



Infrastructure **is** Code

- with the AWS Cloud Development Kit

Steve Roberts
Developer Advocate, Amazon Web Services
January 2021



This session...

- What is Infrastructure as Code (IaC)?
- Introduction to the AWS Cloud Development Kit (CDK)
- Examples

Declaring Infrastructure – Previous Approaches

Use console, e.g. AWS Management Console

- Not easily repeatable and error prone
- “Clicks don’t scale”
- But...great for learning your way around

Use template approach

- I don’t work in the templated language (json/yaml/whatever) all day, every day – do you?
- Limited expressiveness
- Hard to share resource definitions inside template – copy/paste
- But...reasonably straightforward

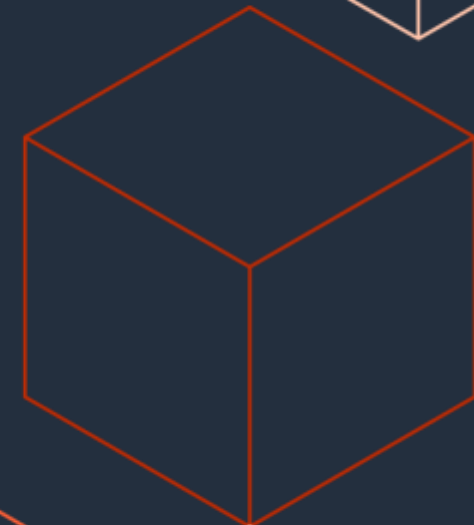
Code against service APIs

- Down at the ‘metal’, need to understand API relationships
- Manually handle ‘wait periods’ on resources to stabilize
- But...hey, I’m a developer so....cool 😊

Template example (AWS CloudFormation)

```
ElasticBeanstalkSample.template
1 {
2   "AWSTemplateFormatVersion" : "2010-09-09",
3
4   "Description" : "AWS CloudFormation Sample Template ElasticBeanstalkSample: Configure and
5
6   "Parameters" : {
7     "KeyName" : {
8       "Description" : "Name of an existing EC2 KeyPair to enable SSH access to the AWS Elast
9       "Type" : "String"
10    }
11  },
12
13  "Resources" : {
14    "sampleApplication" : {
15      "Type" : "AWS::ElasticBeanstalk::Application",
16      "Properties" : {
17        "Description" : "AWS Elastic Beanstalk Sample Application",
18        "ApplicationVersions" : [{
19          "VersionLabel" : "Initial Version",
20          "Description" : "Version 1.0",
21          "SourceBundle" : {
22            "S3Bucket" : { "Fn::Join" : ["-", ["elasticbeanstalk-samples", { "Ref" : "AWS::R
23            "S3Key" : "elasticbeanstalk-sampleapp.war"
24          }
25        }],
26        "ConfigurationTemplates" : [{
27          "TemplateName" : "DefaultConfiguration",
28          "Description" : "Default Configuration Version 1.0 - with SSH access",
29          "SolutionStackName" : "64bit Amazon Linux running Tomcat 7",
```

Introduction to the *AWS* CDK



AWS Cloud Development Kit (CDK)

A multi-language software development framework for modeling cloud infrastructure as reusable components

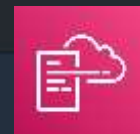


```
using Amazon.CDK;
using Amazon.CDK.AWS.EC2;
using Amazon.CDK.AWS.ECS;
using Amazon.CDK.AWS.ECS.Patterns;

namespace EcsFargateSample
{
    public class EcsFargateSampleStack : Stack
    {
        public EcsFargateSampleStack(Construct parent, string id, IStackProps props) : base(parent, id, props)
        {
            var vpc = new Vpc(this, "myVpc", new VpcProps
            {
                MaxAzs = 2
            });

            var myCluster = new Cluster(this, "myCluster", new ClusterProps
            {
                Vpc = vpc
            });

            // Define an ECS Fargate Service, defined in the ECS.Patterns library,
            // that runs two instances of a Docker image
            new ApplicationLoadBalancedFargateService(this, "myService", new ApplicationLoadBalancedFargateServiceProps
            {
                Cluster = myCluster,
                DesiredCount = 2,
                TaskImageOptions = new ApplicationLoadBalancedTaskImageOptions
                {
                    Image = ContainerImage.FromRegistry("amazon/amazon-ecs-sample")
                }
            });
        }
    }
}
```



AWS CloudFormation



Hello (CDK) world!

