Welcome to Week 2

AWS & Azure Academy 2024

PLURALSIGHT

Hello

About me...

- 27+ years in the industry
- 23+ years in teaching
- Certified Cloud architect
- Passionate about learning
- Also, passionate about Reese's Cups!

Agenda

- IaC using CloudFormation in AWS
- IaC using Azure Resource Manager (ARM) templates
- IaC using Cloud Development Kit (CDK) in AWS
- The 3 pillars of observability – logging, metrics, and tracing
- Monitoring & alerting in the Cloud
- Application resiliency patterns

How we're going to work together

- Slides and words to highlight key concepts
- Demos to bring those concepts "to life"
- Lab work (which will take place in sandboxes provided by "A Cloud Guru") for hands-on reinforcement
- NOTE: I welcome being interrupted – if you need more info, or clarification, or anything else, just break in and ask. I am here to help you.

Deployment Using IaC

IaC – What is it?

- As the name implies, the definition & configuration of our infrastructure IN code
- Instead of manually creating (inefficient) → automated in scripts that run “at the push of a button”

IaC – Why is it valuable?

- If only creating a handful of resources, manual is (probably) fine
- Creating hundreds (or even thousands), not so much!
- Modern DevOps is built around automation – quickly tearing down and rebuilding entire sets of infrastructure as and when required

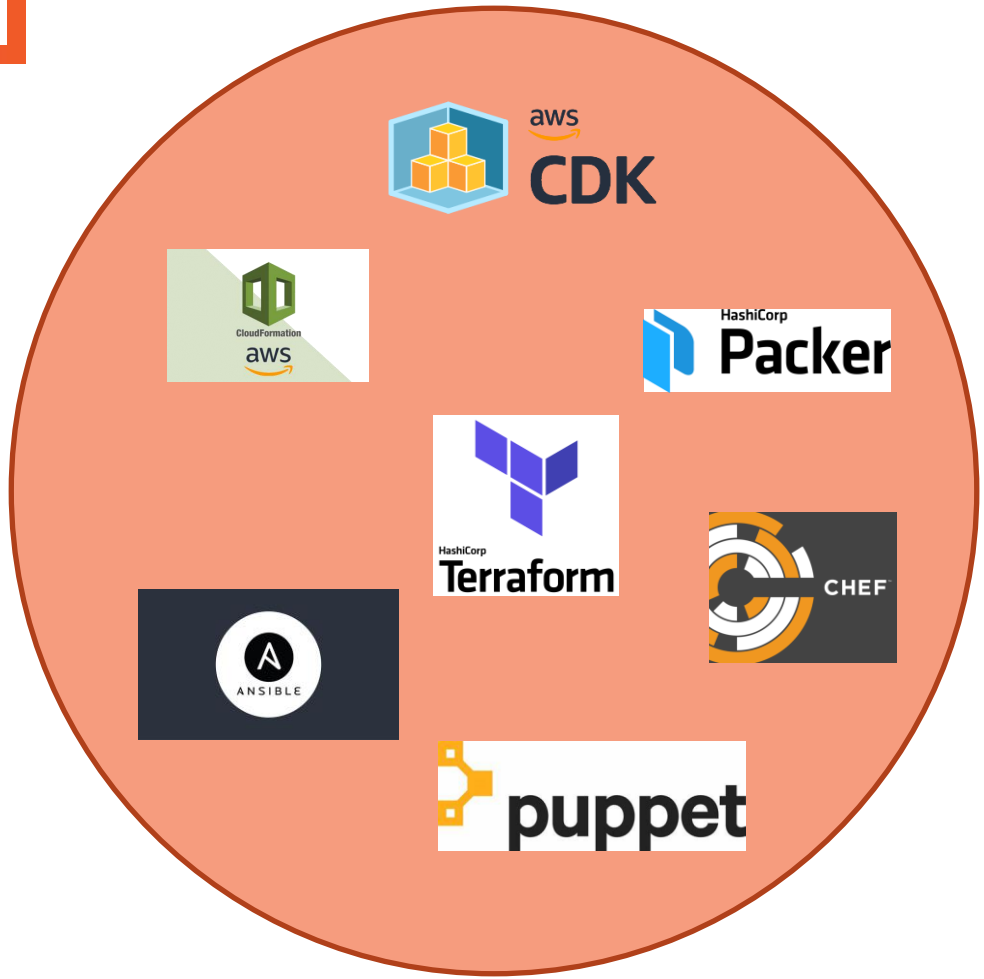
IaC – Advantages?

