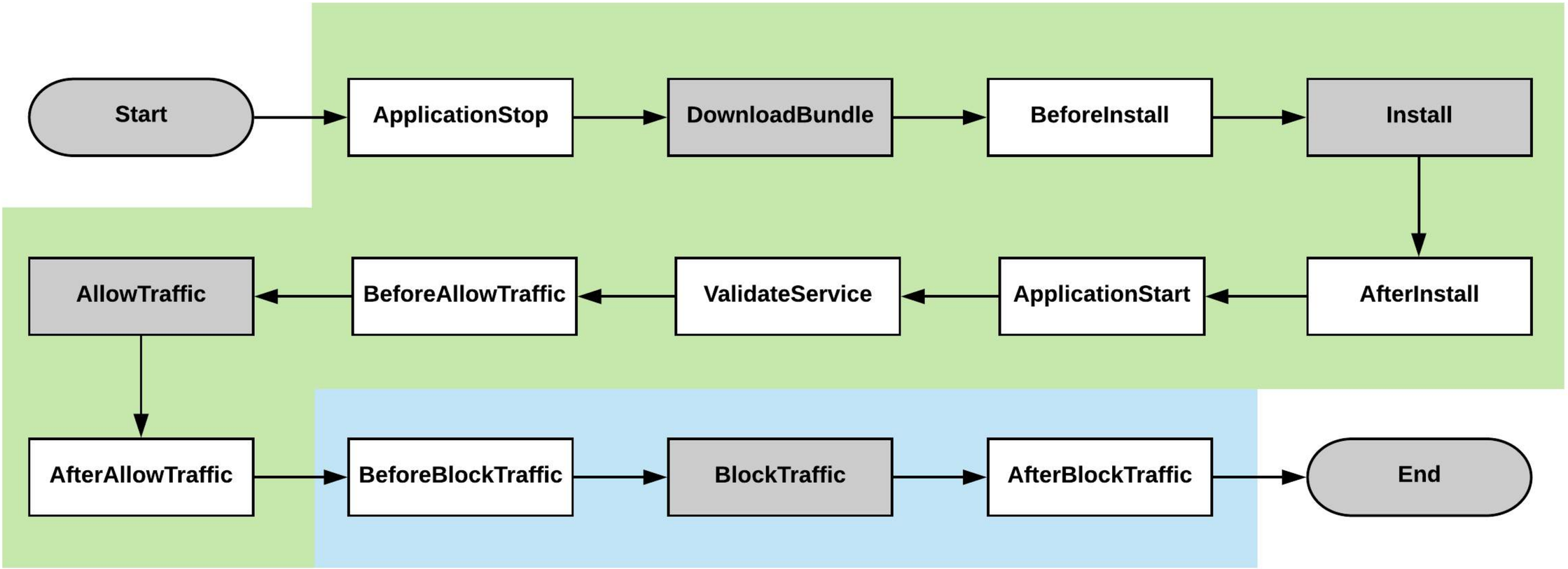
```yaml
    type:
      - file
```

https://docs.aws.amazon.com/codedeploy/latest/userguide/reference-appspec-file-structure-hooks.html

CodeDeploy - appspec.yml

```yaml
version: 0.0
os: linux
files:
  - source: /
    destination: /var/www/demo-application/
permissions:
  - object: /var/www/demo-application
    owner: root
    group: nginx
    mode: 750
    type:
      - directory
  - object: /var/www/demo-application
    owner: root
    group: nginx
    mode: 640
    type:
      - file
hooks:
  BeforeInstall:
    - location: .aws/codedeploy/hooks/before-install.sh
      timeout: 300
      runas: root
```

CodeDeploy - `appspec.yml` - Event lifecycle



CodeDeploy - Deploy application

Application

DemoApplication

Deployment group

DemoApplicationTest

CodeDeploy - Deploy application

Revision type

- ○ My application is stored in Amazon S3
- ◉ My application is stored in GitHub

GitHub token name

Select the name of the token associated to an account you have already connected, or grant AWS CodeDeploy permission to access a different account. To connect to a GitHub account for the first time, type an alias for the account, and then choose Connect to GitHub

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

Connected

Application DemoApplication successfully bound to XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX GitHub token

CodeDeploy - Deploy application

Repository name

<organization-name>/<repository-name>

Commit ID

<full 40 character git commit hash>

CodeDeploy - Deploy application

Deployment description - optional

Add a brief description about the deployment

Our first deployment!

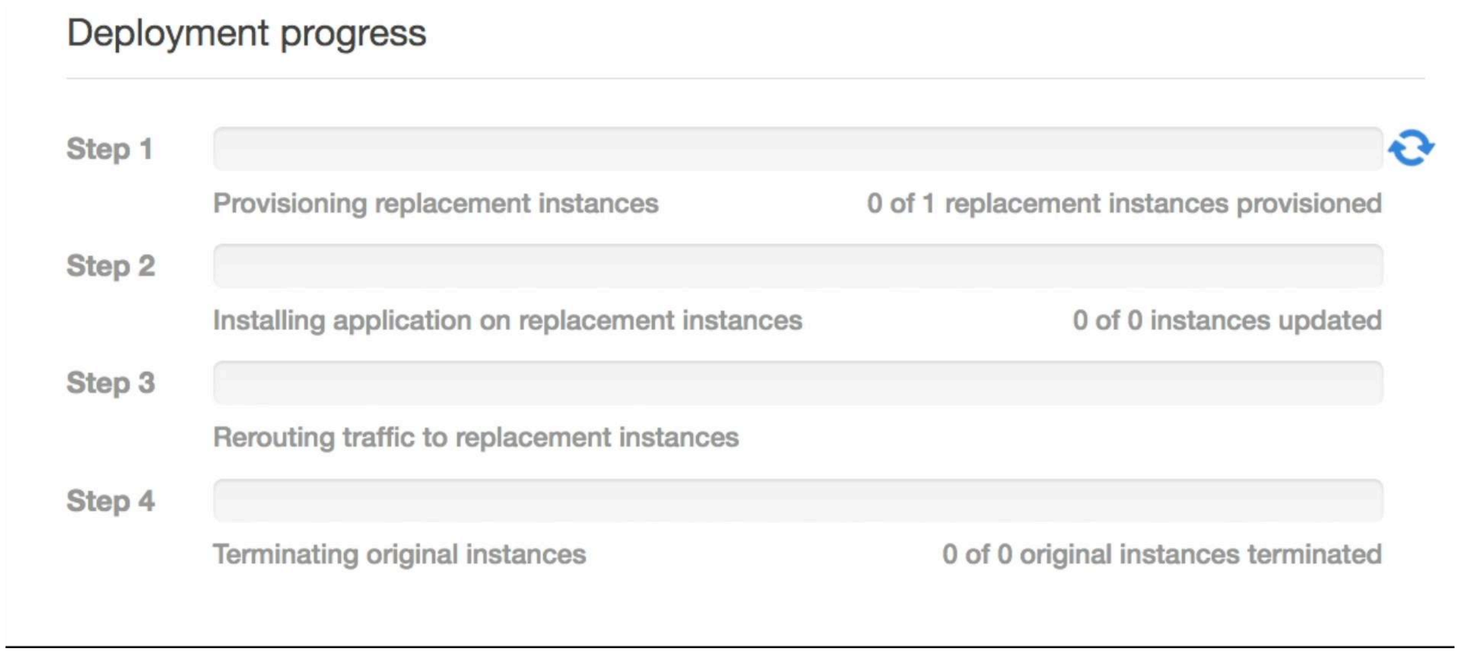
CodeDeploy - Deploy application

Content options

Choose what to do during a deployment when a file on a target instance has the same name as a file in the application revision

- ○ Fail the deployment
  An error is reported and the deployment status is changed to Failed.
- ○ Overwrite the content
  The file in the application revision is copied to the target location on the instance, replacing the previous file.
- ○ Retain the content
  The file in the application revision is not copied to the instance. The existing file is kept at the target location and treated as part of the new deployment.

CodeDeploy - Deploy application



Other capabilities

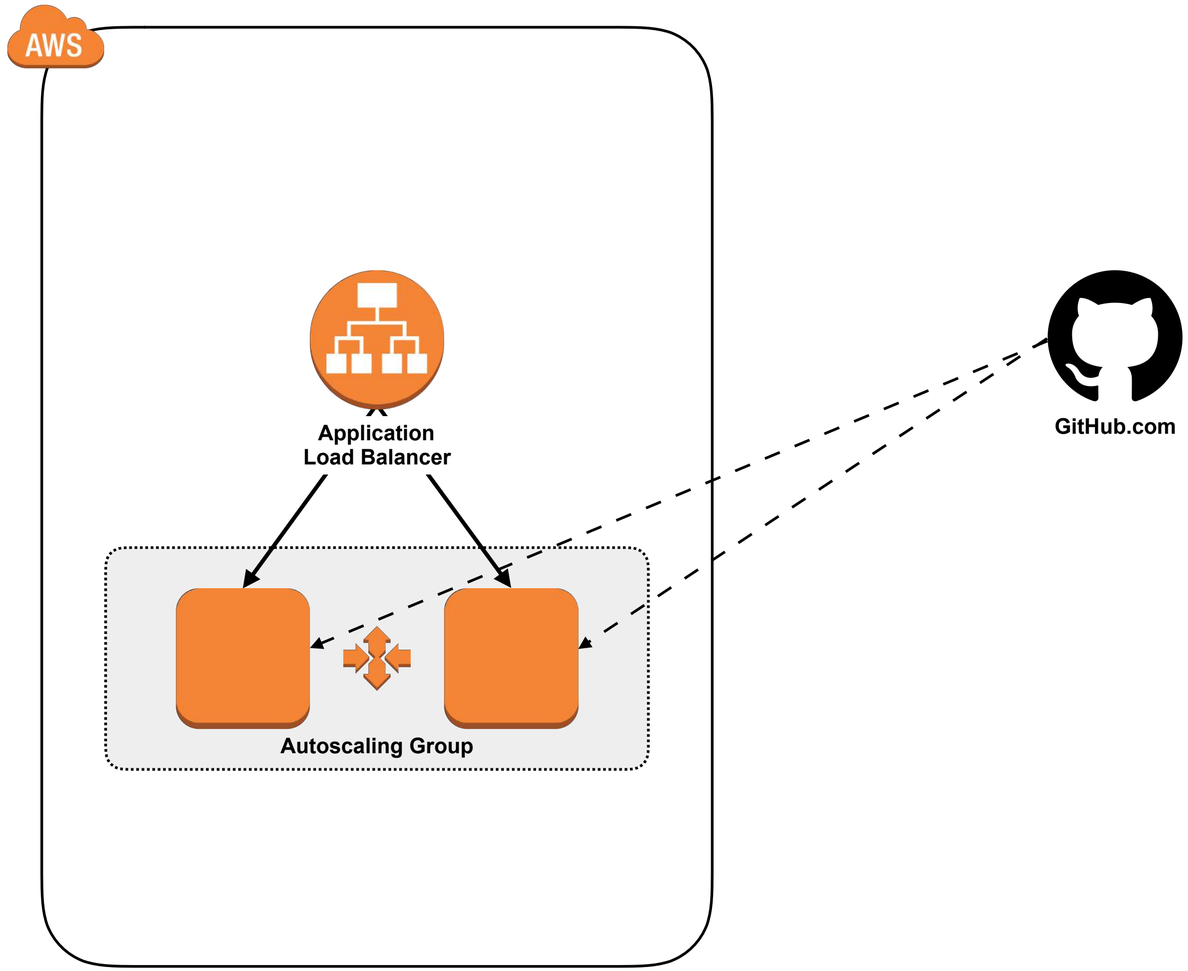
CodeDeploy - Other capabilities

- **SNS Triggers**
  - Deployment (`start|succeeds|fails|stops|ready|rollback`)
  - Instance (`starts|succeeds|fails|ready`)
- **CloudWatch Alarms**
  - Stop deployments if metric threshold is breached
- **Automatic Rollback**

