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

BOA316

Terraform providers using AWS CloudFormation custom resources

Sohan Maheshwar

Developer Advocate
Amazon Web Services

Martijn van Dongen

AWS Community Hero & Cloud Evangelist
Schuberg Philis

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

Based on a true story

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

Customer 1: Automated lift and shift migration



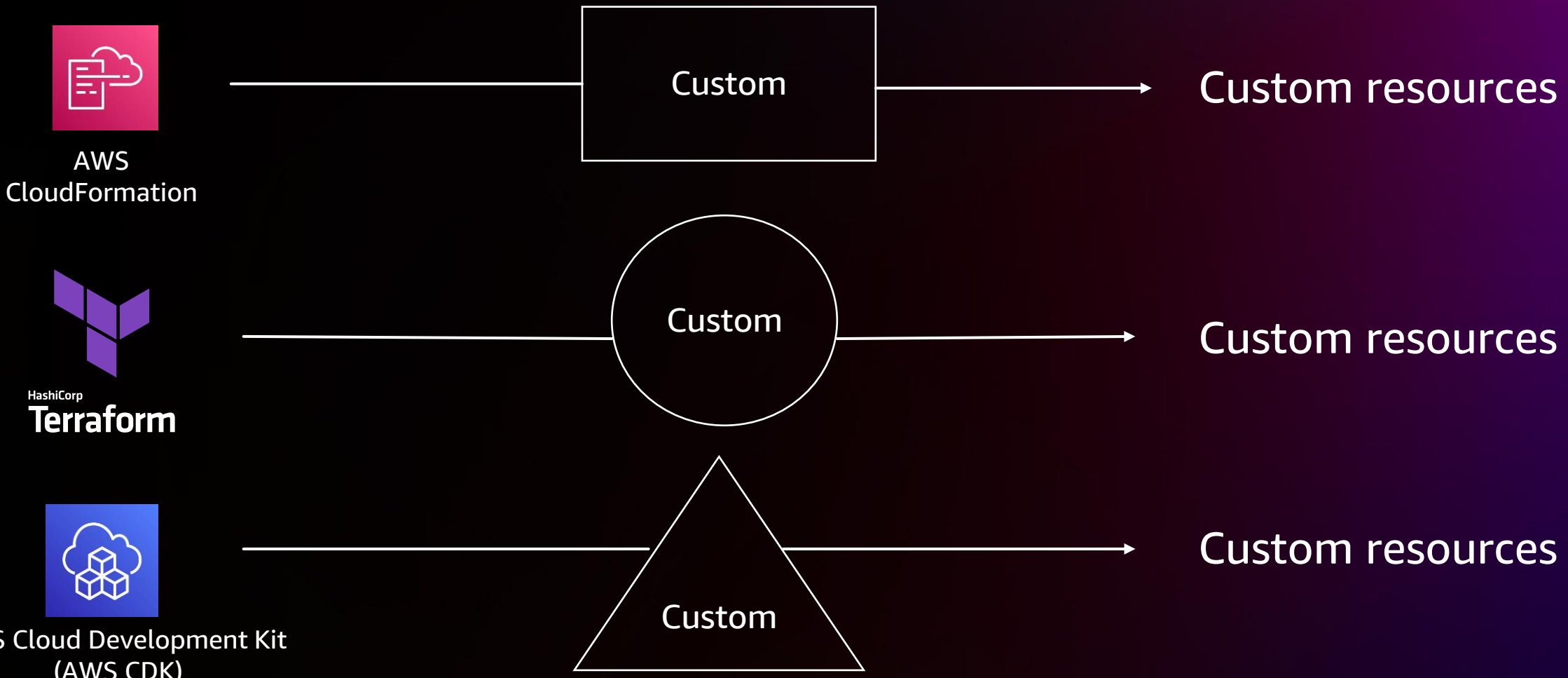
Legacy data center
with 1000+ workloads

10K+ engineers
Variance in tooling

AWS Cloud

Resources

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



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



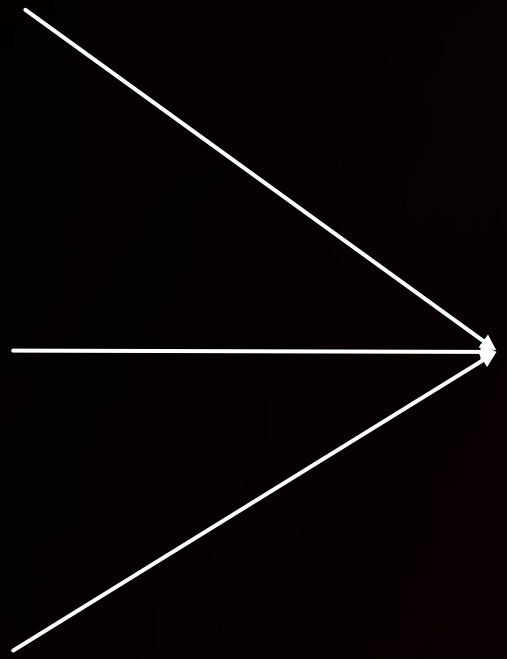
AWS
CloudFormation



HashiCorp
Terraform



AWS Cloud Development Kit
(AWS CDK)