Testable

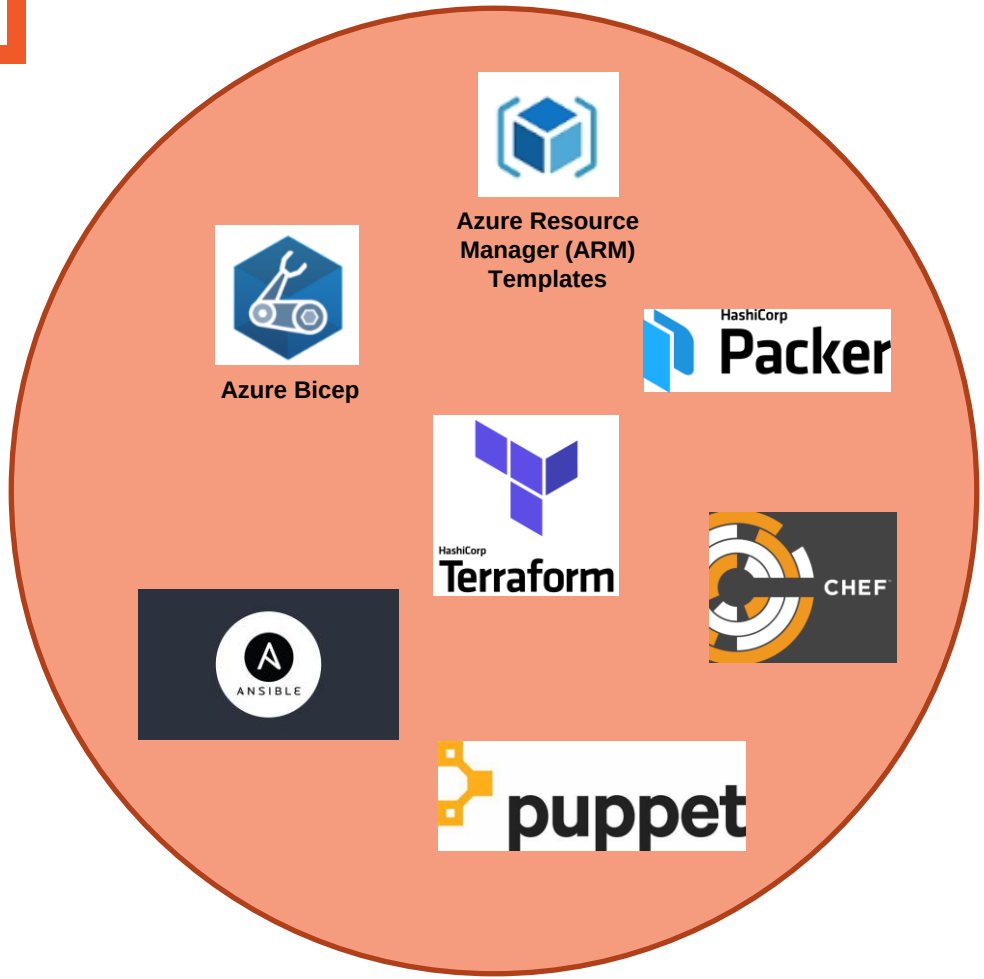
Repeatable

Auditable

IaC – AWS


aws
CDK
CloudFormation
aws
HashiCorp
Packer
HashiCorp
Terraform
CHEF
ANSIBLE
puppet

IaC – Azure


Azure Resource Manager (ARM) Templates
Azure Bicep
HashiCorp Packer
HashiCorp Terraform
CHEF
ANSIBLE
puppet

AWS CloudFormation

AWS CloudFormation

Works off 3 main concepts:

- Templates
- Stacks
- Change Sets

AWS CloudFormation

Works off 3 main concepts:

Formatted text files written in JSON or YAML that describe the “blueprint” for the AWS resources to be built

- Templates
- Stacks
- Change Sets

AWS CloudFormation

Works off 3 main concepts:

Templates

Stacks

Change Sets

A grouping of the complete set of resources provisioned by execution of a CloudFormation template

AWS CloudFormation

Works off 3 main concepts:

Templates

Stacks

Change Sets

Provides a summary of proposed changes that will be made to a set of running resources through execution of an updated template – before those updates are made

AWS CloudFormation

For additional concepts, see:

https://github.com/PacktPublishing/Mastering-AWS-CloudFormation/blob/master/Chapter2/core.yaml

https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/intrinsic-function-reference-foreach-example-outputs.html