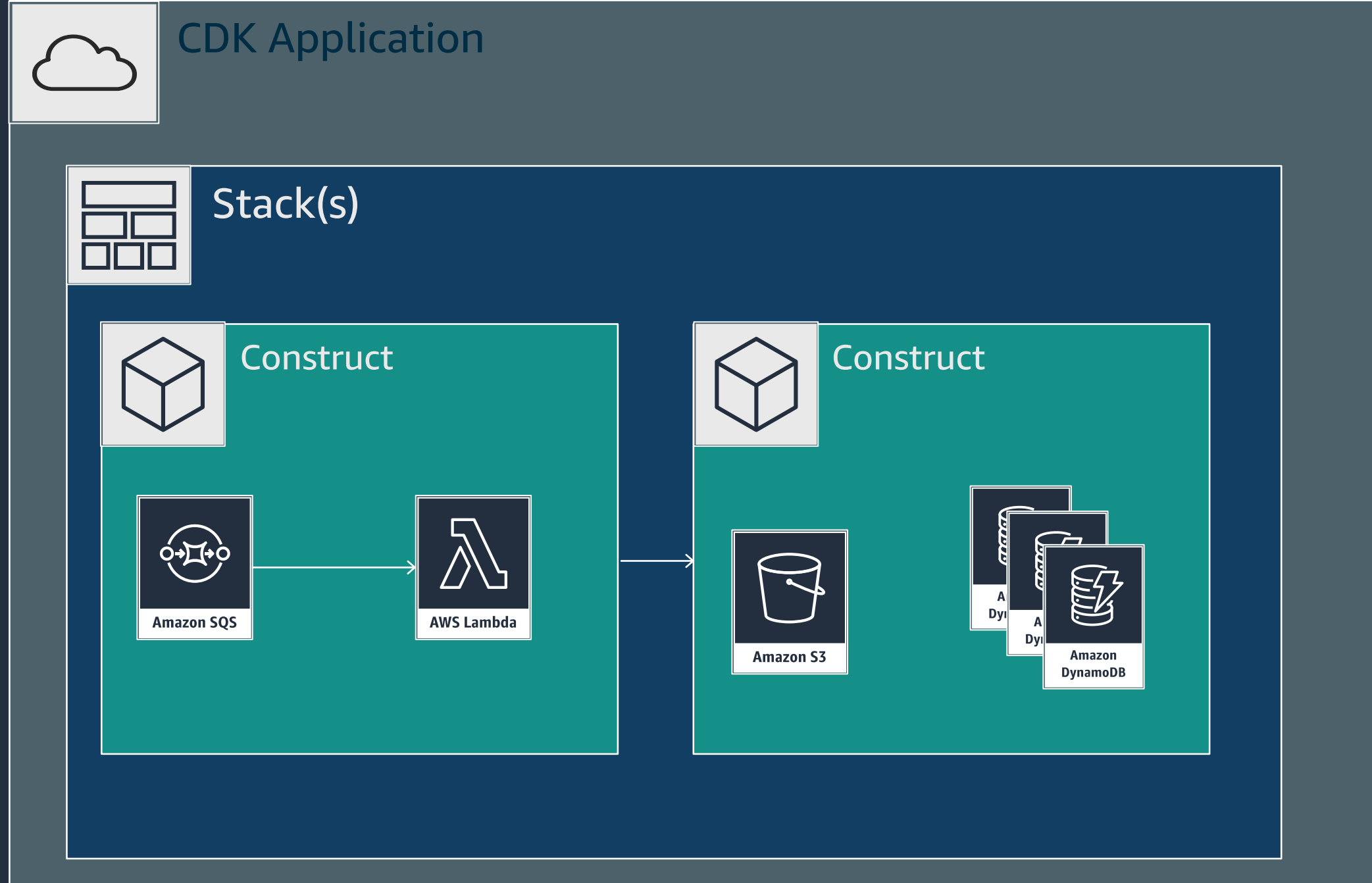
```
using Amazon.CDK;
using Amazon.CDK.AWS.SQS;
using Amazon.CDK.AWS.SNS;
using Amazon.CDK.AWS.SNS.Subscriptions;

namespace SnsTopicSample
{
    public class SnsTopicSampleStack : Stack
    {
        public SnsTopicSampleStack(Construct parent, string id, IStackProps props) : base(parent, id, props)
        {
            var queue = new Queue(this, "myFirstQueue", new QueueProps
            {
                // Duration is in Amazon.CDK
                VisibilityTimeout = Duration.Seconds(300)
            });

            var topic = new Topic(this, "myFirstTopic");

            topic.AddSubscription(new SqsSubscription(queue));
        }
    }
}
```