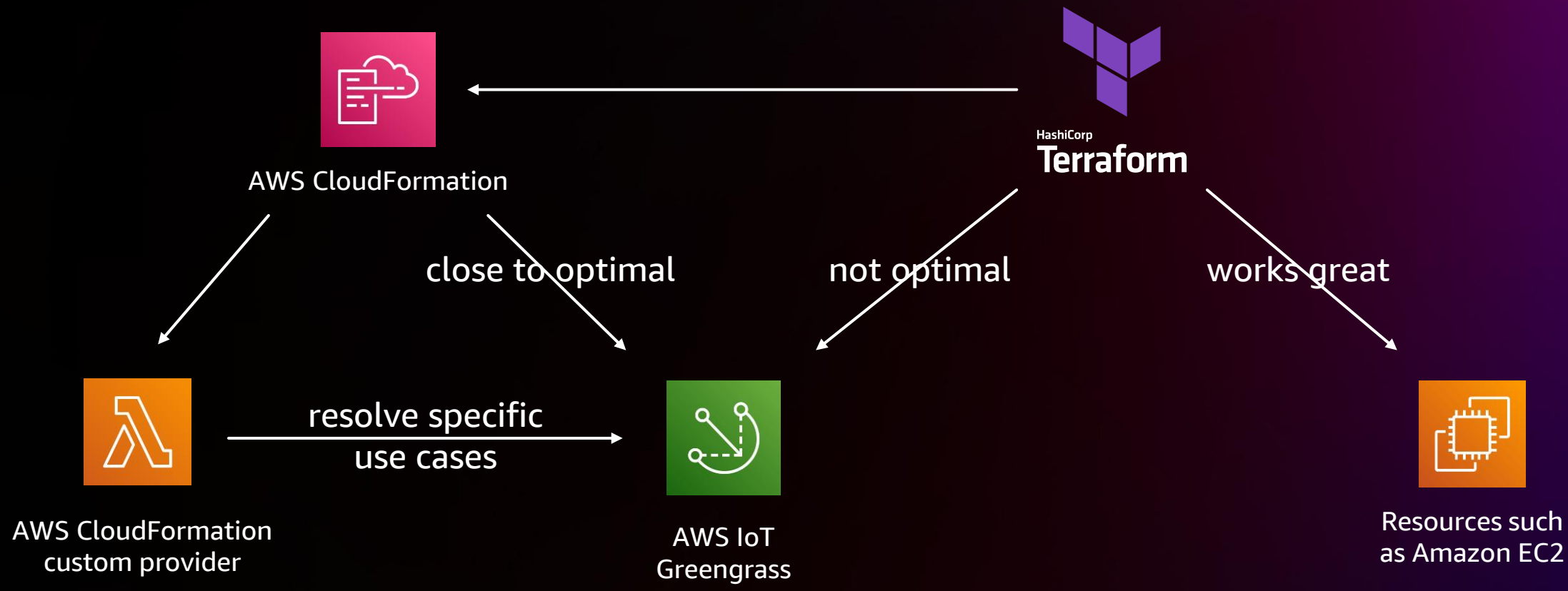
Common
implementation



Custom resources

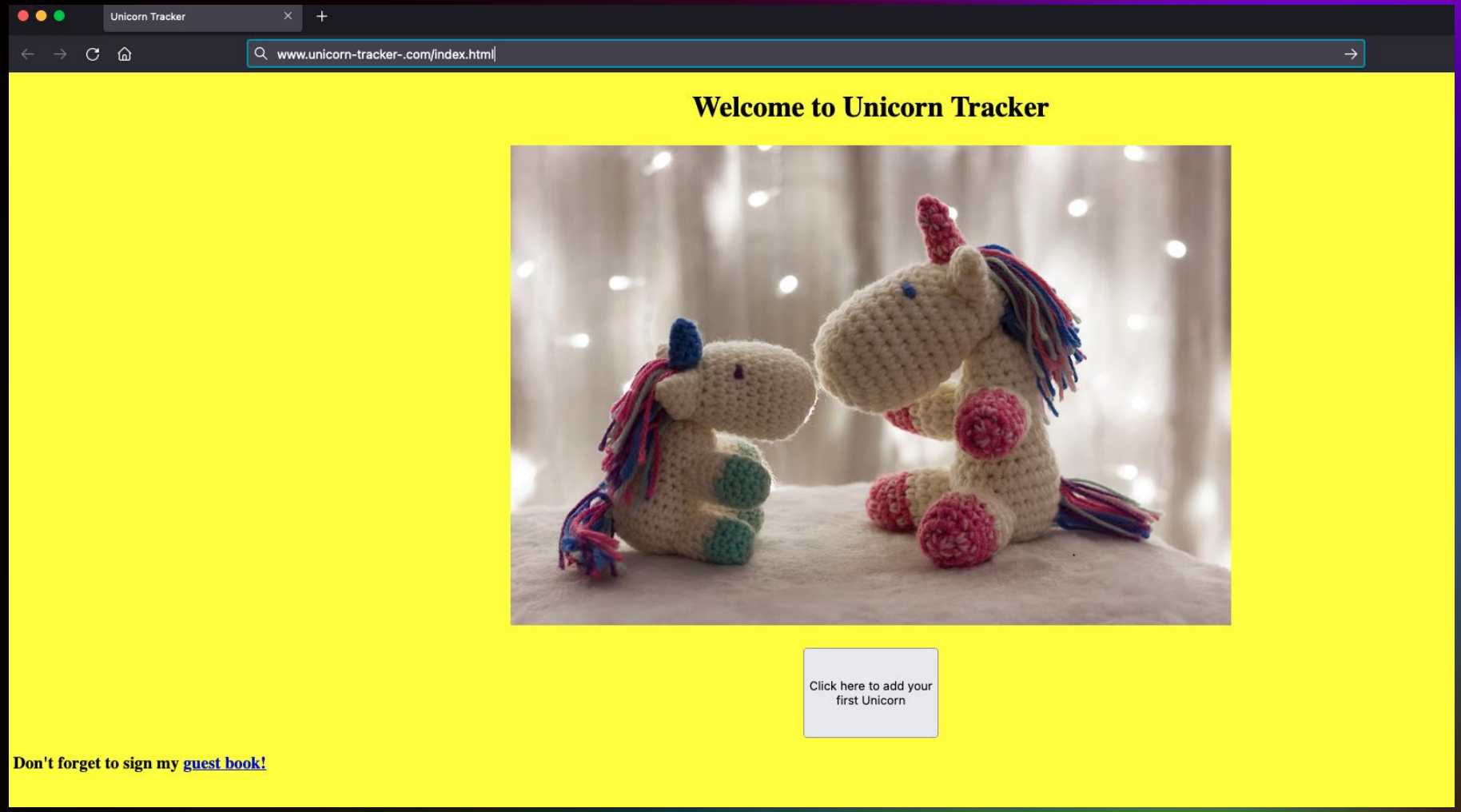
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Customer 2: 100% infrastructure as code



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

Let’s look at a legacy application

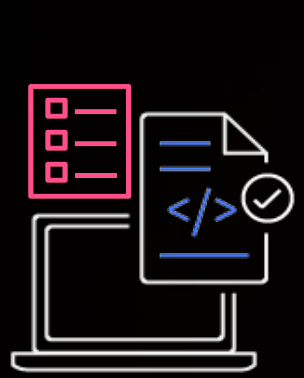


© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS CloudFormation custom resources

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

AWS CloudFormation



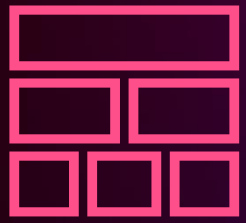
Code in YAML or JSON directly or use sample templates



Upload local files or from an S3 bucket



Create stack using API via AWS CloudFormation



Stacks and resources are provisioned as a running environment

aws

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

Custom resources

- Write custom provisioning logic in CloudFormation templates
- Include resources that aren't available as AWS CloudFormation resource types

