
AWS
re:Invent
NOV. 28 – DEC. 2, 2022 | LAS VEGAS, NV

SVS303

Building real-world serverless applications with AWS SAM

Eric Johnson

Principal Developer Advocate
AWS

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Who am I?

- Principal Developer Advocate – AWS
- Father of 5
- Brand new grandpa
- Drummer
- Foodie
- Twitter → @edjgeek



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Today’s agenda

- Intro to AWS SAM
- AWS SAM local
- AWS SAM Accelerate
- Code deployment
- New (or recent) features
- The development cycle
- Best practices

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Intro to AWS SAM

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.



Serverless Application Model

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS SAM has two parts

Transform templates

Shorthand syntax expresses resources and event source mappings; it provides infrastructure as code (IaC) for serverless applications

AWS Command Line Interface (AWS CLI)

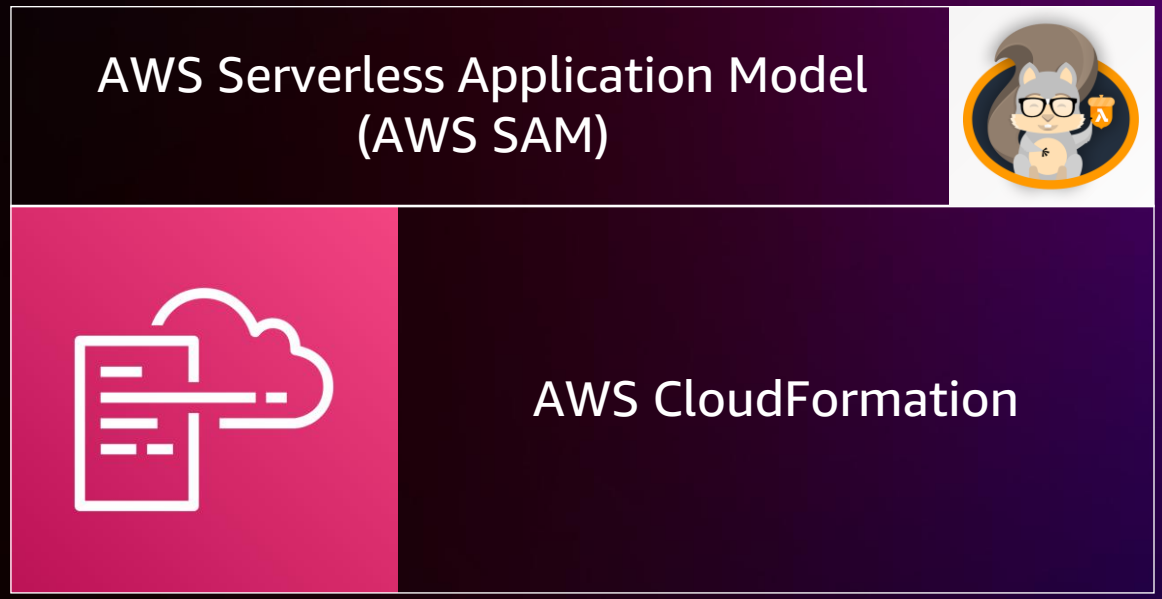
Provides tooling for local development with rapid iteration, debugging, build, packaging, and deployment for serverless applications



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS SAM transform templates

Built on top of
AWS CloudFormation



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS SAM transform templates

```
AWSTemplateFormatVersion: '2010-09-09'
Transform: AWS::Serverless-2016-10-31
Resources:
  GetProductsFunction:
    Type: AWS::Serverless::Function
    Properties:
      CodeUri: src/
      Handler: app.handler
      Runtime: python3.9
      Policies:
        - DynamoDBReadPolicy:
            TableName: !Ref ProductTable
      Events:
        GetProductsEvent:
          Type: Api
          Properties:
            Path: /products
            Method: get
  ProductTable:
    Type: AWS::Serverless::SimpleTable
```

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS SAM transform templates

```
AWSTemplateFormatVersion: '2010-09-09'
Transform: AWS::Serverless-2016-10-31
Resources:
  GetProductsFunction:
    Type: AWS::Serverless::Function
    Properties:
      CodeUri: src/
      Handler: app.handler
      Runtime: python3.9
      Policies:
        - DynamoDBReadPolicy:
            TableName: !Ref ProductTable
      Events:
        GetProductsEvent:
          Type: Api
          Properties:
            Path: /products
            Method: get
  ProductTable:
    Type: AWS::Serverless::SimpleTable
```



aws
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS SAM transform templates

```
AWSTemplateFormatVersion: '2010-09-09'
Transform: AWS::Serverless-2016-10-31
Resources:
  GetProductsFunction:
    Type: AWS::Serverless::Function
    Properties:
      CodeUri: src/
      Handler: app.handler
      Runtime: python3.9
      Policies:
        - DynamoDBReadPolicy:
            TableName: !Ref ProductTable
      Events:
        GetProductsEvent:
          Type: Api
          Properties:
            Path: /products
            Method: get
  ProductTable:
    Type: AWS::Serverless::SimpleTable
```



aws
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS SAM transform templates

```
AWSTemplateFormatVersion: '2010-09-09'
Transform: AWS::Serverless-2016-10-31
Resources:
  GetProductsFunction:
    Type: AWS::Serverless::Function
    Properties:
      CodeUri: src/
      Handler: app.handler
      Runtime: python3.9
      Policies:
        - DynamoDBReadPolicy:
            TableName: !Ref ProductTable
      Events:
        GetProductsEvent:
          Type: Api
          Properties:
            Path: /products
            Method: get
  ProductTable:
    Type: AWS::Serverless::SimpleTable
```





aws
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS SAM transform templates

```
AWSTemplateFormatVersion: '2010-09-09'
Transform: AWS::Serverless-2016-10-31
Resources:
  GetProductsFunction:
    Type: AWS::Serverless::Function
    Properties:
      CodeUri: src/
      Handler: app.handler
      Runtime: python3.9
      Policies:
        - DynamoDBReadPolicy:
            TableName: !Ref ProductTable
      Events:
        GetProductsEvent:
          Type: Api
          Properties:
            Path: /products
            Method: get
  ProductTable:
    Type: AWS::Serverless::SimpleTable
```

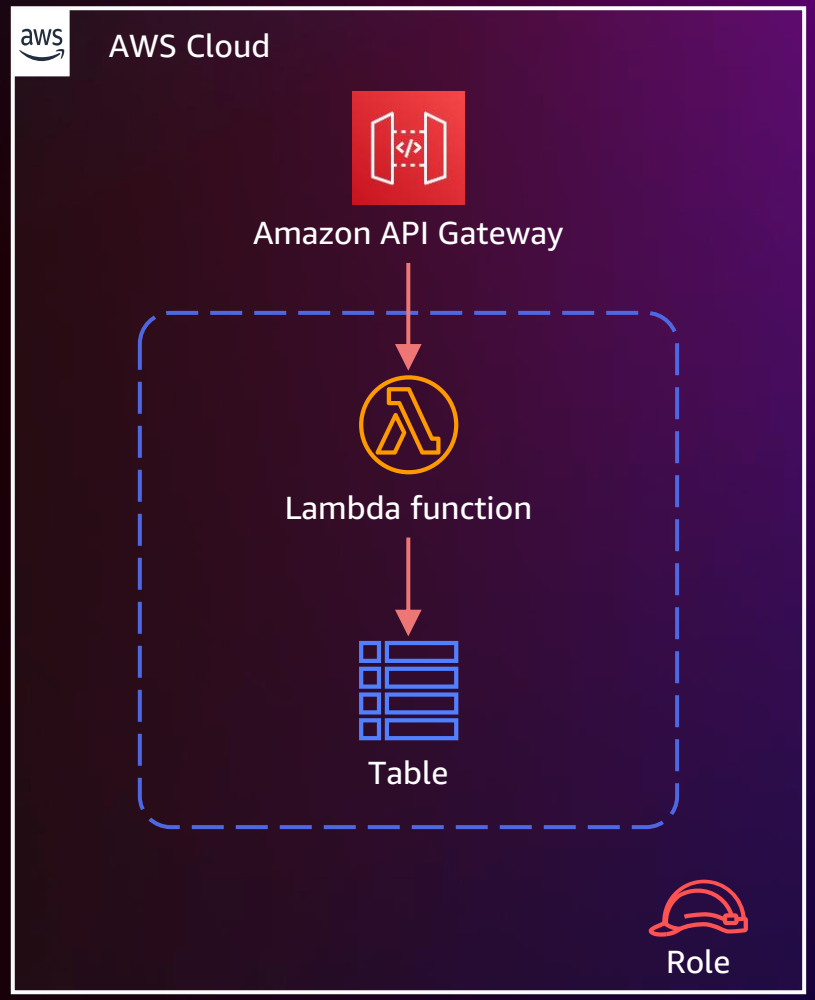


aws
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS SAM transform templates

```
AWSTemplateFormatVersion: '2010-09-09'
Transform: AWS::Serverless-2016-10-31
Resources:
  GetProductsFunction:
    Type: AWS::Serverless::Function
    Properties:
      CodeUri: src/
      Handler: app.handler
      Runtime: python3.9
      Policies:
        - DynamoDBReadPolicy:
            TableName: !Ref ProductTable
      Events:
        GetProductsEvent:
          Type: Api
          Properties:
            Path: /products
            Method: get
  ProductTable:
    Type: AWS::Serverless::SimpleTable
```

Deployment



aws
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS SAM serverless resources

8 serverless resource types

AWS::Serverless::Function

AWS::Serverless::Api

AWS::Serverless::HttpApi

AWS::Serverless::SimpleTable

AWS::Serverless::LayerVersion

AWS::Serverless::Application

AWS::Serverless::StateMachine

AWS::Serverless::Connector

SAM Version 2016-10-31

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Lambda function event sources

16

function event source types supported in AWS SAM

- Amazon S3
- Amazon SNS
- Amazon Kinesis
- Amazon DynamoDB
- Amazon SQS
- Api
- HttpApi
- Schedule
- CloudWatchEvent
- CloudWatchLogs
- IoTRule
- Alexa skills
- Amazon Cognito
- EventBridgeRule
- Amazon MSK
- Amazon MQ

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS SAM serverless resources – IAM policies

75+

managed templates,
with more being added

s12d.com/sam-policies

Policy template table

The following is a table of the available policy templates.

Policy Template	Description
AcmGetCertificatePolicy	Gives a permission to read a certificate from AWS Certificate Manager.
AMIDescribePolicy	Gives permission to describe Amazon Machine Images (AMIs).
AthenaQueryPolicy	Gives permissions to execute Athena queries.
AWSSecretsManagerGetSecretValuePolicy	Gives permission to get the secret value for the specified AWS Secrets Manager secret.
AWSSecretsManagerRotationPolicy	Gives permission to rotate a secret in AWS Secrets Manager.
CloudFormationDescribeStacksPolicy	Gives permission to describe AWS CloudFormation stacks.
CloudWatchDashboardPolicy	Gives permissions to put metrics to operate on CloudWatch dashboards.
CloudWatchDescribeAlarmHistoryPolicy	Gives permission to describe CloudWatch alarm history.
CloudWatchPutMetricPolicy	Gives permission to send metrics to CloudWatch.
CodeCommitCrudPolicy	Gives permissions to create/read/update/delete objects within a specific CodeCommit repository.
CodeCommitReadPolicy	Gives permissions to read objects within a specific CodeCommit repository.
CodePipelineLambdaExecutionPolicy	Gives permission for a Lambda function invoked by CodePipeline to report the status of the job.
CodePipelineReadOnlyPolicy	Gives read permission to get details about a CodePipeline pipeline.