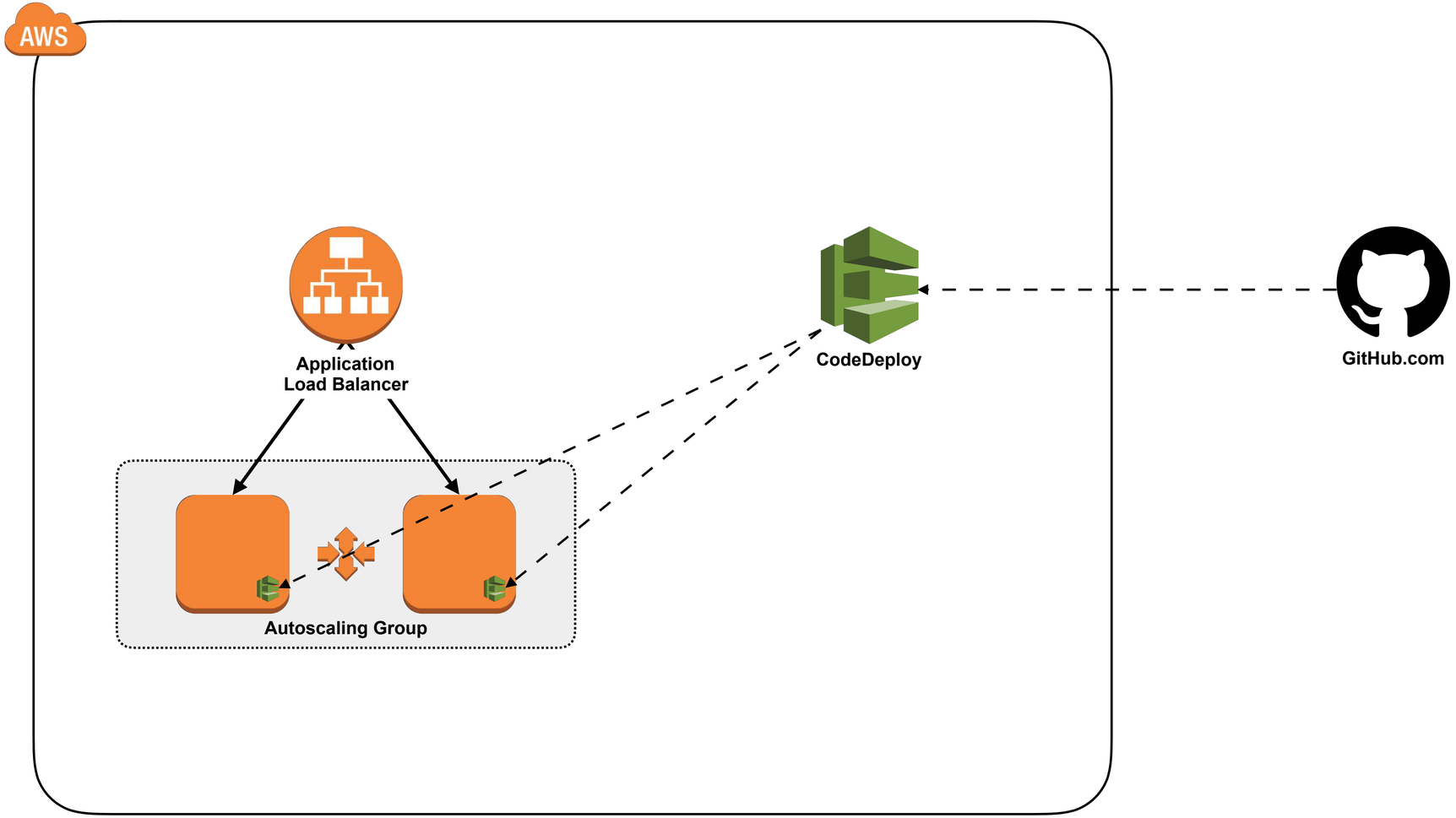
Objective

Set up CICD for a non-trivial PHP application running on AWS, and deploy it with zero downtime.

Objective


AWS
Application
Load Balancer
Autoscaling Group
GitHub.com

Objective


AWS
Application
Load Balancer
CodeDeploy
GitHub.com
Autoscaling Group

Objective

Set up CICD for a non-trivial PHP application running on AWS, and deploy it with zero downtime.

Objective

Set up CICD for a ***non-trivial*** PHP application running on AWS, and deploy it with zero downtime.

CodeBuild

CodeBuild

- Fully managed build service
  - Compiles, tests, and packages code
- Scales continuously
- Pay-as-you-go
- *Also* highly customizable!

CodeBuild - Get started

Create AWS CodeBuild project

Get started with AWS CodeBuild by creating your first build project.

Create project

CodeBuild - Get started

Project name

DemoApplication

A project name must be 2 to 255 characters. It can include the letters A-Z and a-z, the numbers 0-9, and the special characters - and _.

Description - *optional*

An AWS CodeDeploy PHP demo application!

Build badge - *optional*

☐ Enable build badge

CodeBuild - Get started

Project name

DemoApplication

A project name must be 2 to 255 characters. It can include the letters A-Z and a-z, the numbers 0-9, and the special characters - and _.

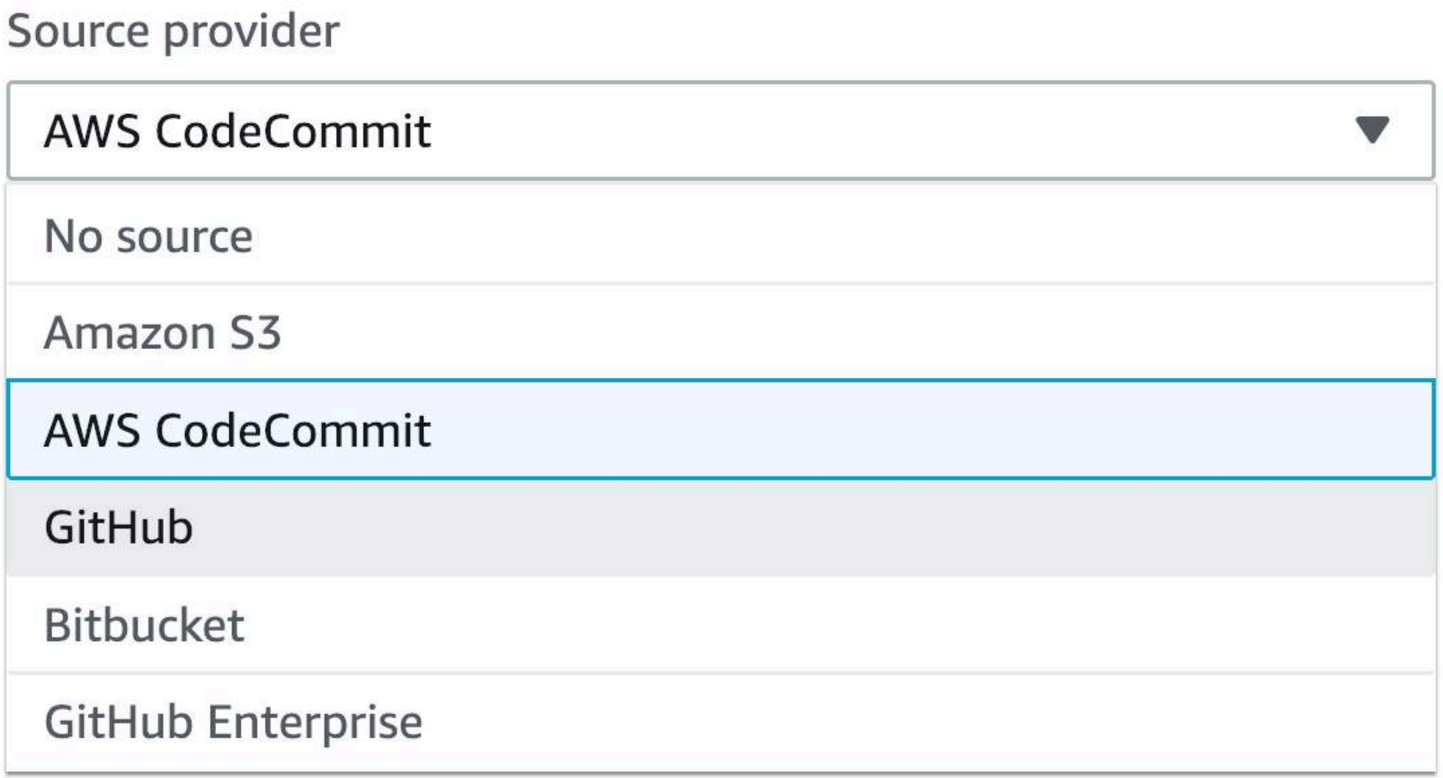
Description - *optional*

An AWS CodeDeploy PHP demo application!

