Authoring and Deploying Serverless Applications

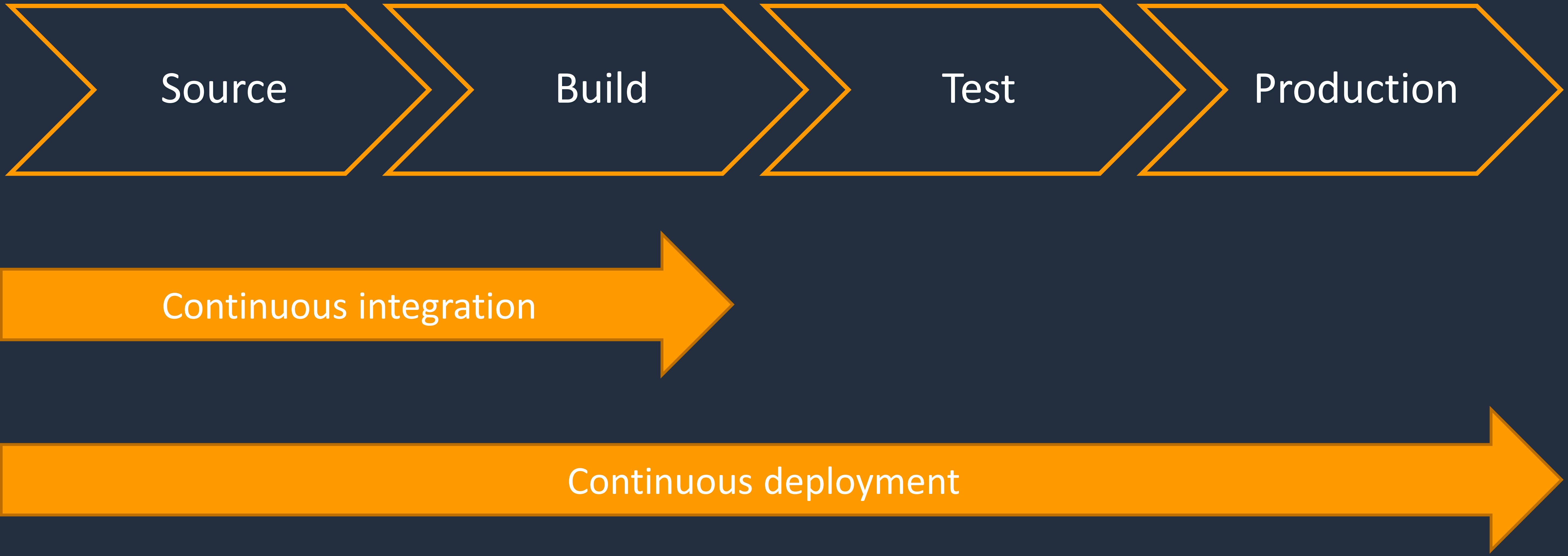
Sriram Dhandapani, Technical Account Manager

2021

You don’t want to deploy this by hand..

Photo by jesse orrico on Unsplash

Take it through a pipeline



Source

Build

Test

Production

Continuous integration

Continuous deployment

AWS Serverless Application Model (SAM)

AWS Serverless Application Model (SAM)

- CloudFormation extension optimized for serverless
- Shorthand syntax to express functions, APIs, databases, and event source mappings
- Simplifies IAM policy and Event trigger management
- Model with YAML, deploy using AWS CloudFormation
- Open source!

https://aws.amazon.com/serverless/sam/
https://github.com/awslabs/serverless-application-model

AWS SAM Template

```
AWSTemplateFormatVersion: '2010-09-09’
Transform: AWS::Serverless-2016-10-31
Resources:
 GetHtmlFunction:
  Type: AWS::Serverless::Function
  Properties:
   CodeUri: s3://sam-demo-bucket/todo_list.zip
   Handler: index.handler
   Runtime: nodejs12.x
   Policies: DynamoDBReadPolicy
   Events:
    GetToDo:
     Type: Api
     Properties:
      Path: /todo/{id}
      Method: GET
 Listable:
  Type: AWS::Serverless::Simple Table
```

SAM template transform

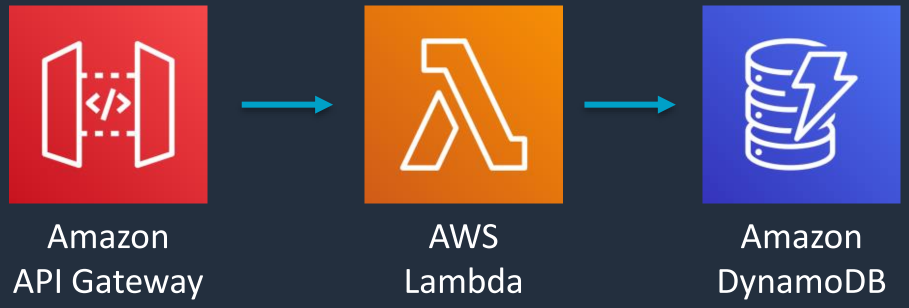
Creates:

- Lambda function
  - Runtime
  - Execution Policy
  - Code
  - Handler
- API Gateway
  - API endpoint
  - Permissions

Create DynamoDB table with sane defaults

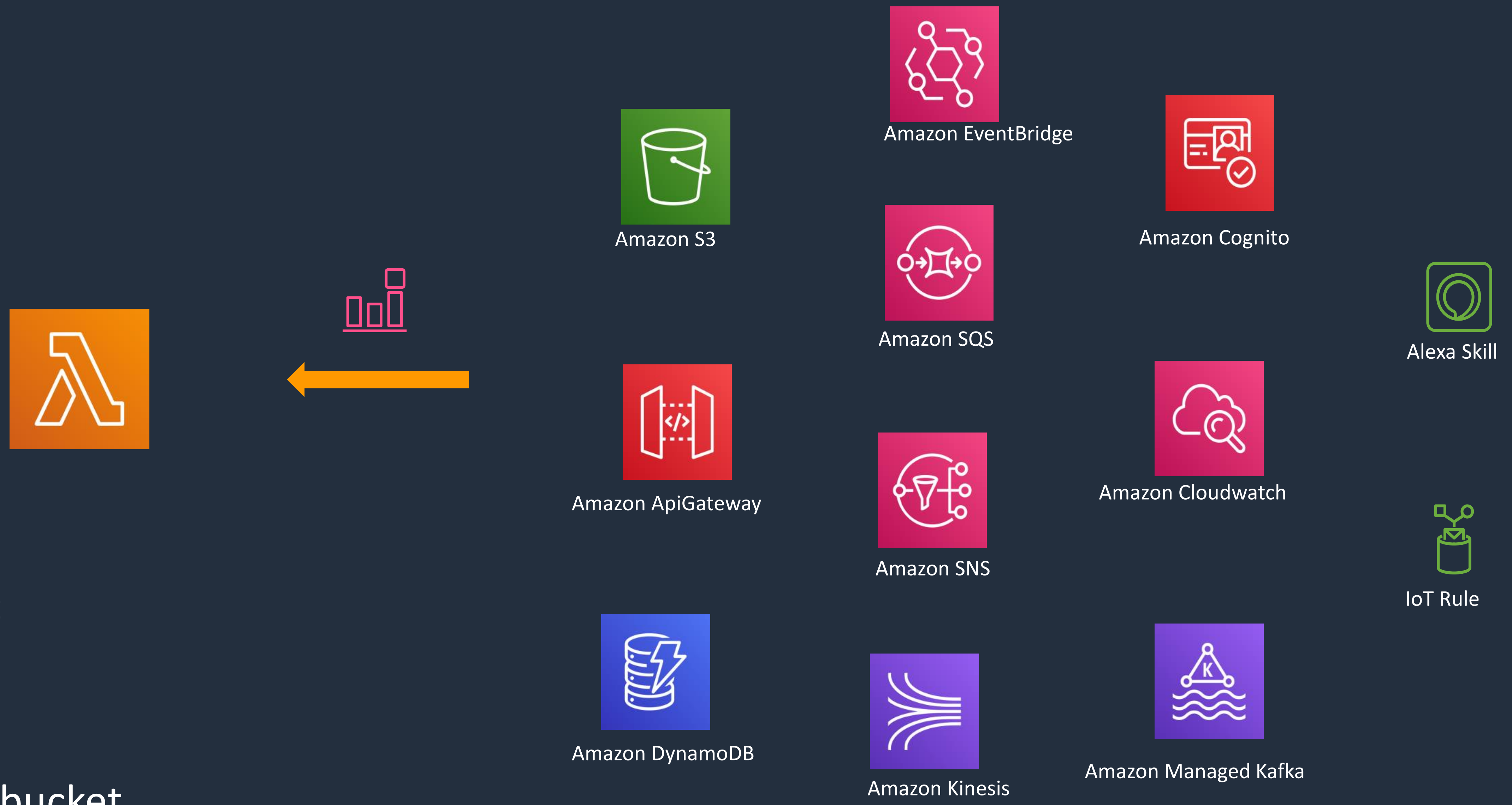
AWS SAM template

```
AWSTemplateFormatVersion: '2010-09-09’
Transform: AWS::Serverless-2016-10-31
Resources:
  GetHtmlFunction:
    Type: AWS::Serverless::Function
    Properties:
      CodeUri: s3://sam-demo-bucket/todo_list.zip
      Handler: index.handler
      Runtime: nodejs12.x
      Policies: DynamoDBReadPolicy
      Events:
        GetToDo:
          Type: Api
          Properties:
            Path: /todo/{id}
            Method: GET

  ListTable:
    Type: AWS::Serverless::SimpleTable
```