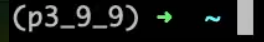
aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS SAM CLI

App development
- init
- build
- local
- logs
- traces
- sync

Stack management
- validate
- package
- delete
- deploy
- pipeline
- publish*



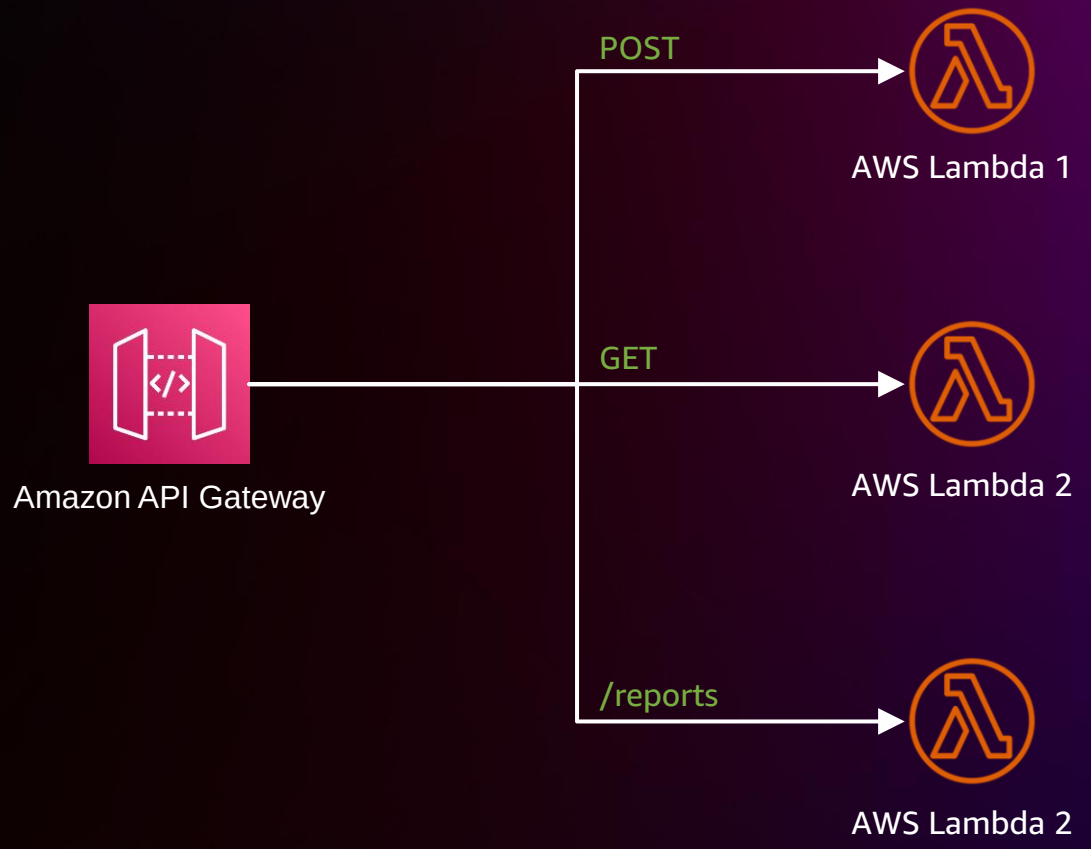
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS SAM local

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS SAM local – start-api

Emulates an Amazon API Gateway endpoint backed by one or more Lambda functions



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS SAM local – invoke

Invokes a Lambda function directly emulating a service emitting an event and the Lambda service reacting


Amazon S3


AWS Step Functions


Amazon Lex


Amazon DynamoDB


AWS Lambda


Amazon Cognito


Amazon SageMaker


AWS Batch


Amazon CloudWatch

CLI flags
--event -e
--shutdown

aws
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS SAM local – generate-event

Generates a mock event from each of the supported services

```
sam local generate-event batch get-job
{
  "jobName": "hello_world",
  "jobQueue": "default",
  "jobDefinition": "hello_world"
}
```

alb
alexa-skills-kit
alexa-smart-home
apigateway
appsync
batch
cloudformation
cloudfront
cloudwatch
codecommit
codepipeline
cognito
config
connect
dynamodb
kinesis
lex
lex-v2
rekognition
s3
sagemaker
ses
sns
sqs
stepfunctions
workmail

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS SAM local – start-lambda

Starts a local endpoint that emulates the Lambda service

```
$ aws lambda invoke --function-name ”GetProductsFunction" --endpoint-url
"http://127.0.0.1:3001" --no-verify-ssl out.txt
```

Use this to test AWS CLI invocations and AWS SDK invocations

CLI flags
--event -e
--shutdown

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS SAM Accelerate

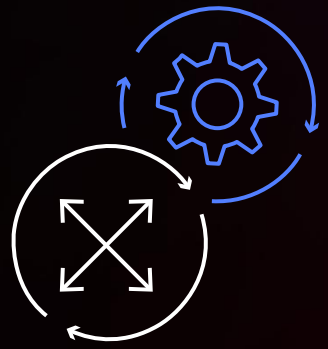
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS SAM Accelerate