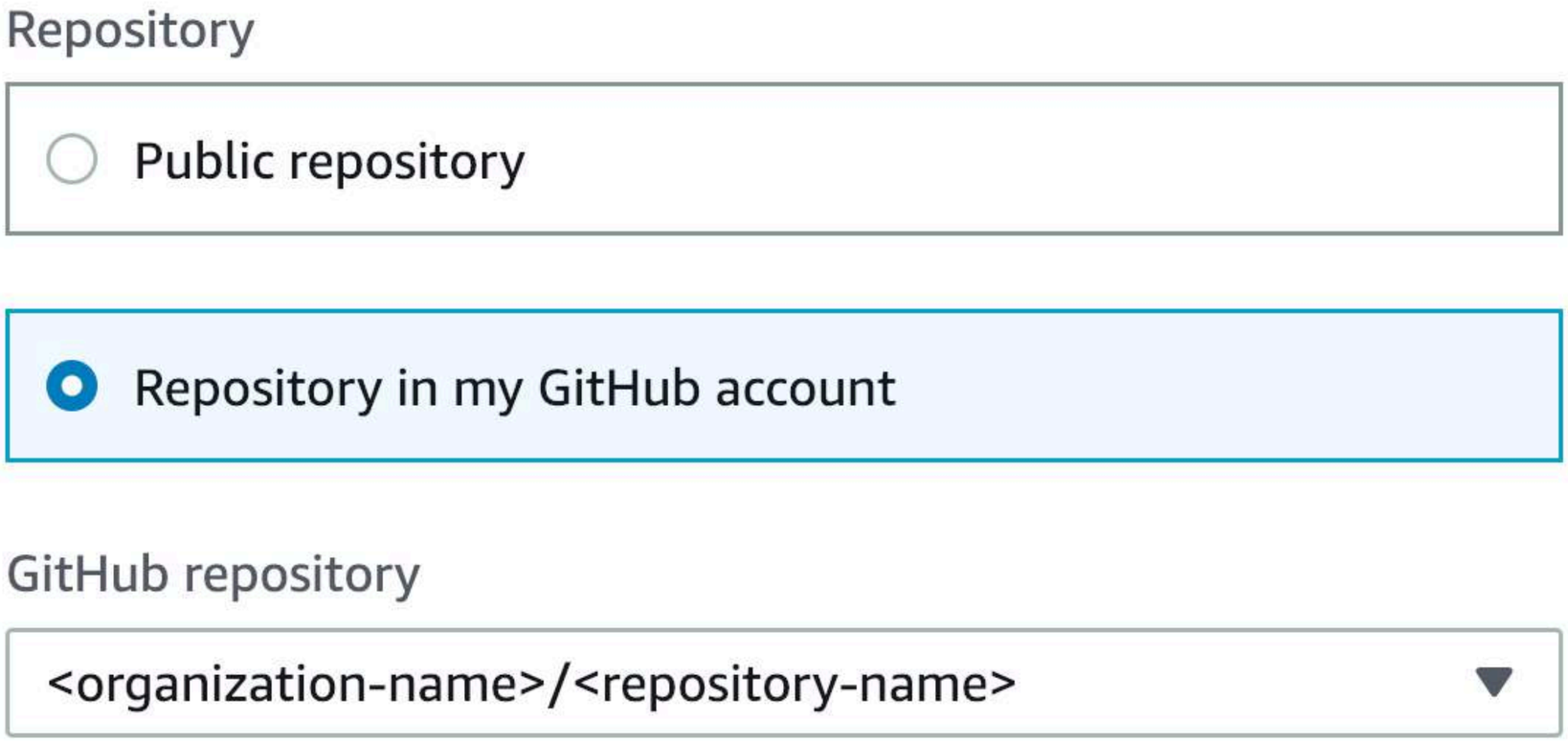
Build badge - *optional*

☑ Enable build badge

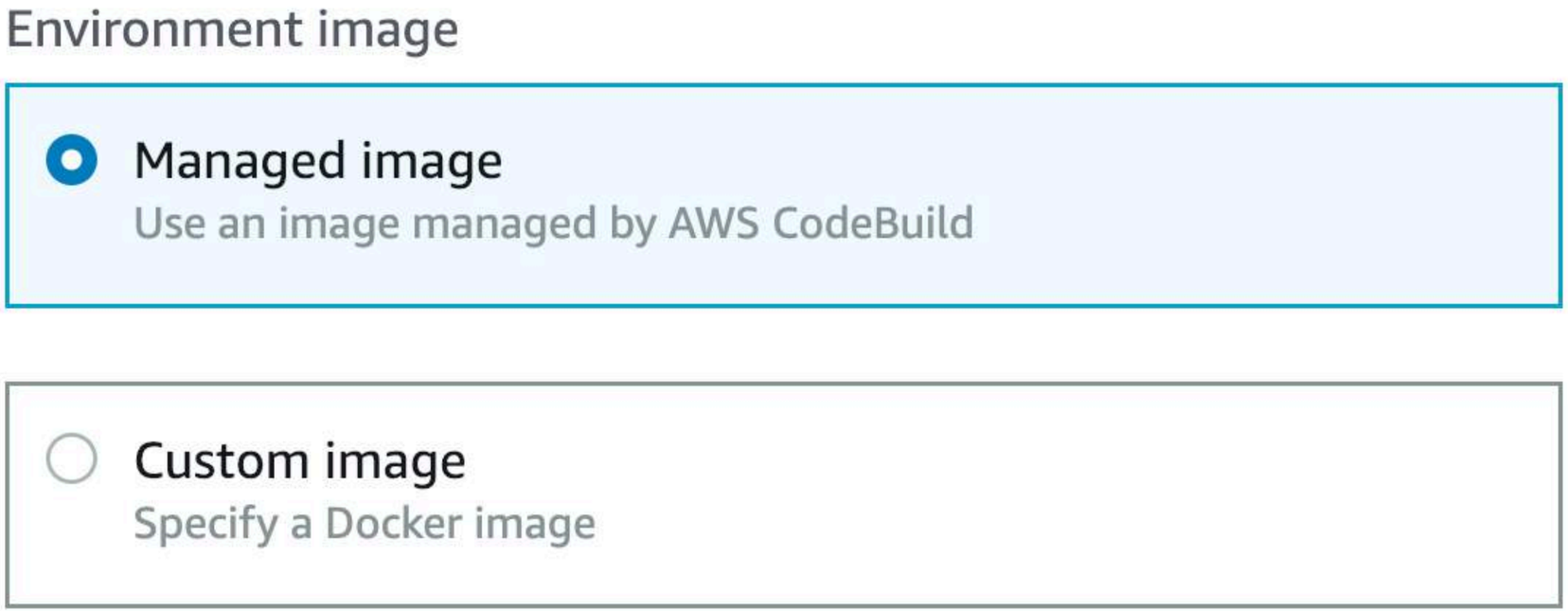
CodeBuild - Source



CodeBuild - Source



CodeBuild - Environment



CodeBuild - Environment

Operating system

Ubuntu

Runtime

PHP

CodeBuild - Environment

CodeBuild - Environment

Environment image

Managed image
Use an image managed by AWS CodeBuild

Custom image
Specify a Docker image

Environment type
Choose an environment type

Linux

CodeBuild - Environment

Image registry

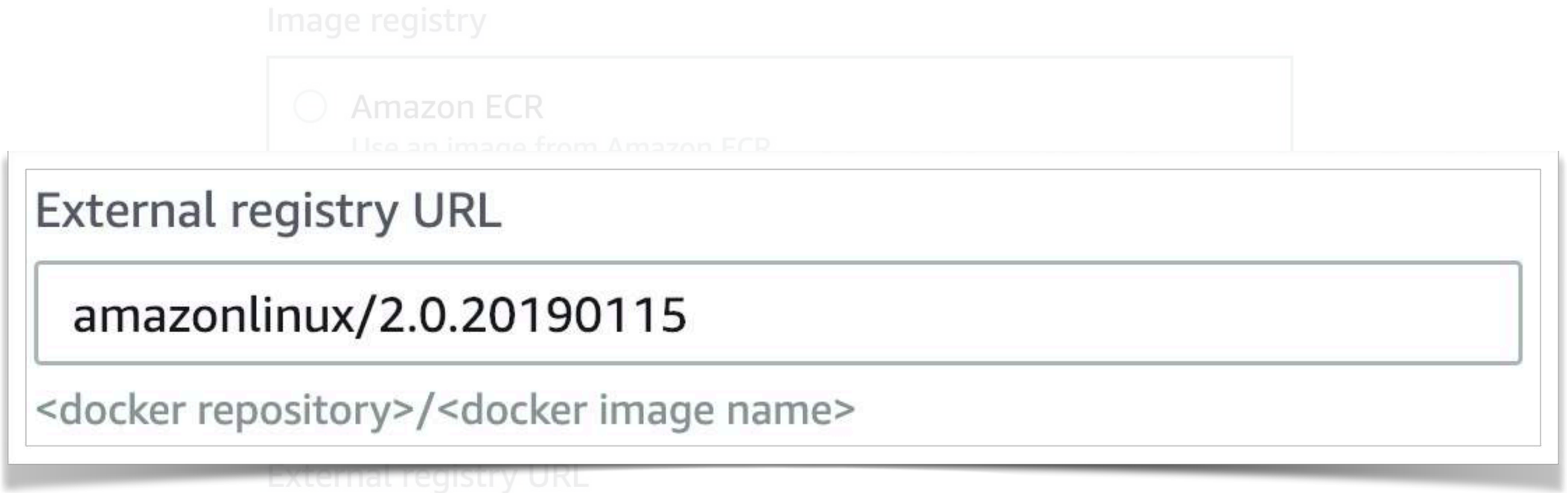
- ○ Amazon ECR
  Use an image from Amazon ECR
- ◉ Other registry
  Use an image hosted in an external Docker registry

External registry URL

amazonlinux/2.0.20190115

<docker repository>/<docker image name>

CodeBuild - Environment



CodeBuild - Environment

Image registry

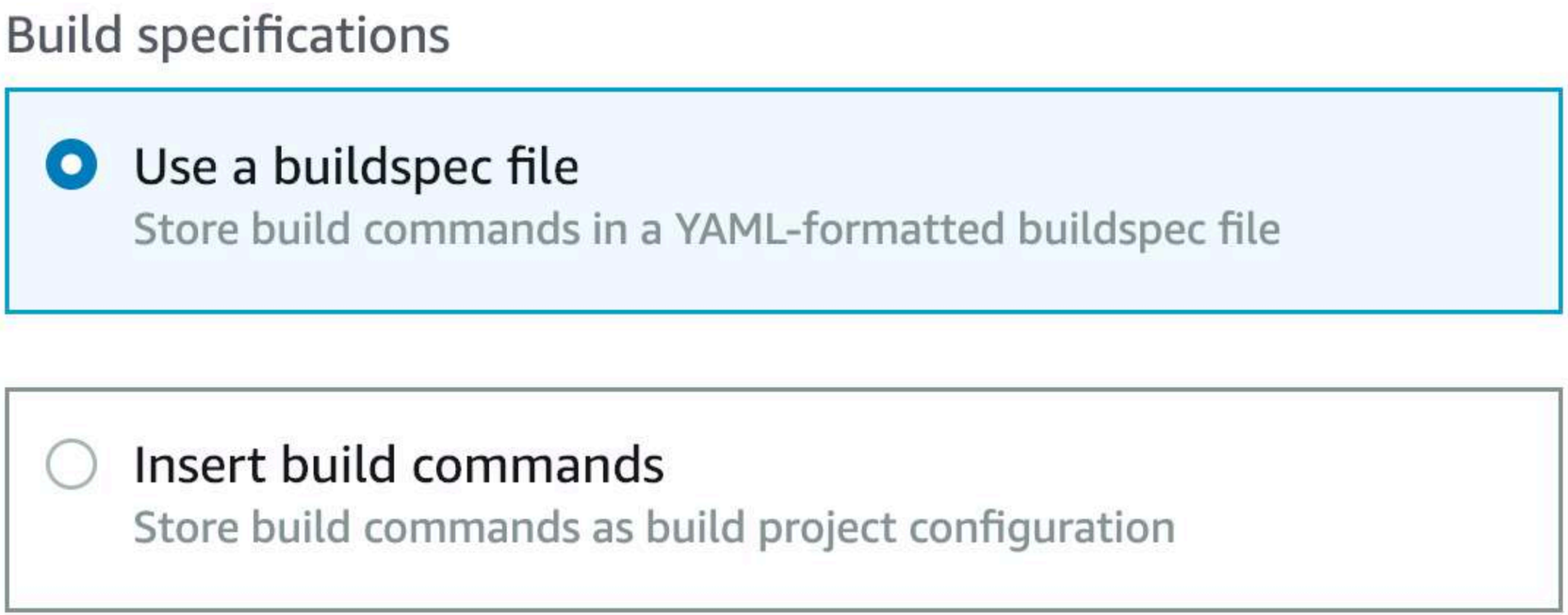
- ○ Amazon ECR
  Use an image from Amazon ECR
- ◉ Other registry
  Use an image hosted in an external Docker registry

External registry URL

amazonlinux/2.0.20190115

<docker repository>/<docker image name>

CodeBuild - `buildspec.yml`



CodeBuild - buildspec.yml

```yaml
version: 0.2

phases:
  install:
      commands:
        - yum update -y
        - amazon-linux-extras install php7.2 -y
        - yum install git-core php-mbstring php-xml zip unzip php-zip -y
        - curl -sS https://getcomposer.org/installer -o /tmp/composer-setup.php
        - php /tmp/composer-setup.php --install-dir=/usr/local/bin --filename=composer
  pre_build:
    commands:
      - composer install --no-interaction
  build:
    commands:
      - composer cs-check
      - composer test
  post_build:
    commands:
      - composer install --no-dev --optimize-autoloader --prefer-dist --no-interaction
artifacts:
  files:
    - '**/*'
```

CodeBuild - buildspec.yml

```
version: 0.2

phases:
  install:
      commands:
        - yum update -y
        - amazon-linux-extras install php7.2 -y
        - yum install git-core php-mbstring php-xml zip unzip php-zip -y
        - curl -sS https://getcomposer.org/installer -o /tmp/composer-setup.php
        - php /tmp/composer-setup.php --install-dir=/usr/local/bin --filename=composer
  pre_build:
    commands:
      - composer install --
  build:
    commands:
      - composer cs-check
      - composer test
  post_build:
    commands:
      - composer install --no-dev --optimize-autoloader --prefer-dist --no-interaction
artifacts:
  files:
    - '**/*'
```

version: 0.2

CodeBuild - buildspec.yml

```yaml
phases:
  install:
      commands:
        - yum update -y
        - amazon-linux-extras install php7.2 -y
        - yum install git-core php-mbstring php-xml zip unzip php-zip -y
        - curl -sS https://getcomposer.org/installer -o /tmp/composer-setup.php
        - php /tmp/composer-setup.php --install-dir=/usr/local/bin --filename=composer
```

CodeBuild - buildspec.yml

```yaml
phases:
  install:
      commands:
        - yum update -y
        - amazon-linux-extras install php7.2 -y
        - yum install git-core php-mbstring php-xml zip unzip php-zip -y
        - curl -sS https://getcomposer.org/installer -o /tmp/composer-setup.php
        - php /tmp/composer-setup.php --install-dir=/usr/local/bin --filename=composer
```

CodeBuild - buildspec.yml

```yaml
phases:
  install:
      commands:
        - yum update -y
        - amazon-linux-extras install php7.2 -y
        - yum install git-core php-mbstring php-xml zip unzip php-zip -y
        - curl -sS https://getcomposer.org/installer -o /tmp/composer-setup.php
        - php /tmp/composer-setup.php --install-dir=/usr/local/bin --filename=composer
```

CodeBuild - buildspec.yml

```yaml
version: 0.2

phases:
  install:
    commands:
      - yum update -y
  pre_build:
    commands:
      - composer install --no-interaction
  post_build:
    commands:
      - composer install --no-dev --optimize-autoloader --prefer-dist --no-interaction
artifacts:
  files:
    - '**/*'
```

CodeBuild - buildspec.yml

```
version: 0.2

phases:
  install:
      commands:
        - yum upd
        - amazon-
        - yum ins                                                    -y
        - curl -s                                                    setup.php
        - php /tm                                                    ilename=composer
  pre_build:
    commands:
      - composer
  build:
    commands:
      - composer
      - composer
  post_build:
    commands:
      - composer install --no-dev --optimize-autoloader --prefer-dist --no-interaction
artifacts:
  files:
    - '**/*'
```

```
phases:

  build:
    commands:
      - composer cs-check
      - composer test
```

CodeBuild - buildspec.yml

```yaml
version: 0.2

phases:
  install:
      commands:
        - yum update -y
        - amazon-linux-extras install php7.2 -y
```

```yaml
phases:

  post_build:
    commands:
      - composer install --no-dev --optimize-autoloader --prefer-dist --no-interaction
```

```yaml
        - composer cs-check
        - composer test
  post_build:
    commands:
      - composer install --no-dev --optimize-autoloader --prefer-dist --no-interaction
artifacts:
  files:
    - '**/*'
```

rackspace
the #1 managed cloud company

CodeBuild - buildspec.yml

```
version: 0.2

phases:
  install:
      commands:
        - yum update -y
        - amazon-linux-extras install php7.2 -y
        - yum install git-core php mbstring php-xml zip unzip php-zip -y
        - curl -sS https://ge                                    mp/composer-setup.php
        - php /tmp/composer-s                                    ocal/bin --filename=composer
  pre_build:
    commands:
      - composer install --no
  build:
    commands:
      - composer cs-check
      - composer test
  post_build:
    commands:
      - composer install --no-dev --optimize-autoloader --prefer-dist --no-interaction
artifacts:
  files:
    - '**/*'
```

```
artifacts:
  files:
    - '**/*'
```

rackspace
the #1 managed cloud company

CodeBuild - `buildspec.yml`

```
version: 0.2

phases:
  install:
      commands:
        - yum update -y
        - amazon-linux-extras install php7.2 -y
        - composer cs-check
        - composer test
  post_build:
    commands:
      - composer install --no-dev --optimize-autoloader --prefer-dist --no-interaction
artifacts:
  files:
    - '**/*'
```

