AWS USER GROUPS MUMBAI

presents

SHETALKS

- TECH TALKS
- WORKSHOP
- FIRECHAT
- QUIZ

9TH APRIL,2022

2-5 PM IST

In association with

AWS User Groups Pune

AWS User Groups Nashik

Our Community Partners

pyladies Mumbai

WOMEN IN DATA SCIENCE MUMBAI & KERALA

WOMEN IN BIG DATA INDIA

WOMEN IN BIG DATA BANGLADESH

Speakers

Spandana Y.

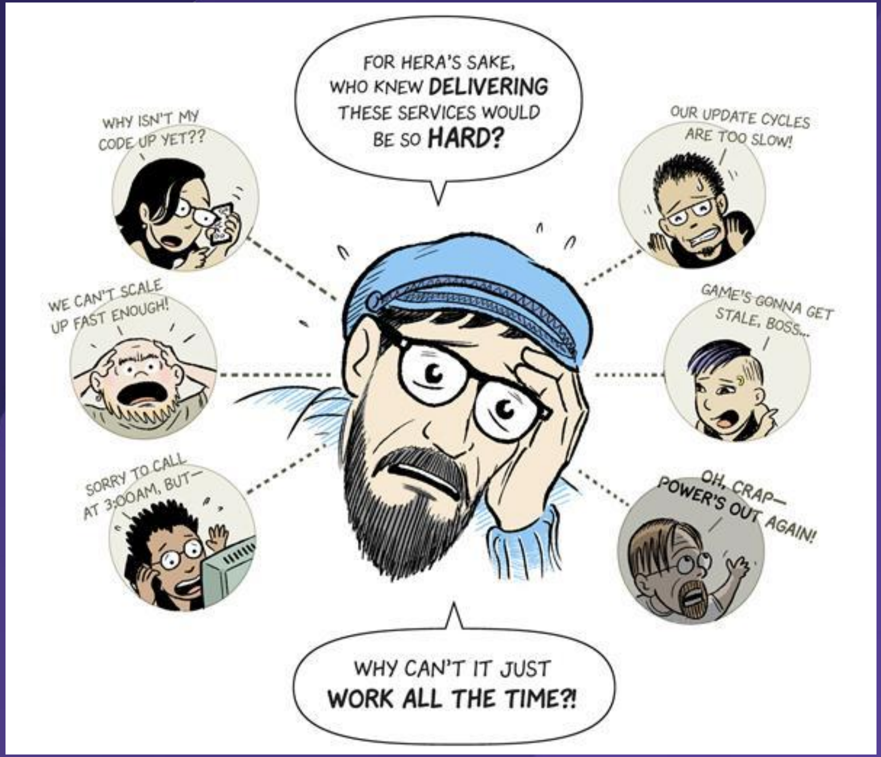
Sr. Cloud Engineer - AWS at Quantiphi

AWS
User Groups

Agenda

- CICD Tools
- Cloudformation

Life Without Devops


FOR HERA'S SAKE, WHO KNEW DELIVERING THESE SERVICES WOULD BE SO HARD?
WHY ISN'T MY CODE UP YET??
OUR UPDATE CYCLES ARE TOO SLOW!
WE CAN'T SCALE UP FAST ENOUGH!
GAME'S GONNA GET STALE, BOSS...
SORRY TO CALL AT 3:00AM, BUT—
OH, CRAP— POWER'S OUT AGAIN!
WHY CAN'T IT JUST WORK ALL THE TIME?!