sam build

- Parallel builds
- Incremental builds
- Temporary layers



sam sync

- Update code without redeploying
- Watch for changed files



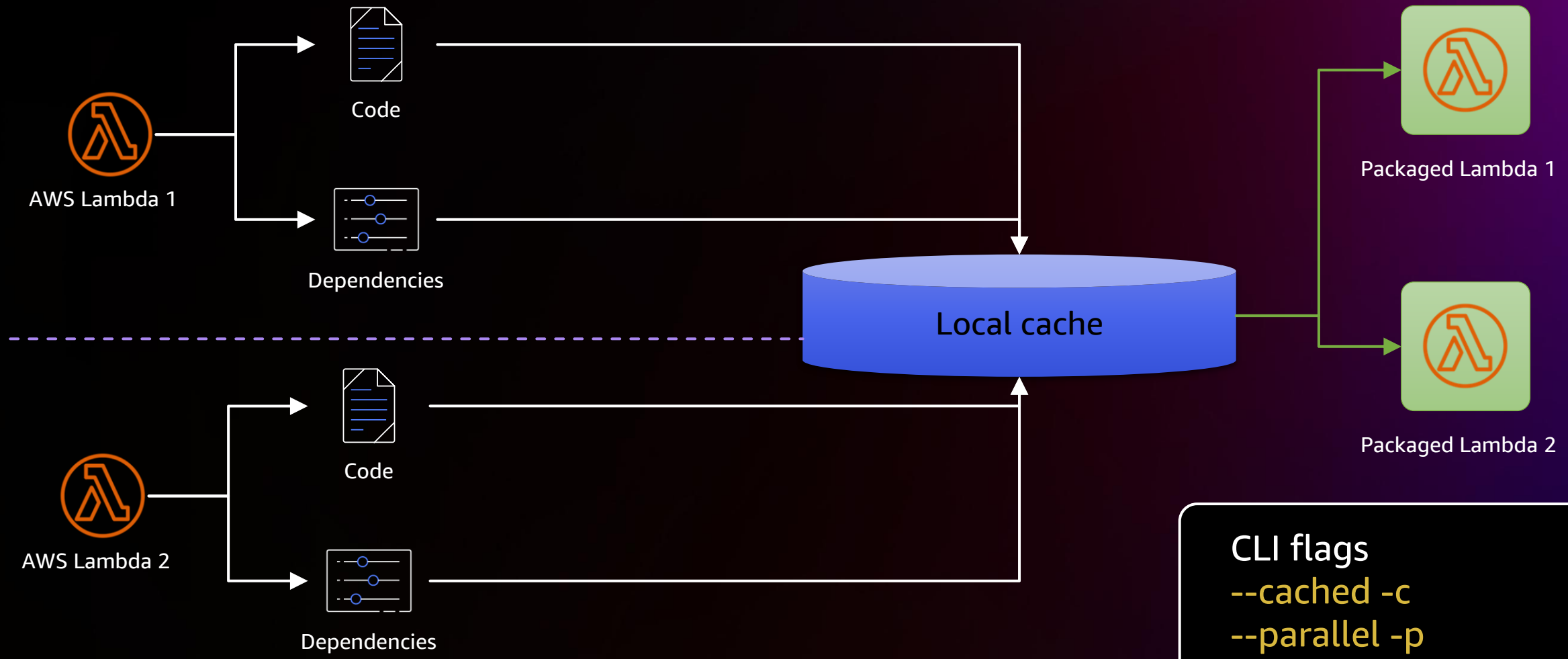
sam logs

- Get aggregated logs in your local terminal
- Filter logs
- Include traces

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

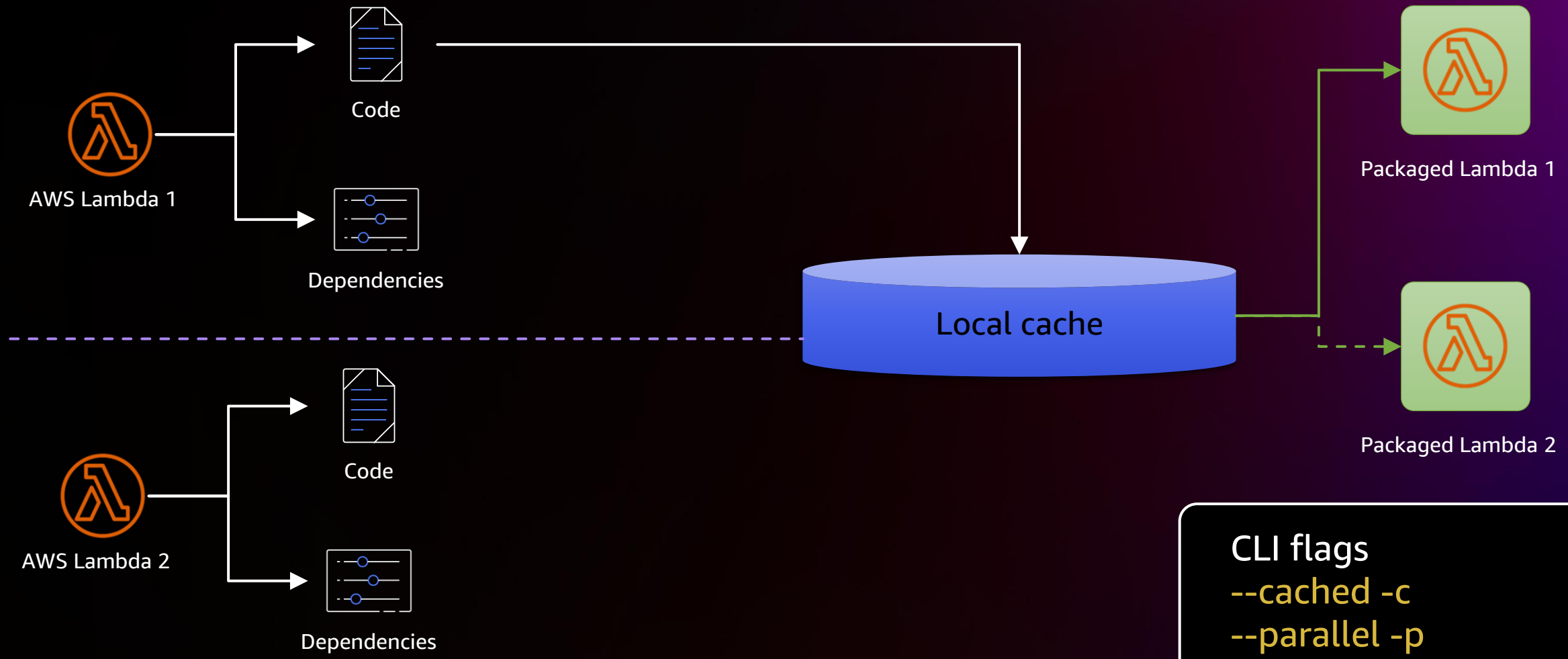
AWS SAM Accelerate – sam build



CLI flags
--cached -c
--parallel -p
--use-container -u

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

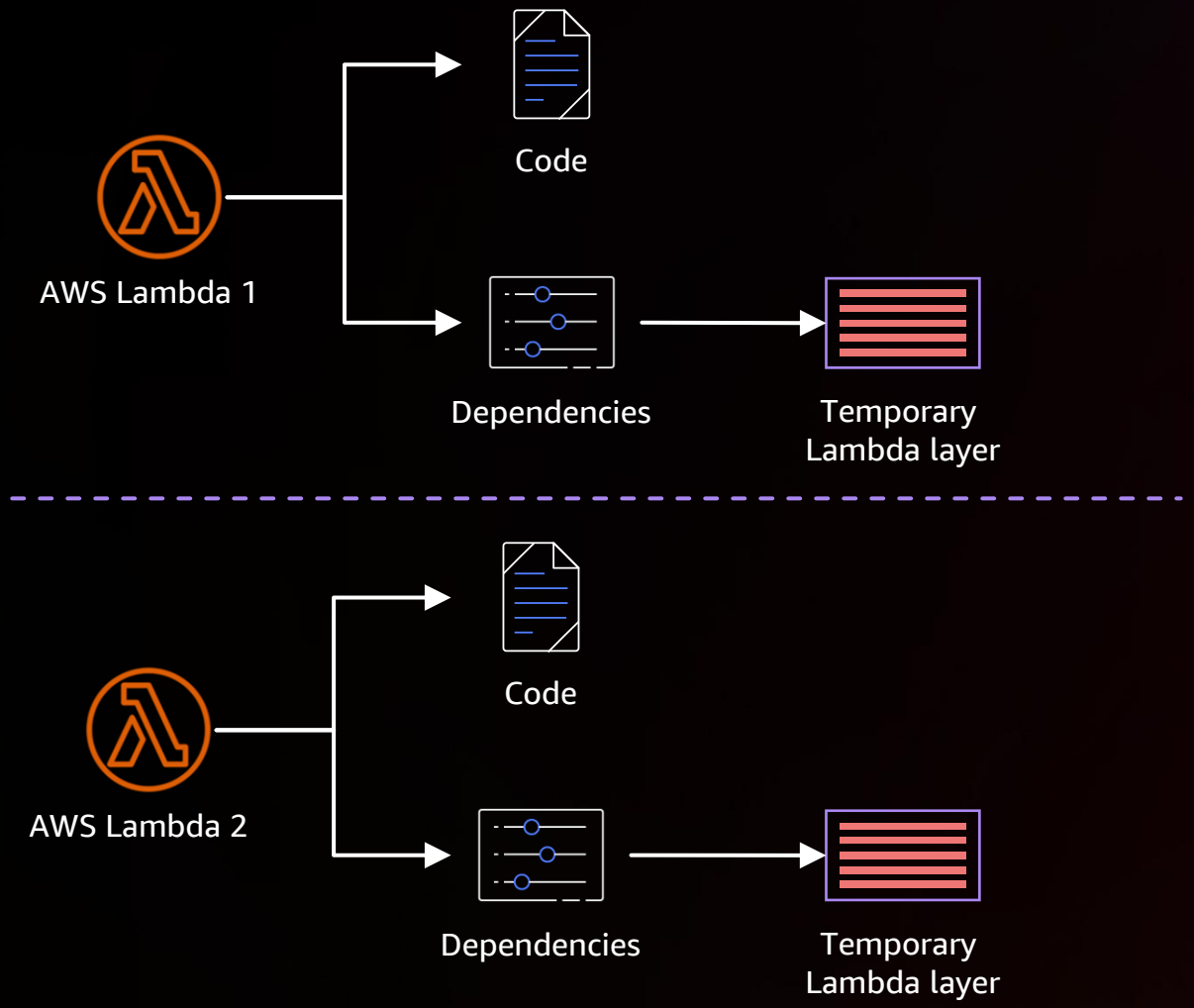
AWS SAM Accelerate – sam build



CLI flags
--cached -c
--parallel -p
--use-container -u

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS SAM Accelerate – sam sync

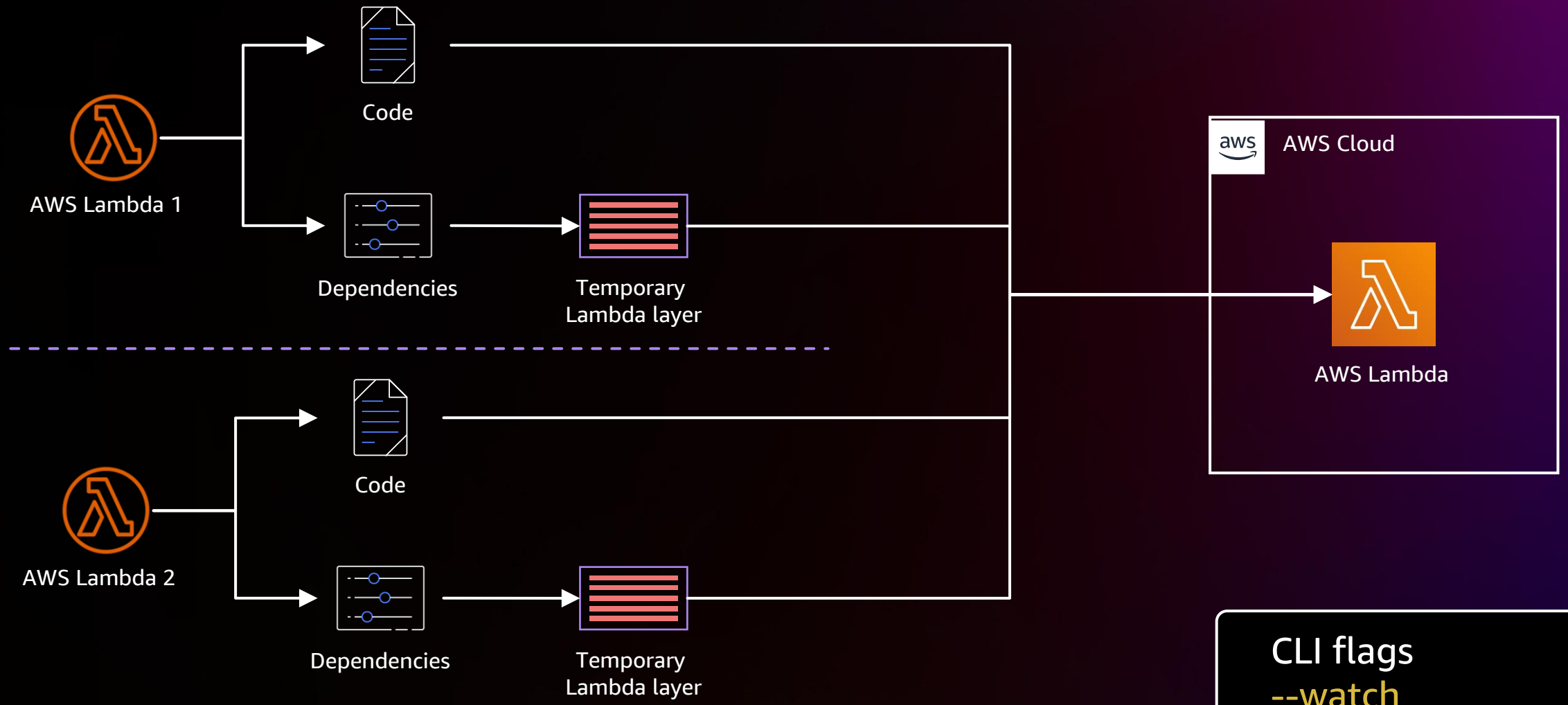


Note: the sync command automatically calls AWS SAM to build for you

CLI flags
--watch

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

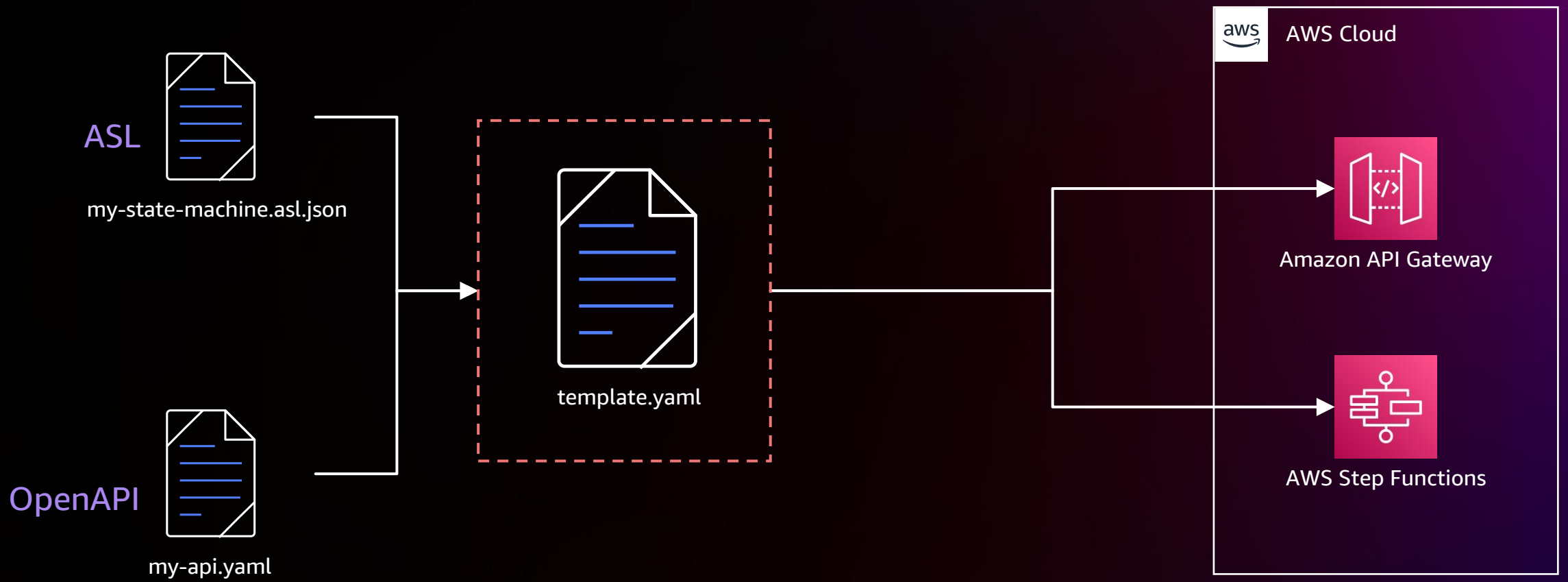
AWS SAM Accelerate – sam sync



CLI flags
--watch

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

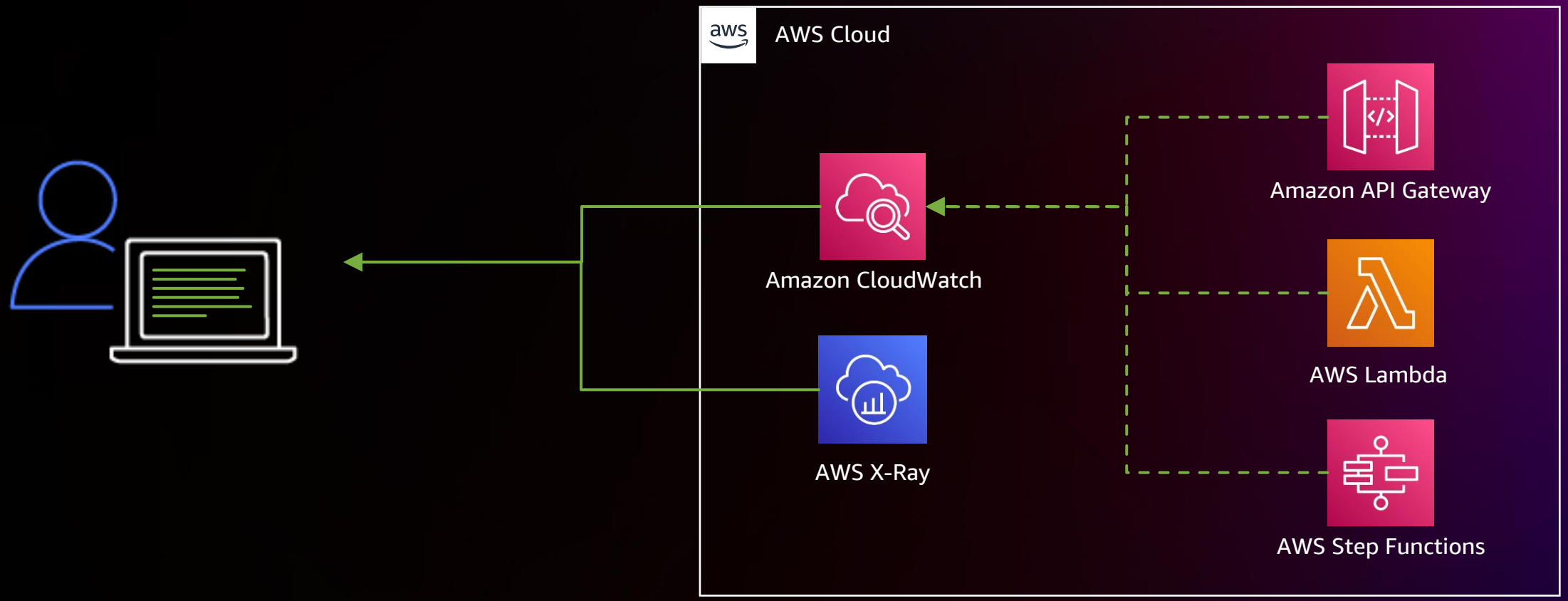