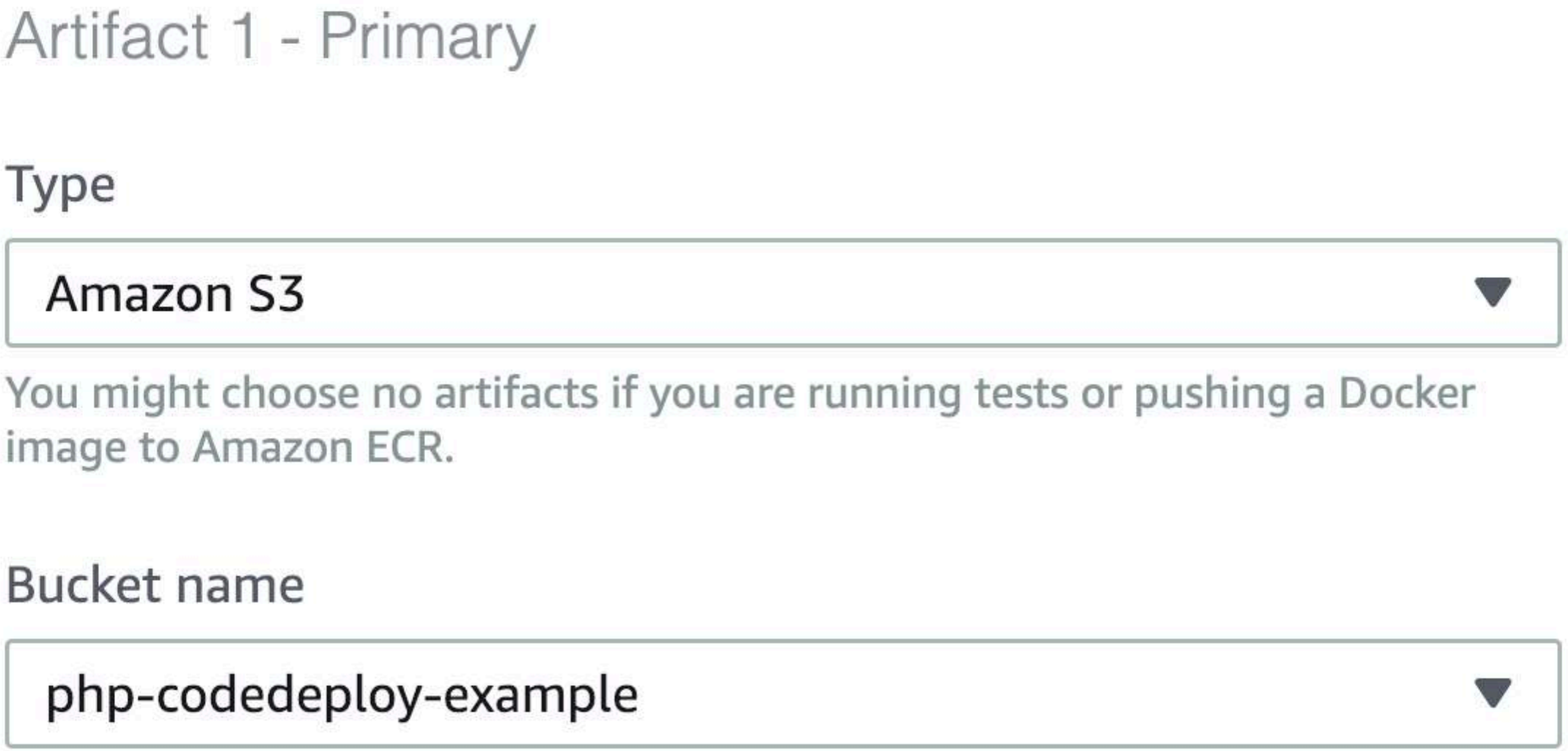
https://docs.aws.amazon.com/codebuild/latest/userguide/build-spec-ref.html

CodeBuild - buildspec.yml

```yaml
version: 0.2

phases:
  install:
      commands:
        - yum update -y
        - amazon-linux-extras install php7.2 -y
        - yum install git-core php-mbstring php-xml zip unzip php-zip -y
        - curl -sS https://getcomposer.org/installer -o /tmp/composer-setup.php
        - php /tmp/composer-setup.php --install-dir=/usr/local/bin --filename=composer
  pre_build:
    commands:
      - composer install --no-interaction
  build:
    commands:
      - composer cs-check
      - composer test
  post_build:
    commands:
      - composer install --no-dev --optimize-autoloader --prefer-dist --no-interaction
artifacts:
  files:
    - '**/*'
```

CodeBuild - Artifacts



CodeBuild - Artifacts

Name - *optional*

The name of the folder or compressed file in the bucket that will contain your output artifacts. Use Artifacts packaging under Additional configuration to choose whether to use a folder or compressed file.

DemoApplication.zip

☐ Enable semantic versioning

Use the artifact name specified in the buildspec file

Path - *optional*

The path to the build output ZIP file or folder.

build-artifacts

Example: MyPath/MyArtifact.zip.

CodeBuild - Artifacts

Name - *optional*

The name of the folder or compressed file in the bucket that will contain your output artifacts. Use Artifacts packaging under Additional configuration to choose whether to use a folder or compressed file.

DemoApplication.zip

Enable semantic versioning

Use the artifact name specified in the buildspec file

Path - *optional*

The path to the

build-artifact

Example: MyPath

```
artifacts:
  files:
    - '**/*'
  name: DemoApplication-$(date +%Y-%m-%d).zip
```

CodeBuild - Artifacts

Name - *optional*

The name of the folder or compressed file in the bucket that will contain your output artifacts. Use Artifacts packaging under Additional configuration to choose whether to use a folder or compressed file.

DemoApplication.zip

☐ Enable semantic versioning

Use the artifact name specified in the buildspec file

Path - *optional*

The path to the build output ZIP file or folder.

build-artifacts

Example: MyPath/MyArtifact.zip.

CodeBuild - Artifacts

Namespace type - *optional*

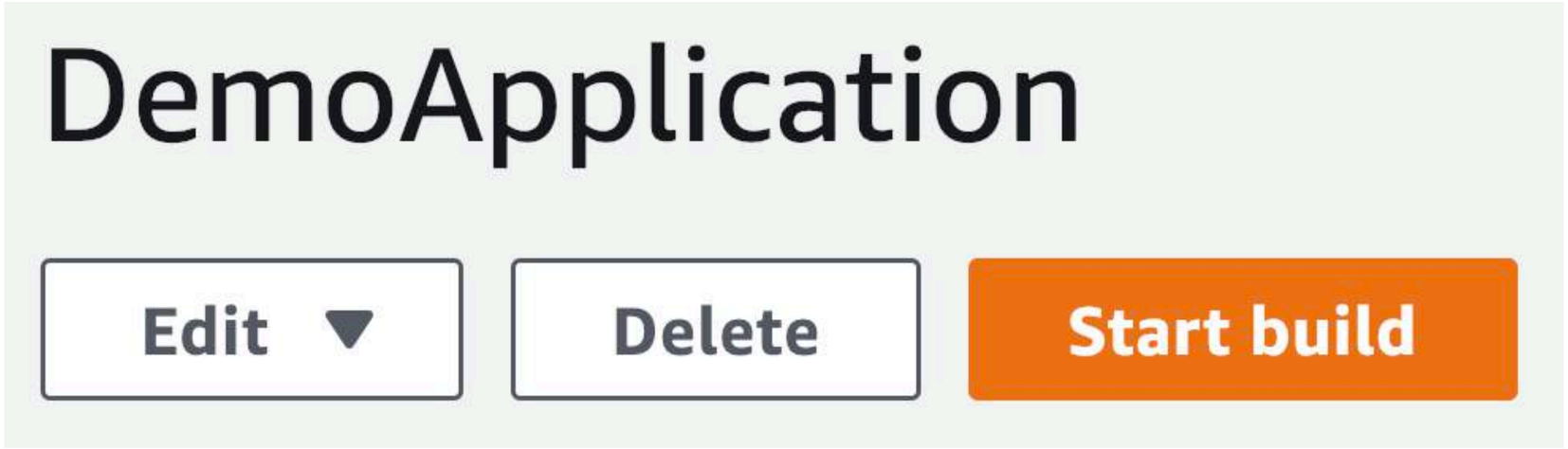
Build ID

Choose Build ID to insert the build ID into the path to the build output ZIP file or folder, e.g. MyPath/MyBuildID/MyArtifact.zip. Otherwise, choose None.

Artifacts packaging

- None
  The artifact files will be uploaded to the bucket.
- Zip
  AWS CodeBuild will upload artifacts into a compressed file that is put into the specified bucket.

CodeBuild - Start build



CodeBuild - Start build

Source version - *optional*

<commit ID, pull request ID, branch name, or tag name>

CodeBuild - Success!

Name	Status	Context	Duration	Start time	End time
SUBMITTED	Succeeded	-	<1 sec	Feb 5, 2019 9:06 PM	Feb 5, 2019 9:06 PM
QUEUED	Succeeded	-	1 sec	Feb 5, 2019 9:06 PM	Feb 5, 2019 9:06 PM
PROVISIONING	Succeeded	-	18 secs	Feb 5, 2019 9:06 PM	Feb 5, 2019 9:07 PM
DOWNLOAD_SOURCE	Succeeded	-	<1 sec	Feb 5, 2019 9:07 PM	Feb 5, 2019 9:07 PM
INSTALL	Succeeded	-	36 secs	Feb 5, 2019 9:07 PM	Feb 5, 2019 9:07 PM
PRE_BUILD	Succeeded	-	6 secs	Feb 5, 2019 9:07 PM	Feb 5, 2019 9:07 PM
BUILD	Succeeded	-	1 sec	Feb 5, 2019 9:07 PM	Feb 5, 2019 9:07 PM
POST_BUILD	Succeeded	-	<1 sec	Feb 5, 2019 9:07 PM	Feb 5, 2019 9:07 PM
UPLOAD_ARTIFACTS	Succeeded	-	<1 sec	Feb 5, 2019 9:07 PM	Feb 5, 2019 9:07 PM
FINALIZING	Succeeded	-	2 secs	Feb 5, 2019 9:07 PM	Feb 5, 2019 9:07 PM
COMPLETED	Succeeded	-	-	Feb 5, 2019 9:07 PM	-

CodeBuild - Success!