AWS
User Groups

CICD Overview

- Continuous Integration
- Continuous Deployment
- Benefits of CICD
  - Small and frequent code changes
  - Faster Release Rate
  - Easier Rollbacks
  - Increase Team Transparency and Accountability

AWS CICD Tools

AWS
CodeCommit

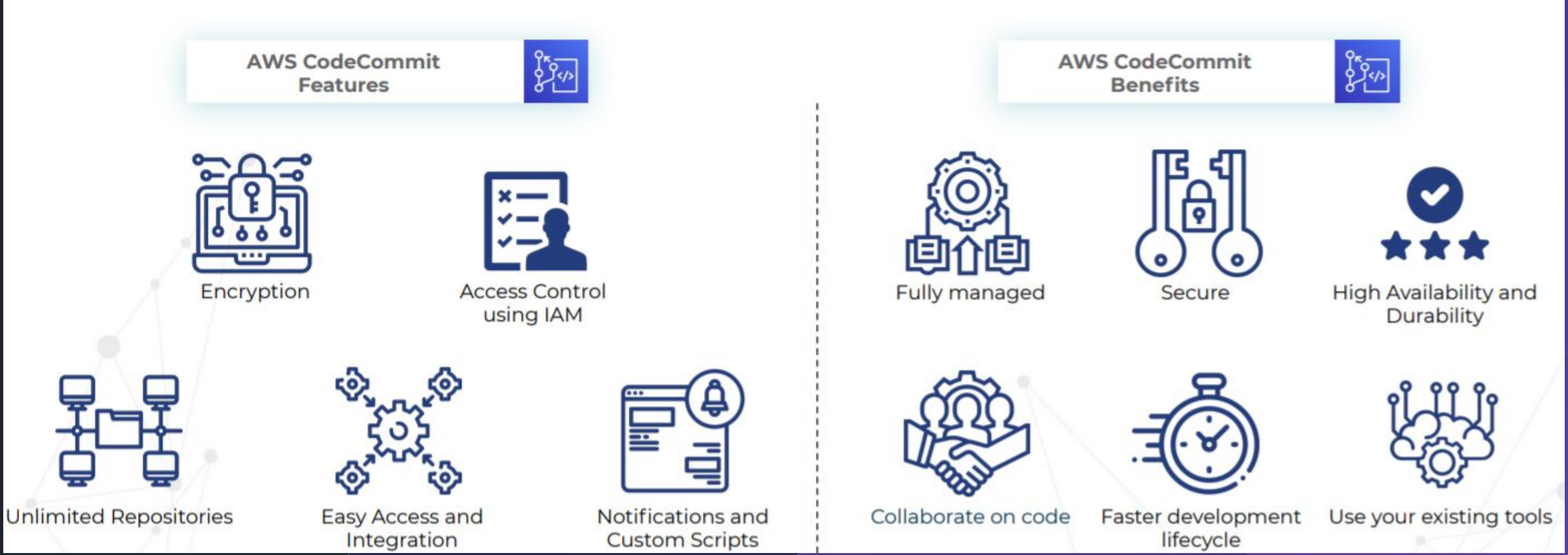
AWS
CodeBuild

AWS
CodeDeploy

AWS
CodePipeline

CodeCommit

AWS CodeCommit Overview



AWS CodeCommit Features

- Encryption
- Access Control using IAM
- Unlimited Repositories
- Easy Access and Integration
- Notifications and Custom Scripts

AWS CodeCommit Benefits

- Fully managed
- Secure
- High Availability and Durability
- Collaborate on code
- Faster development lifecycle
- Use your existing tools

