

CodePipeline: Overview

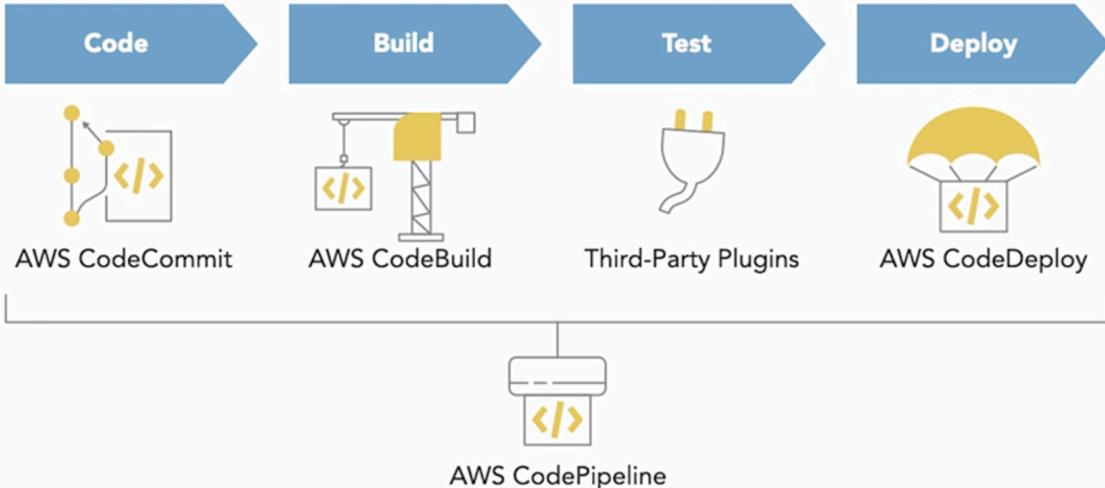
- Continuous delivery
- Visual workflow
- Source: GitHub, CodeCommit, or Amazon S3
- Build: CodeBuild, Jenkins, and so on
- Load testing: third-party tools
- Deploy: CodeDeploy, Beanstalk, CloudFormation, ECS, and so on

CodePipeline: Stages

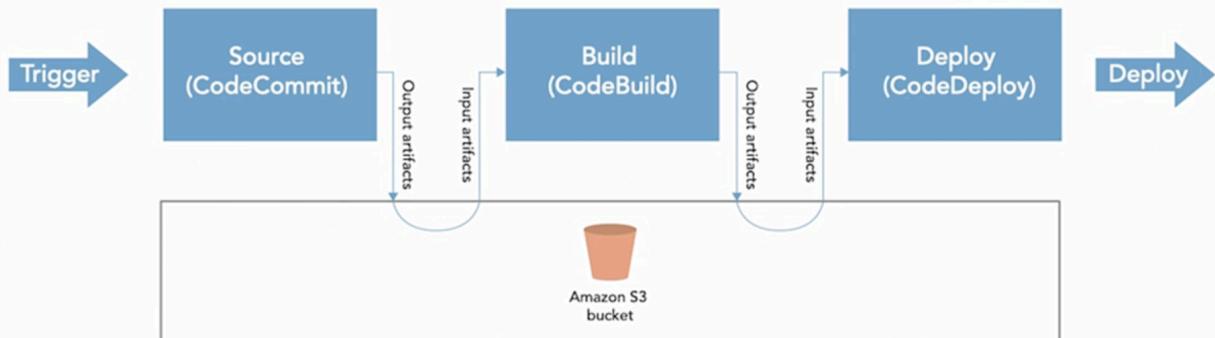
- Each stage can have sequential or parallel actions
- Stages can be build, test, deploy, load test, and so on
- Manual approvals can be defined at any stage