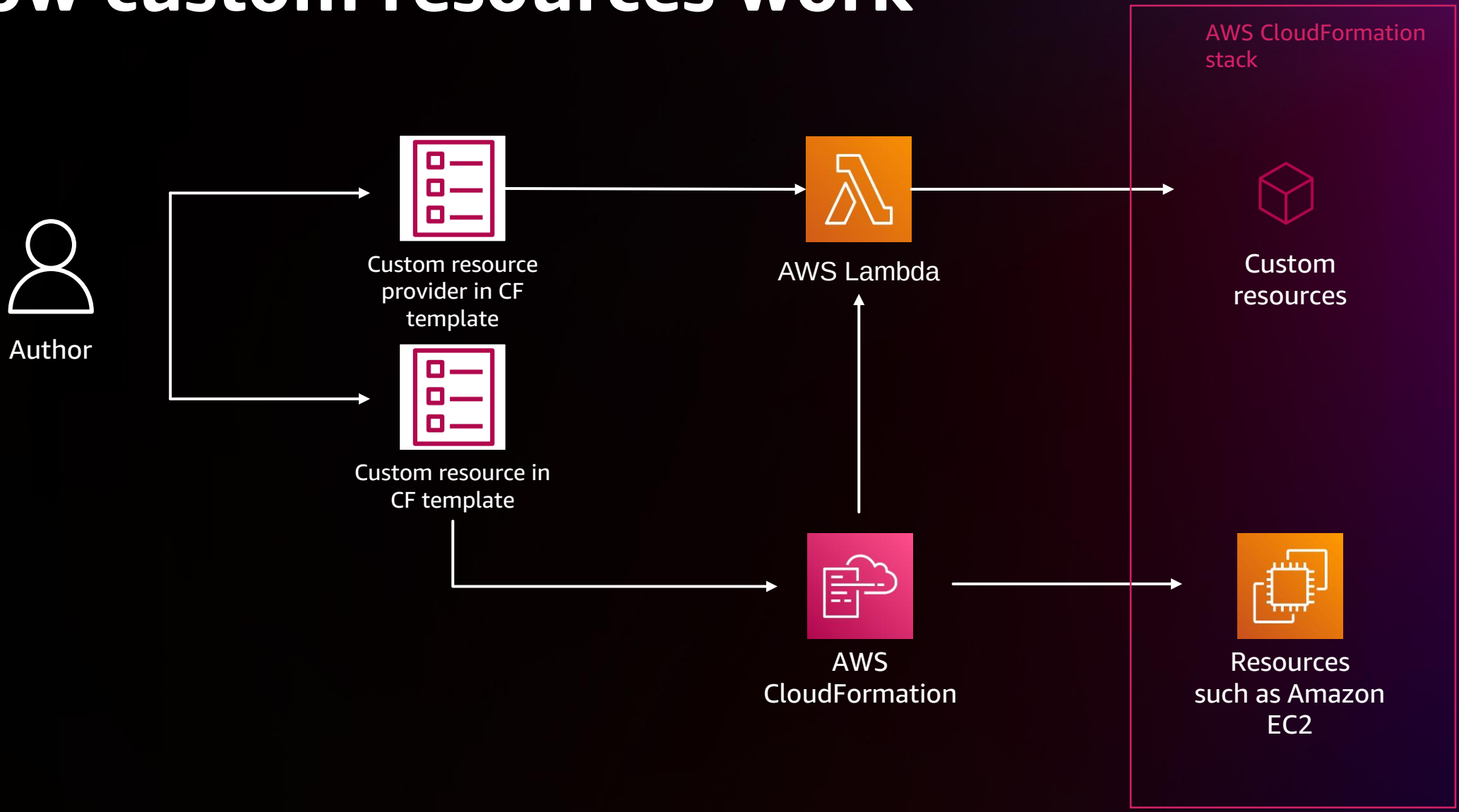
```
Resources:
  Server:
    Type: “AWS:EC2::Instance”
    Properties:
      InstanceType: m1.small
...
```

```
Resources:
  MyCustomResource:
    Type: “Custom::UnicornTracker”
    Properties:
        ServiceToken: 'arn:aws:lambda:us-east..’
...
```

aws

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

How custom resources work



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

AWS Lambda code

```python
#backend_api = "https://abcxxx123.execute-api.eu-west-1.amazonaws.com/prod"
backend_api = os.getenv('UT_ENDPOINT')
helper = CfnResource()

@helper.create
def create(event, _):
    if 'UnicornName' in event['ResourceProperties']:
        response = requests.put(f"{backend_api}/items",
                    json = {'name': event['ResourceProperties']['UnicornName'],
                            'length': 1234})
    else:
        raise ValueError("UnicornName is a required property.")
    res_dict = json.loads(response.json())
    helper.Data["UnicornId"] = res_dict['id']
    return res_dict['id']

@helper.delete
def delete(event, _):
    id = event['PhysicalResourceId']
    response = requests.delete(f"{backend_api}/items/{id}")

def handler(event, context):
    helper(event, context)
```