Demo

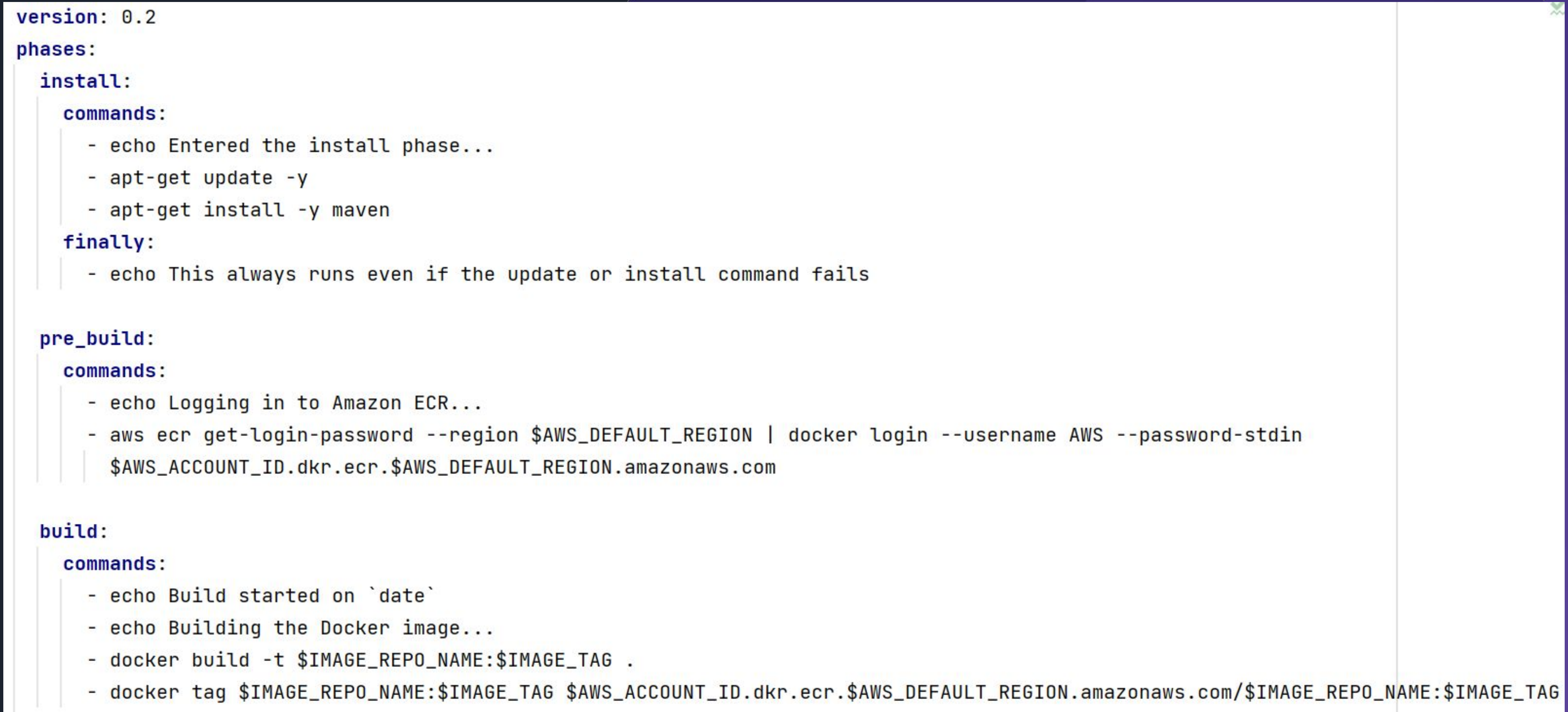
- Console walkthrough

CodeBuild

CodeBuild Overview

- Fully Managed Build service(alternative to tools like jenkins)
- Serverless
- Secure,Highly scalable
- Continuous scaling(no servers to manage)
- Pay only for the build time.
- Supports custom Docker images
- Build instructions defined in buildspec.yml file