AWS SAM Accelerate – sam sync



CLI flags
--watch

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

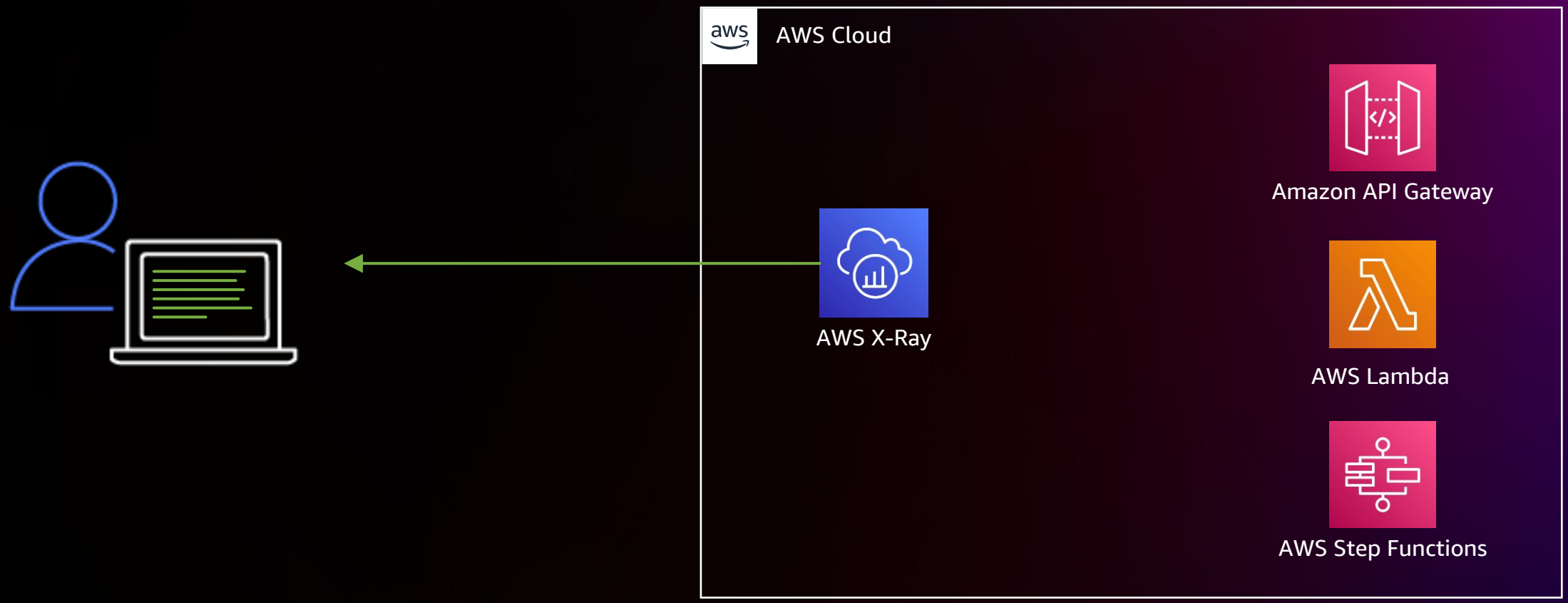
AWS SAM Accelerate – sam logs



CLI flags
--tail -t

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS SAM Accelerate – sam traces



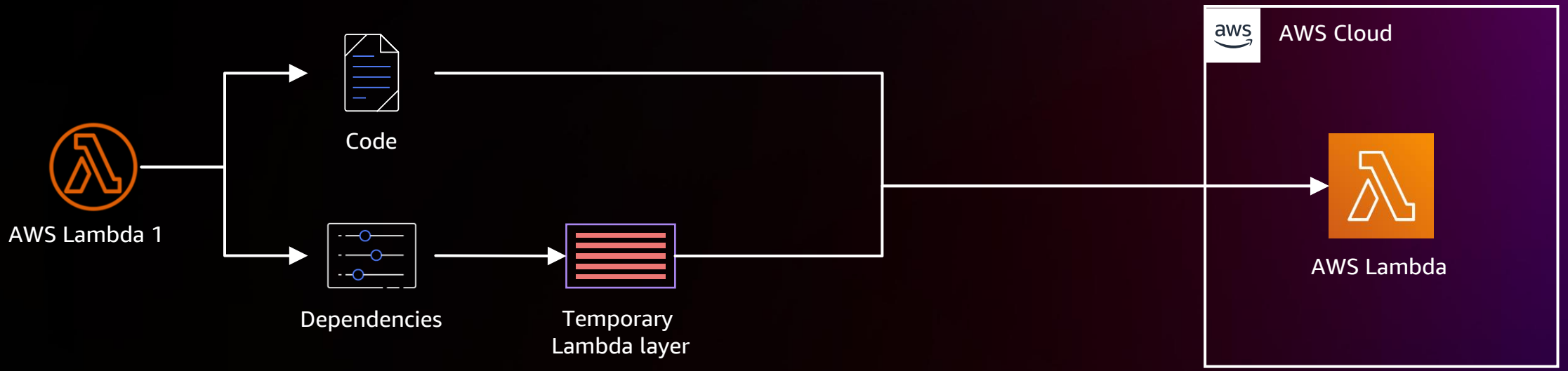
CLI flags
--tail -t

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Code deployment

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS SAM sync

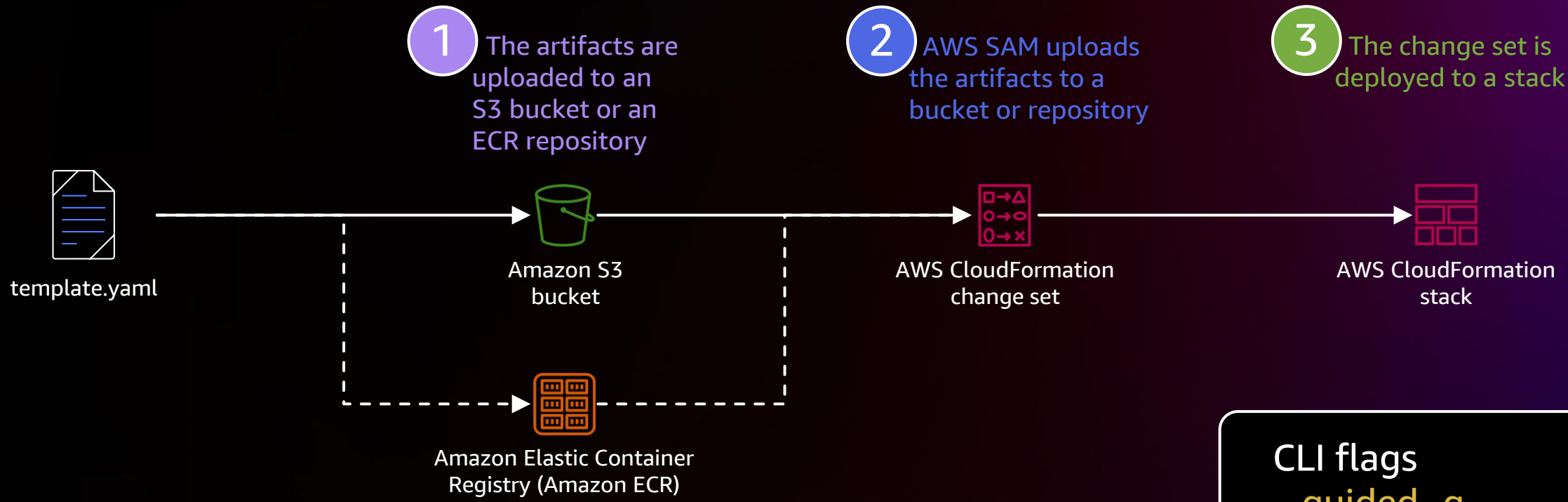


STOP: for development only

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS SAM deploy

Deploys or updates the current stack using AWS CloudFormation

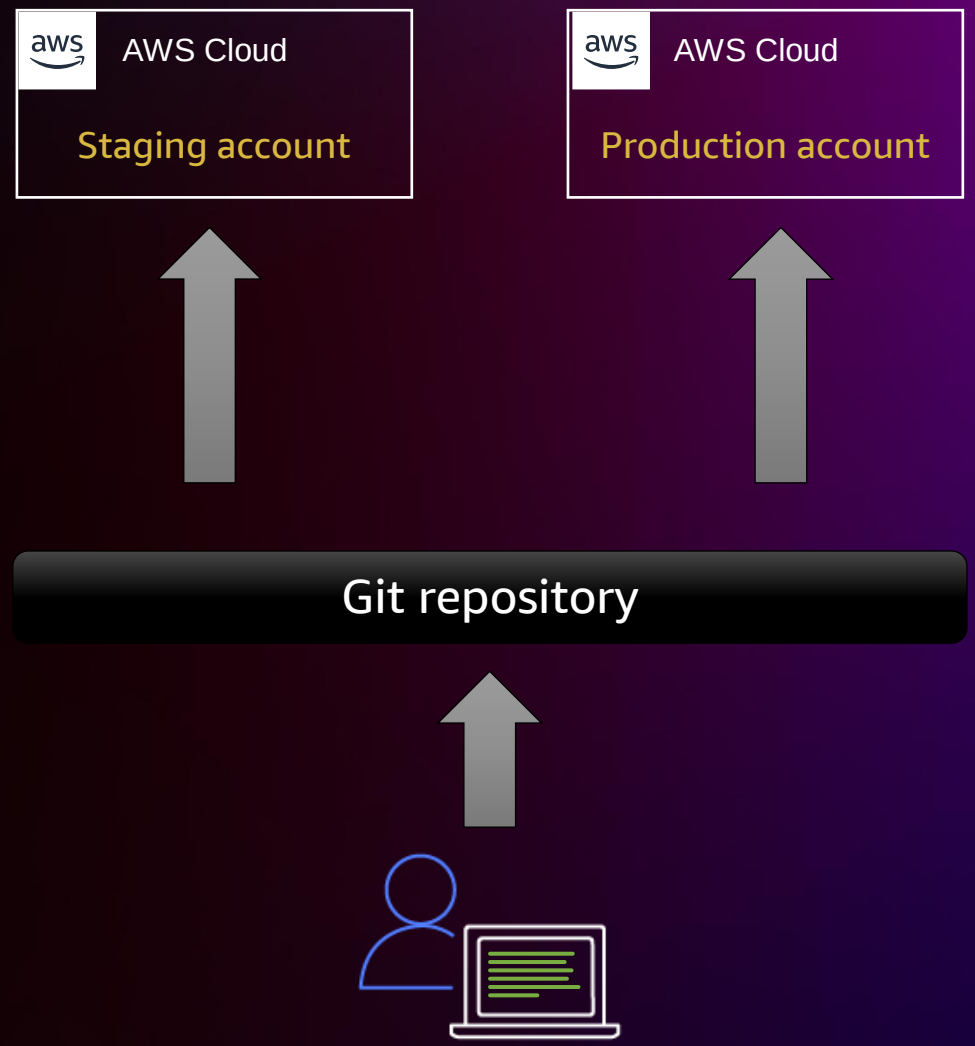


CLI flags
--guided -g
--config-env
--on-failure

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS SAM pipeline

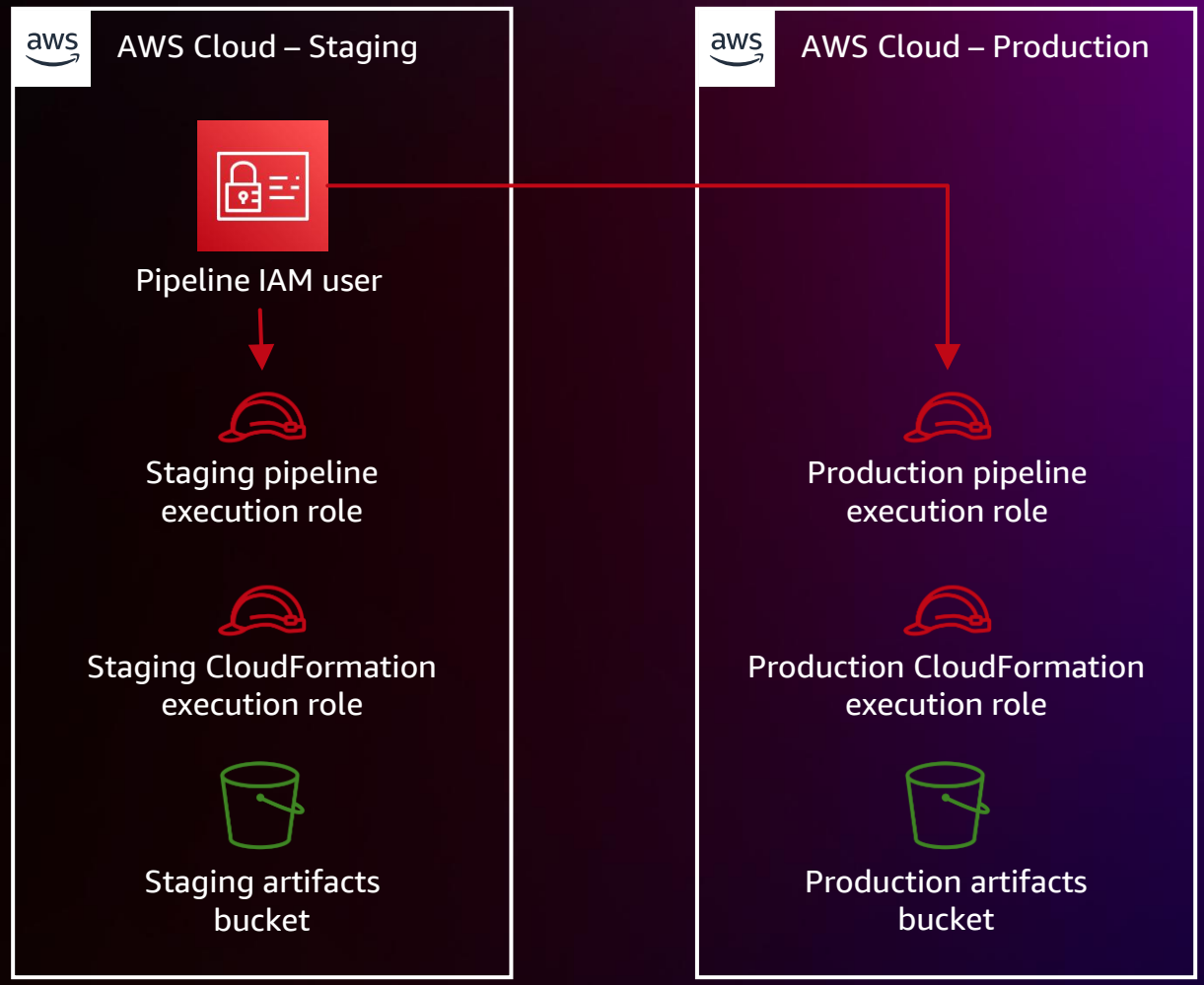
- Generates the resource and configuration required to build a continuous integration and continuous delivery (CI/CD) pipeline
- Supports
  - AWS CodeSuite
  - GitHub Actions
  - Jenkins CI
  - GitLab