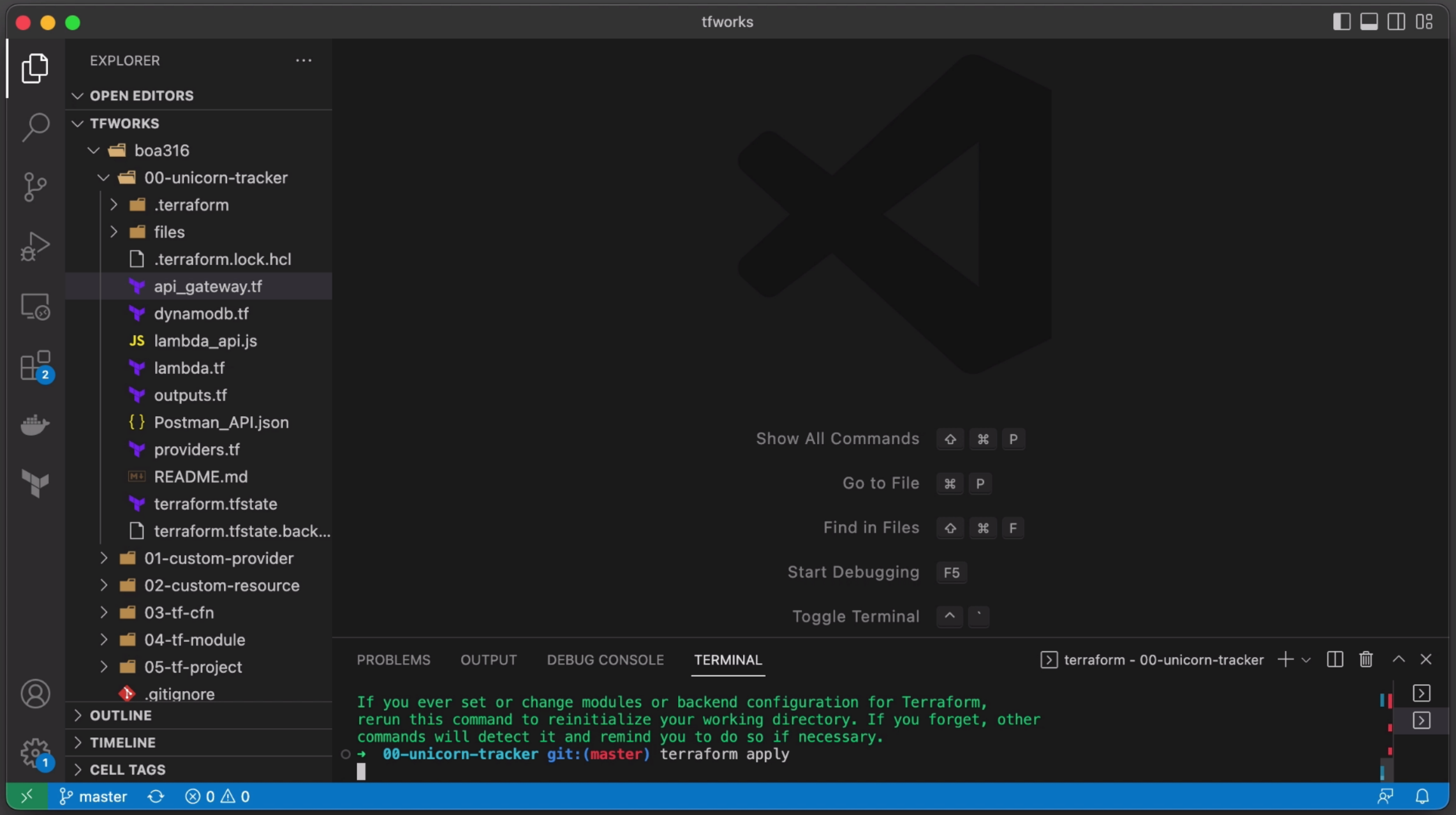
aws

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

CloudFormation code

```
Transform: AWS::Serverless-2016-10-31
Resources:
  LambdaLayer:
    Type: AWS::Serverless::LayerVersion
    Properties:
      ContentUri: "lambda_layer/lambda_layer.zip"
  LambdaFunction:
    Type: AWS::Serverless::Function
    Properties:
      FunctionName: CustomUnicornTracker
      Handler: function.handler
      Runtime: python3.9
      CodeUri: lambda/
      Layers:
        - !Ref LambdaLayer
      Environment:
        Variables:
          UT_ENDPOINT: "https://abcxxx123.execute-api.eu-west-1.amazonaws.com/prod"
      Policies:
        - AWSLambdaExecute
```

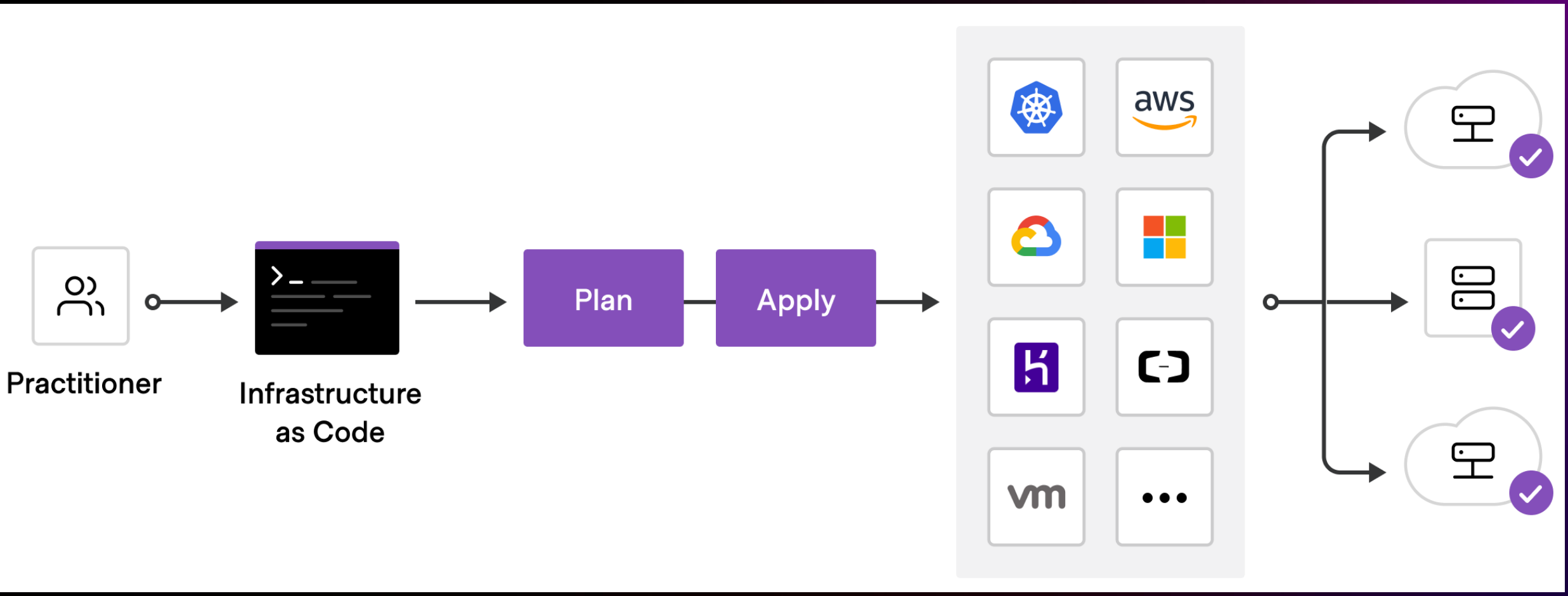
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.


tfworks
EXPLORER
OPEN EDITORS
TFWORKS
boa316
00-unicorn-tracker
.terraform
files
.terraform.lock.hcl
api_gateway.tf
dynamodb.tf
lambda_api.js
lambda.tf
outputs.tf
Postman_API.json
providers.tf
README.md
terraform.tfstate
terraform.tfstate.back...
01-custom-provider
02-custom-resource
03-tf-cfn
04-tf-module
05-tf-project
.gitignore
OUTLINE
TIMELINE
CELL TAGS
Show All Commands ⇧ ⌘ P
Go to File ⌘ P
Find in Files ⇧ ⌘ F
Start Debugging F5
Toggle Terminal ^ `
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
terraform - 00-unicorn-tracker
If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
➜ 00-unicorn-tracker git:(master) terraform apply
master
0 ⚠ 0