buildspec.yml structure

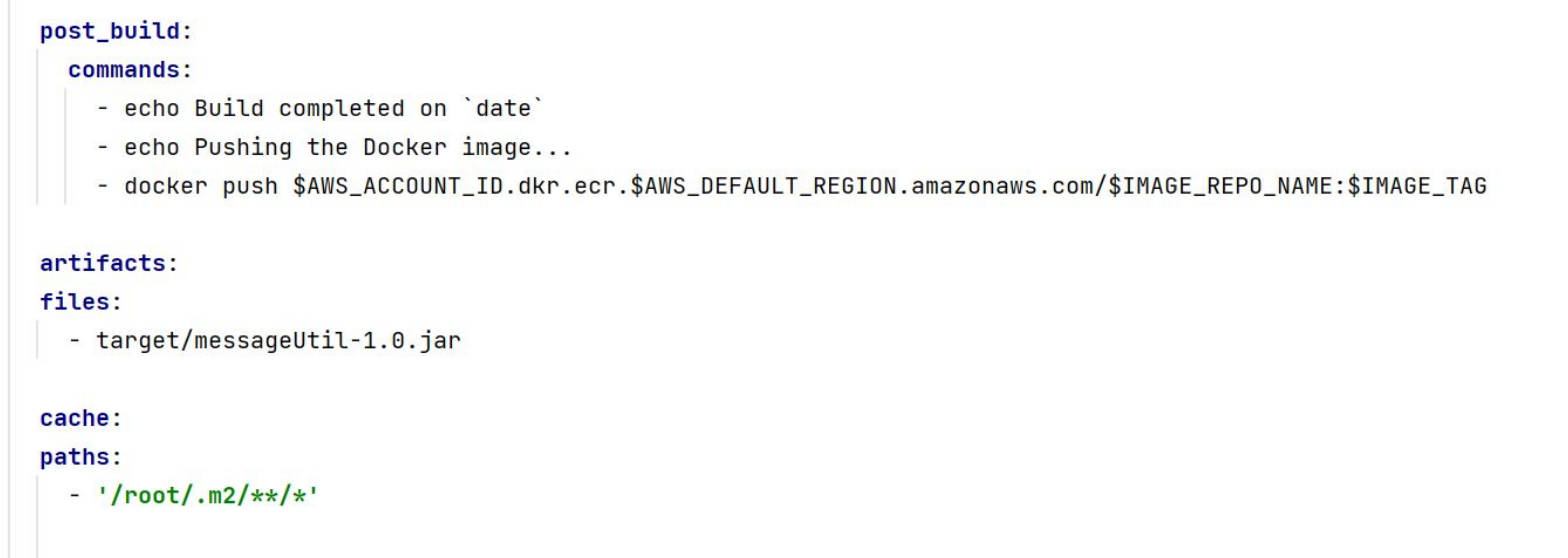


```yaml
version: 0.2
phases:
  install:
    commands:
      - echo Entered the install phase...
      - apt-get update -y
      - apt-get install -y maven
    finally:
      - echo This always runs even if the update or install command fails

  pre_build:
    commands:
      - echo Logging in to Amazon ECR...
      - aws ecr get-login-password --region $AWS_DEFAULT_REGION | docker login --username AWS --password-stdin
        $AWS_ACCOUNT_ID.dkr.ecr.$AWS_DEFAULT_REGION.amazonaws.com

  build:
    commands:
      - echo Build started on `date`
      - echo Building the Docker image...
      - docker build -t $IMAGE_REPO_NAME:$IMAGE_TAG .
      - docker tag $IMAGE_REPO_NAME:$IMAGE_TAG $AWS_ACCOUNT_ID.dkr.ecr.$AWS_DEFAULT_REGION.amazonaws.com/$IMAGE_REPO_NAME:$IMAGE_TAG
```

buildspecs.yaml structure



```
post_build:
  commands:
    - echo Build completed on `date`
    - echo Pushing the Docker image...
    - docker push $AWS_ACCOUNT_ID.dkr.ecr.$AWS_DEFAULT_REGION.amazonaws.com/$IMAGE_REPO_NAME:$IMAGE_TAG

artifacts:
files:
  - target/messageUtil-1.0.jar

cache:
paths:
  - '/root/.m2/**/*'
```

link: buildspec.yml syntax

Demo

- Console Walkthrough

CodeDeploy

CodeDeploy Overview

- Automates application deployment to EC2,Lambda,ECS
- Also supports deployment On-Premise servers
- Secure,Highly scalable,Fully managed
- Deploy instructions defined in appspec.yml file
- Automated Rollback