Build status

Current Phase	Status	Start time
COMPLETED	Succeeded	Feb 5, 2019 9:06 PM

End time	Output artifacts
Feb 5, 2019 9:07 PM	View artifacts

CodeBuild - Artifacts

s3://php-codedeploy-example

/build-artifacts

/41dd4014-e610-43ed-a5d6-551c47570c8d

/DemoApplication.zip

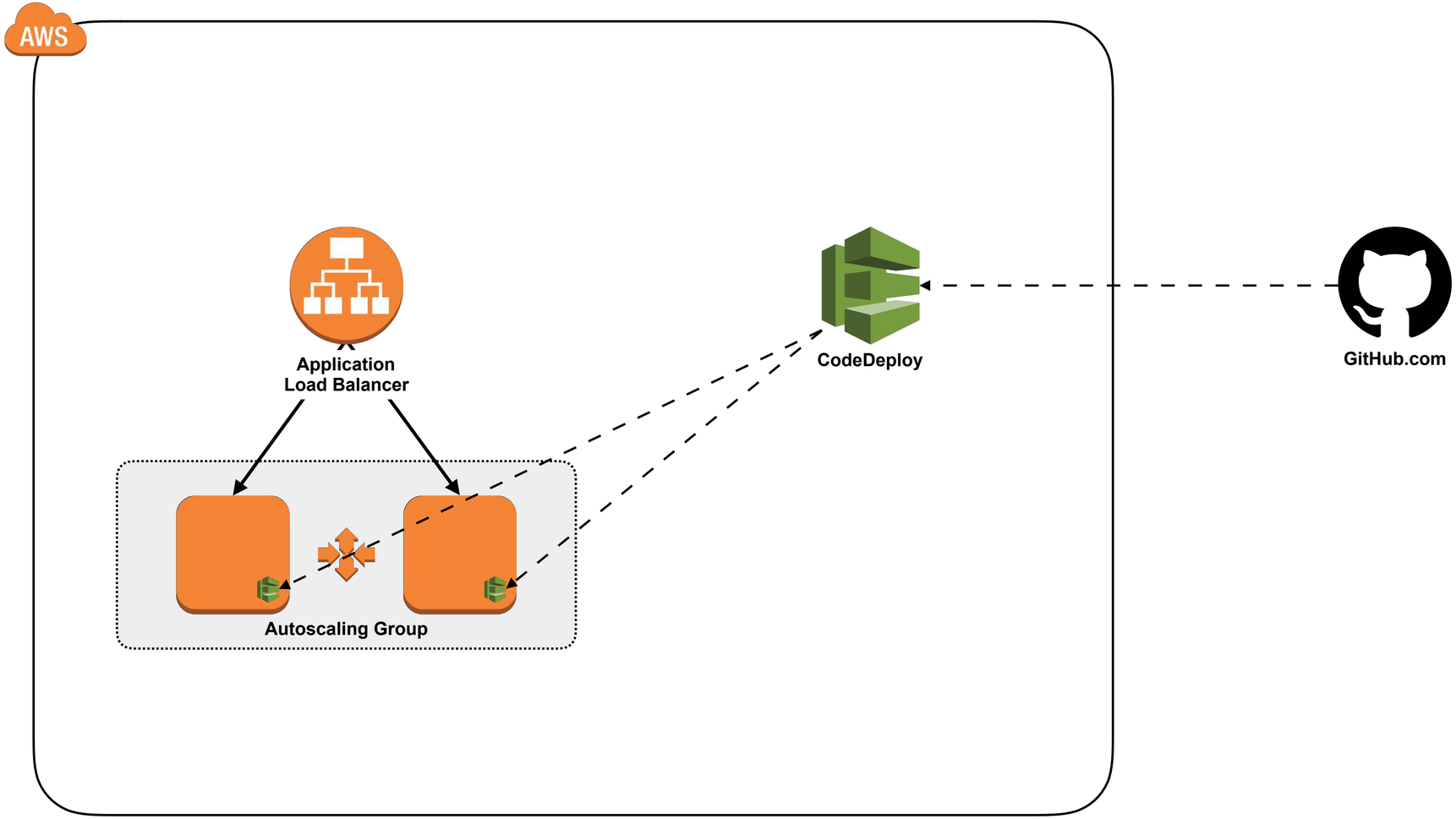
Other capabilities

CodeBuild - Other capabilities

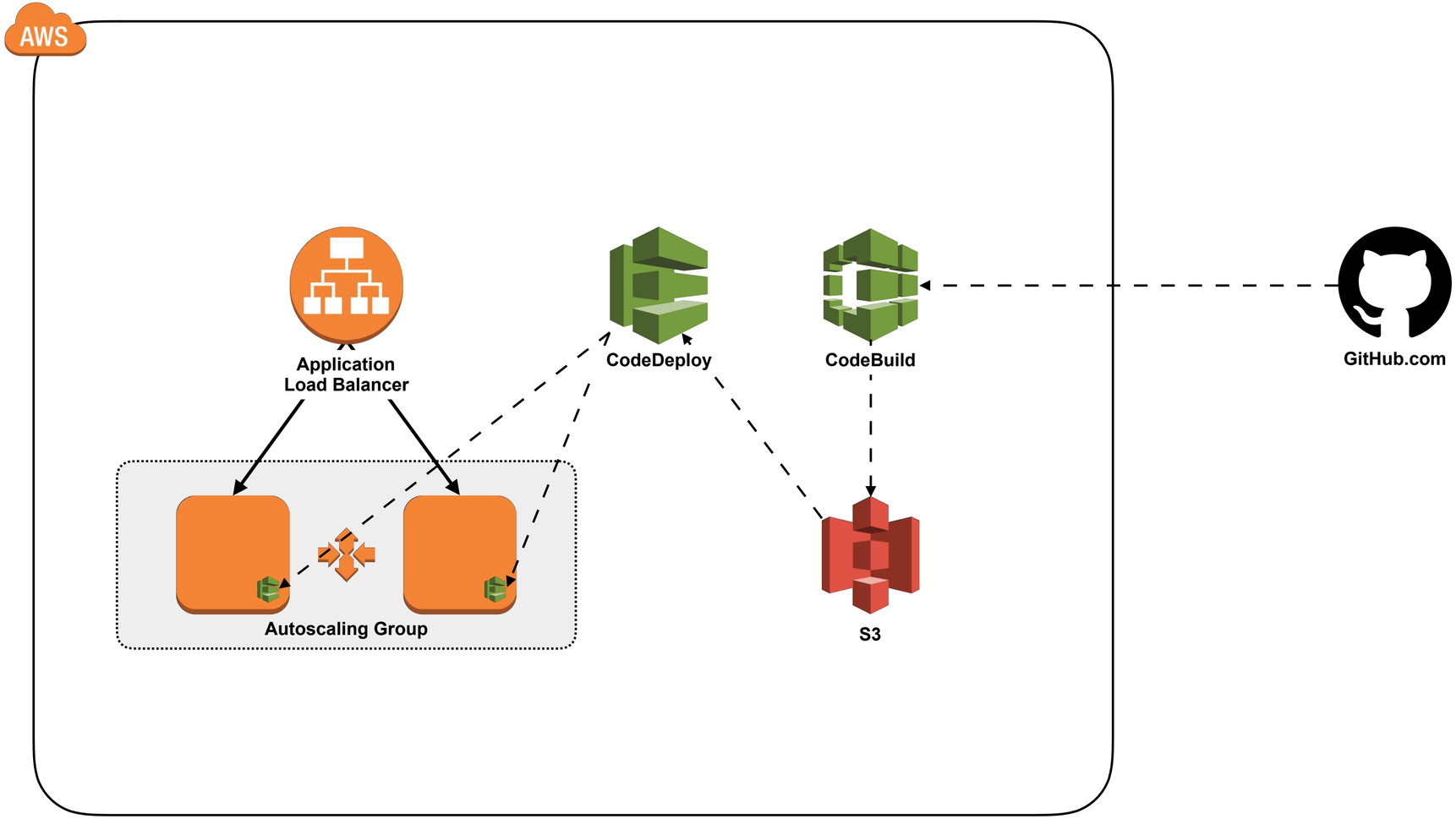
- Dependency caching
- VPC support
- Environment variables & secure parameters
- Build Docker images
- Larger build instance sizes

CodeBuild → CodeDeploy

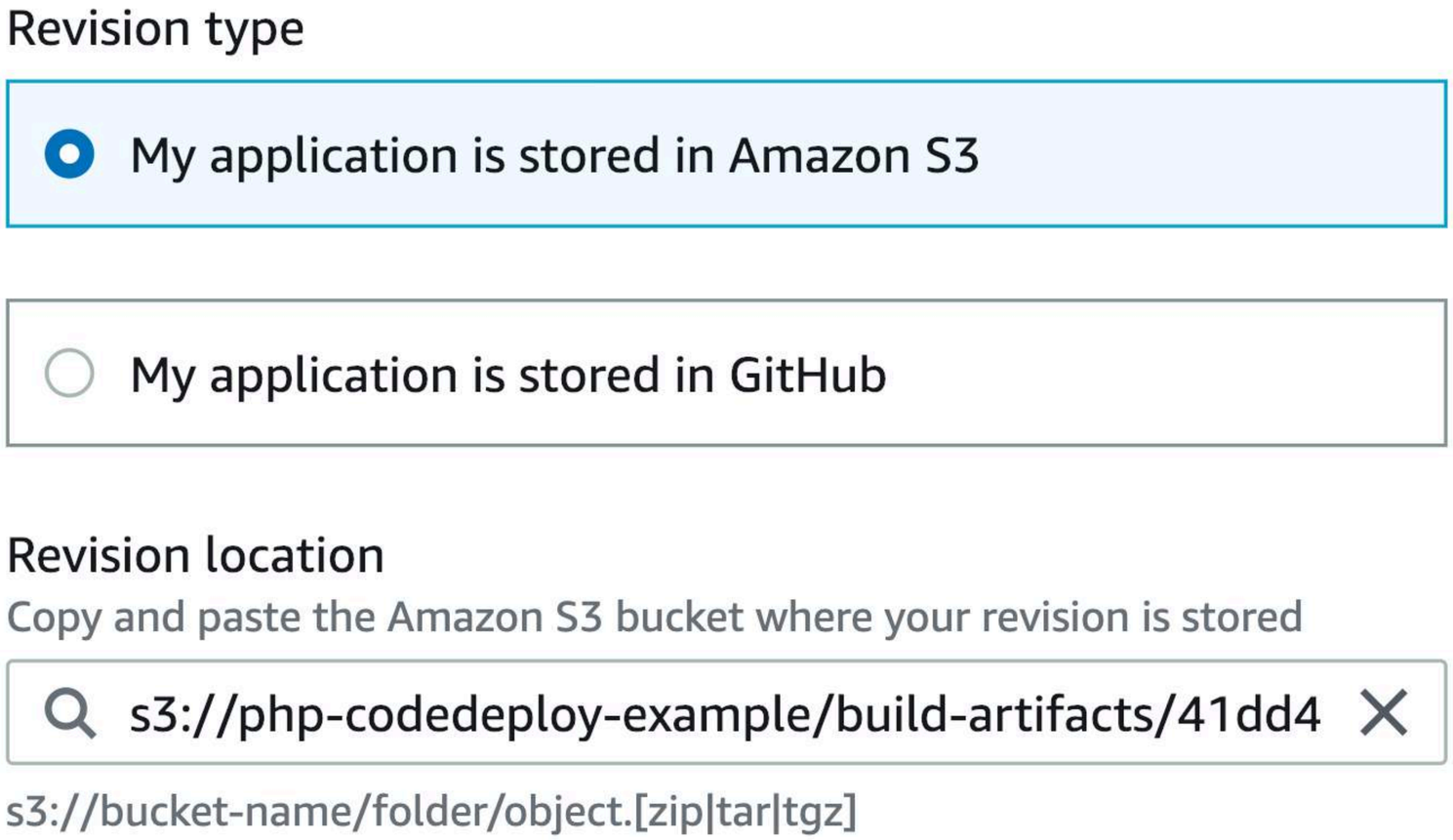
CodeBuild → CodeDeploy



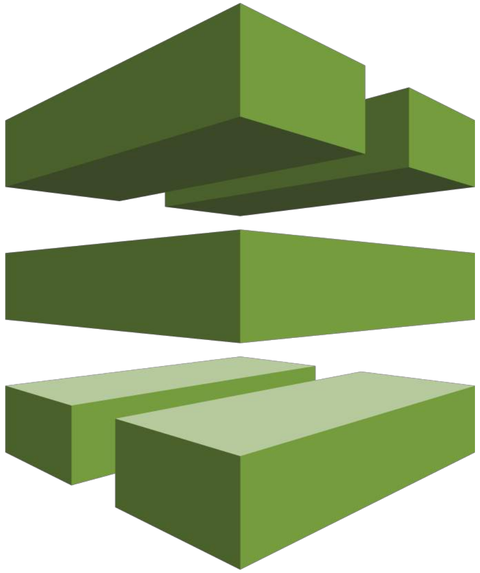
CodeBuild → CodeDeploy



CodeBuild → CodeDeploy

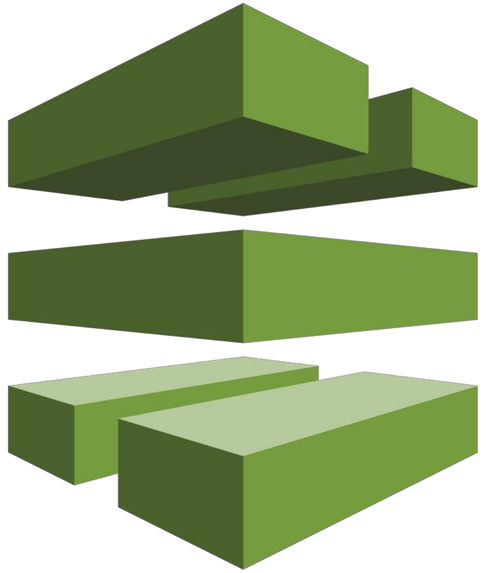


CodePipeline



CodePipeline

- Continuous integration & continuous delivery service
  - Model, visualize, and automate steps
- AWS & third-party integration support
- Fully configurable workflow
- Cheap!