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS SAM pipeline bootstrap

Creates the AWS resources and permissions required to deploy application artifacts from your code repository into your AWS environments



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

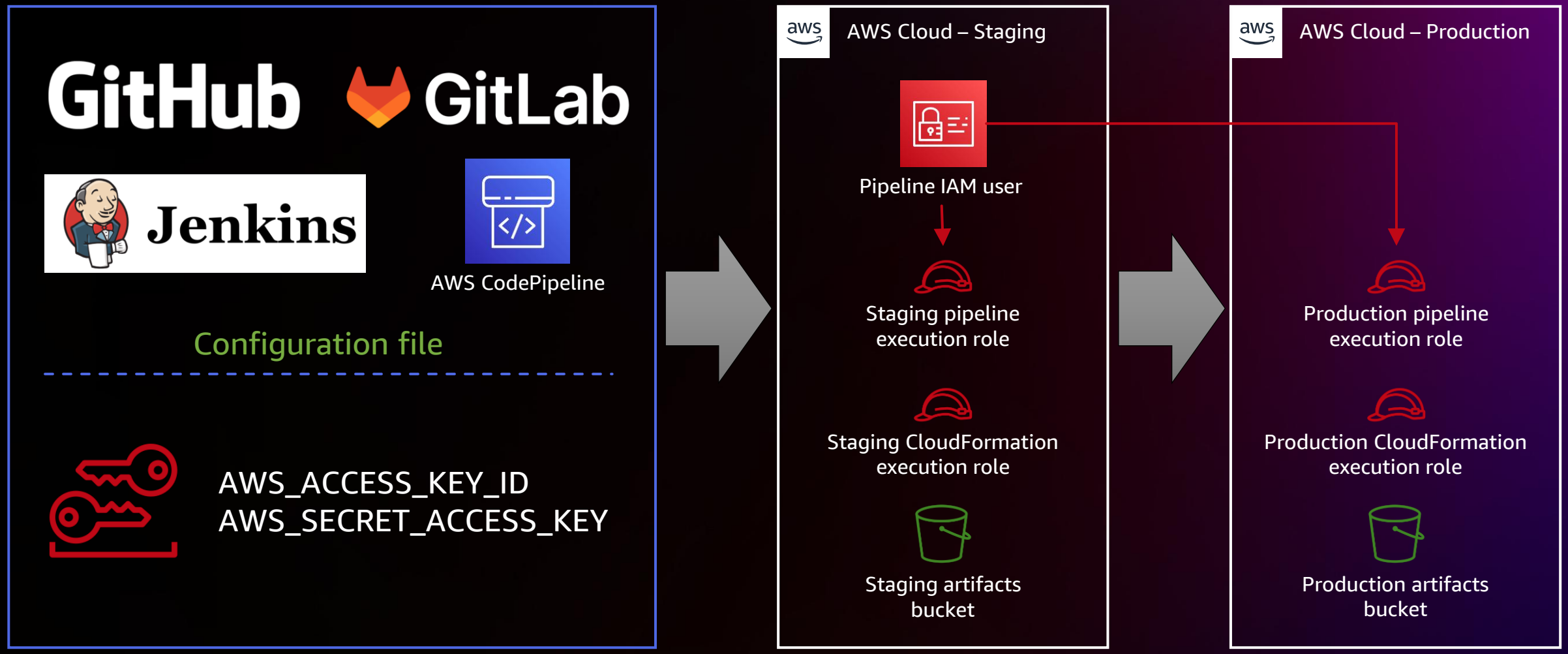
AWS SAM pipeline init



Generates a pipeline configuration file that your CI/CD system can use to deploy serverless applications using AWS SAM

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS SAM pipeline



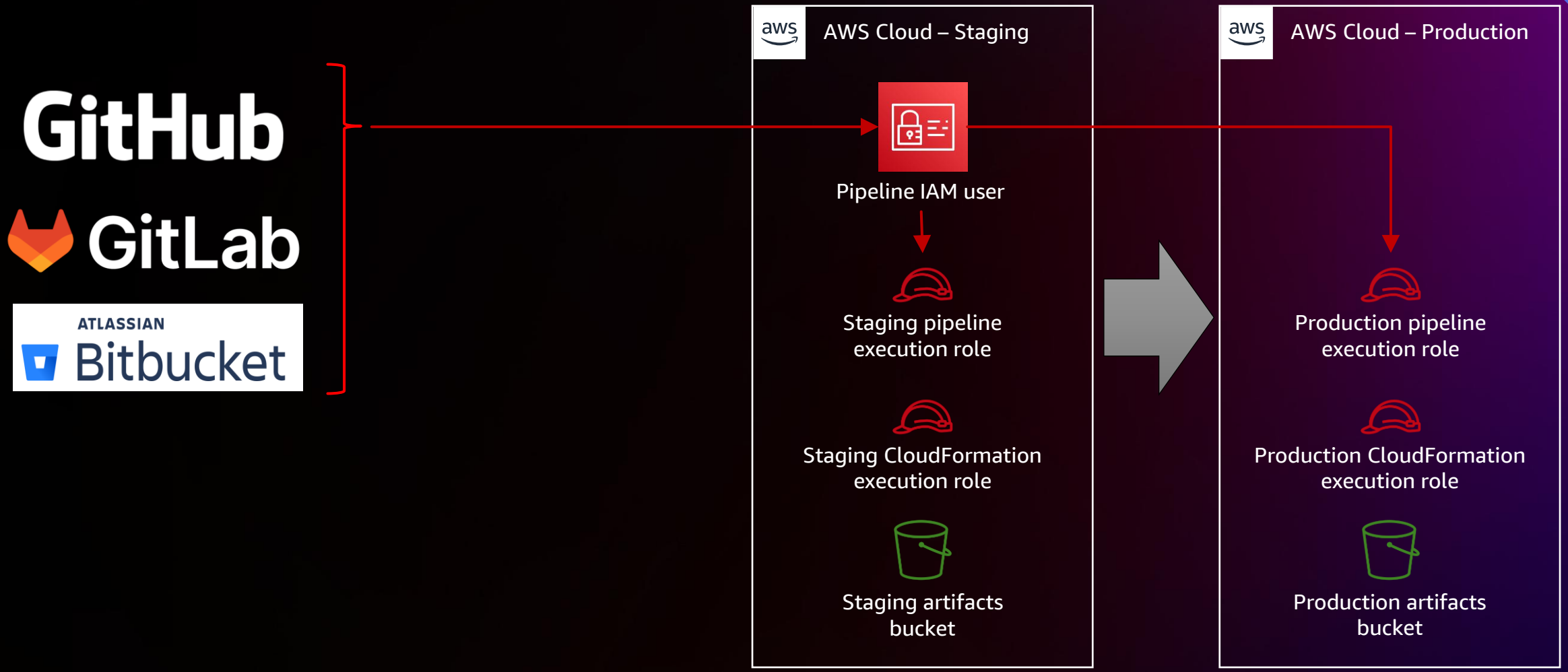
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

New (or recent) features

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS SAM pipelines OIDC authorizer support

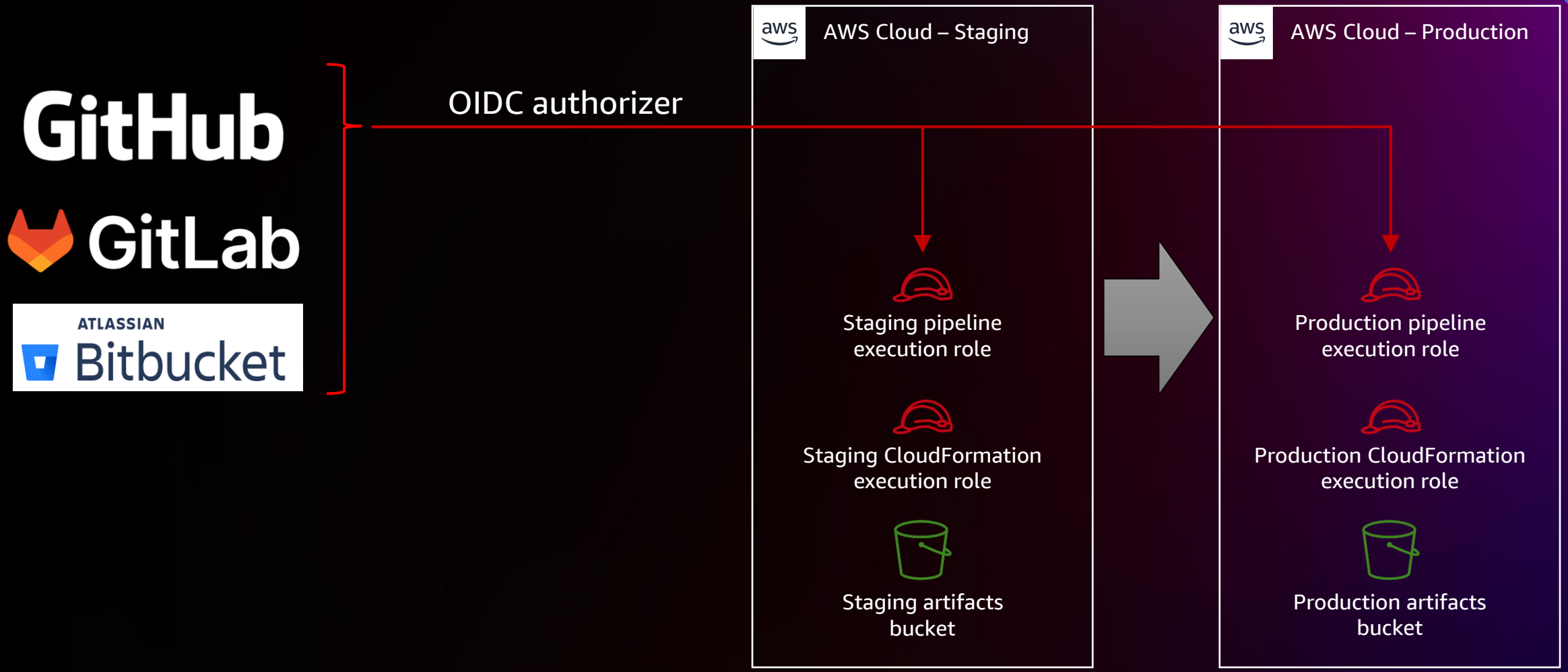
Preview



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS SAM pipelines OIDC authorizer support

