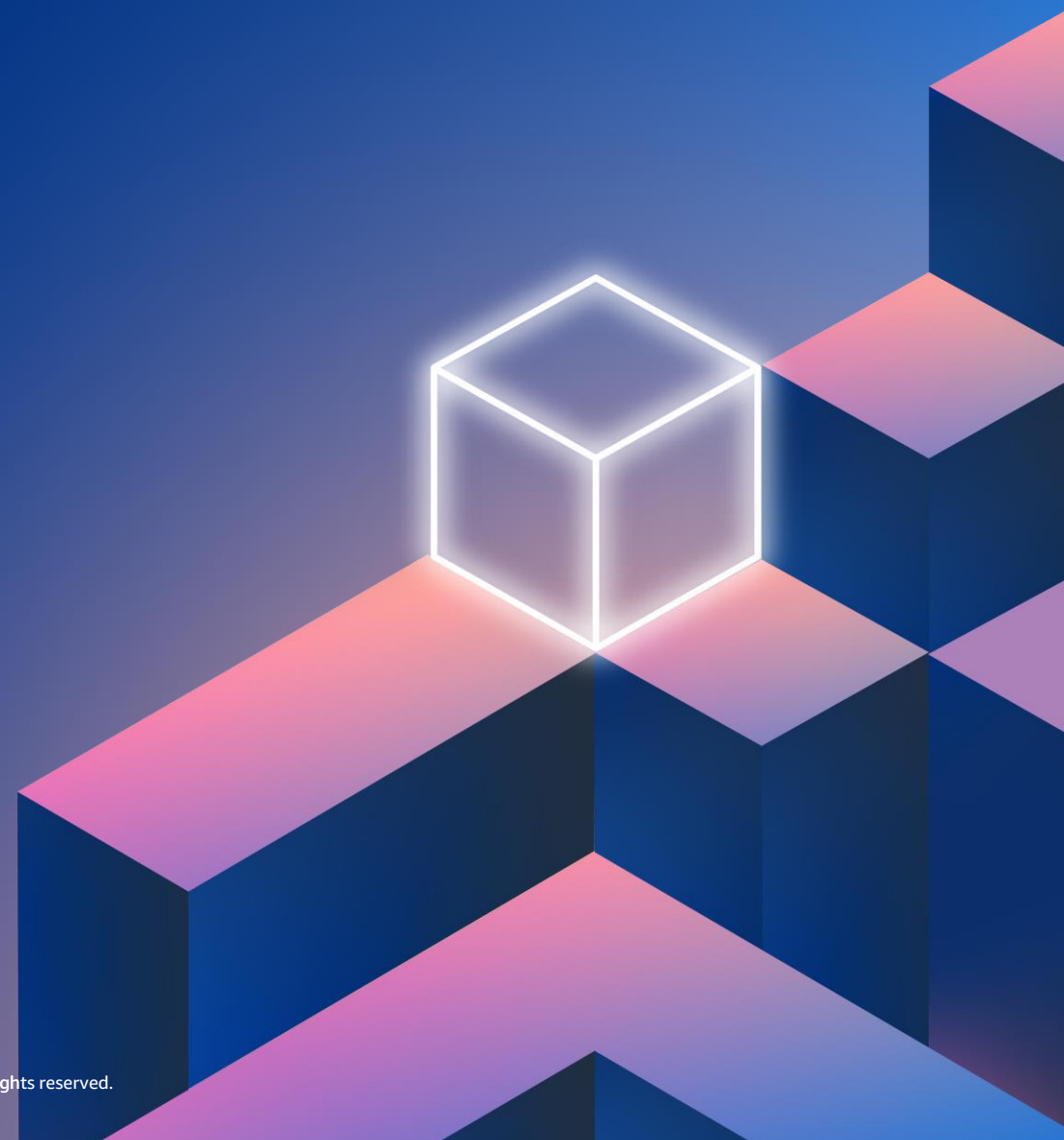# AWS Builders Online Series

Deploy Infrastructure as a Code on AWS

Nelli Lovchikova

Solutions Architect Manager
Amazon Web Services

aws

Agenda

1. Infrastructure as a code (IaC) overview

2. AWS CloudFormation

3. AWS Cloud Development Kit (AWS CDK)

4. Terraform with AWS CDK

5. Recap

6. Next steps – resources to get started

Challenge: Environmental Drift

- Each environment maintains individual settings
- Snowflake effect
- Causes by
  - Manual changes
  - Poor configuration
  - Application mischief