A Terraform module for our unicorns

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

Terraform refresher



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

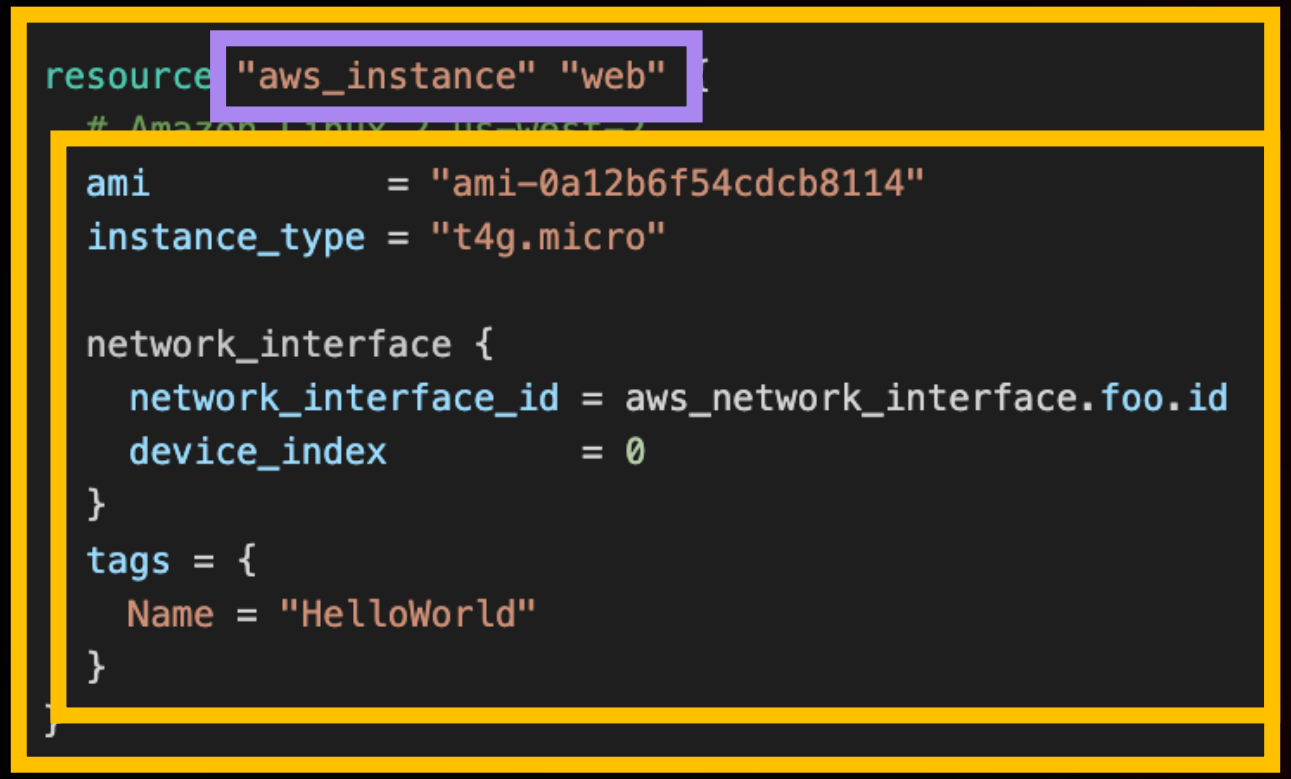
Terraform providers

- Terraform relies on plugins called providers to interact with cloud providers, SaaS providers, and other APIs
- Each provider adds a set of resource types and/or data sources that Terraform can manage
- Provider can orchestrate complex operations on dependent services



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

Terraform language (HCL)



Resource type and resource name

Resource attributes

Resource from AWS provider

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

Using CloudFormation with Terraform

```
provider "aws" {
  region = "eu-west-1"
}

variable "name" {
  type = string
  description = "Enter a name for the Unicorn"
}

data "aws_region" "current" {}

data "aws_caller_identity" "current" {}

resource "aws_cloudformation_stack" "unicorn" {
  name = "ut-tf-resource"

  parameters = {
    UnicornName = var.name
  }

  template_body = <<STACK
{
  "Parameters" : {
    "UnicornName" : {
      "Type" : "String"
    }
  },
  "Resources" : {
    "Unicorn": {
      "Type" : "Custom::Unicorn",
      "Properties" : {
        "ServiceToken" : "arn:aws:lambda:${data.aws_region.current.name}...",
        "UnicornName": { "Ref": "UnicornName" }
      }
    }
  }
}
STACK
}
```