Preview



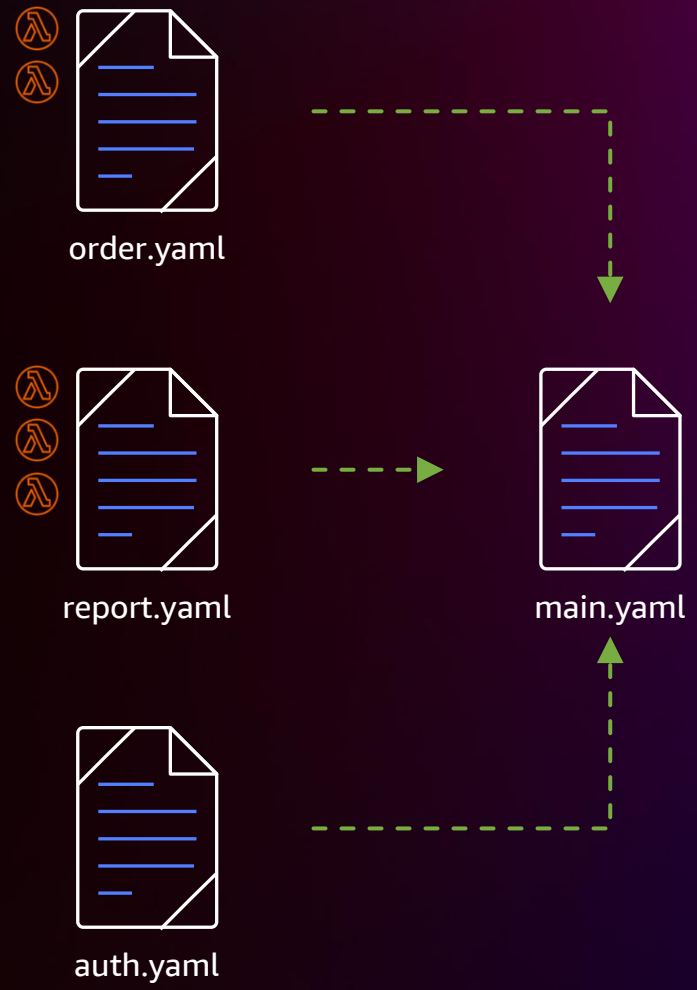
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Nested stack support for AWS SAM Accelerate

Recent

```
OrderStack
  Type: AWS::Serverless::Application
  Properties:
  Location: ./orders.yaml
  Parameters:
    AppId: 112d43

ReportStack
  Type: AWS::Serverless::Application
  Properties:
  Location: ./reports.yaml
  Parameters:
    AppId: 112d43
```

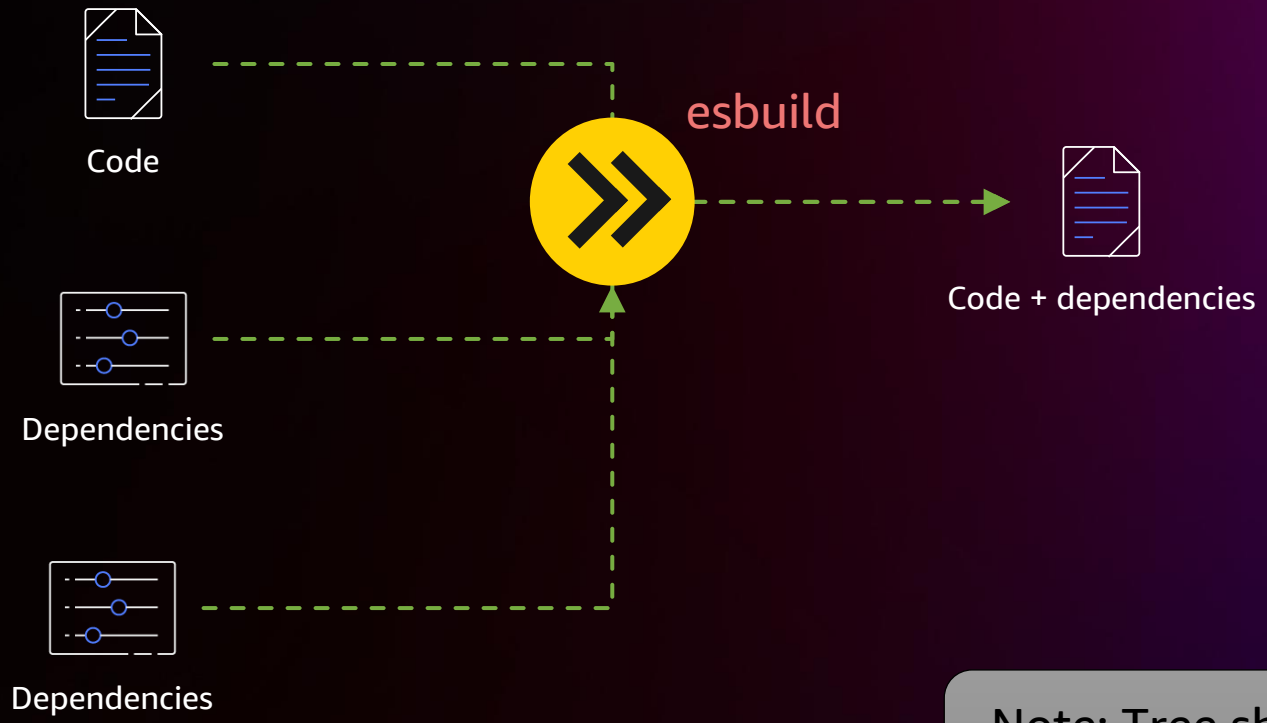


aws
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Native esbuild support for AWS SAM build

Recent

```
Metadata:
  BuildMethod: esbuild
  BuildProperties:
    Minify: true
    Target: es2020
    EntryPoints:
      - app.ts
```



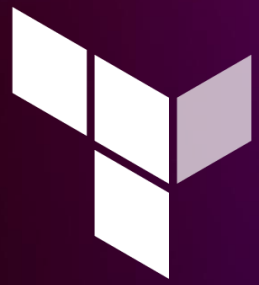
Note: Tree shaking is a term commonly used within a JavaScript context to describe the removal of dead code

aws
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS SAM local support for HashiCorp Terraform

Preview

```
resource "aws_lambda_function" "test_lambda" {
  filename = "lambda_function_payload.zip”
  function_name = "lambda_function_name”
  role = aws_iam_role.iam_for_lambda.arn
  handler = "index.test”
  source_code_hash = filebase64sha256("payload.zip")
  runtime = "nodejs16.x”
  environment {
    variables = { foo = "bar" }
  }
}
```



HashiCorp Terraform

Commands
sam local invoke
sam local start-lambda

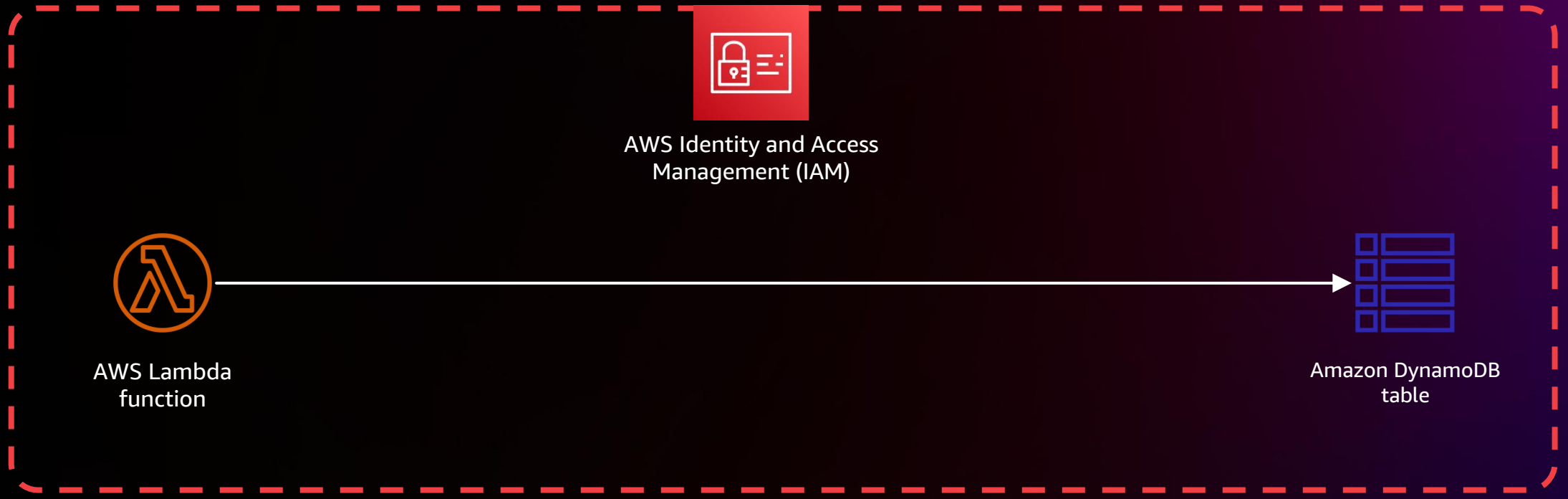
aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS SAM connectors

Preview

Connectors allow the developer to describe how data and events flow between two resources and the level of permissions required



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS SAM connectors

Preview

Source

```
HelloWorldFunction:
  Type: AWS::Serverless::Function
  Properties:
    CodeUri: hello-world/
    Handler: app.lambdaHandler
    Runtime: nodejs16.x
    Events:
      HelloWorld:
        Type: Api
        Properties:
          Path: /hello
          Method: get
```

Connector

```
MyTableConnector:
  Type: AWS::Serverless::Connector
  Properties:
    Source:
      Id: HelloWorldFunction
    Destination:
      Id: HelloWorldTable
    Permissions:
      - Write
      - Read
```

Destination

```
HelloWorldTable:
  Type: AWS::Serverless::SimpleTable
```

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS SAM connectors

Preview

Read permissions

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "dynamodb:GetItem",
                "dynamodb:Query",
                "dynamodb:Scan",
                "dynamodb:BatchGetItem",
                "dynamodb:ConditionCheckItem",
                "dynamodb:PartiQLSelect"
            ],
            "Resource": [
                "arn:aws:dynamodb:us-west-2:088483494489:table/connectors-HelloWorldTable-NVNO16YXY86W",
                "arn:aws:dynamodb:us-west-2:088483494489:table/connectors-HelloWorldTable-NVNO16YXY86W/index/*"
            ],
            "Effect": "Allow”
. . .
```

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS SAM connectors

Preview

Write permissions