/todo/{id} - GET

Serverless Function Event source types



```
Events:
  UploadEvent:
    Type: S3
    Properties:
      Bucket: mybucket
```

AWS SAM CLI

- CLI tool for local building, validating, testing of serverless apps
- Works with Lambda functions and “proxy-style” APIs
- Response object and function logs available on your local machine
- Mimic Lambda’s execution environment with Dockers images
  - Emulates timeout, memory limits, runtimes

https://github.com/aws/aws-sam-cli

Getting Started with SAM CLI

- sam init

  Generates a preconfigured AWS SAM template and example application code in the language that you choose

- sam package

  Bundles your application code and dependencies into a "deployment package"

- sam build

  prepares it for subsequent steps like deploy or local testing

- sam deploy

  Deploys your serverless application to the AWS Cloud

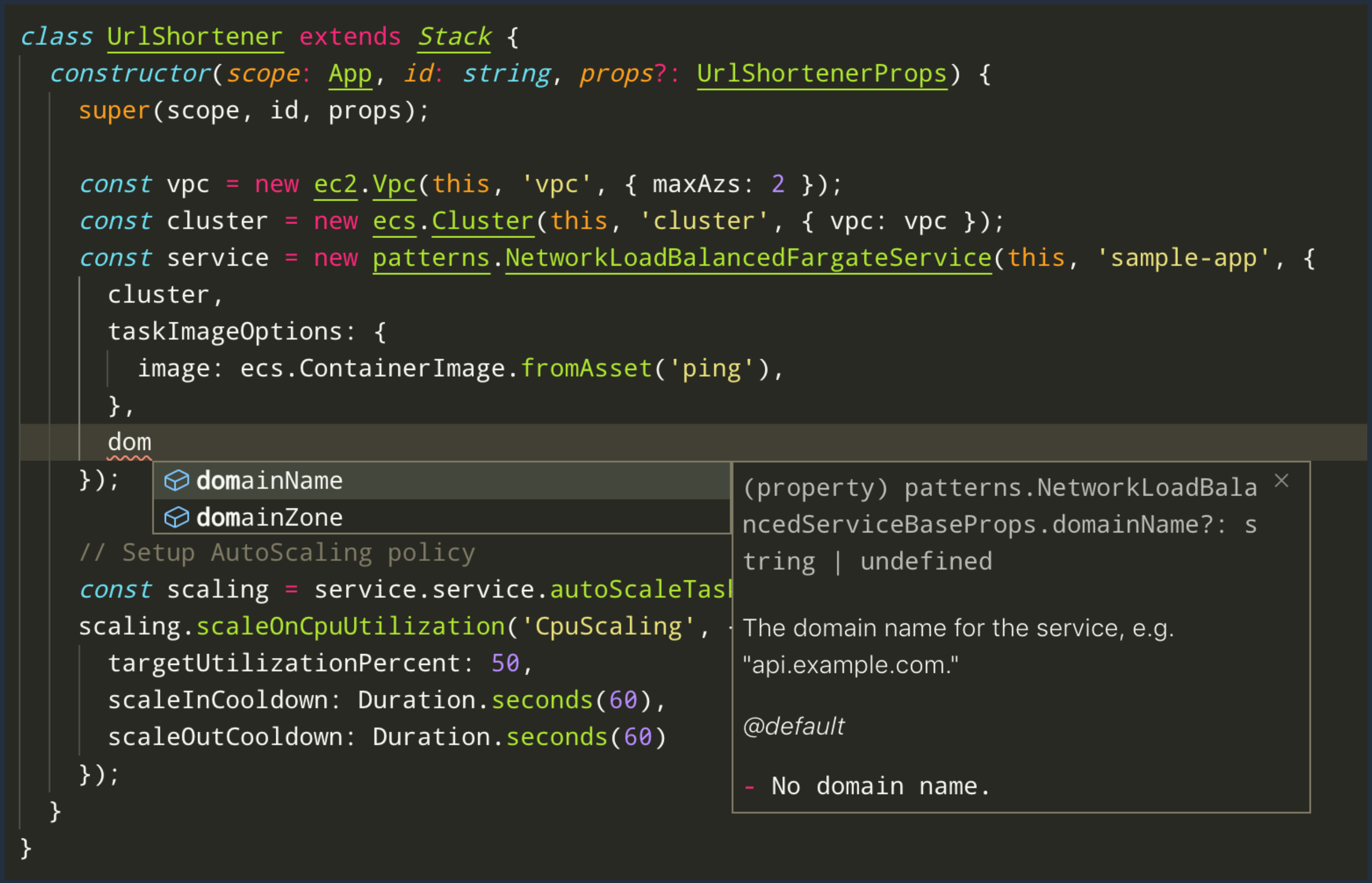
- sam local

  Test your application code locally

AWS Cloud Development Kit (AWS CDK)

AWS Cloud Development Kit (AWS CDK)

Multi-language framework for modeling infrastructure as reusable components



```
class UrlShortener extends Stack {
  constructor(scope: App, id: string, props?: UrlShortenerProps) {
    super(scope, id, props);

    const vpc = new ec2.Vpc(this, 'vpc', { maxAzs: 2 });
    const cluster = new ecs.Cluster(this, 'cluster', { vpc: vpc });
    const service = new patterns.NetworkLoadBalancedFargateService(this, 'sample-app', {
      cluster,
      taskImageOptions: {
        image: ecs.ContainerImage.fromAsset('ping'),
      },
      dom
    });

    // Setup AutoScaling policy
    const scaling = service.service.autoScaleTas
    scaling.scaleOnCpuUtilization('CpuScaling',
      targetUtilizationPercent: 50,
      scaleInCooldown: Duration.seconds(60),
      scaleOutCooldown: Duration.seconds(60)
    });
  }
}
```

```
(property) patterns.NetworkLoadBalancedServiceBaseProps.domainName?: string | undefined
```

The domain name for the service, e.g. "api.example.com."

