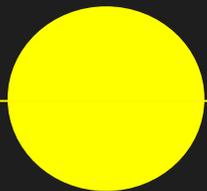


# Os princípios de DevOps, Infraestrutura como código e o AWS Cloud Development Kit (CDK)



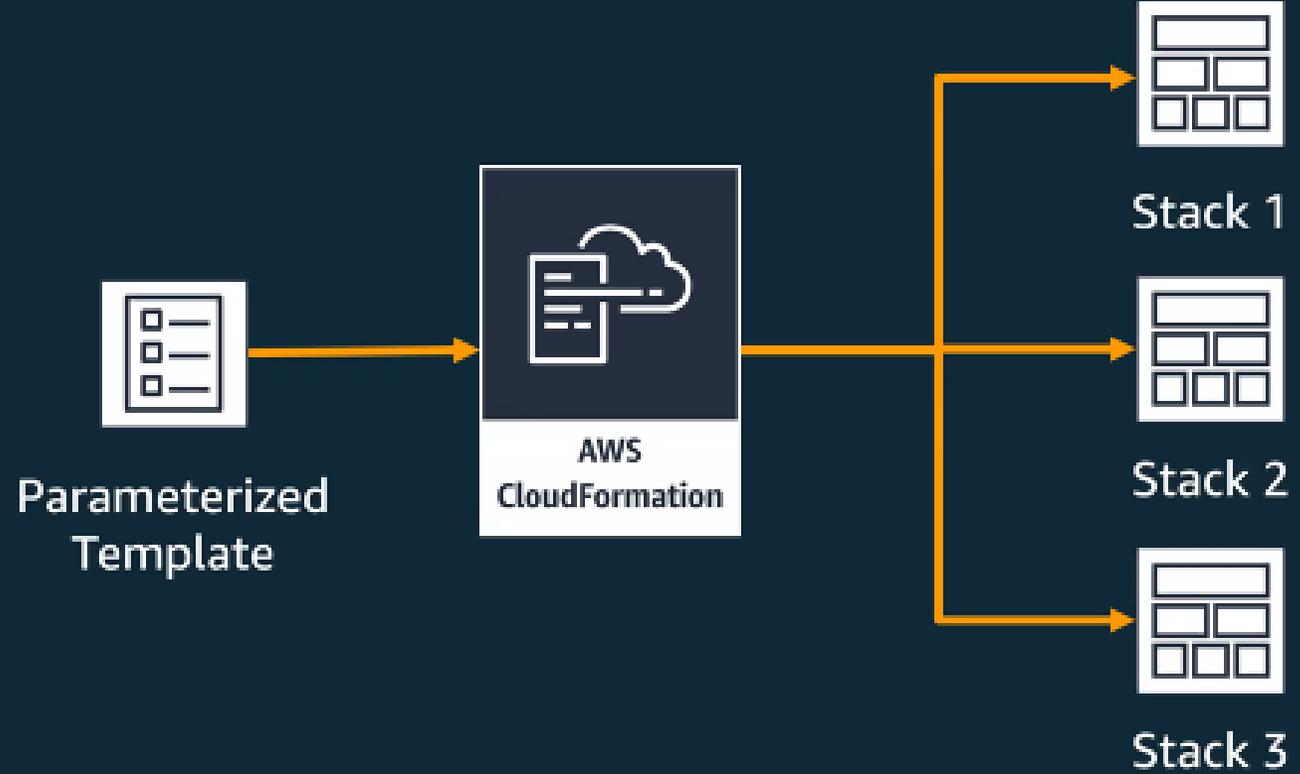
Para não perder  
de vista **as**  
**motivações**

*Reduzir o tempo da entrega de valor em  
produção, aumentar o **feedback** entre os times  
e fornecer um **ambiente seguro** para **experimentações**  
e **inovações de negócio**.*

# Motivações ainda mais específicas

- Custo
- Produtividade
- Risco

# Infraestrutura como código na AWS?



# Pedaco de um template de CloudFormation

```
{
  "Resources": {
    "Videos88AE8DA0": {
      "Type": "AWS::S3::Bucket",
      "Properties": {
        "LifecycleConfiguration": {
          "Rules": [
            {
              "AbortIncompleteMultipartUpload": {
                "DaysAfterInitiation": 1
              },
              "ExpirationInDays": 1,
              "Status": "Enabled"
            }
          ]
        },
        "PublicAccessBlockConfiguration": {
          "BlockPublicAcls": true,
          "BlockPublicPolicy": true,
          "IgnorePublicAcls": true,
          "RestrictPublicBuckets": true
        }
      },
      "UpdateReplacePolicy": "Delete",
      "DeletionPolicy": "Delete",
      "Metadata": {
        "aws:cdk:path": "sayit/Videos/Resource"
      }
    },
    "VideosNotifications193C6AA8": {
      "Type": "Custom::S3BucketNotifications",
      "Properties": {
        "ServiceToken": {
          "Fn::GetAtt": [
            "BucketNotificationsHandler050a0587b7544547bf325f094a3db8347ECC3691",
            "Arn"
          ]
        }
      },
      "BucketName": {
        "Ref": "Videos88AE8DA0"
      },
      "NotificationConfiguration": {
        "LambdaFunctionConfigurations": [
          {
            "Events": [
              "s3:ObjectCreated:*"
            ],
            "LambdaFunctionArn": {
              "Fn::GetAtt": [
                "StartVideoInputtingStateMachineLambda48B5C82F",
                "Arn"
              ]
            }
          }
        ]
      }
    }
  }
}
```