Challenge: Human Error

- Idempotence – same operation produces the same result
- Lost knowledge
- Emergency situations

Infrastructure as a code (IaC)
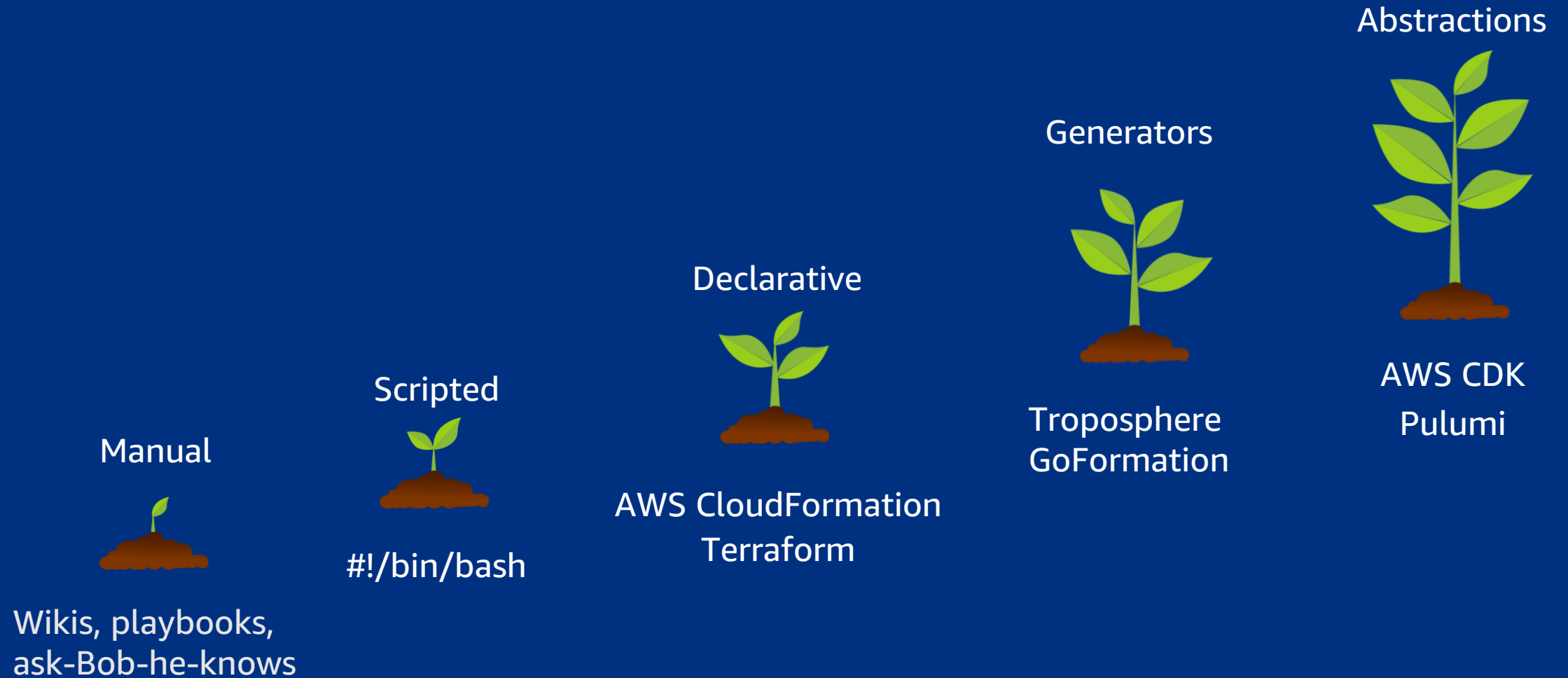

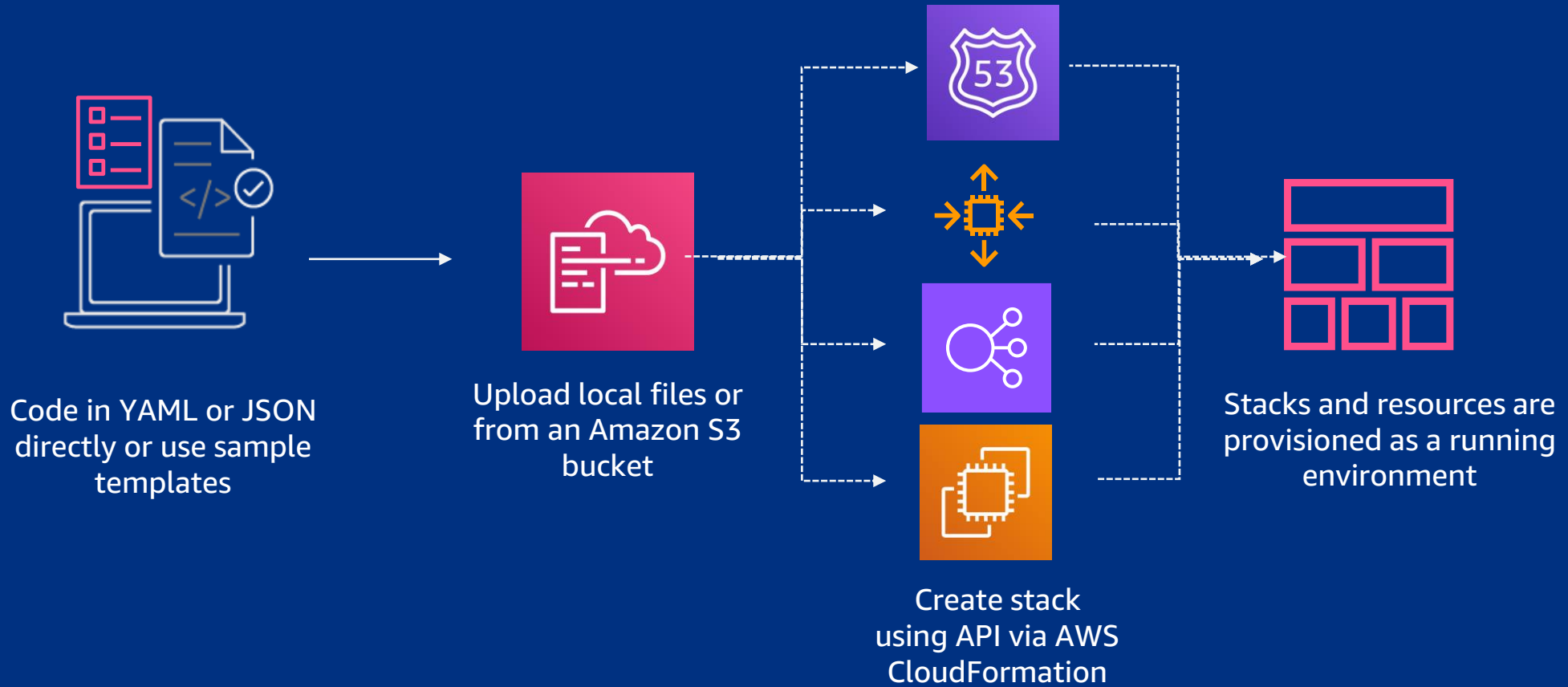
Consistent

Lower risk of human error

Lower cost

Self-documented

Evolution of Infrastructure as Code (IaC)