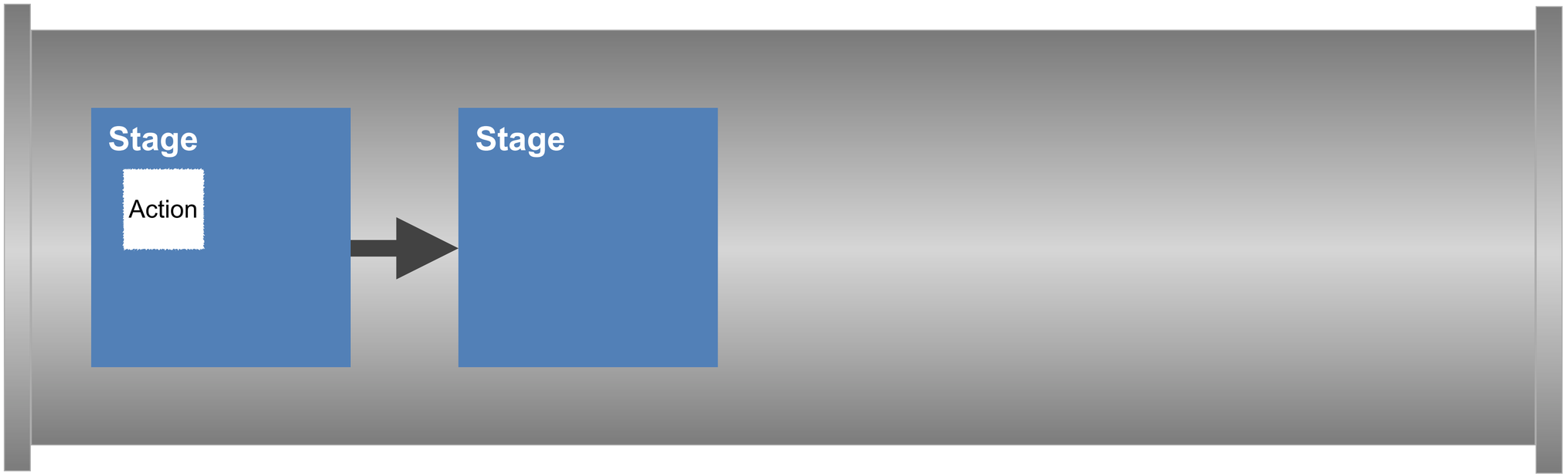


CodePipeline

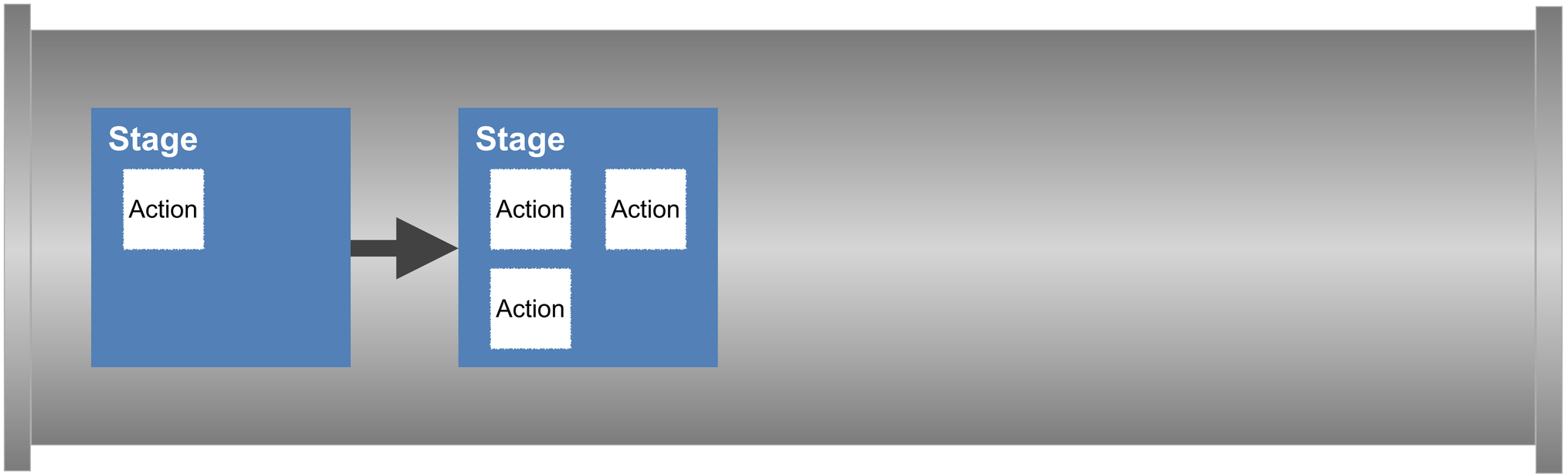
CodePipeline

CodePipeline

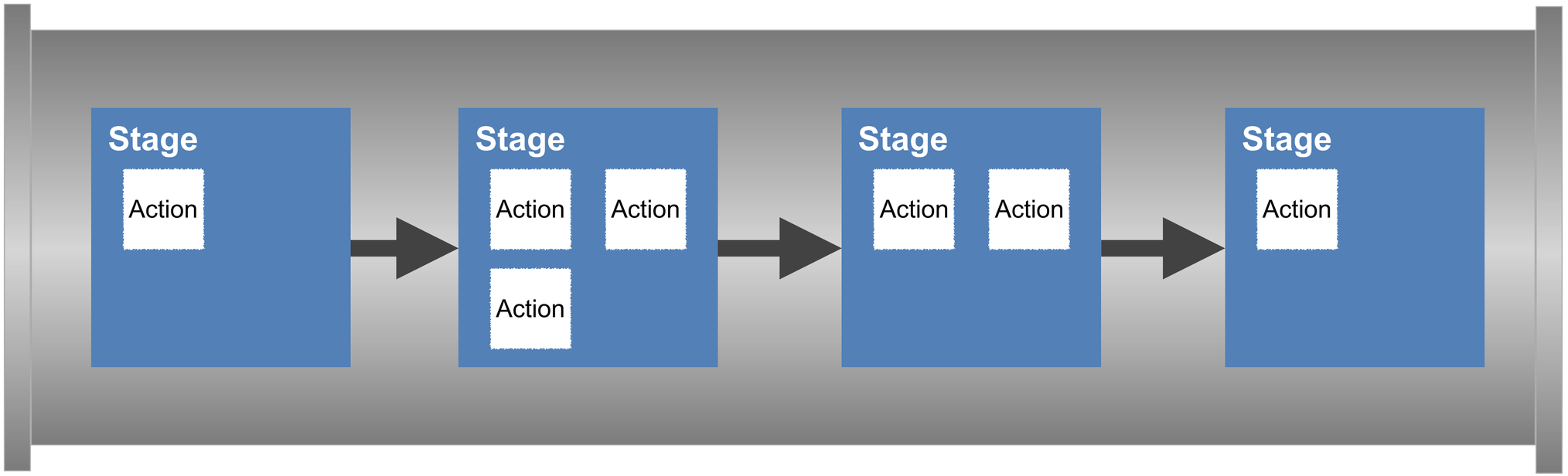
CodePipeline



CodePipeline



CodePipeline



CodePipeline - Create pipeline

Pipeline name

Enter the pipeline name. You cannot edit the pipeline name after it is created.

DemoApplication

No more than 100 characters

CodePipeline - Create pipeline

Service role

New service role
Create a service role in your account

Existing service role
Choose an existing service role from your account

CodePipeline - Create pipeline

Role name

AWSCodePipelineServiceRole-us-east-1-DemoApplication

Type your service role name

- [x] Allow AWS CodePipeline to create a service role so it can be used with this new pipeline

CodePipeline - Create pipeline

Artifact store

Default location
Create a default S3 bucket in your account.

Custom location
Choose an existing S3 location from your account in the same region and account as your pipeline

CodePipeline - Source location

CodePipeline - Source location

Repository

Branch

CodePipeline - Source location

Change detection options

Choose a detection mode to automatically start your pipeline when a change occurs in the source code.

- (•) **GitHub webhooks (recommended)**
  Use webhooks in GitHub to automatically start my pipeline when a change occurs

- ( ) **AWS CodePipeline**
  Use AWS CodePipeline to check periodically for changes

CodePipeline - Build provider

Build provider

This is the tool of your build project. Provide build artifact details like operating system, build spec file, and output file names.

AWS CodeBuild

Project name

Choose a build project that you have already created in the AWS CodeBuild console. Or create a build project in the AWS CodeBuild console and then return to this task.

DemoApplication

Create project

CodePipeline - Deployment provider

Deploy provider

Choose how you deploy to instances. Choose the provider, and then provide the configuration details for that provider.

AWS CodeDeploy

Application name

Choose an application that you have already created in the AWS CodeDeploy console. Or create an application in the AWS CodeDeploy console and then return to this task.

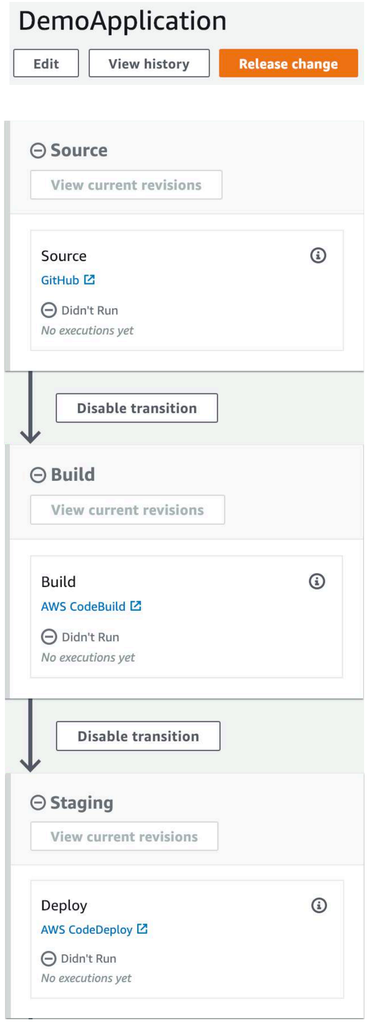
DemoApplication

Deployment group

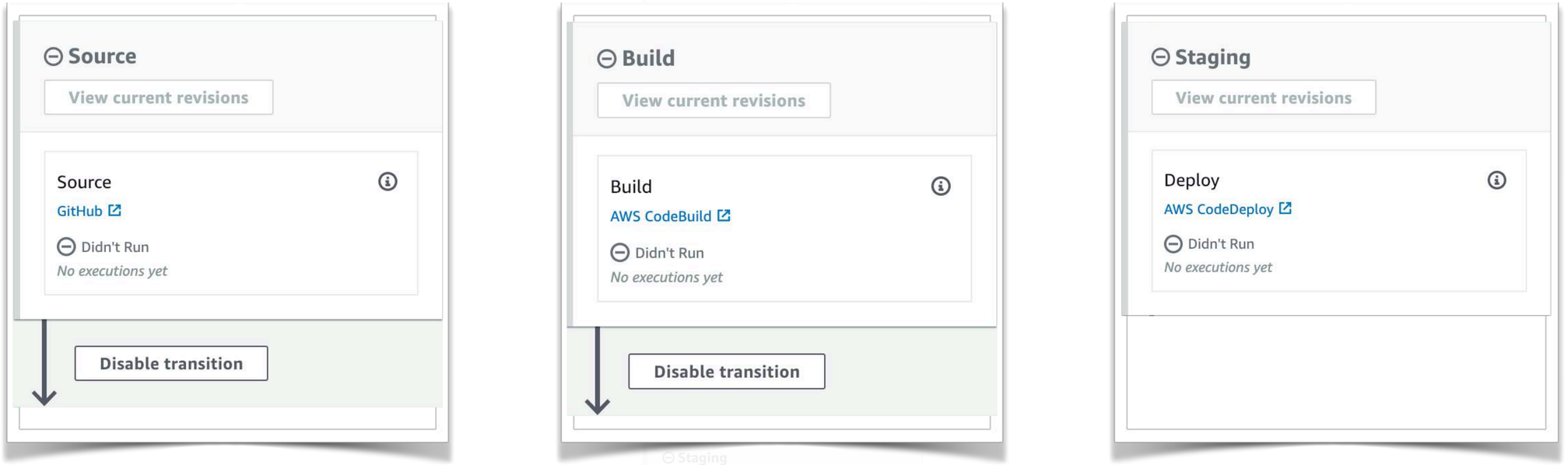
Choose a deployment group that you have already created in the AWS CodeDeploy console. Or create a deployment group in the AWS CodeDeploy console and then return to this task.