Ainda estamos  
falando de  
código

~~*Infraestrutura como código*~~

*Infraestrutura É código*

Declarando a  
infraestrutura  
como código  
com  
**CloudFormation**

- 😊 Reproduzível/Previsível
- 😊 Fácil de automatizar
- 😊 Possui gerenciamento de estado
- 😞 Curva de aprendizado acentuada
- 😞 Pouca ou nenhuma abstração
- 😞 Difícil compartilhar padrões com o time
- 😞 (CloudFormation) Difícil de modularizar
- 😞 (CloudFormation) Não tem fluxos de controle embutidos

**Problemas que  
aparecem com  
frequência**

Baixa produtividade para escrever templates

Dificuldade em modularizar e compartilhar soluções

Dificuldade em utilizar fluxos de controle

# AWS Cloud Development Kit (CDK)

O AWS CDK é um framework de multi-linguagem para modelar infraestrutura de nuvem como **componentes reutilizáveis**



```
1 import sns = require('@aws-cdk/aws-sns');
2 import sqs = require('@aws-cdk/aws-sqs');
3 import cdk = require('@aws-cdk/cdk');
4
5 export class HelloCdkStack extends cdk.Stack {
6   constructor(scope: cdk.App, id: string, props?: cdk.StackProps) {
7     super(scope, id, props);
8
9     const queue = new sqs.Queue(this, 'HelloCdkQueue', {
10      visibilityTimeoutSec: 300
11    });
12
13    const topic = new sns.Topic(this, 'HelloCdkTopic');
14
15    topic.subscribeQueue(queue);
16  }
17 }
18
```



AWS CloudFormation



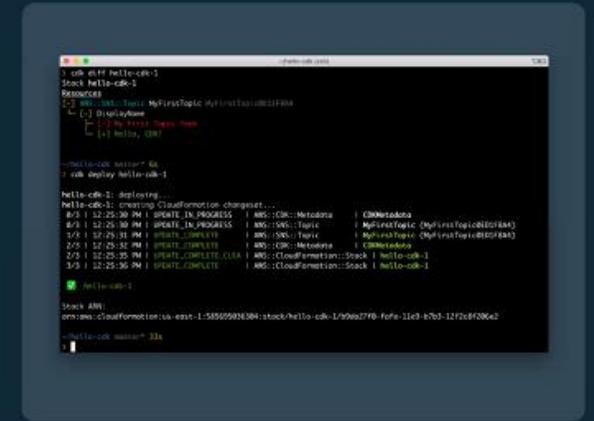
# Componentes principais e fluxo de construção



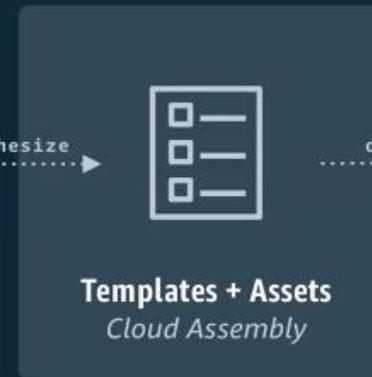
Core Framework



AWS Construct Library



AWS CDK CLI



# A AWS Construct Library



# Menos código, para fazer mais

```
// Example of an ECS-patterns construct  
new patterns.ApplicationLoadBalancedFargateService(stack, 'MyFargateService',{  
  taskImageOptions: {  
    image: ecs.ContainerImage.fromRegistry("amazon/amazon-ecs-sample")  
  }  
});
```



# Ambiente de desenvolvimento

```
vpc = Vpc(self, "ScrumDayBrazil2020Vpc", max_azs=2)
```

```
vpc.
```

m	__init__(self, scope, id, cidr, default_instance_te...	Vpc
p	stack	Resource
	while	while expr
m	__dir__(self)	object
f	__doc__	Vpc
m	add_dynamo_db_endpoint	Vpc
m	add_flow_log	Vpc
m	add_gateway_endpoint	Vpc
m	add_interface_endpoint	Vpc
m	add_s3_endpoint	Vpc
m	add_vpn_connection	Vpc
p	availability_zones	Vpc