Abstractions

Generators

Declarative

Scripted

Manual

AWS CDK
Pulumi

Troposphere
GoFormation

AWS CloudFormation
Terraform

#!/bin/bash

Wikis, playbooks,
ask-Bob-he-knows

aws

AWS CloudFormation

Code in YAML or JSON directly or use sample templates

Upload local files or from an Amazon S3 bucket

Create stack using API via AWS CloudFormation

Stacks and resources are provisioned as a running environment

```yaml
AWSTemplateFormatVersion: 2010-09-09
Description: 'Serverless application for Innovate 2023'
Parameters:
    EnvName:
        Type: String
        Description: Name of the environment

Outputs:
  LambdaFunctionName:
    Value:
      Ref: Compute

Resources:
  Storage:
    Type: 'AWS::DynamoDB::Table'
    Properties:
      TableName: 'storage'
      AttributeDefinitions:
        - AttributeName: id
          AttributeType: S
      KeySchema:
        - AttributeName: id
          KeyType: HASH
      ProvisionedThroughput:
        ReadCapacityUnits: 2
        WriteCapacityUnits: 2
```

OUTLINE

- AWSTemplateForma...
- Description Serverle...
- {} Parameters
- {} Outputs
- {} Resources
  - {} Storage
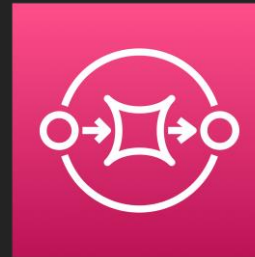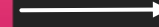 - {} Compute
  - {} ComputeRole

AWS Cloud Development Kit (CDK)

App

Stack (s)

Construct

AWS Lambda

Amazon DynamoDB

Construct

Amazon SQS

Amazon S3 Bucket

Development workflow

AWS CDK
Source Code

— execute →

AWS CDK
CLI

— synthesize →

Template and
assets

— deploy →

AWS
CloudFormation

— provision →

Cloud Resources
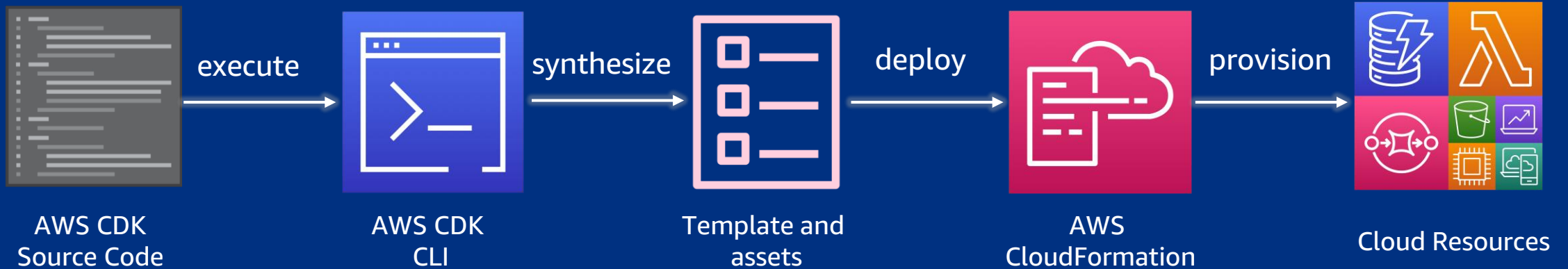
📓 **cdk init**   // **create new project**

🛠️ **npm run build**   // **build project**

🧬 **cdk synth**   // **create templates and assets**

🔍 **cdk diff**   // **check what will change**

🚀 **cdk deploy**   // **push changes to your account**

Construct levels

L3   Patterns

L2   AWS Constructs

L1   AWS CloudFormation resources

Level 1

```
new CfnBucket(this, 'MyBucket', {bucketName: 'my-bucket'})
```

cdk synth

```
Resources:
  myBucket:
    Type: AWS::S3::Bucket
    Properties:
      BucketName: my-bucket
```