```
. . .
        {
            "Action": [
                "dynamodb:PutItem",
                "dynamodb:UpdateItem",
                "dynamodb:DeleteItem",
                "dynamodb:BatchWriteItem",
                "dynamodb:PartiQLDelete",
                "dynamodb:PartiQLInsert",
                "dynamodb:PartiQLUpdate"
            ],
            "Resource": [
                "arn:aws:dynamodb:us-west-2:088483494489:table/connectors-HelloWorldTable-NVNO16YXY86W",
                "arn:aws:dynamodb:us-west-2:088483494489:table/connectors-HelloWorldTable-NVNO16YXY86W/index/*"
            ],
            "Effect": "Allow"
        }
    ]
}
```

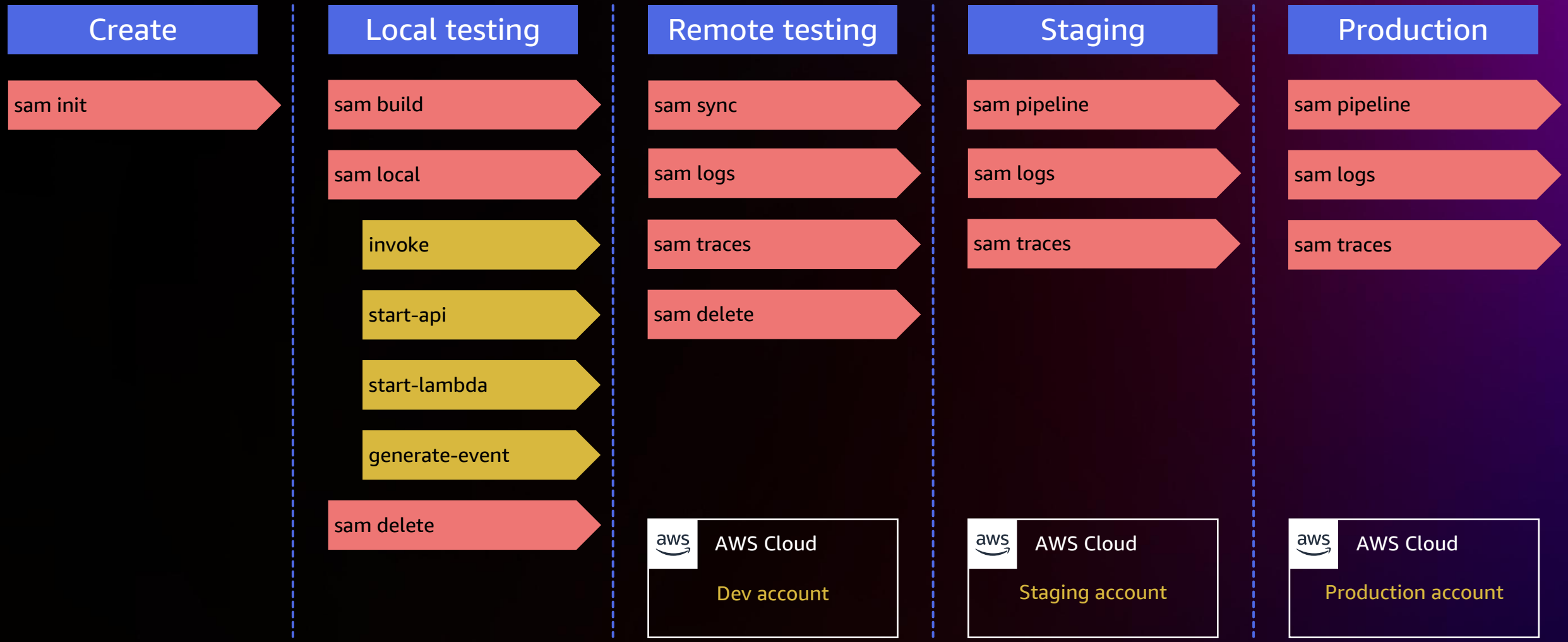
aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

The development cycle

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

The development cycle



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

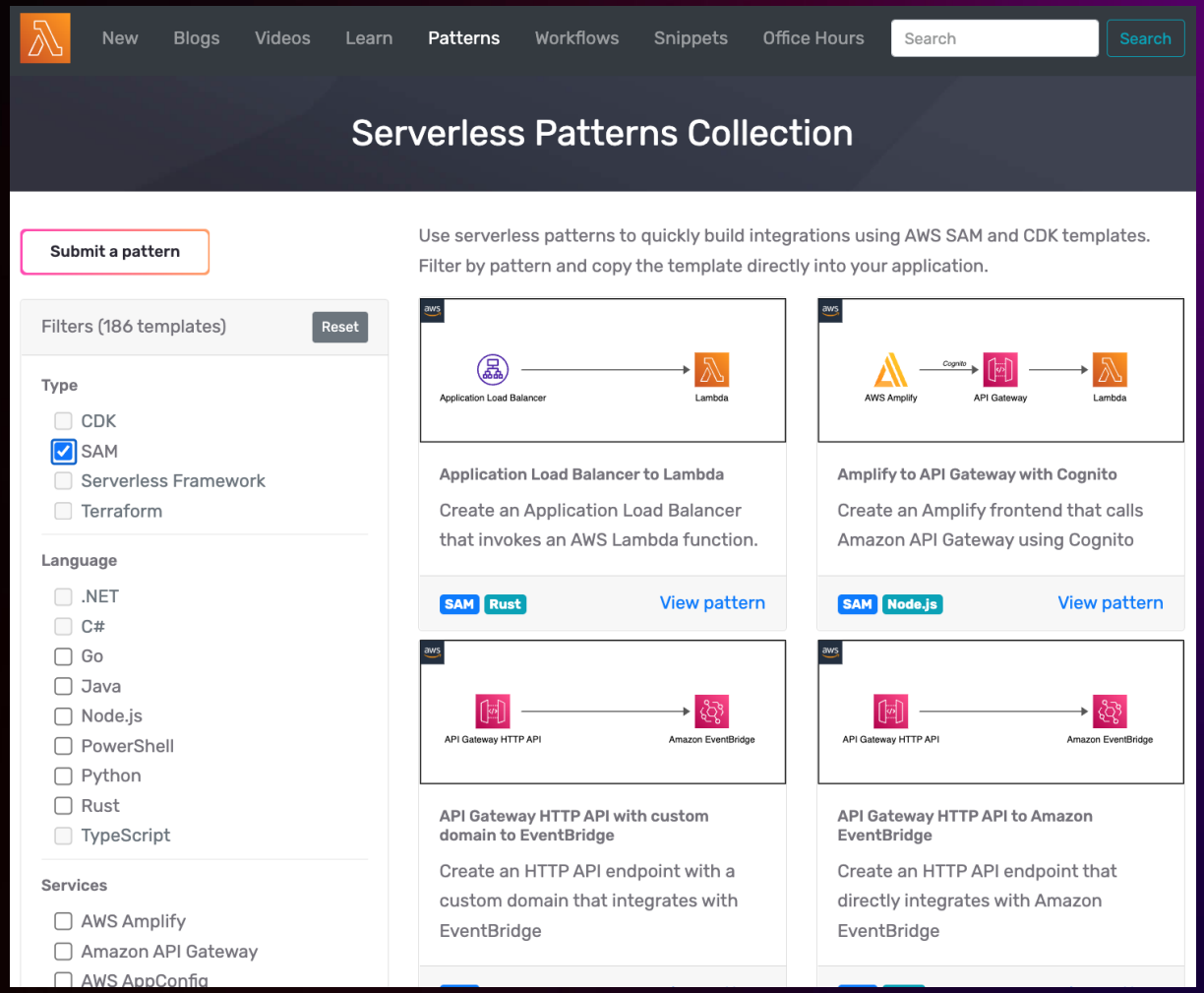
Best practices

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Start with a pattern

https://serverlessland.com

- Filter by IaC
- Filter by language
- Filter by service
- To date, there are 186 AWS SAM template patterns

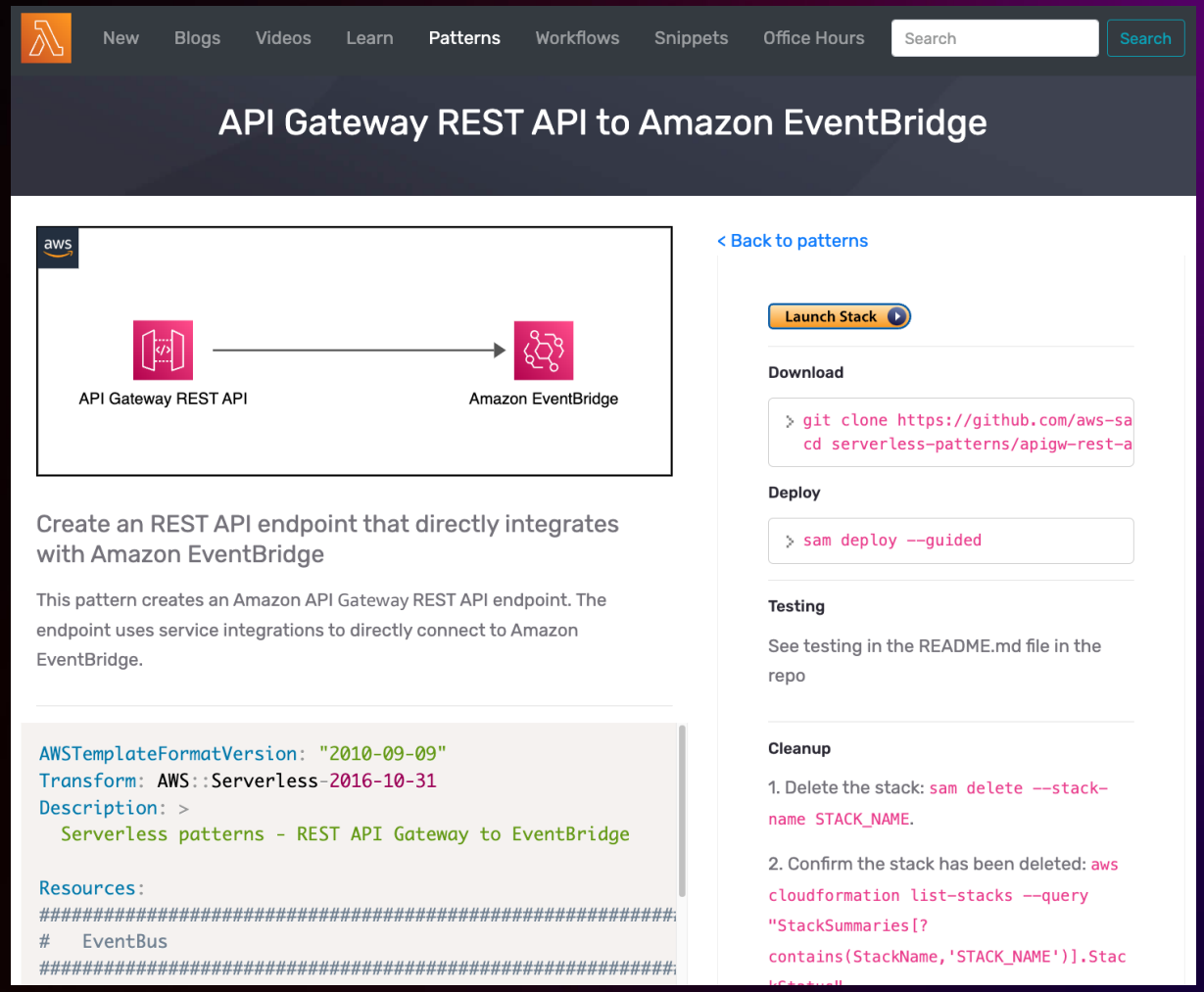


© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Start with a pattern

Each pattern has

- The AWS SAM template
- A link to the repository
- Deploy and cleanup instructions
- Additional resources



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Use the config.toml file

```
version = 0.1
[default]
[default.deploy]
[default.deploy.parameters]
stack_name = ”staging-app"
region = "us-west-2"
capabilities = "CAPABILITY_IAM"
profile = ”staging-profile”

[default.build]
[default.build.parameters]
beta_features="true”
Cached=“true”
Parallel=“true”

[prod]
[prod.deploy]
[prod.deploy.parameters]
stack_name = ”production-app"
region = "us-east-1"
capabilities = "CAPABILITY_IAM"
profile = ”production-profile"
```

The config.toml file enables developers to set defaults for each AWS SAM command

- Differentiate between environments
- If you pass it as a flag, you can set a default
- Flags override configuration settings

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Use managed policies when possible

AWS SAM connectors

```
MyTableConnector:
  Type: AWS::Serverless::Connector
  Properties:
    Source:
      Id: HelloWorldFunction
    Destination:
      Id: HelloWorldTable
    Permissions:
      - Write
      - Read
```

AWS SAM policy templates

Policy template table

The following is a table of the available policy templates.

Policy Template	Description
AcmGetCertificatePolicy	Gives a permission to read a certificate from AWS Certificate Manager.
AMIDescribePolicy	Gives permission to describe Amazon Machine Images (AMIs).
AthenaQueryPolicy	Gives permissions to execute Athena queries.
AWSSecretsManagerGetSecretValuePolicy	Gives permission to get the secret value for the specified AWS Secrets Manager secret.
AWSSecretsManagerRotationPolicy	Gives permission to rotate a secret in AWS Secrets Manager.
CloudFormationDescribeStacksPolicy	Gives permission to describe AWS CloudFormation stacks.
CloudWatchDashboardPolicy	Gives permissions to put metrics to operate on CloudWatch dashboards.
CloudWatchDescribeAlarmHistoryPolicy	Gives permission to describe CloudWatch alarm history.
CloudWatchPutMetricPolicy	Gives permission to send metrics to CloudWatch.
CodeCommitCrudPolicy	Gives permissions to create/read/update/delete objects within a specific CodeCommit repository.
CodeCommitReadPolicy	Gives permissions to read objects within a specific CodeCommit repository.
CodePipelineLambdaExecutionPolicy	Gives permission for a Lambda function invoked by CodePipeline to report the status of the job.
CodePipelineReadOnlyPolicy	Gives read permission to get details about a CodePipeline pipeline.

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Let AWS SAM name your resources

- Don’t hardcode names
- Helps to avoid naming clashes

```yaml
Resources:
  HelloWorldFunction:
    Type: AWS::Serverless::Function
    Properties:
      FunctionName: MyFunction
      CodeUri: hello-world/
      Handler: app.lambdaHandler
      Runtime: nodejs16.x
      Events:
        HelloWorld:
          Type: Api
          Properties:
            Path: /hello
            Method: get
    Metadata:
      BuildMethod: esbuild
      BuildProperties:
        Minify: true
        Target: es2020
        EntryPoints:
          - app.ts
```

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Working with Step Functions

1. Use sam sync –stack-name [Your stack name] --watch to deploy the bare minimum application

statemachine.asl.json