@default

- No domain name.

CDK Benefits

Familiar

Your language
Just classes and methods

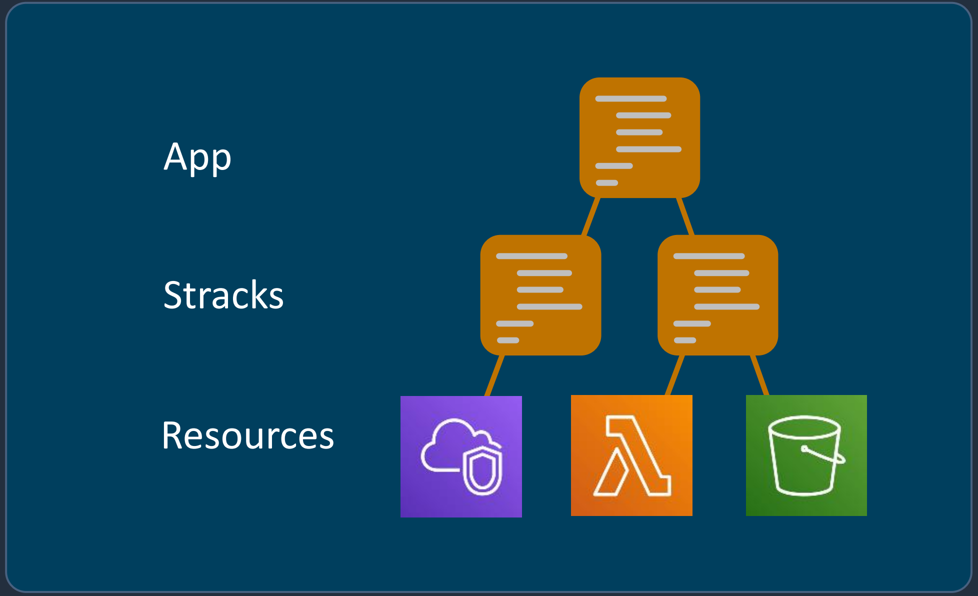
Tool Support

Auto complete
Inline documentation

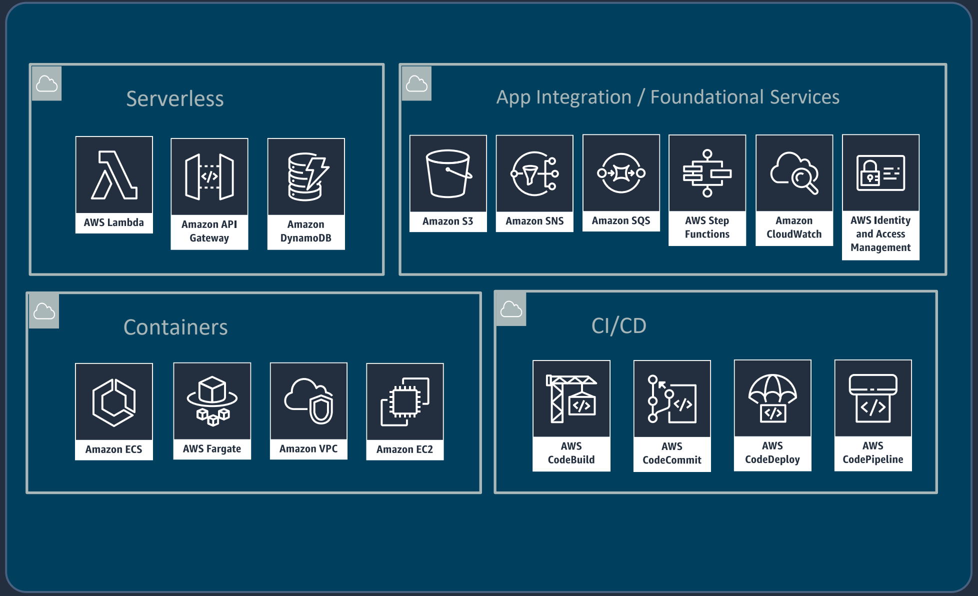
Abstraction

Sane defaults
Reusable classes

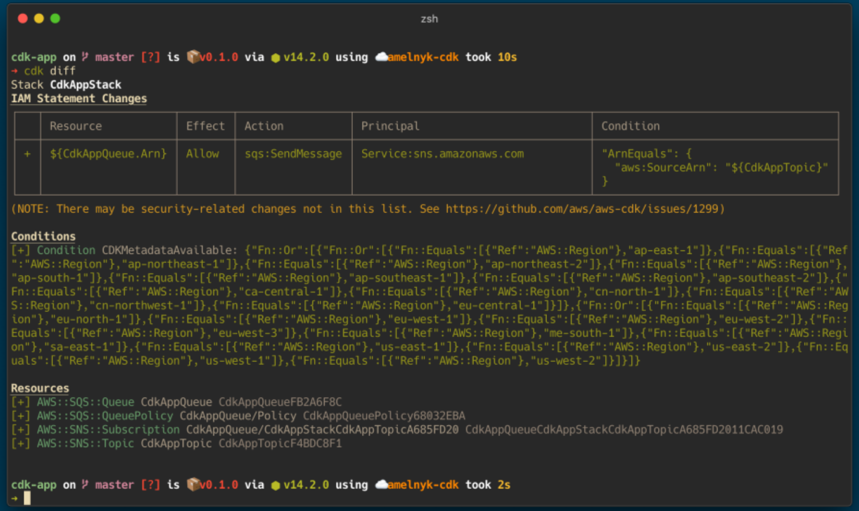
Main Components



Core Framework



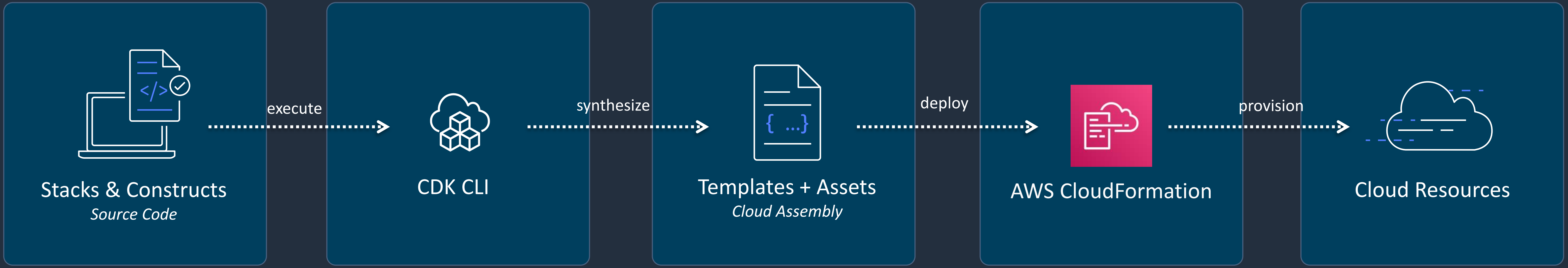
AWS Construct Library



```
cdk-app on  master [?] is v0.1.0 via v14.2.0 using amelnyk-cdk took 10s
→ cdk diff
Stack CdkAppStack
IAM Statement Changes
┌───┬─────────────────────┬────────┬─────────────────┬───────────────────────────────┬─────────────────────────────────────┐
│   │ Resource            │ Effect │ Action          │ Principal                     │ Condition                           │
├───┼─────────────────────┼────────┼─────────────────┼───────────────────────────────┼─────────────────────────────────────┤
│ + │ ${CdkAppQueue.Arn}  │ Allow  │ sqs:SendMessage │ Service:sns.amazonaws.com     │ "ArnEquals": {                      │
│   │                     │        │                 │                               │   "aws:SourceArn": "${CdkAppTopic}" │
│   │                     │        │                 │                               │ }                                   │
└───┴─────────────────────┴────────┴─────────────────┴───────────────────────────────┴─────────────────────────────────────┘
(NOTE: There may be security-related changes not in this list. See https://github.com/aws/aws-cdk/issues/1299)

Conditions
[+] Condition CDKMetadataAvailable: {"Fn::Or":[{"Fn::Or":[{"Fn::Equals":[{"Ref":"AWS::Region"},"ap-east-1"]},{"Fn::Equals":[{"Ref