- Generated mappings from CloudFormation Specification

- abc.CfnXyz → AWS::ABC::XYZ CloudFormation Resource

- ec2.CfnInstance → AWS::EC2::Instance

- kms.CfnKey → AWS::KMS::Key

Level 2

```
new ec2.Vpc(this, 'MyVPC')
```

cdk synth

- Ready-to-use VPC setup

- 65536 IPs split equally between 4 subnets

- If you provide a region → adjusted to 3 AZs

- Everything is optional, change any parameter

- Sane default values

Level 3

```
1    new ecs_patterns.ApplicationLoadBalancedFargateService(this, "FargateService", {
2      cluster,
3      taskImageOptions: {
4        image: ecs.ContainerImage.fromRegistry("amazon/amazon-ecs-sample");
5      }
6    });
```

829 line
AWS CloudFormation
template

AWS VPC

- Subnets
- EIP
- NAT gateways
- Internet gateway
- Route
- Route table

Elastic Load Balancer

- Security group
- Security group egress
- Security group ingress
- Task definition
- Listener
- Target group

AWS Fargate

- IAM roles
- IAM policies
- Log group
- Configuration

Amazon ECS task definition

- Image
- CPU
- Memory
- Port

"amazon-ecs-sample"
image

```
PS /Users/nul/Downloads/Innovate 2023/demo/demo2> cdk init --language typescript --app demo2
```

Terraform

Terraform   ←→   Terraform provider   ←→   Target API

Terraform configuration

A *Terraform configuration* is a complete document in the *Terraform language* that tells Terraform how to manage a given collection of infrastructure.

A configuration can consist of multiple files and directories.

Key concepts: Providers, resources, variables, outputs, and modules.

```hcl
terraform {
  required_providers {
    aws = {
      source  = "hashicorp/aws"
      version = "~> 4.16"
    }
  }

  required_version = ">= 1.2.0"
}


provider "aws" {
  region  = "us-west-2"
}


resource "aws_instance" "app_server" {
  ami           = "ami-830c94e3"
  instance_type = "t2.micro"

  tags = {
    Name = "ExampleAppServerInstance"
  }
}
```