What makes up a CDK application?



AWS Construct Library Levels

Level 1 (“L1”): CloudFormation resources (known as CFN resources)

- Generated from CloudFormation resource specs
- Follow name pattern *CfnResource*, e.g. *CfnBucket*, *CfnInstance*, etc
- Must configure all resource properties yourself

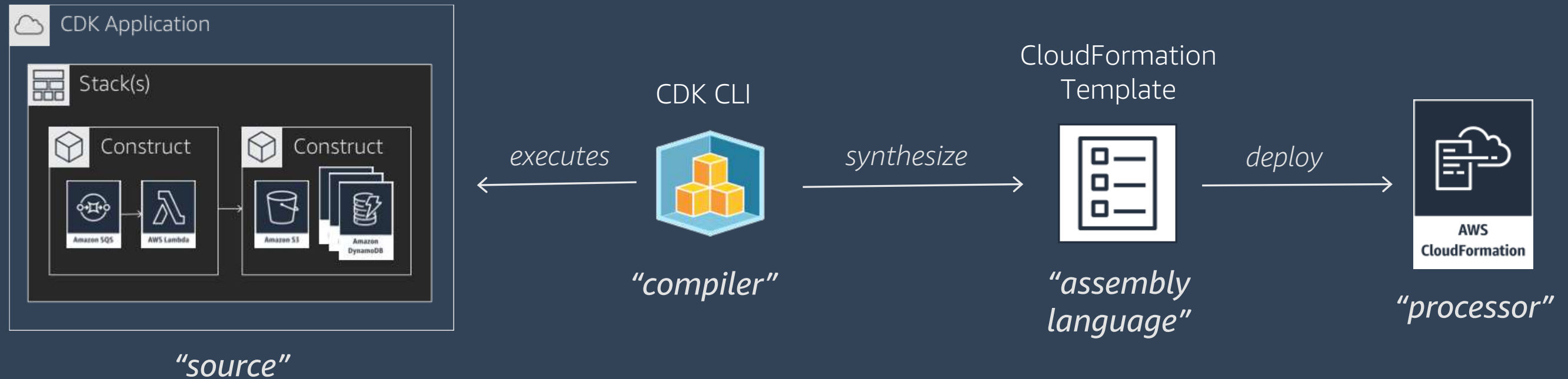
L2: Higher-level, intent-based resources

- Adds common defaults for properties, and useful methods
- e.g. *AddLifecycleRule(new Rule(...))* on an S3 bucket instance