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

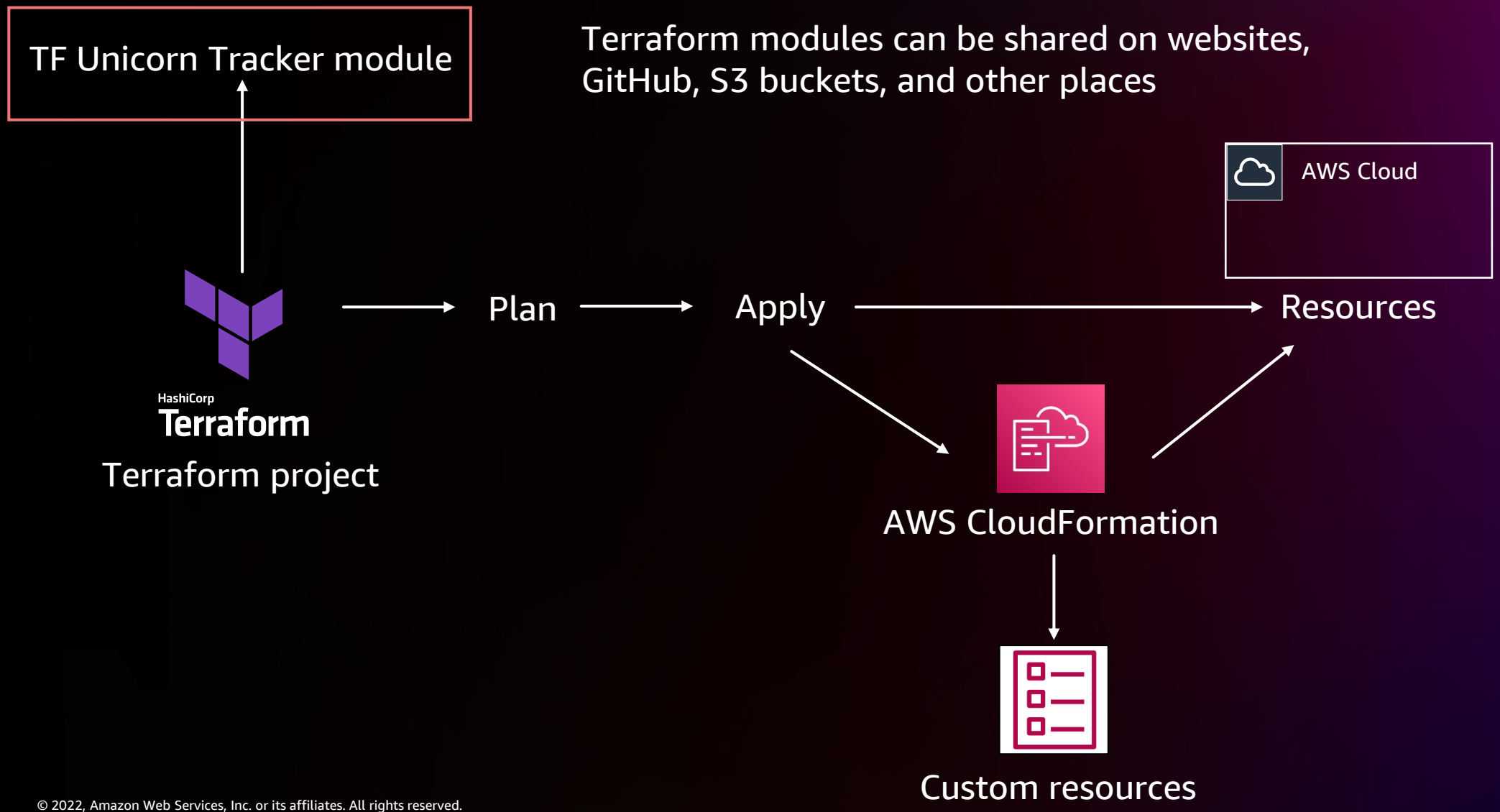
Terraform modules

- A set of Terraform configuration files in a single directory
- When you run Terraform commands directly from such a directory, it is considered the **root module,** so in this sense, every Terraform configuration is part of a module
- Modules can be loaded from either the local file system or a remote source
- Terraform Registry
- Best Practices
  - Name your provider `terraform-<PROVIDER>-<NAME>`
  - Start writing your configuration with modules in mind

aws

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

Creating and sharing a module



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

Using a Terraform module

```
provider "aws" {}

data "aws_region" "current" {}

data "aws_caller_identity" "current" {}

module "unicorn" {
  source = "https://url.s3.amazonaws.com/tfmodule.zip"

  name = "Andy"
}
```

aws

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

Terraform module

```
provider "aws" {
  region = "eu-west-1"
}

variable "name" {
  type = string
  description = "Enter a name for the Unicorn"
}

data "aws_region" "current" {}

data "aws_caller_identity" "current" {}

resource "aws_cloudformation_stack" "unicorn" {
  name = "ut-tf-resource"

  parameters = {
    UnicornName = var.name
  }

  template_body = <<STACK
{
  "Parameters" : {
    "UnicornName" : {
      "Type" : "String"
    }
  },
  "Resources" : {
    "Unicorn": {
      "Type" : "Custom::Unicorn",
      "Properties" : {
        "ServiceToken" : "arn:aws:lambda:${data.aws_region.current.name}...",
        "UnicornName": { "Ref": "UnicornName" }
      }
    }
  }
}
STACK
}
```

aws

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

Share a Terraform module

```
provider "aws" {}
provider "archive" {}

data "aws_region" "current" {}

data "aws_caller_identity" "current" {}

locals {
  s3_bucket = "ut-${data.aws_caller_identity.current.account_id}..."
}

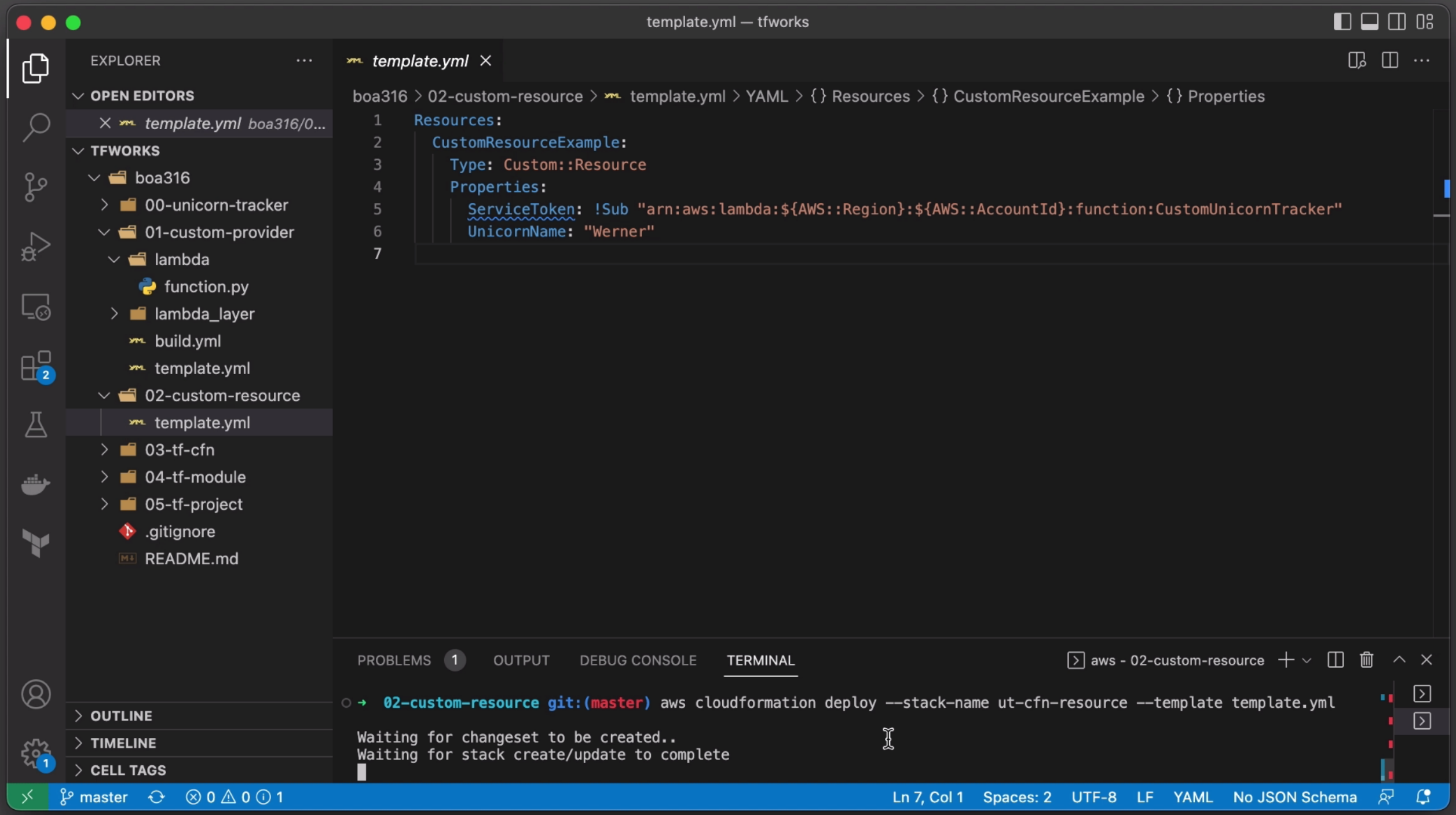
data "archive_file" "lambda_zip" {
  type        = "zip"
  source_dir  = "module_src/"
  output_path = "tfmodule.zip"
}

resource "aws_s3_bucket" "bucket" {
  bucket = local.s3_bucket
}

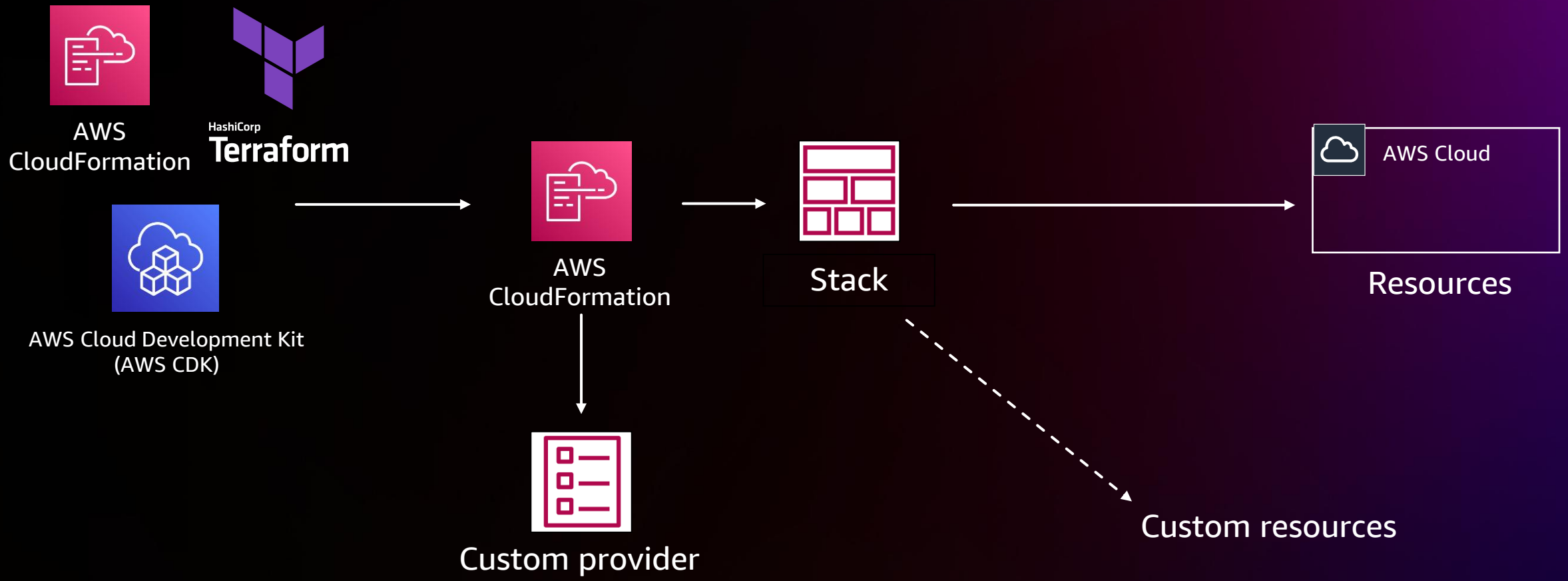
resource "aws_s3_object" "data_zip" {
  bucket = aws_s3_bucket.bucket.id
  key    = data.archive_file.lambda_zip.output_path
  acl    = "public-read"
  source = data.archive_file.lambda_zip.output_path
  etag   = data.archive_file.lambda_zip.output_md5
}

output "url" {
  value = "https://${aws_s3_bucket.bucket.bucket_domain_name}/..."
}
```