AWS CDK supports Terraform
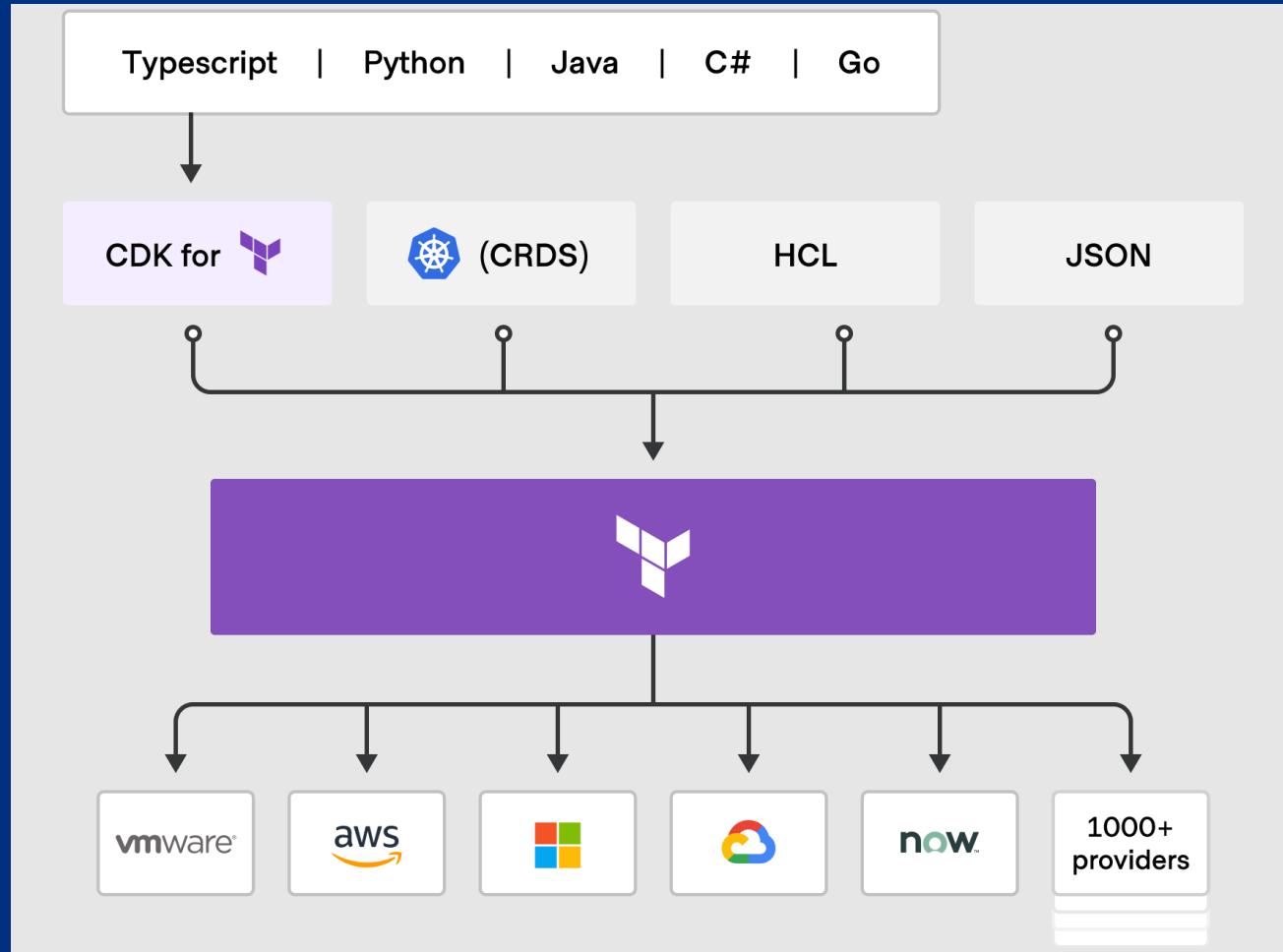
Conclusion

- Infrastructure as a code and it's evolution
- AWS CloudFormation
- AWS CDK – concepts, levels and abstractions
- Terraform

Next steps: Get hands-on

1. https://github.com/aws-cloudformation

2. https://github.com/aws/aws-cdk

3. https://github.com/aws-samples/aws-cdk-examples

4. https://cdkworkshop.com/

5. https://docs.aws.amazon.com/cdk/api/v2/docs/aws-construct-library.html

6. https://cdkworkshop.com/20-typescript/70-advanced-topics/200-pipelines/2000-create-repo.html

7. https://www.hashicorp.com/blog/cdk-for-terraform-now-generally-available

8. https://developer.hashicorp.com/terraform/tutorials/cdktf

Visit the AWS Modern Applications Resource Hub

Dive deeper with these resources to help you innovate fast, reduce risk, and accelerate time to market

- Build modern applications on AWS

- Building event-driven architectures on AWS

- Seamless Kubernetes on premises and in the cloud

- Unlock digital transformation by modernizing with containers

- Unleash the power of modern apps with generative AI on AWS

- Accelerate full-stack web and mobile app development on AWS

- Determining the total cost of ownership: Comparing serverless and server-based technologies

… and more!

https://tinyurl.com/modern-apps-aws

Visit resource hub

AWS Training & Certification

Access 600+ free digital courses with AWS Skill Builder

Focus on the cloud skills and services that are most relevant to you across 30+ AWS solutions, including digital self-paced learning plans and ramp-up guides

- Build your future in the AWS Cloud at your own pace
- Advance your skills and knowledge with learning plans
- Validate your cloud expertise with AWS Certification

https://skillbuilder.aws/

BUILD IN-DEMAND CLOUD SKILLS

Thank you for attending
AWS Builders Online Series

We hope you found it interesting! A kind reminder to **complete the survey.**
Let us know what you thought of today's event and how we can improve the event experience for you in the future.

aws-apj-marketing@amazon.com

twitter.com/AWSCloud

facebook.com/AmazonWebServices

youtube.com/user/AmazonWebServices

linkedin.com/company/amazon-web-services

twitch.tv/aws

Thank you!

Nelli Lovchikova

Solutions Architect Manager
Amazon Web Services

aws