L3: “Pattern” based resources

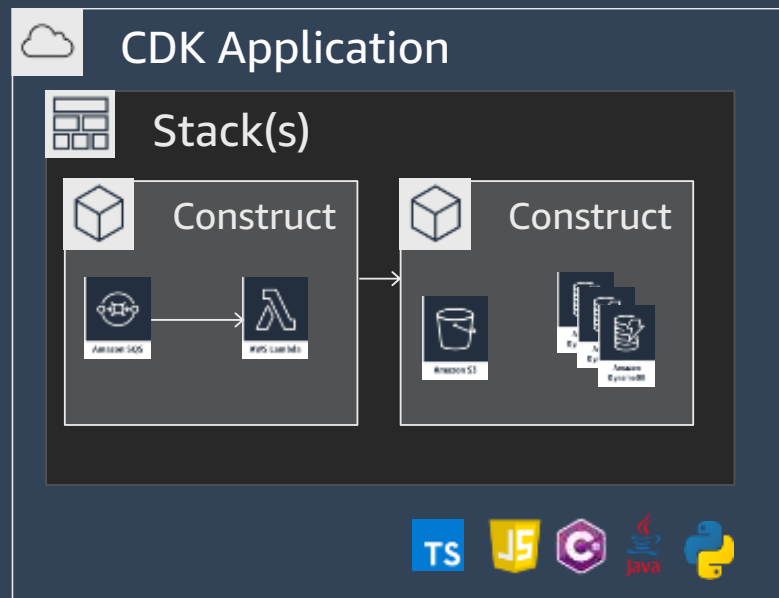
- Uses *composition* to define constructs, usually involve multiple coordinating resources
- E.g. *ApplicationLoadBalancedFargateService*, *LambdaRestApi*