```
{
  "Comment": ”Minimal state machine",
  "StartAt": "Pass",
  "States": {
    "Pass": {
      "Type": "Pass",
      "End": true
    }
  }
}
```

template.yaml

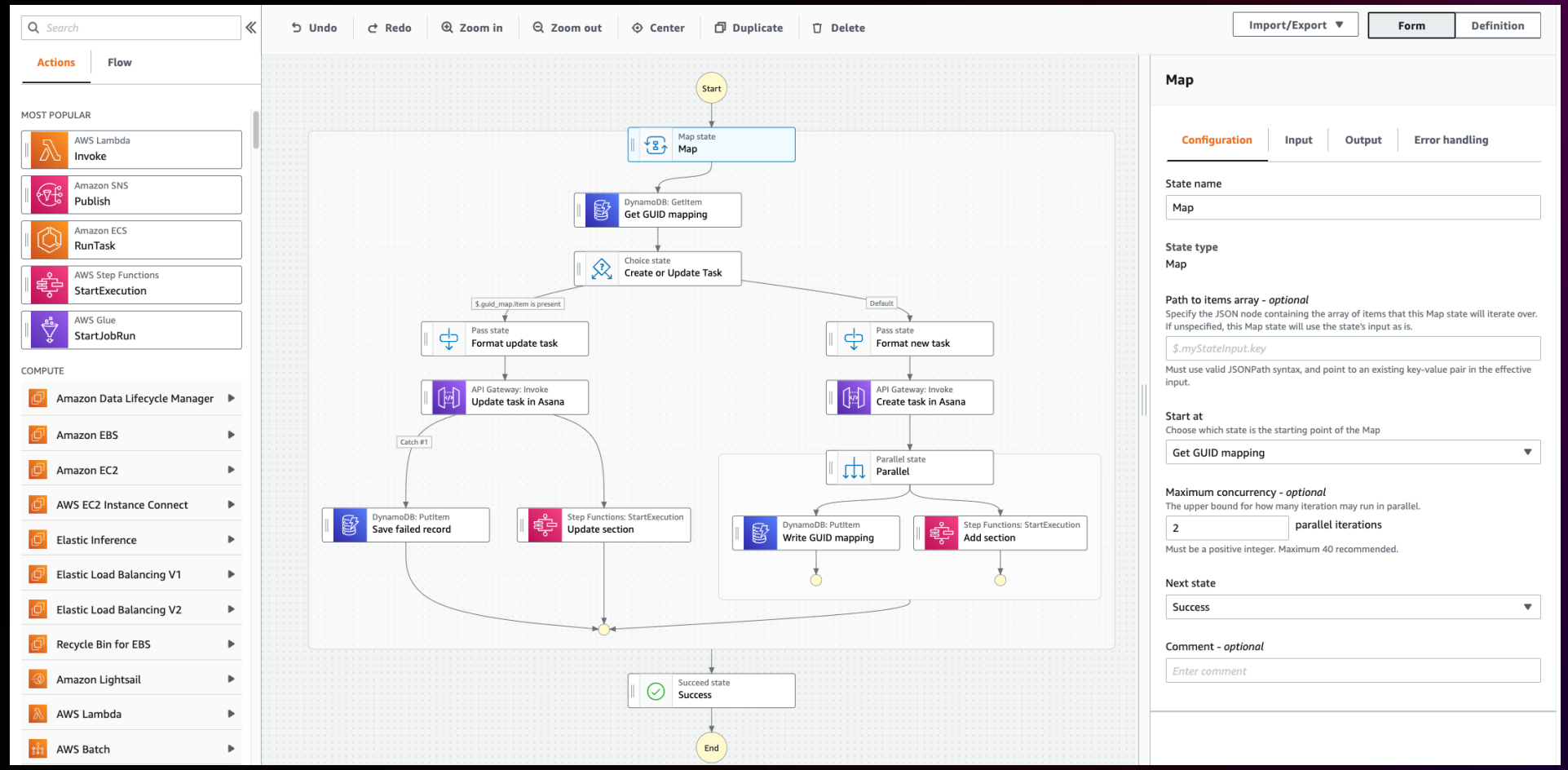
```
...
Orchestrator:
  Type: AWS::Serverless::StateMachine
  Properties:
    Type: EXPRESS
    DefinitionUri: ./statemachine.asl.json
    Logging:
      Destinations:
        - CloudWatchLogsLogGroup:
            LogGroupArn: !GetAtt OrchestratorLogGroup.Arn
      IncludeExecutionData: true
      Level: ALL
    Policies:
      - DynamoDBReadPolicy:
        TableName: !Ref ReportTable
      - DynamoDBWritePolicy:
        TableName: !Ref ReportTable
```

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Working with Step Functions

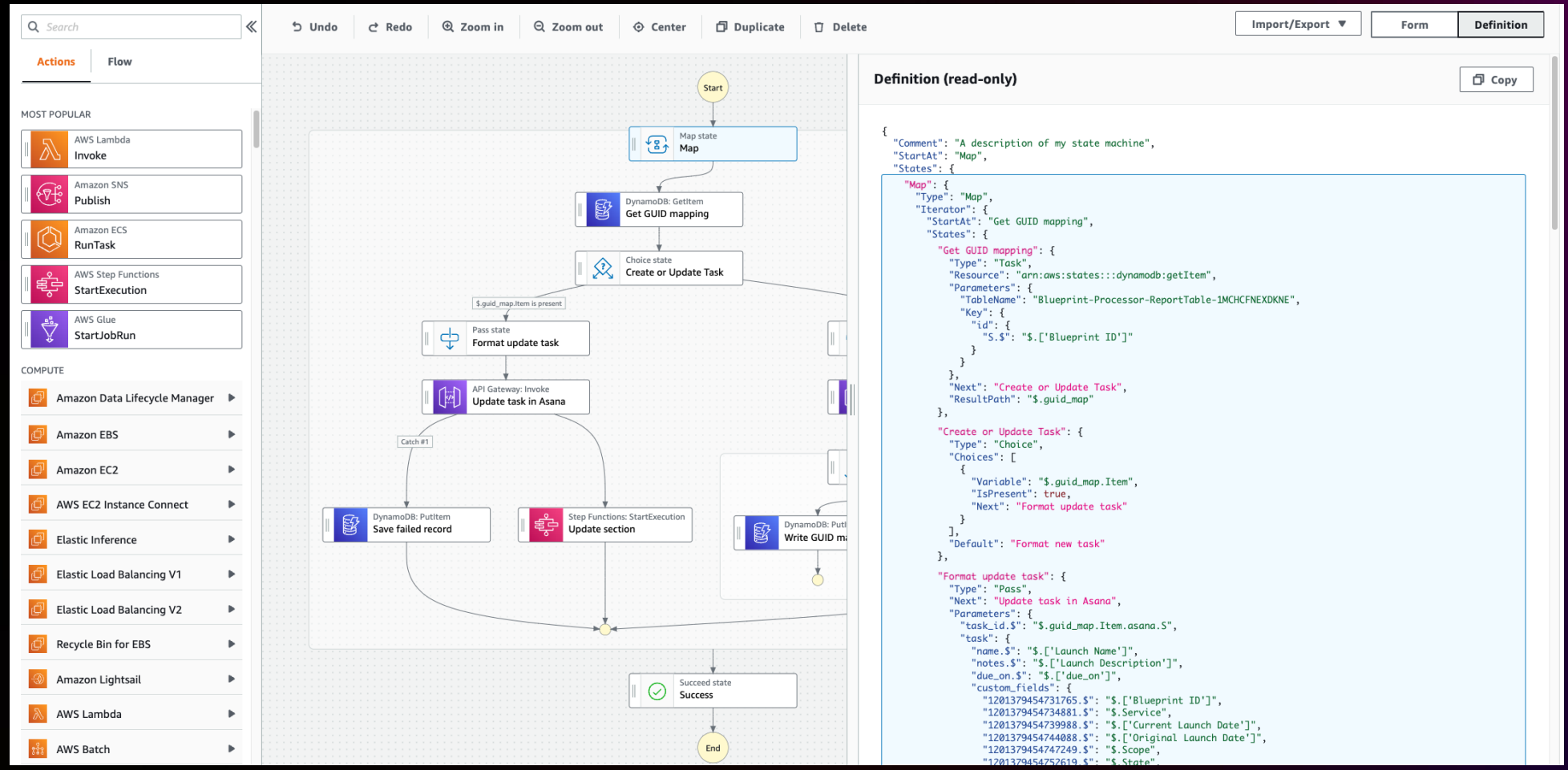
2 Use the AWS Step Functions Workflow Studio to build the state machine



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Working with Step Functions

3 Copy the definition



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Working with Step Functions

4 Paste to the ASL file in your AWS SAM application, save it, and let AWS SAM sync

statemachine.asl.json

```
{
  "Comment": "A description of my state machine",
  "StartAt": "Map",
  "States": {
    "Map": {
      "Type": "Map",
      "Iterator": {
        "StartAt": "Get GUID mapping",
        "States": {
          "Get GUID mapping": {
            "Type": "Task",
            "Resource": "arn:aws:states:::dynamodb:getItem",
            "Parameters": {
. . .
}
```

template.yaml

```
...
Orchestrator:
  Type: AWS::Serverless::StateMachine
  Properties:
    Type: EXPRESS
    DefinitionUri: ./statemachine.asl.json
    Logging:
      Destinations:
        - CloudWatchLogsLogGroup:
            LogGroupArn: !GetAtt OrchestratorLogGroup.Arn
      IncludeExecutionData: true
      Level: ALL
    Policies:
      - DynamoDBReadPolicy:
        TableName: !Ref ReportTable
      - DynamoDBWritePolicy:
        TableName: !Ref ReportTable
```

aws

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Always use AWS SAM delete for cleanup

- Removes all associated companion stacks
- Removes artifact ZIP files
- Removes Amazon ECR repository images and repositories
- Removes the stack
- Prompts before deleting anything

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

More resources

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

More resources

https://s12d.com/svs303-22



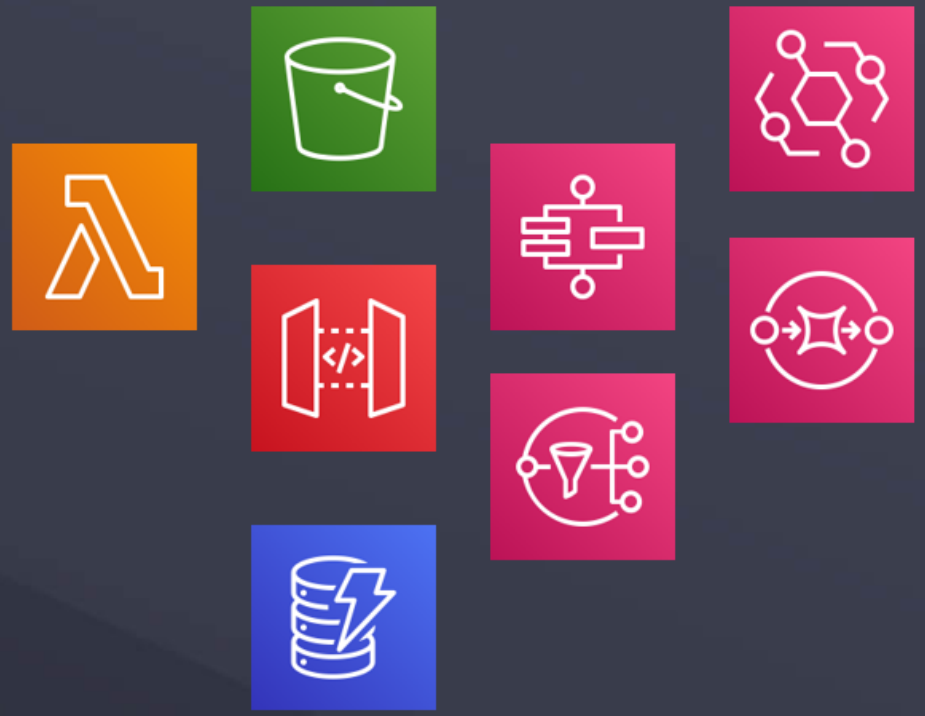
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

New Blogs Videos Learn Events Patterns About Search Search

Welcome to **Serverless** Land

This site brings together all the latest blogs, videos, and training for AWS Serverless. Learn to use and build apps that scale automatically on low-cost, fully-managed serverless architecture.

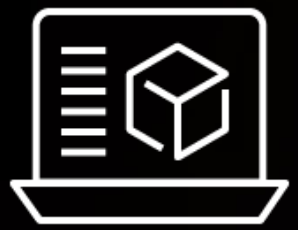
Learn More



For more serverless learning resources, visit:
https://serverlessland.com

Continue your AWS SAM learning

Learn at your own pace



Expand your serverless skills with our Learning Plan on **AWS Skill Builder**

Increase your knowledge



Use our **AWS Ramp-Up Guides** to build your serverless knowledge

Earn AWS SAM badges



Demonstrate your knowledge by achieving **digital badges**

https://s12d.com/serverless-learning

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Thank you!

Eric Johnson
Twitter: @edjgeek



Please complete the session survey in the **mobile app**

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.