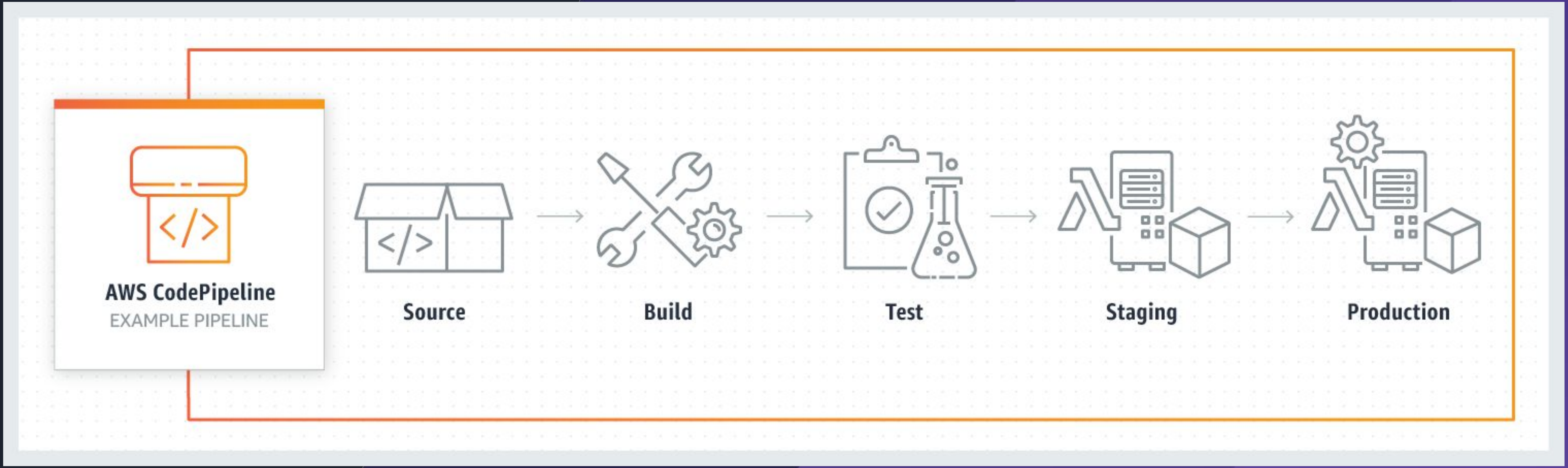
Note: CodeDeploy does not provision resources