The big picture – app to provisioned infrastructure

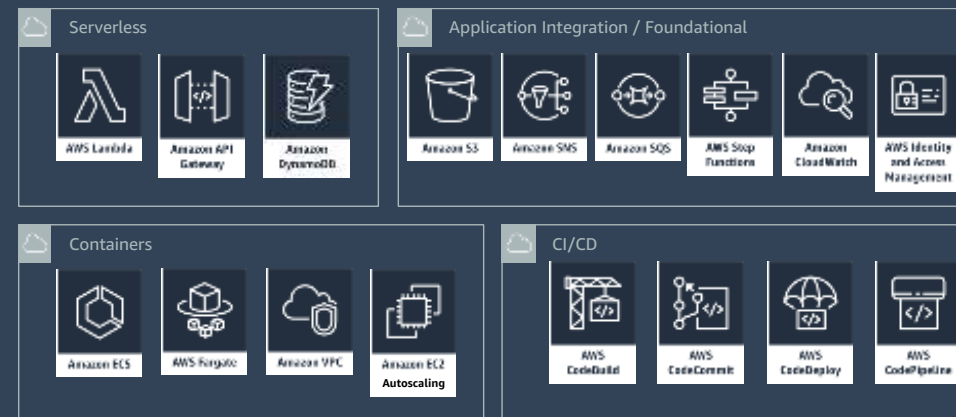


Main components of the CDK

Core Framework



AWS Construct Library



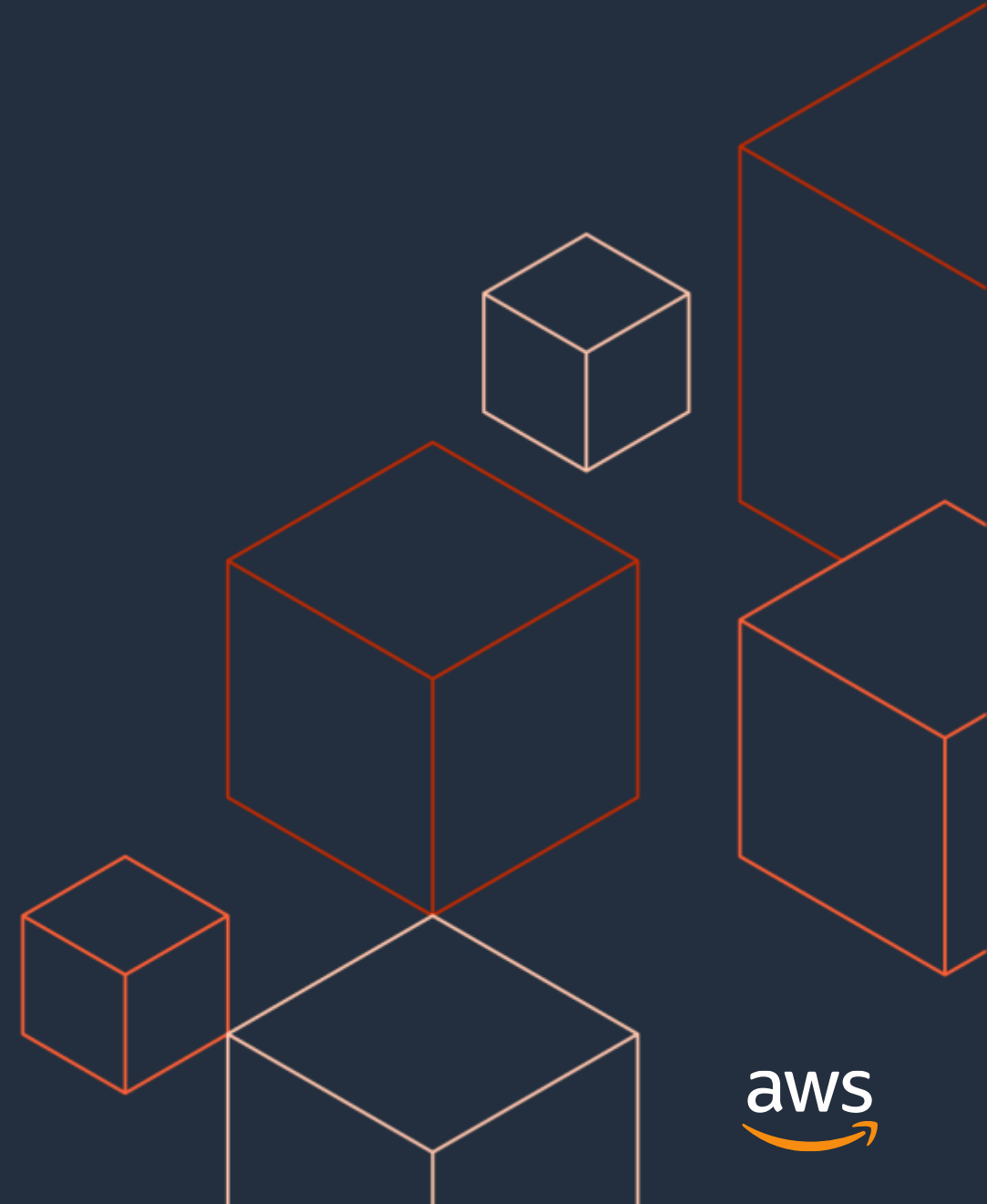
CDK CLI

```
> cdk init --language typescript sample-app
Applying project template sample-app for 'typescript'...
Initializing a new git repository...
Executing npm install...
npm notice created a lockfile as package-lock.json. You should commit this file.
npm WARN hello-cdk@0.1.0 No repository field.
npm WARN hello-cdk@0.1.0 No license field.

# Useful commands

* 'npm run build'    compile typescript to js
* 'npm run watch'    watch for changes and compile
* 'cdk deploy'       deploy this stack to your default AWS account/region
* 'cdk diff'         compare deployed stack with current state
* 'cdk synth'        emits the synthesized CloudFormation template
```

Examples



Session recap

- IaC with declarative templates, what works, what does not
- CDK concepts and tools
- CDK bindings for .NET developers
 - Collection of .NET Standard 2.0 NuGet packages
 - *Amazon.CDK.** on NuGet
- Examples
 - Simple web site hosting on Amazon S3
 - Windows IIS Web Server fleet & simple deployment setup

Useful resources

Session Samples

<https://github.com/steveataws/talk-samples> (./cdk folder)

CDK Developer Guide

<https://docs.aws.amazon.com/cdk/latest/guide/home.html>

CDK API Reference

<https://docs.aws.amazon.com/cdk/api/latest/docs/aws-construct-library.html>

CDK Repository

<https://github.com/aws/aws-cdk>

CDK Samples

<https://github.com/aws-samples/aws-cdk-examples>



Thanks!

Steve Roberts

@bellevuesteve

www.linkedin.com/in/steven-j-roberts

Follow .NET on AWS: @dotnetonAWS