AWS CI/CD Stack

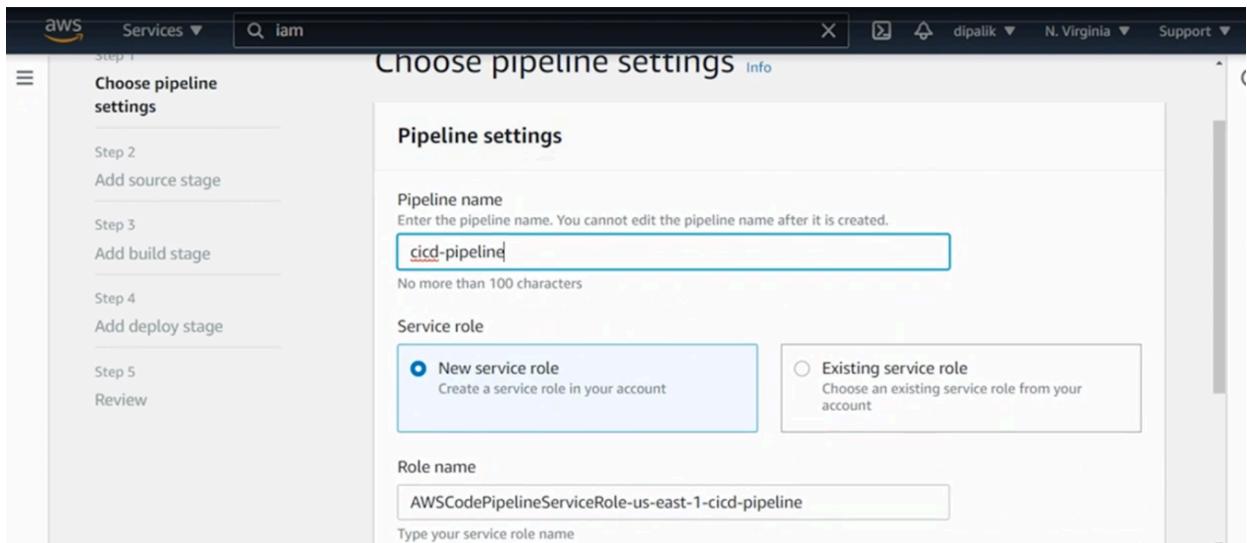
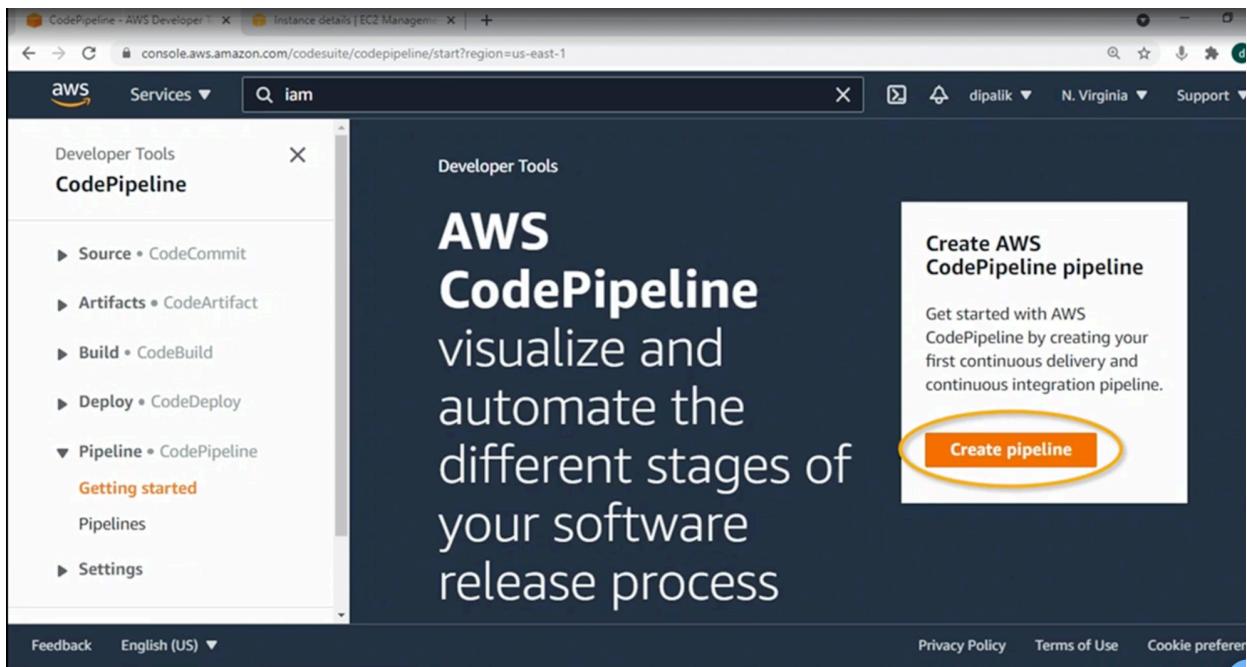
CI/CD with AWS