Demo

- Console Walkthrough

CodePipeline

CodePipeline Overview

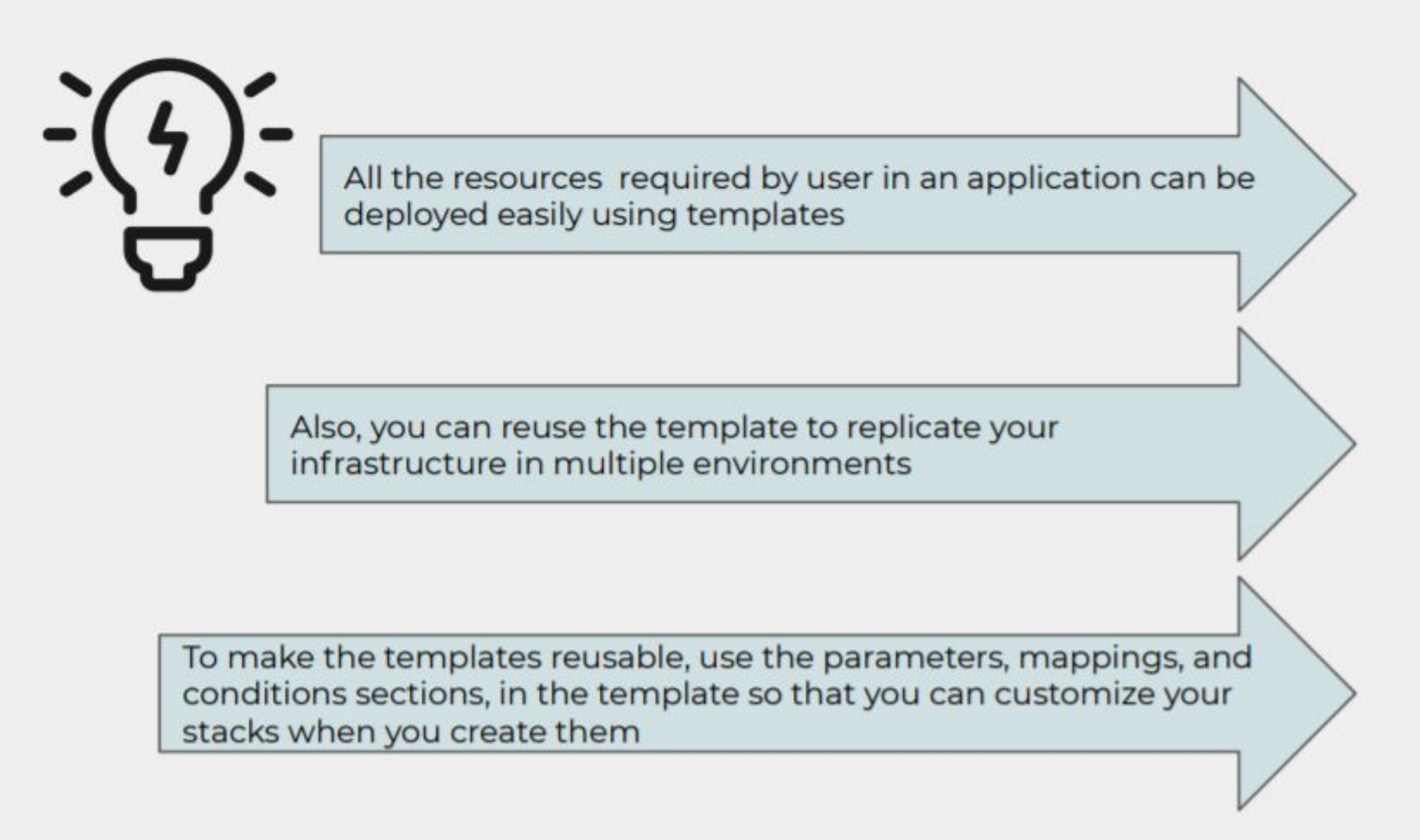


https://aws.amazon.com/codepipeline/

Demo

- Console Walkthrough

CloudFormation



All the resources required by user in an application can be deployed easily using templates

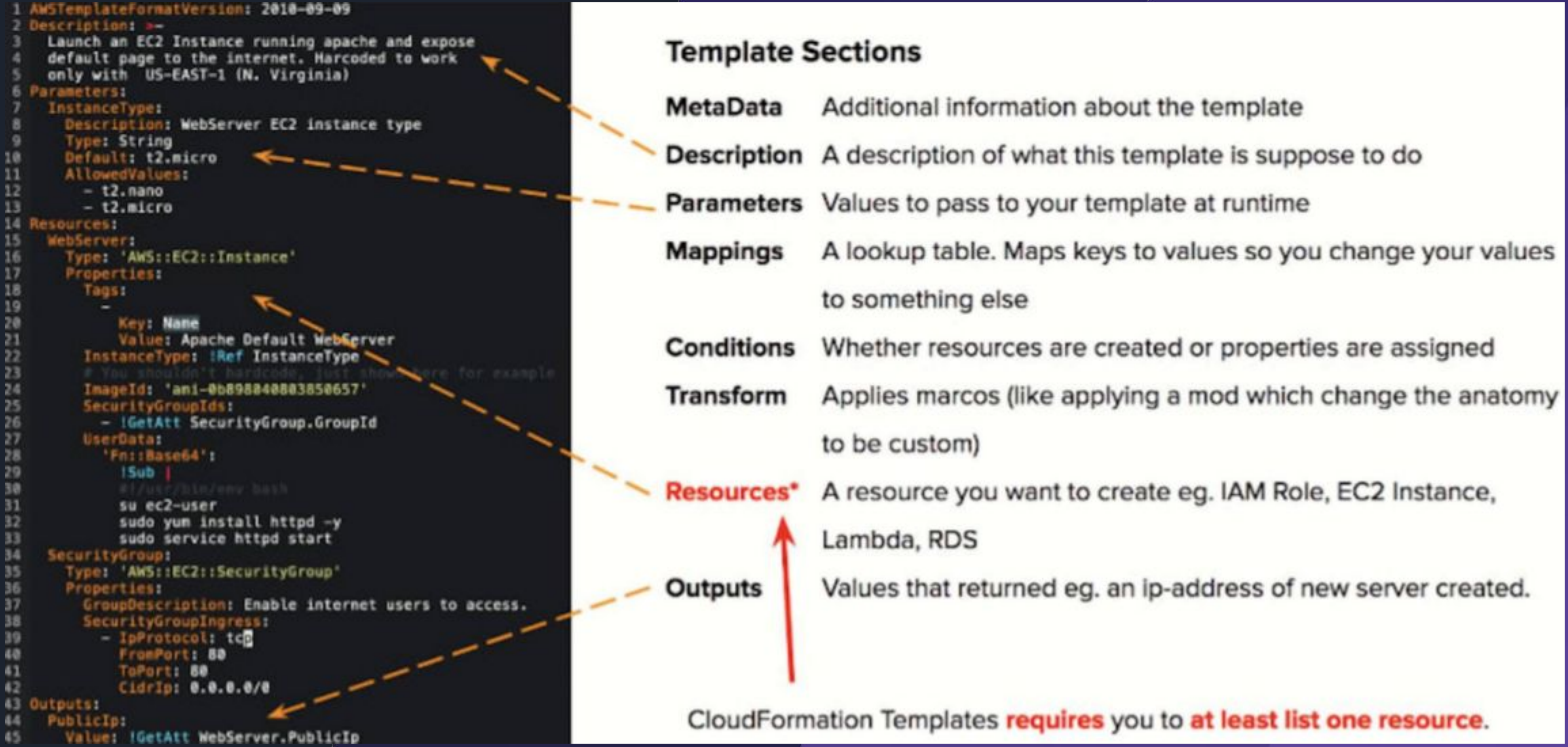
Also, you can reuse the template to replicate your infrastructure in multiple environments