":"AWS::Region"},"ap-northeast-1"]},{"Fn::Equals":[{"Ref":"AWS::Region"},"ap-northeast-2"]},{"Fn::Equals":[{"Ref":"AWS::Region"},
"ap-south-1"]},{"Fn::Equals":[{"Ref":"AWS::Region"},"ap-southeast-1"]},{"Fn::Equals":[{"Ref":"AWS::Region"},"ap-southeast-2"]},{"
Fn::Equals":[{"Ref":"AWS::Region"},"ca-central-1"]},{"Fn::Equals":[{"Ref":"AWS::Region"},"cn-north-1"]},{"Fn::Equals":[{"Ref":"AW
S::Region"},"cn-northwest-1"]},{"Fn::Equals":[{"Ref":"AWS::Region"},"eu-central-1"]}]},{"Fn::Or":[{"Fn::Equals":[{"Ref":"AWS::Reg
ion"},"eu-north-1"]},{"Fn::Equals":[{"Ref":"AWS::Region"},"eu-west-1"]},{"Fn::Equals":[{"Ref":"AWS::Region"},"eu-west-2"]},{"Fn::
Equals":[{"Ref":"AWS::Region"},"eu-west-3"]},{"Fn::Equals":[{"Ref":"AWS::Region"},"me-south-1"]},{"Fn::Equals":[{"Ref":"AWS::Regi
on"},"sa-east-1"]},{"Fn::Equals":[{"Ref":"AWS::Region"},"us-east-1"]},{"Fn::Equals":[{"Ref":"AWS::Region"},"us-east-2"]},{"Fn::Eq
uals":[{"Ref":"AWS::Region"},"us-west-1"]},{"Fn::Equals":[{"Ref":"AWS::Region"},"us-west-2"]}]}]}

Resources
[+] AWS::SQS::Queue CdkAppQueue CdkAppQueueFB2A6F8C
[+] AWS::SQS::QueuePolicy CdkAppQueue/Policy CdkAppQueuePolicy68032EBA
[+] AWS::SNS::Subscription CdkAppQueue/CdkAppStackCdkAppTopicA685FD20 CdkAppQueueCdkAppStackCdkAppTopicA685FD2011CAC019
[+] AWS::SNS::Topic CdkAppTopic CdkAppTopicF4BDC8F1


cdk-app on  master [?] is v0.1.0 via v14.2.0 using amelnyk-cdk took 2s
→
```

AWS CDK CLI

Development Workflow

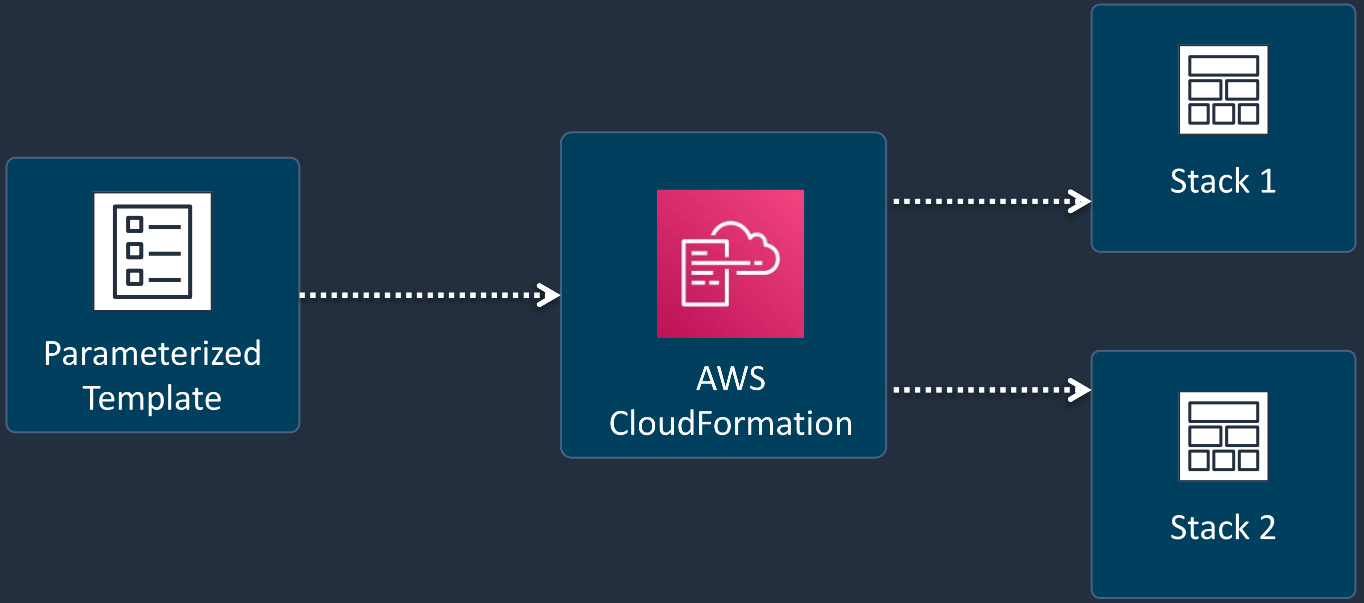


```
cdk init            // create new project
npm run build       // build project
cdk synth           // create templates and assets
cdk diff            // check what will change
cdk deploy          // push changes to the cloud
```