CodePipeline: Artifacts



- Each stage can create artifacts
- Artifacts are stored in S3 and passed on to next stage



CodePipeline - AWS Developer Tools Instance details | EC2 Management +

console.aws.amazon.com/codesuite/codepipeline/pipeline/new?region=us-east-1

aws Services ▾ iam X dipalik N. Virginia Support

Advanced settings

Artifact store

Default location
Use the default artifact store (Amazon S3 codepipeline-us-east-1-868401448611) designated in the same region and account as your pipeline

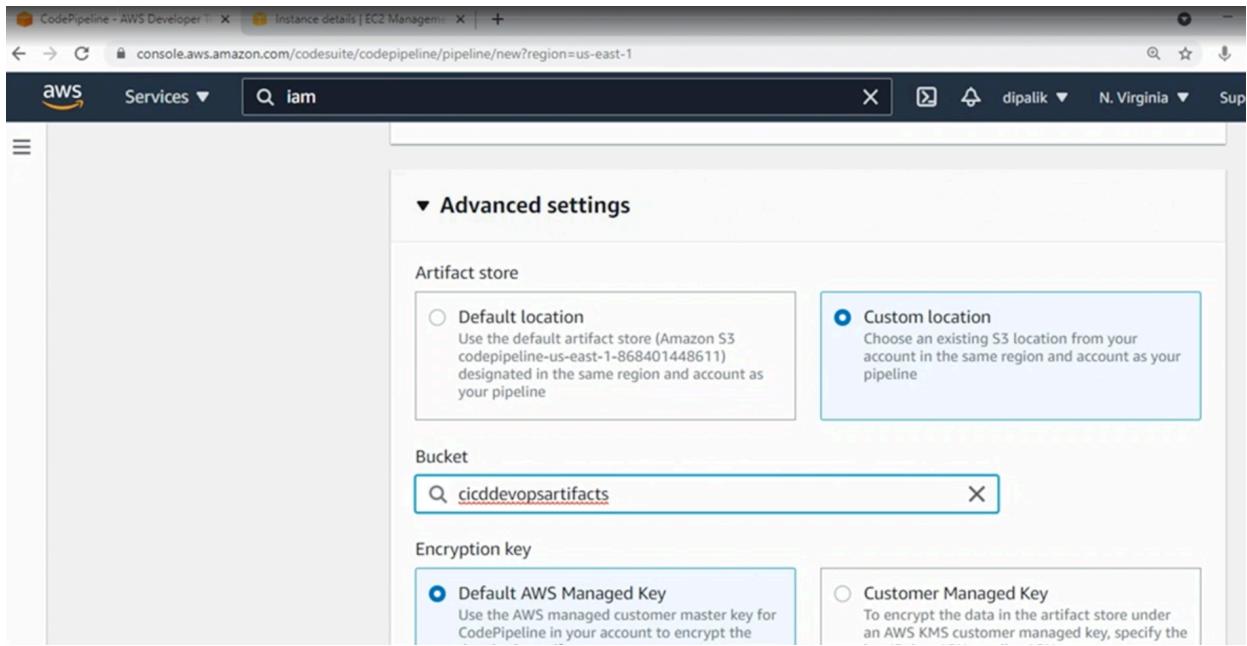
Custom location
Choose an existing S3 location from your account in the same region and account as your pipeline

Bucket

Encryption key

Default AWS Managed Key
Use the AWS managed customer master key for CodePipeline in your account to encrypt the