To make the templates reusable, use the parameters, mappings, and conditions sections, in the template so that you can customize your stacks when you create them

CloudFormation Overview

- Infrastructure as a code(IAAC)
- One Click Creation and Deletion
- Declarative in nature
- Rollback through version control
- Supports Yaml/Json
- Custom Resource

Template Structure



```
AWSTemplateFormatVersion: 2010-09-09
Description: >-
  Launch an EC2 Instance running apache and expose
  default page to the internet. Harcoded to work
  only with  US-EAST-1 (N. Virginia)
Parameters:
  InstanceType:
    Description: WebServer EC2 instance type
    Type: String
    Default: t2.micro
    AllowedValues:
      - t2.nano
      - t2.micro
Resources:
  WebServer:
    Type: 'AWS::EC2::Instance'
    Properties:
      Tags:
        -
          Key: Name
          Value: Apache Default WebServer
      InstanceType: !Ref InstanceType
      # You shouldn't hardcode, just shown here for example
      ImageId: 'ami-0b898040803850657'
      SecurityGroupIds:
        - !GetAtt SecurityGroup.GroupId
      UserData:
        'Fn::Base64':
          !Sub |
          #!/usr/bin/env bash
          su ec2-user
          sudo yum install httpd -y
          sudo service httpd start
  SecurityGroup:
    Type: 'AWS::EC2::SecurityGroup'
    Properties:
      GroupDescription: Enable internet users to access.
      SecurityGroupIngress:
        - IpProtocol: tcp
          FromPort: 80
          ToPort: 80
          CidrIp: 0.0.0.0/0
Outputs:
  PublicIp:
    Value: !GetAtt WebServer.PublicIp
```

Template Sections

MetaData	Additional information about the template
Description	A description of what this template is suppose to do
Parameters	Values to pass to your template at runtime
Mappings	A lookup table. Maps keys to values so you change your values to something else
Conditions	Whether resources are created or properties are assigned
Transform	Applies marcos (like applying a mod which change the anatomy to be custom)
Resources*	A resource you want to create eg. IAM Role, EC2 Instance, Lambda, RDS
Outputs	Values that returned eg. an ip-address of new server created.

CloudFormation Templates **requires** you to **at least list one resource.**

Demo

- Console Walkthrough

Q&A

Quiz Time !

Quiz

KAHOOT!

Note: Please use the same name via which you have registered!

Thank you!

AWS
User Groups