DemoApplicationTest

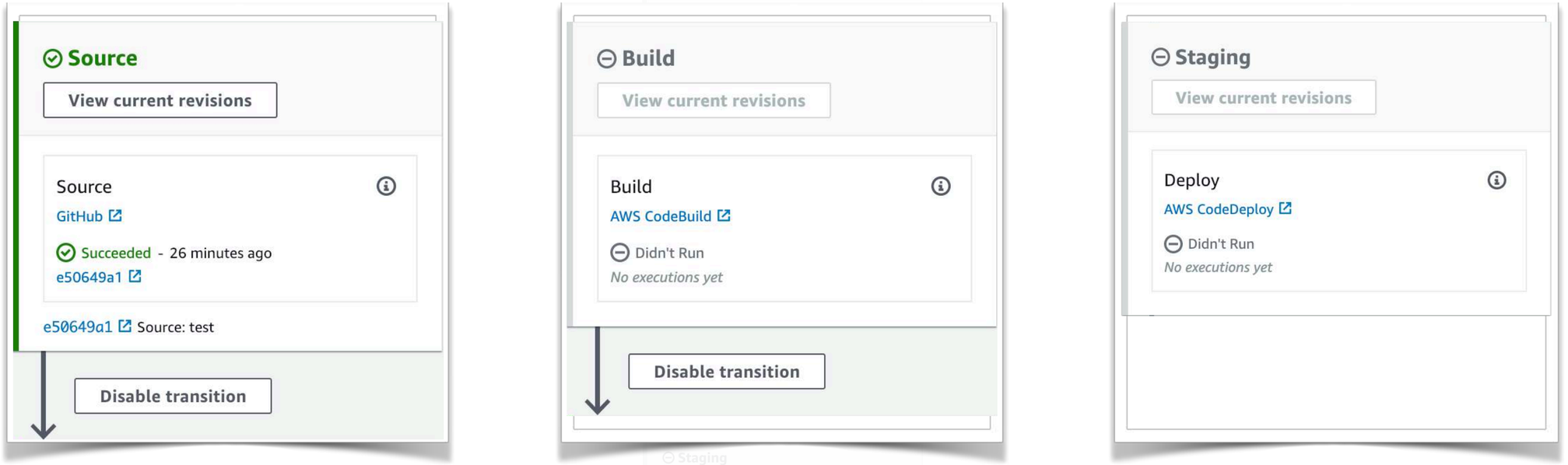
CodePipeline



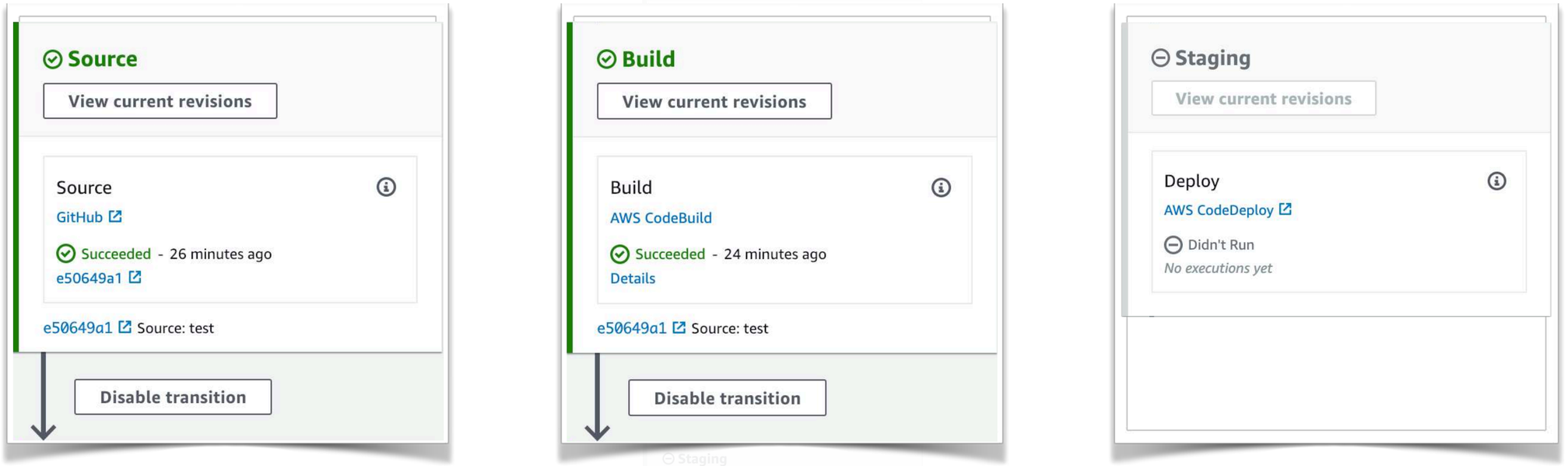
CodePipeline



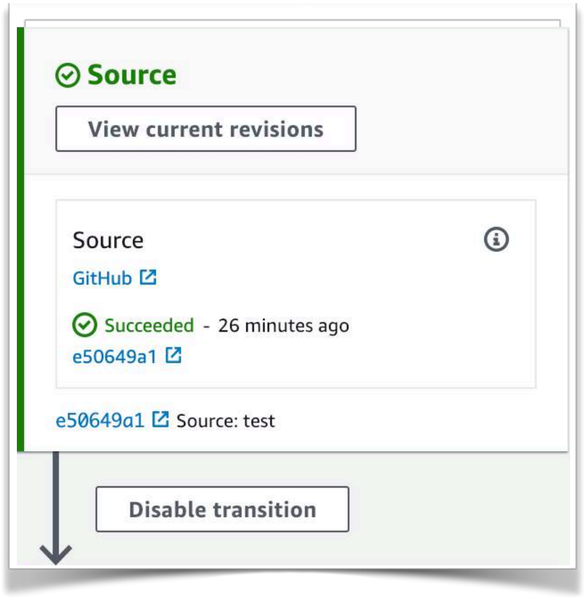
CodePipeline

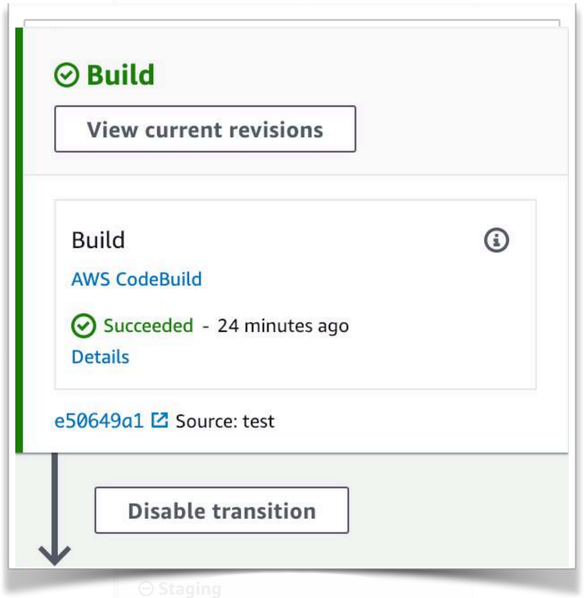


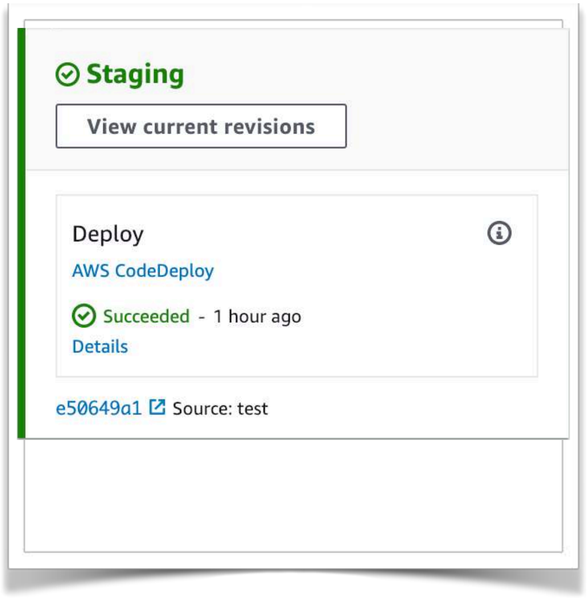
CodePipeline



CodePipeline







Other actions

CodePipeline - Other actions - Test

- Jenkins
- AWS CodeBuild
- BlazeMeter
- Ghost Inspector UI Testing
- HPE StormRunner Load
- Runscope API Monitoring

CodePipeline - Other actions - Approval

SNS topic ARN - *optional*

arn:aws:sns:us-east-1:012345678910:demo-application-build-…

URL for review - *optional*

Type the URL you want to provide to the reviewer as part of the approval request. The URL must begin with 'http://' or 'https://'.

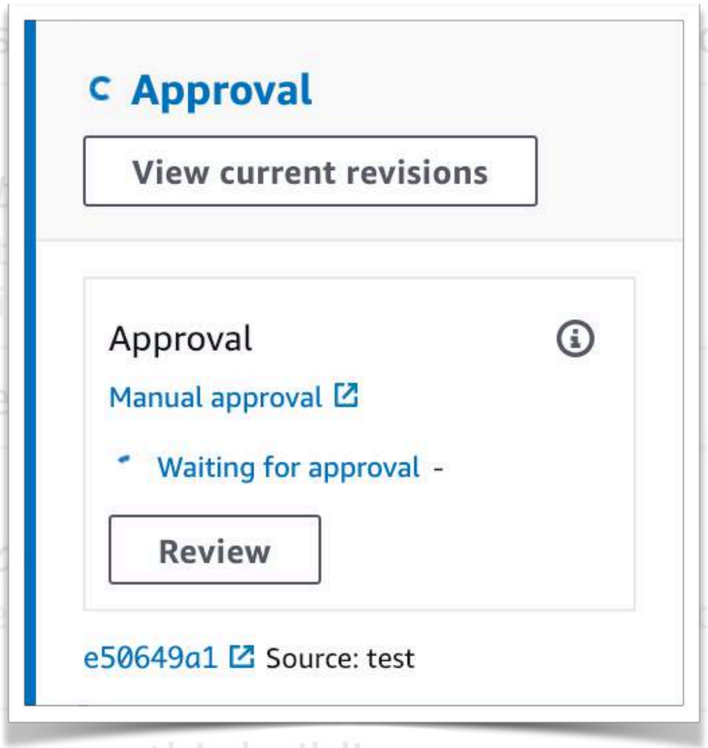
https://staging.demo-application.com

Comments - *optional*

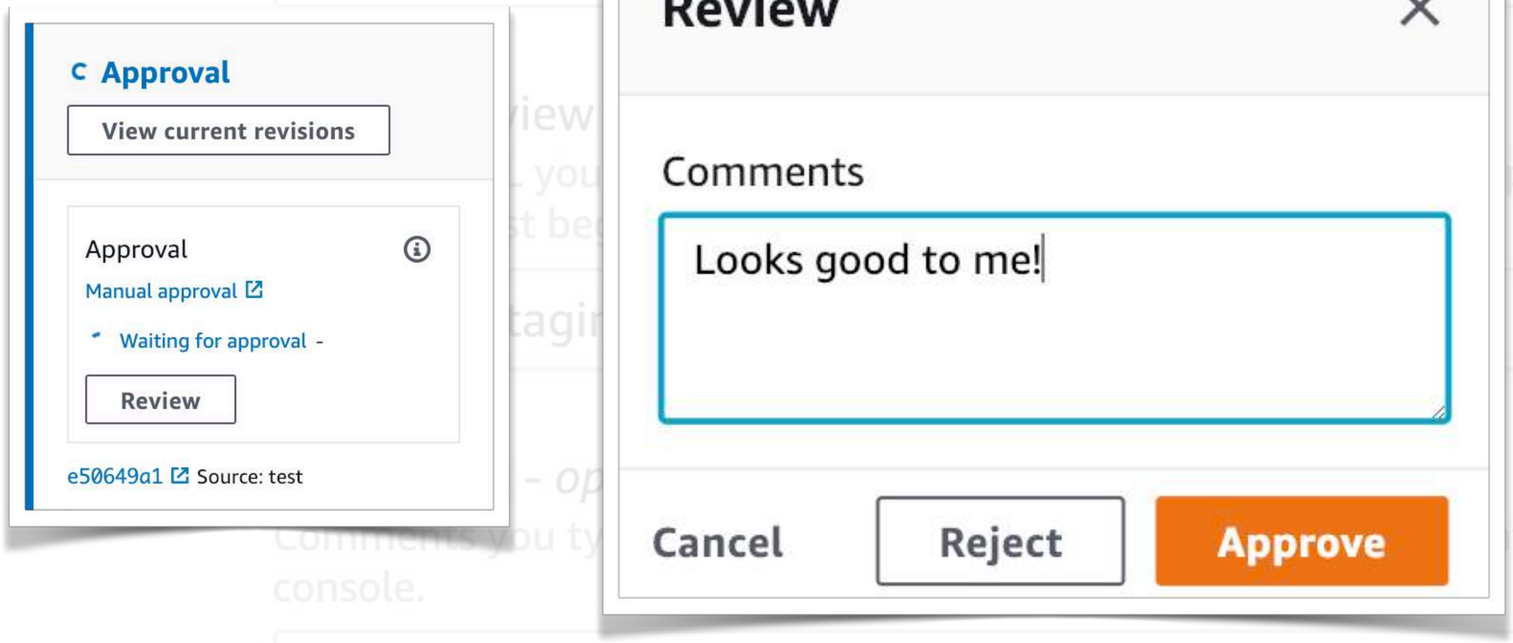
Comments you type here display for the reviewer in email notifications or the console.

Please test and approve this build!