Paradigm Shift

CloudFormation

Parameters and
intrinsic functions



Parameterized Template

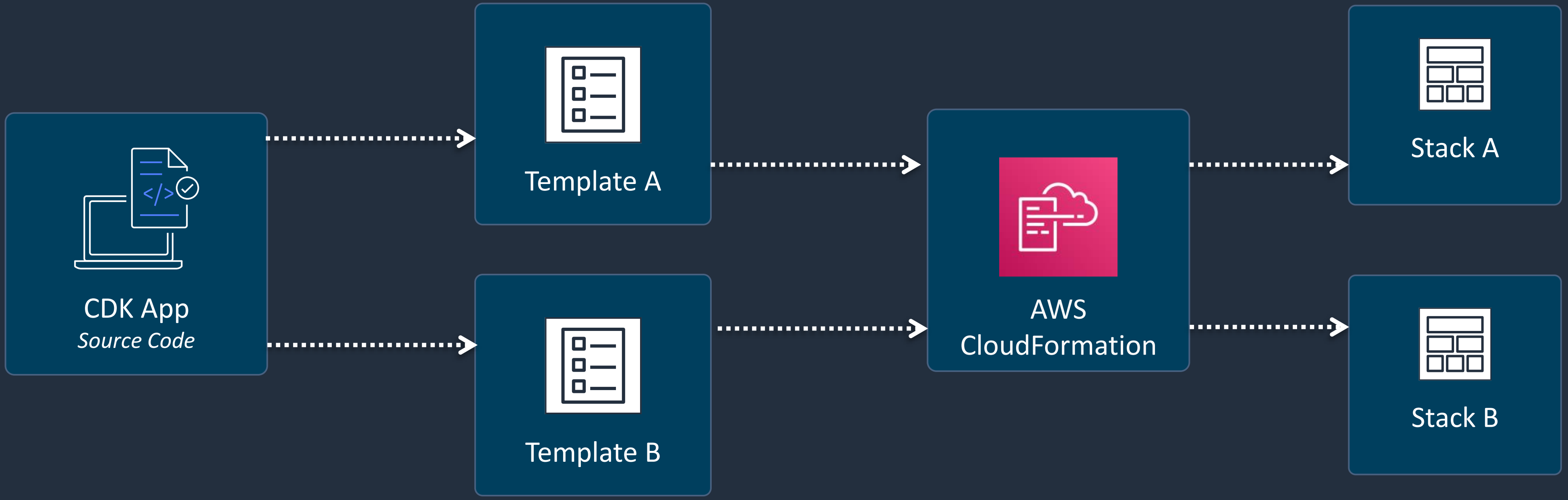
AWS CloudFormation

Stack 1

Stack 2

CDK

Typed OO language:
loops, conditions,
inheritence, etc



CDK App
Source Code

Template A

Template B

AWS CloudFormation

Stack A

Stack B

Other frameworks for building serverless applications

Terraform

- Infrastructure as code
- Access to broad array of resources

serverless

- Local testing, deployment management
- Rich plug-in ecosystem
- Open source and professional offerings

CHALICE

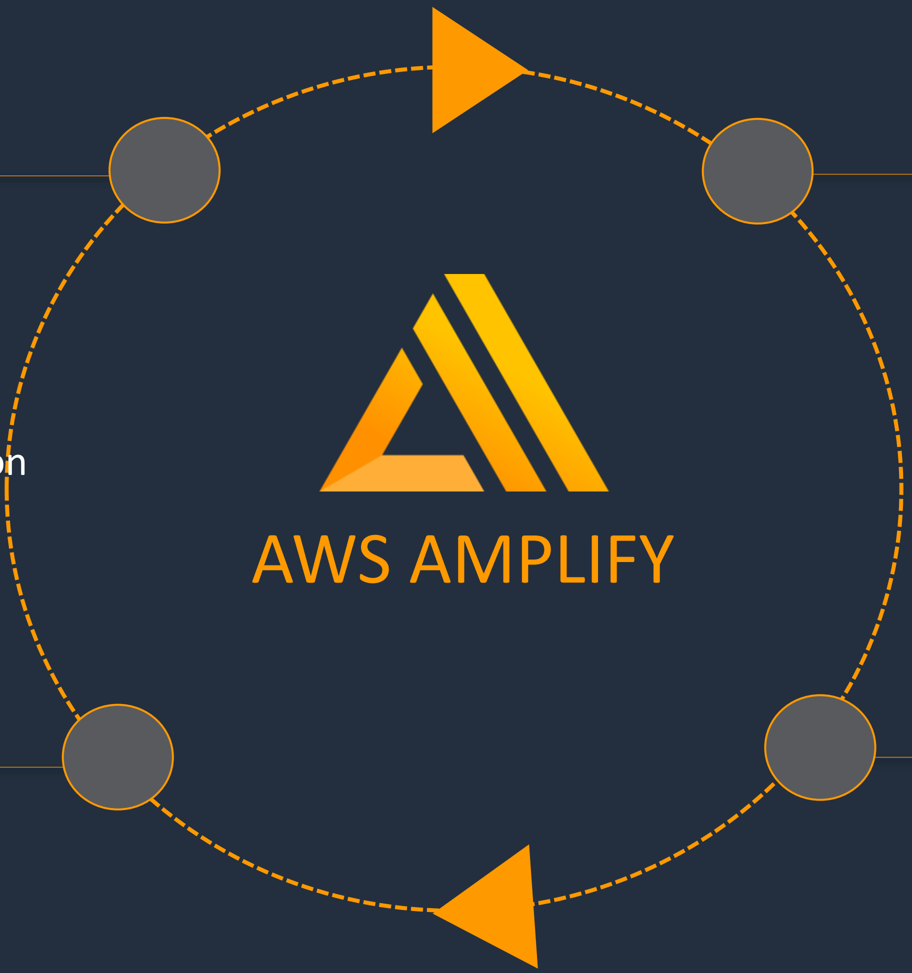
- Python only
- Decorator API for integrations
- CLI to manage deployments

Architect

- Focus on business logic
- Simple, plain text definition
- Extensible with CloudFormation

AWS Amplify

AWS Amplify
Designed for the Development Lifecycle



ENGAGE/MEASURE

- Multi-channel (push/sms/email/voice)
- Behavior-based and personalized audience segments
- Engagement performance measurements
- Real-time customer data for immediate optimization

DEVELOP

- Open source libraries
- iOS/Android Native
- Javascript/React/Vue/Angular
- UI components
- Escape hatches