Press Enter to insert, Tab to replace [Next Tip](#)

# Testes de unidade e integração

Bash

```
PASS test/dead-letter-queue.test.js
```

```
✓ dlq creates an alarm (51ms)
```

```
> 1 snapshot updated.
```

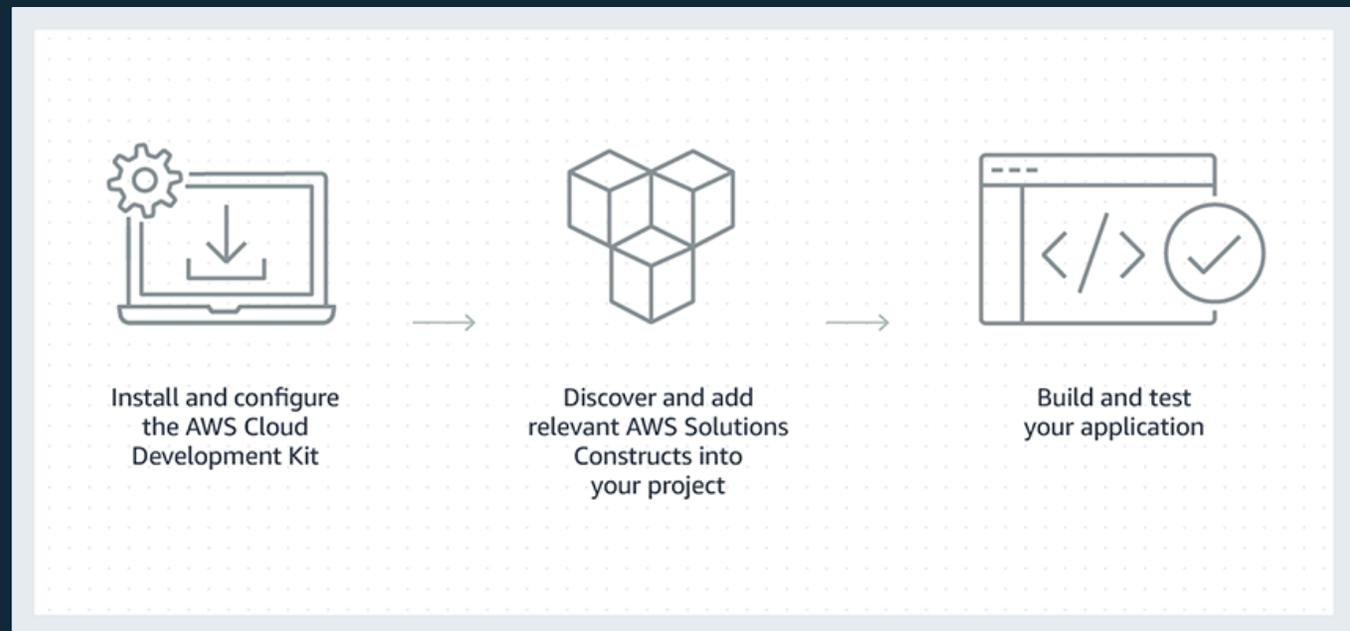
```
Snapshot Summary
```

```
> 1 snapshot updated
```

# AWS Solutions Constructs

Os “Constructs” de soluções da AWS são criados e mantidos pela AWS, usando práticas recomendadas estabelecidas pelo **AWS Well-Architected Framework**.

<https://aws.amazon.com/solutions/constructs/>  
<https://aws.amazon.com/solutions/constructs/patterns/>



# CDK Patterns

<https://cdkpatterns.com>

<https://github.com/cdk-patterns/serverless>

<https://twitter.com/CdkPatterns>

<https://www.youtube.com/c/DeconstructingAWSCDKPatterns>



aws

**CDK Patterns**

## Links úteis

- <https://aws.amazon.com/pt/cdk>
- <https://github.com/aws/aws-cdk>
- <https://docs.aws.amazon.com/cdk/api/latest/docs/aws-construct-library.html>
- <https://aws.amazon.com/solutions/constructs/>
- <https://aws.amazon.com/solutions/constructs/patterns/>
- <https://cdkpatterns.com>
- <https://github.com/cdk-patterns/serverless>
- <https://twitter.com/CdkPatterns>
- <https://www.youtube.com/c/DeconstructingAWSCDKPatterns>
- <https://medium.com/@mfpalladino>



# A comunidade AWS UG SP



спасибо 谢谢  
GRACIAS 谢谢  
**THANK YOU**  
ありがとうございました MERCI  
DANKE धन्यवाद  
شُكْرًا **OBRIGADO**