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


template.yml — tfworks
EXPLORER
OPEN EDITORS
template.yml boa316/0...
TFWORKS
boa316
00-unicorn-tracker
01-custom-provider
lambda
function.py
lambda_layer
build.yml
template.yml
02-custom-resource
template.yml
03-tf-cfn
04-tf-module
05-tf-project
.gitignore
README.md
OUTLINE
TIMELINE
CELL TAGS
template.yml
boa316 > 02-custom-resource > template.yml > YAML > { } Resources > { } CustomResourceExample > { } Properties
1 Resources:
2 CustomResourceExample:
3 Type: Custom::Resource
4 Properties:
5 ServiceToken: !Sub "arn:aws:lambda:${AWS::Region}:${AWS::AccountId}:function:CustomUnicornTracker"
6 UnicornName: "Werner"
7
PROBLEMS 1
OUTPUT
DEBUG CONSOLE
TERMINAL
aws - 02-custom-resource
02-custom-resource git:(master) aws cloudformation deploy --stack-name ut-cfn-resource --template template.yml
Waiting for changeset to be created..
Waiting for stack create/update to complete
master
0 0 1
Ln 7, Col 1
Spaces: 2
UTF-8
LF
YAML
No JSON Schema

Final implementation



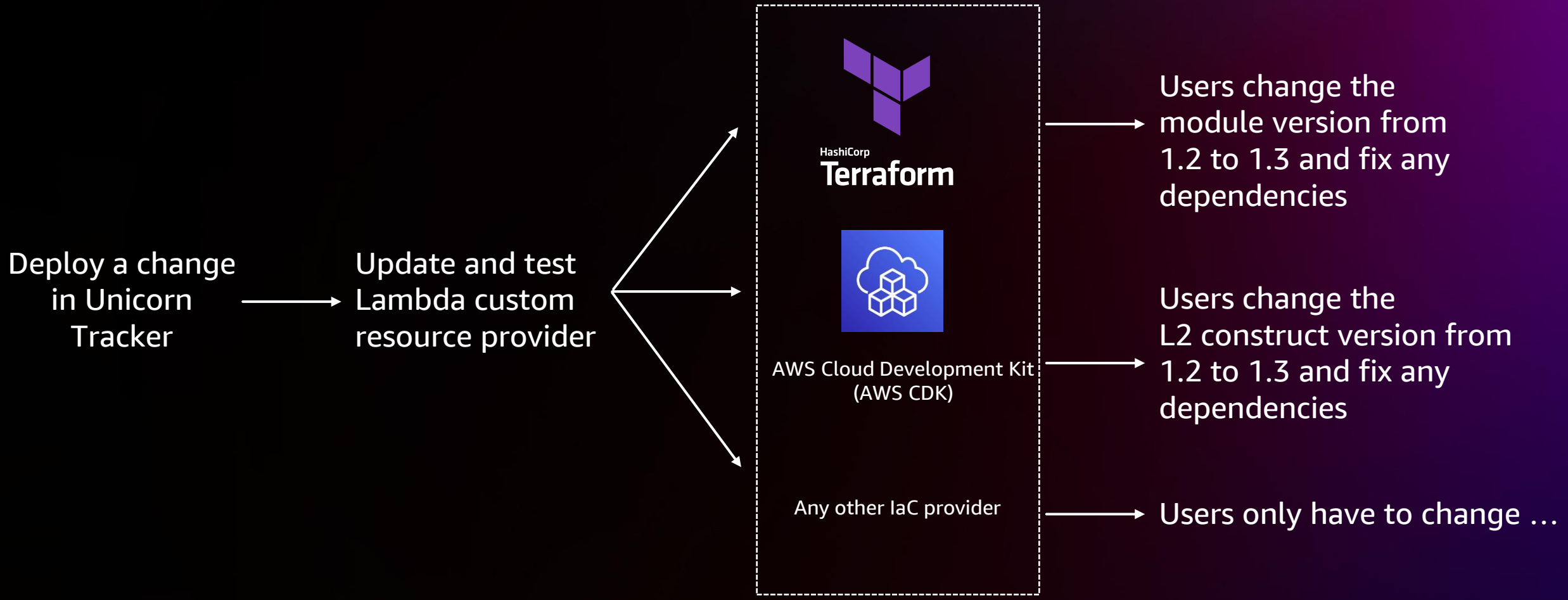
© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Without the solution