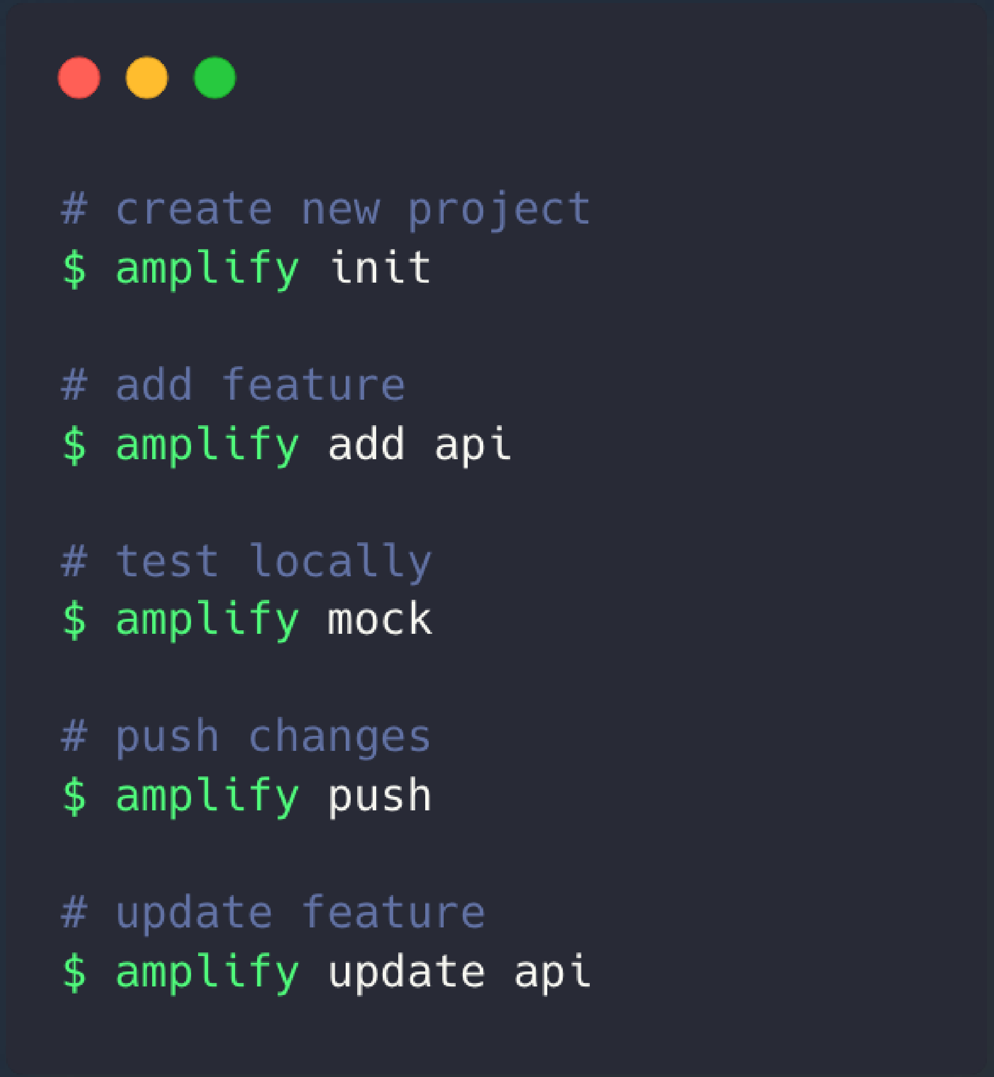
DEPLOY/HOST

- AWS infrastructure
- Ease of deployment with CLI or hosting
- CI/CD capability
- Fully managed global hosting

TEST

- Device Farm
- Test on real devices
- Test on real browsers
- Integrate testing with CI/CD

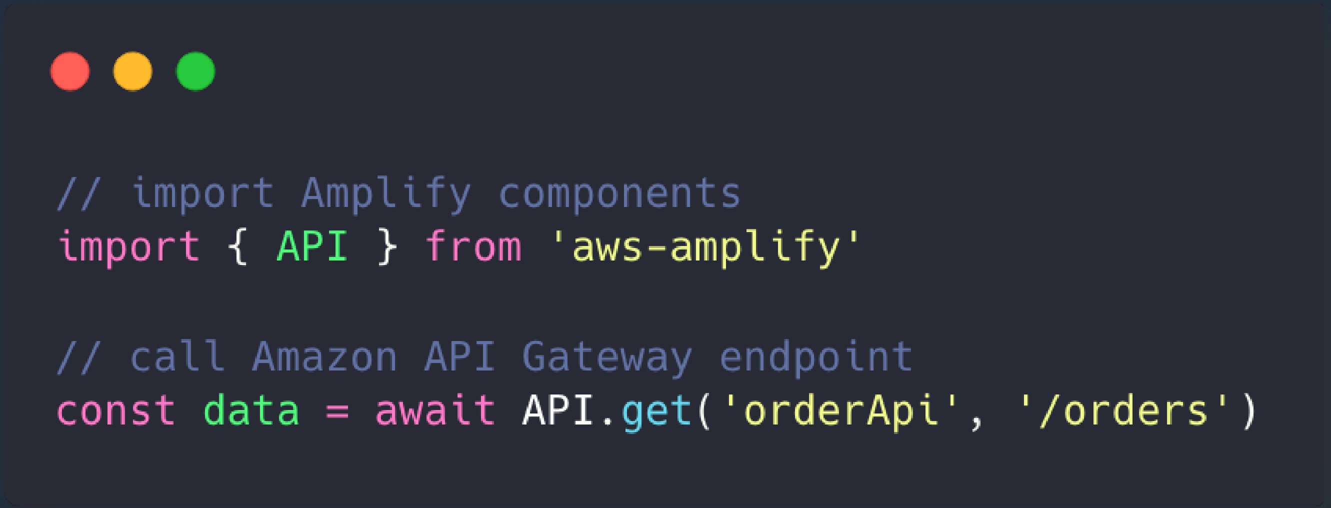
AWS Amplify CLI



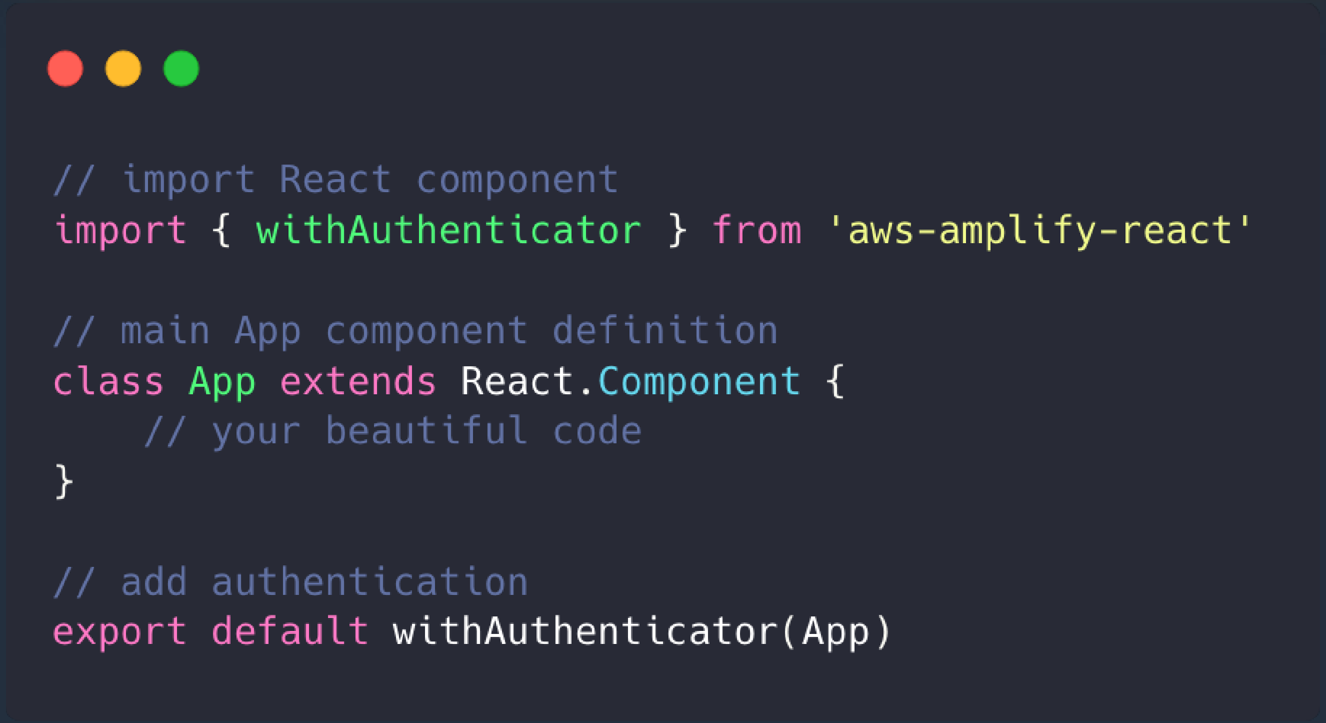
```
# create new project
$ amplify init

# add feature
$ amplify add api

# test locally
$ amplify mock

# push changes
$ amplify push

# update feature
$ amplify update api
```

- Create, update, and delete cloud services
- Manage multiple environments
- GraphQL Transform
- GraphQL Codegen

AWS Amplify Client



```
// import Amplify components
import { API } from 'aws-amplify'

// call Amazon API Gateway endpoint
const data = await API.get('orderApi', '/orders')
```



```
// import React component
import { withAuthenticator } from 'aws-amplify-react'

// main App component definition
class App extends React.Component {
    // your beautiful code
}

// add authentication
export default withAuthenticator(App)
```

Android

iOS

React native

Ionic

Javascript

React

Angular

Vue

Amplify Libraries: Use Case Features

Authentication

Authentication APIs with prebuilt UI components for your app

Data Store

On-device persistent storage that automatically synchronizes data between you apps and the cloud.