Customer Managed Key
To encrypt the data in the artifact store under an AWS KMS customer managed key, specify the



aws Services ▾ iam X dipalik N. Virginia Support

Developer Tools > CodePipeline > Pipelines > Create new pipeline

Step 1 Choose pipeline settings

Step 2 Add source stage

Step 3 Add build stage

Step 4 Add deploy stage

Step 5 Review

A

AWS CodeCommit

Amazon ECR AWS CodeCommit

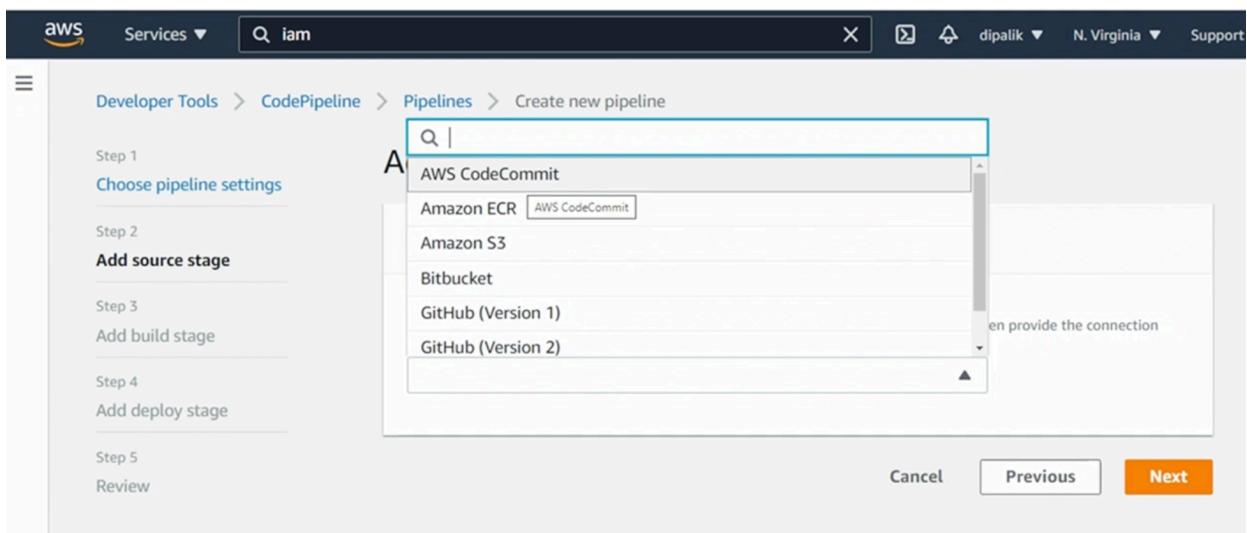
Amazon S3

Bitbucket

GitHub (Version 1)

GitHub (Version 2)

Cancel Previous Next



Screenshot of the AWS CodePipeline Step 5: Review screen.

Step 5: Review

Choose a repository that you have already created where you have pushed your source code.
cicd-repo

Branch name
Choose a branch of the repository
master

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

Amazon CloudWatch Events (recommended)
Use Amazon CloudWatch Events to automatically start my pipeline when a change occurs

AWS CodePipeline
Use AWS CodePipeline to check periodically for changes

Output artifact format
Choose the output artifact format.

CodePipeline default
AWS CodePipeline uses the default zip format for artifacts in the pipeline. Does not include git

Full clone
AWS CodePipeline passes metadata about the repository that allows subsequent actions to do

Feedback English (US) Privacy Policy Terms of Use Cookie preferences

Screenshot of the AWS CodePipeline Create new pipeline screen, Step 4: Add deploy stage.

Developer Tools > CodePipeline > Pipelines > Create new pipeline

Step 1: Choose pipeline settings

Step 2: Add source stage

Step 3: Add build stage

Step 4: Add deploy stage

Step 5: Review

Add deploy stage Info

You cannot skip this stage
Pipelines must have at least two stages. Your second stage must be either a build or deployment stage. Choose a provider for either the build stage or deployment stage.

Deploy

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

Region

Feedback English (US) Privacy Policy Terms of Use Cookie preferences

Screenshot of the AWS IAM console showing the "Add deploy stage" step for a new policy.

Step 5

Review

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

Region
US East (N. Virginia)

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.
CodeDeployDemo

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.
MyDevInstances

Screenshot of the AWS CodePipeline console showing the "Editing: cicd-pipeline" pipeline.

Developer Tools

CodePipeline

- Source • CodeCommit
- Artifacts • CodeArtifact
- Build • CodeBuild
- Deploy • CodeDeploy
- Pipeline • CodePipeline
 - Getting started
 - Pipelines
 - Pipeline**
 - History

Edit: Source

Source **AWS CodeCommit**

+ Add stage

Edit: Deploy

Deploy

Save