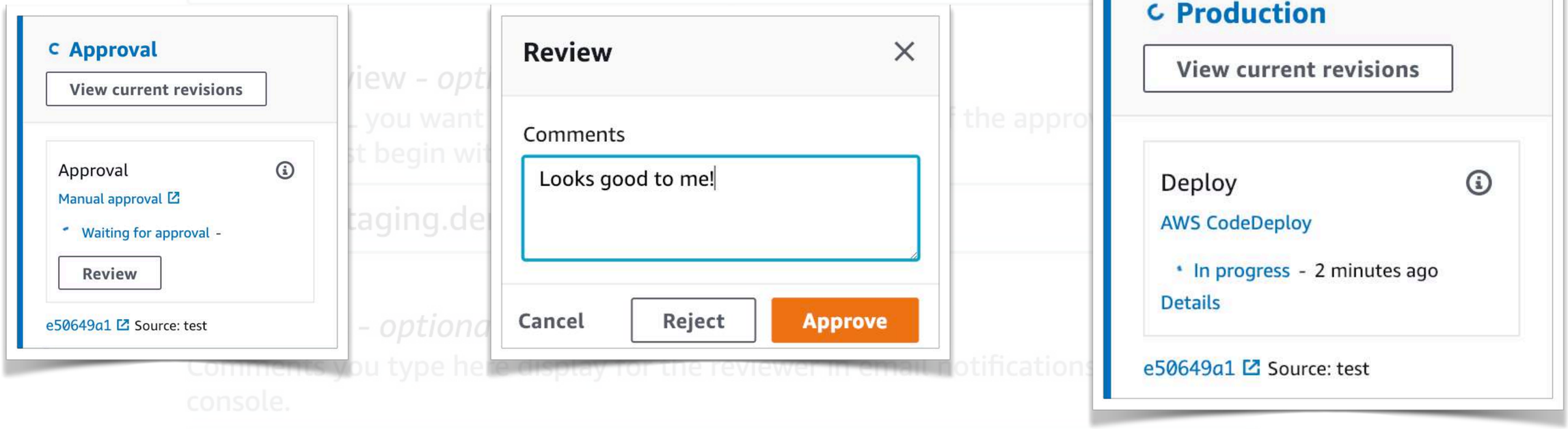
CodePipeline - Other actions - Approval



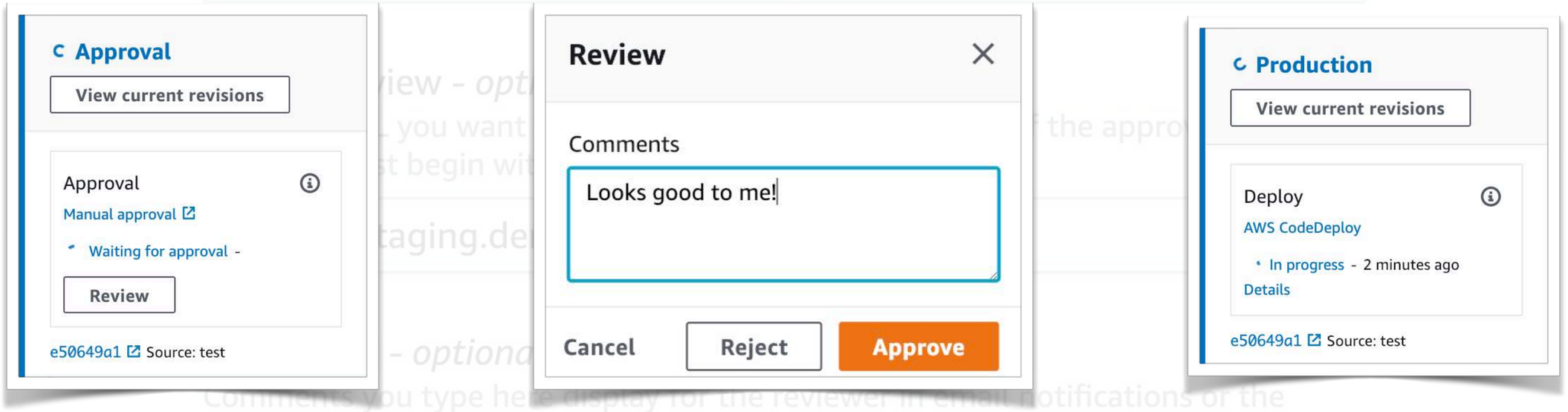
CodePipeline - Other actions - Approval



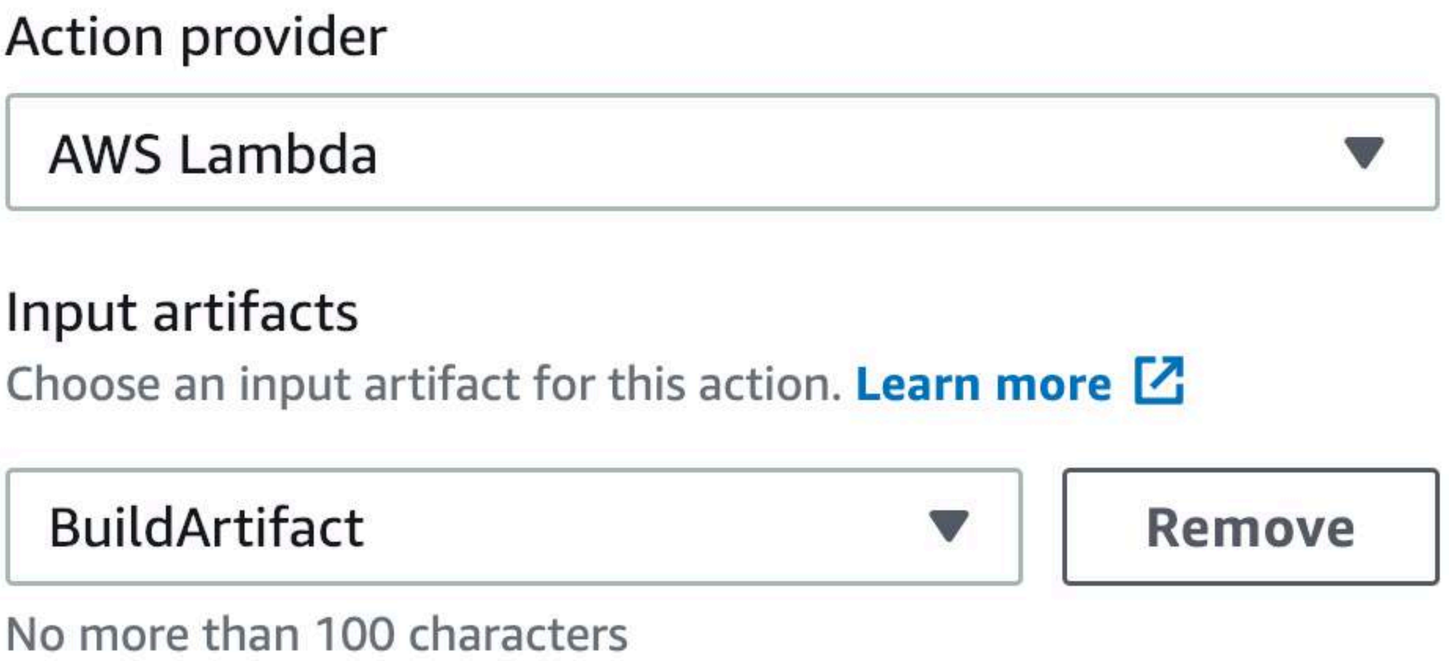
CodePipeline - Other actions - Approval



CodePipeline - Other actions - Approval



CodePipeline - Other actions - Invoke



CodePipeline - Other actions - Invoke

Function name

Choose a function that you have already created in the AWS Lambda console. Or create a function in the Amazon ECS console and then return to this task.

DoSomething

User parameters - *optional*

This string will be used in the event data parameter passed to the handler in AWS Lambda.

{"foo":"bar"}

Output artifacts

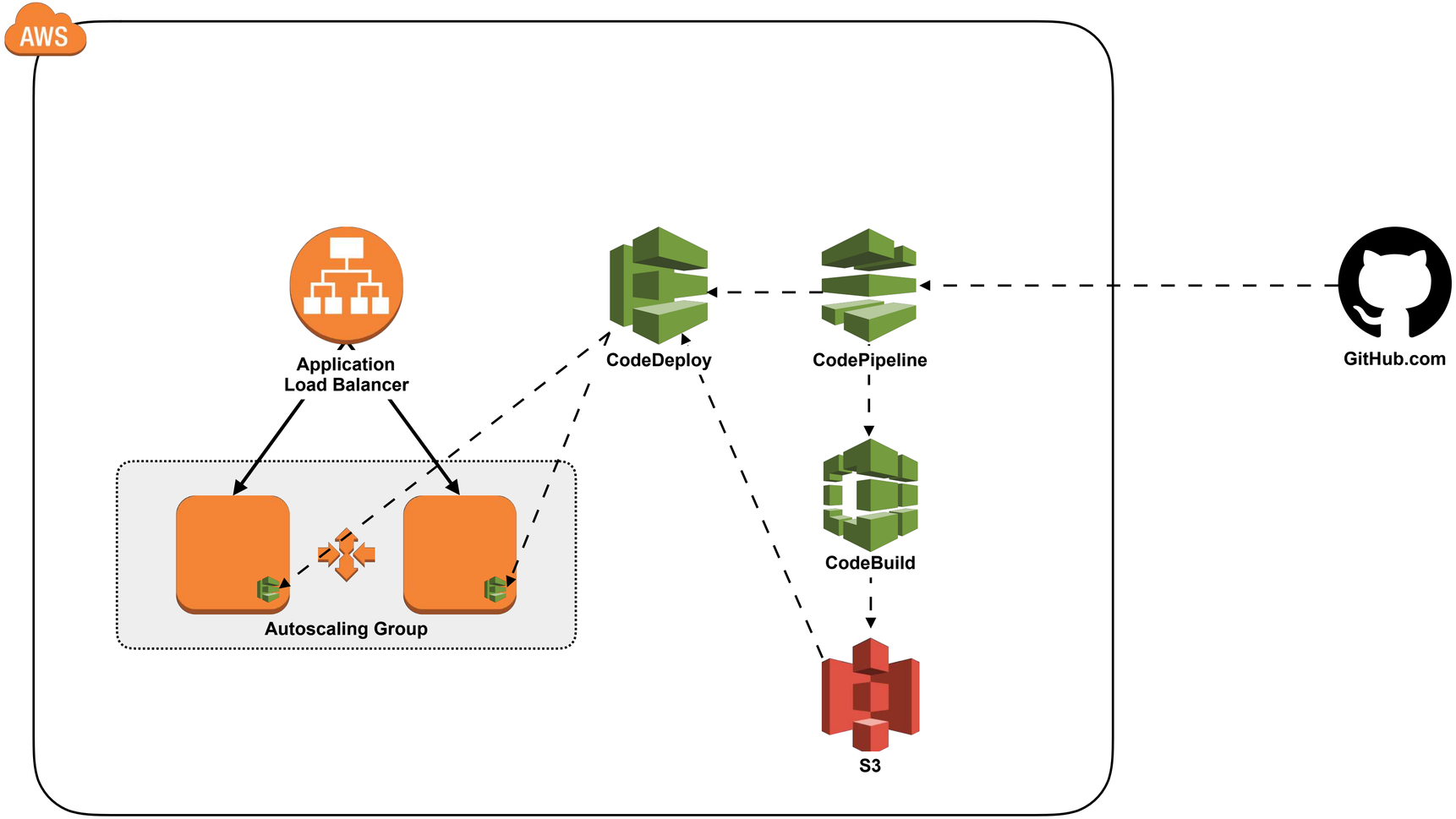
Choose a name for the output of this action.

LambdaOutput.zip

Remove

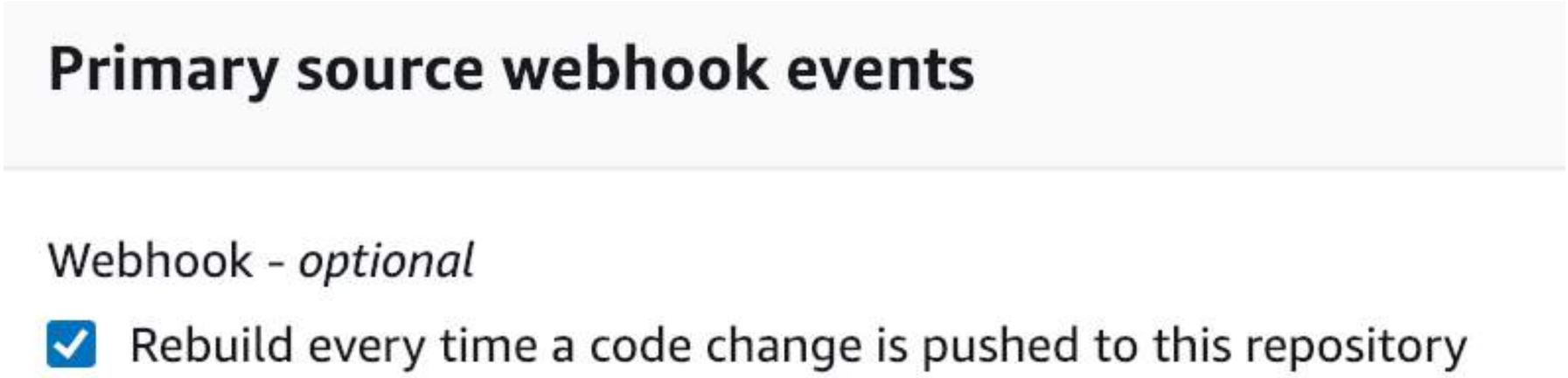
Putting it all together

CodePipeline → CodeBuild → CodeDeploy

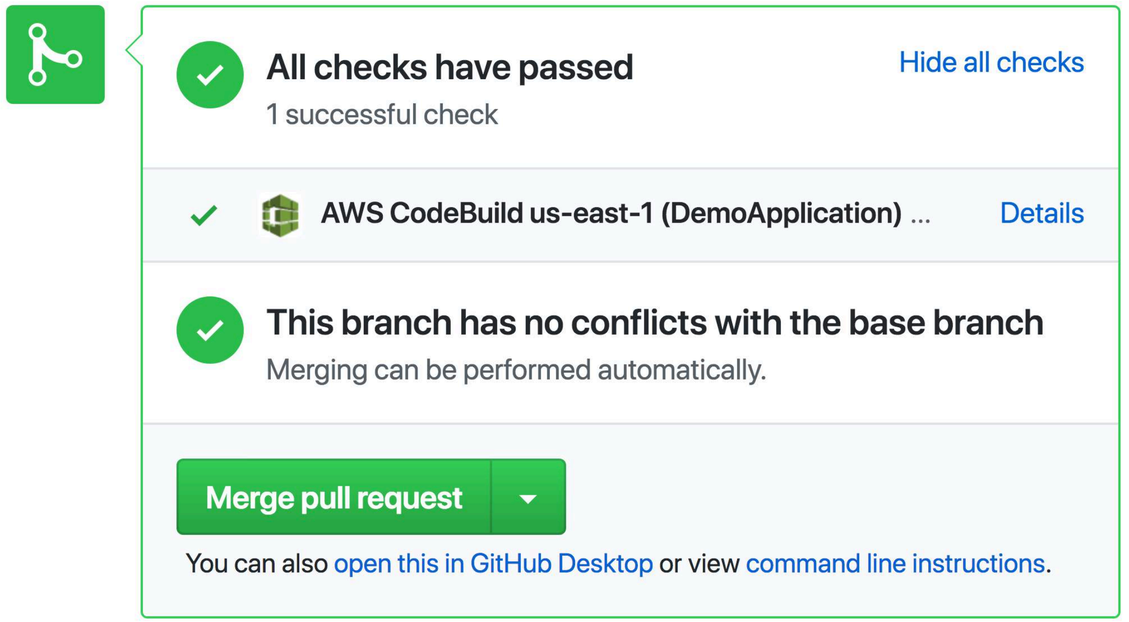


Pull Requests

CodeBuild - Pull Requests



CodeBuild - Pull Requests



https://docs.aws.amazon.com/codebuild/latest/userguide/sample-github-pull-request.html

Closing thoughts

Thanks!

https://joind.in/talk/05470