API

HTTP requests using REST and GraphQL with support for real-time data

Analytics

Track user sessions, custom user attributes and in-app metrics

PubSub

Connect your app to message-oriented middleware on the cloud

Predictions

Add MI/ML capabilities to your app powered by cloud services

Interactions

Conversational bots powered by deep learning technologies

Notifications

Push notifications with campaign analytics and targeting

Storage

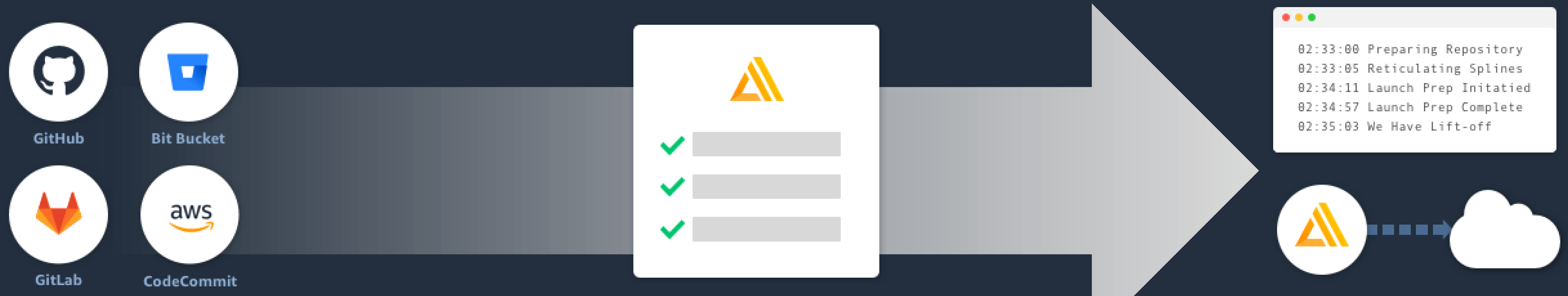
Manage user content securely in public, protected, and private storage.

XR

Work with augmented reality and virtual reality content in your apps

Amplify Console

Build, deploy, and host cloud-powered modern web apps



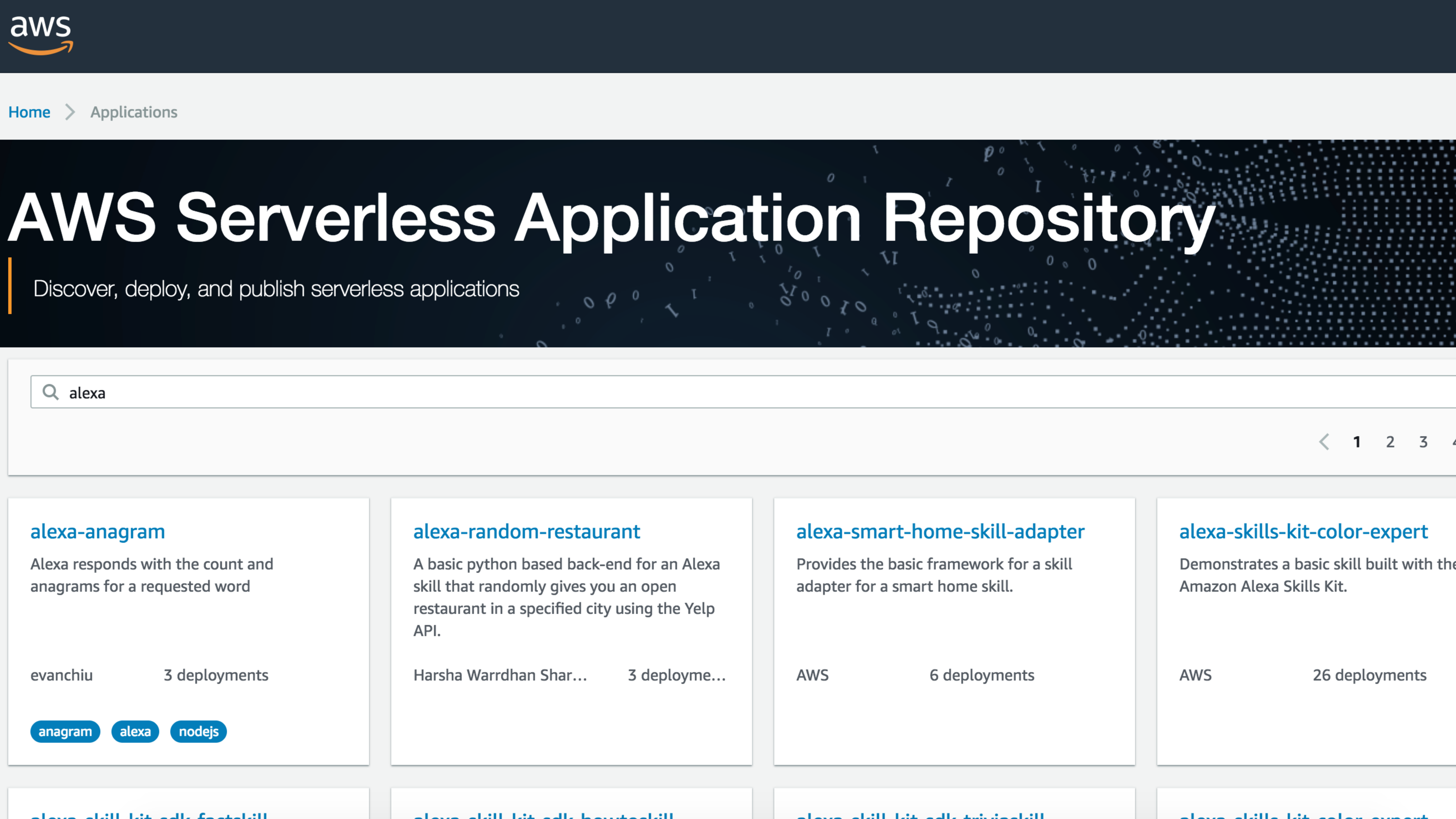
```
02:33:00 Preparing Repository
02:33:05 Reticulating Splines
02:34:11 Launch Prep Initatied
02:34:57 Launch Prep Complete
02:35:03 We Have Lift-off
```

Connect your repository

Configure build settings

Deploy your app

AWS Serverless Application Repository (AWS SAR)



Home > Applications

AWS Serverless Application Repository

Discover, deploy, and publish serverless applications

alexa

< 1 2 3

alexa-anagram

Alexa responds with the count and anagrams for a requested word

evanchiu 3 deployments

anagram alexa nodejs

alexa-random-restaurant

A basic python based back-end for an Alexa skill that randomly gives you an open restaurant in a specified city using the Yelp API.