- Any change in the Unicorn Tracker requires every user’s attention
- Simultaneous rollouts when the Unicorn Tracker changes
- Almost impossible rollbacks
- Learn new coding languages, tools, or patterns
- Over-engineered glue code

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

With the solution



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

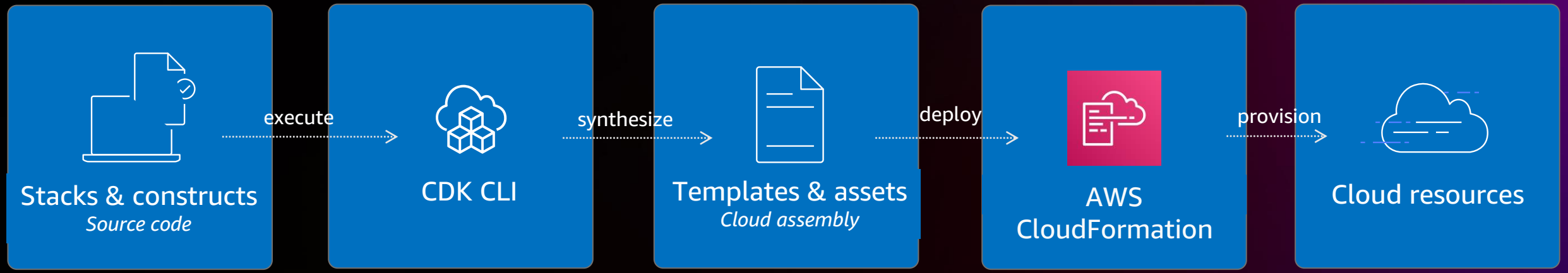
Next steps

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

And now in AWS Cloud Development Kit

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

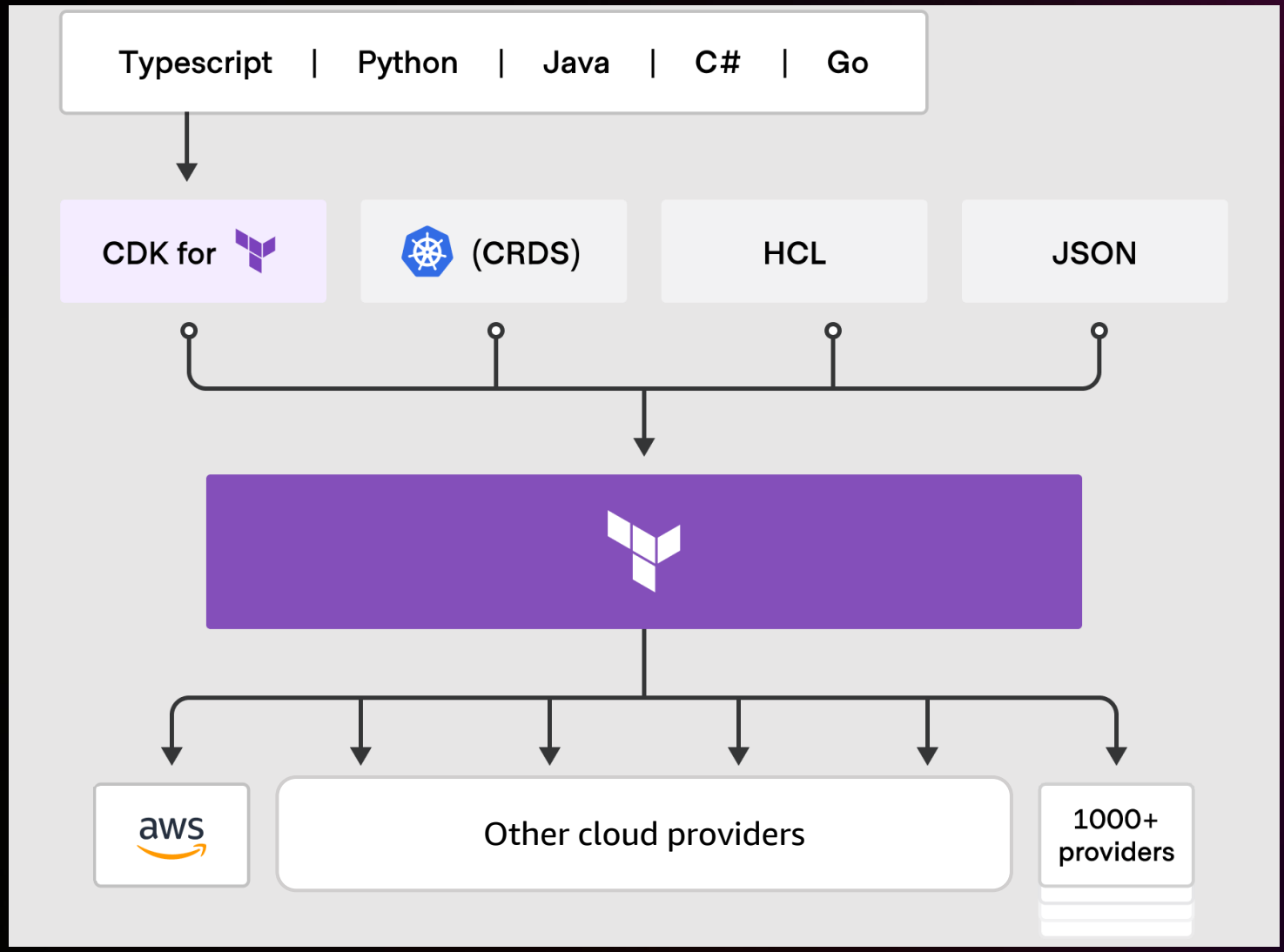
AWS CDK refresher




```
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
```

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

Terraform support in AWS CDK



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

Resources

- Source Code: github.com/tfworks
- Custom Resources Docs: docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/template-custom-resources.html
- Terraform Providers: developer.hashicorp.com/terraform/tutorials/providers

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

Wrap-up

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

Thank you!

Sohan Maheshwar
sohanm@amazon.com
LinkedIn.com/in/sohanmaheshwar

Martijn van Dongen
martijnvandongen@schubergphilis.com
bio.link/mvd



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

aws

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