Harsha Warrdhan Shar... 3 deployme...

alexa-smart-home-skill-adapter

Provides the basic framework for a skill adapter for a smart home skill.

AWS 6 deployments

alexa-skills-kit-color-expert

Demonstrates a basic skill built with the Amazon Alexa Skills Kit.

AWS 26 deployments

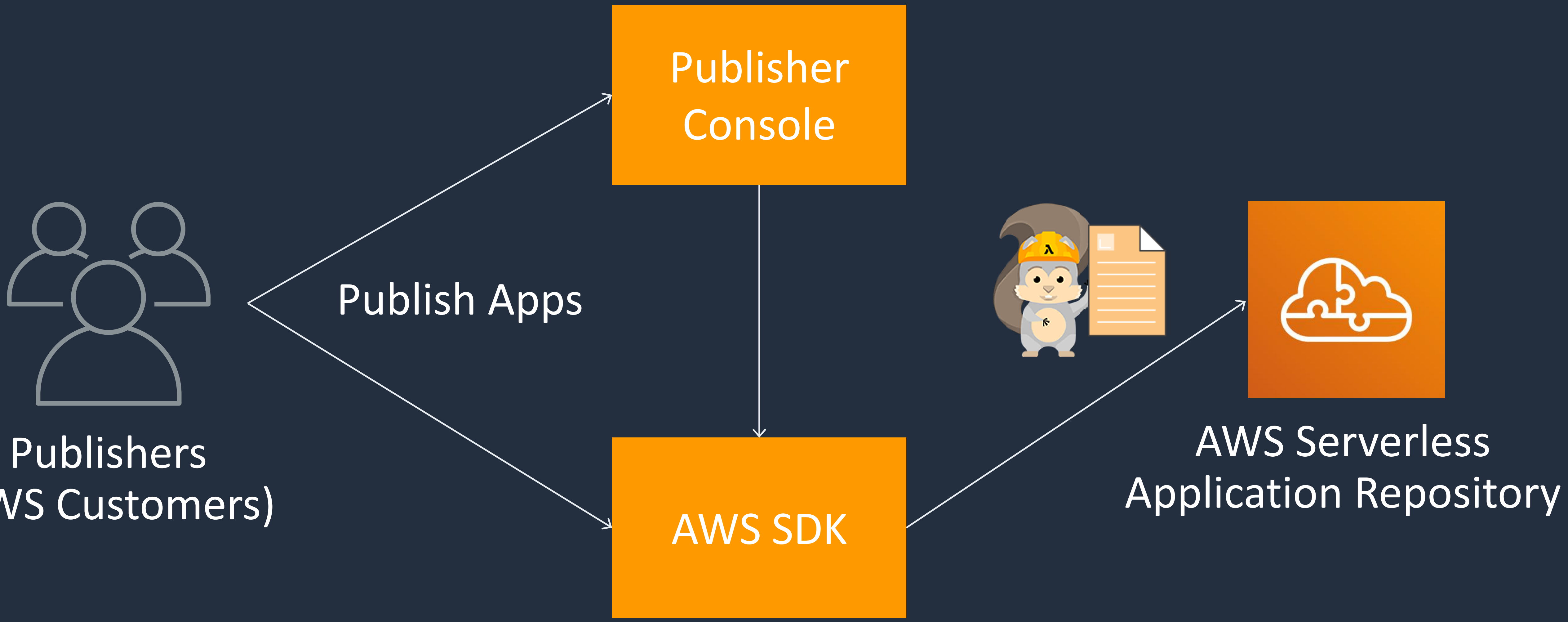
alexa-skill-kit-sdk-factskill

alexa-skill-kit-sdk-howtoskill

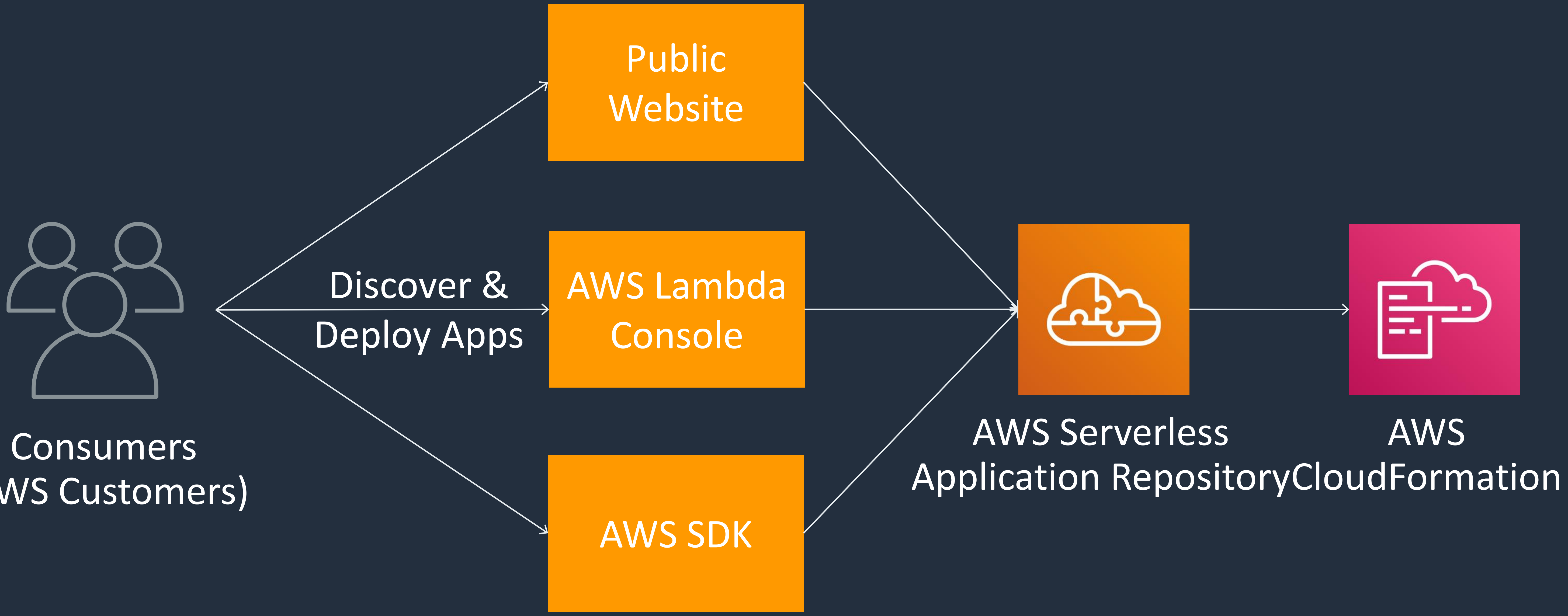
alexa-skill-kit-sdk-triviaskill

alexa-skills-kit-color-expert

AWS SAR - Publishing applications


Publisher Console
Publish Apps
AWS SDK
Publishers (AWS Customers)
AWS Serverless Application Repository

AWS SAR - Consuming applications



Consumers (AWS Customers)

Discover & Deploy Apps

Public Website

AWS Lambda Console

AWS SDK

AWS Serverless Application Repository

AWS CloudFormation

Thank you!

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