DEMO/LAB:

AWS - n-Tier Deployment Using IaC

Execute the “Hands-On” lab available at https://github.com/KernelGamut32/aws_azure_academy_2024_public/tree/main/week02/labs/lab01

IaC with ARM Templates

Azure Resource Manager (ARM) Templates

- Microsoft's JSON-based IaC solution
- Supports definition of Cloud resources for Azure in code that adheres to a predefined schema

```
{
  "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json#",
  "languageVersion": "",
  "contentVersion": "",
  "apiProfile": "",
  "definitions": { },
  "parameters": { },
  "variables": { },
  "functions": [ ],
  "resources": [ ], /* or "resources": { } with languageVersion 2.0 */
  "outputs": { }
}
```

Source: https://learn.microsoft.com/en-us/azure/azure-resource-manager/templates/syntax

$schema Element

- Location of schema file that defines target version of template language
- Used to enforce rules around properties, hierarchy, and values applied to each
- Can be used to automate validation of a given ARM template instance

Source: https://learn.microsoft.com/en-us/azure/azure-resource-manager/templates/syntax

parameters Element

- Optionally allows specification of values that can be inputs at deployment time
- Limited to 256 parameters in a given template

```
"parameters": {
  "<parameter-name>" : {
    "type" : "<type-of-parameter-value>",
    "defaultValue": "<default-value-of-parameter>",
    "allowedValues": [ "<array-of-allowed-values>" ],
    "minValue": <minimum-value-for-int>,
    "maxValue": <maximum-value-for-int>,
    "minLength": <minimum-length-for-string-or-array>,
    "maxLength": <maximum-length-for-string-or-array>,
    "prefixItems": <schema-for-validating-array>,
    "items": <schema-for-validating-array-or-boolean>,
    "properties": <schema-for-validating-object>,
    "additionalProperties": <schema-for-validating-object-or-boolean>,
    "discriminator": <schema-to-apply>,
    "nullable": <boolean>,
    "metadata": {
      "description": "<description-of-the parameter>"
    }
  }
}
```

Source: https://learn.microsoft.com/en-us/azure/azure-resource-manager/templates/syntax

variables Element

- Optionally allows definition of variables that can be used throughout template – similar to variables used in application code
- Can help reduce complex expressions through reusability

```
"variables": {
  "<variable-name>": "<variable-value>",
  "<variable-name>": {
    <variable-complex-type-value>
  },
  "<variable-object-name>": {
    "copy": [
      {
        "name": "<name-of-array-property>",
        "count": <number-of-iterations>,
        "input": <object-or-value-to-repeat>
      }
    ]
  },
  "copy": [
    {
      "name": "<variable-array-name>",
      "count": <number-of-iterations>,
      "input": <object-or-value-to-repeat>
    }
  ]
}
```

Source: https://learn.microsoft.com/en-us/azure/azure-resource-manager/templates/syntax

functions Element

- Optionally allows definition of user-defined, custom functions
- Allows encapsulation of complex expressions/instructions that are callable by name

```
"functions": [
  {
    "namespace": "<namespace-for-functions>",
    "members": {
      "<function-name>": {
        "parameters": [
          {
            "name": "<parameter-name>",
            "type": "<type-of-parameter-value>"
          }
        ],
        "output": {
          "type": "<type-of-output-value>",
          "value": "<function-return-value>"
        }
      }
    }
  }
],
```

Source: https://learn.microsoft.com/en-us/azure/azure-resource-manager/templates/syntax

resources Element

- Required section of template that enables the definition of the actual resources to be deployed
- See source URL for additional info

Source: https://learn.microsoft.com/en-us/azure/azure-resource-manager/templates/syntax

outputs Element

- Defines values returned from a deployment (i.e., outputs from execution of the deployment)
- Can be used to return details from deployed resources (e.g., public DNS value)

```
"outputs": {
  "<output-name>": {
    "condition": "<boolean-value-whether-to-output-value>",
    "type": "<type-of-output-value>",
    "value": "<output-value-expression>",
    "copy": {
      "count": <number-of-iterations>,
      "input": <values-for-the-variable>
    }
  }
}
```

Source: https://learn.microsoft.com/en-us/azure/azure-resource-manager/templates/syntax

Resource Iteration

For additional concepts, see:

https://learn.microsoft.com/en-us/azure/azure-resource-manager/templates/copy-resources

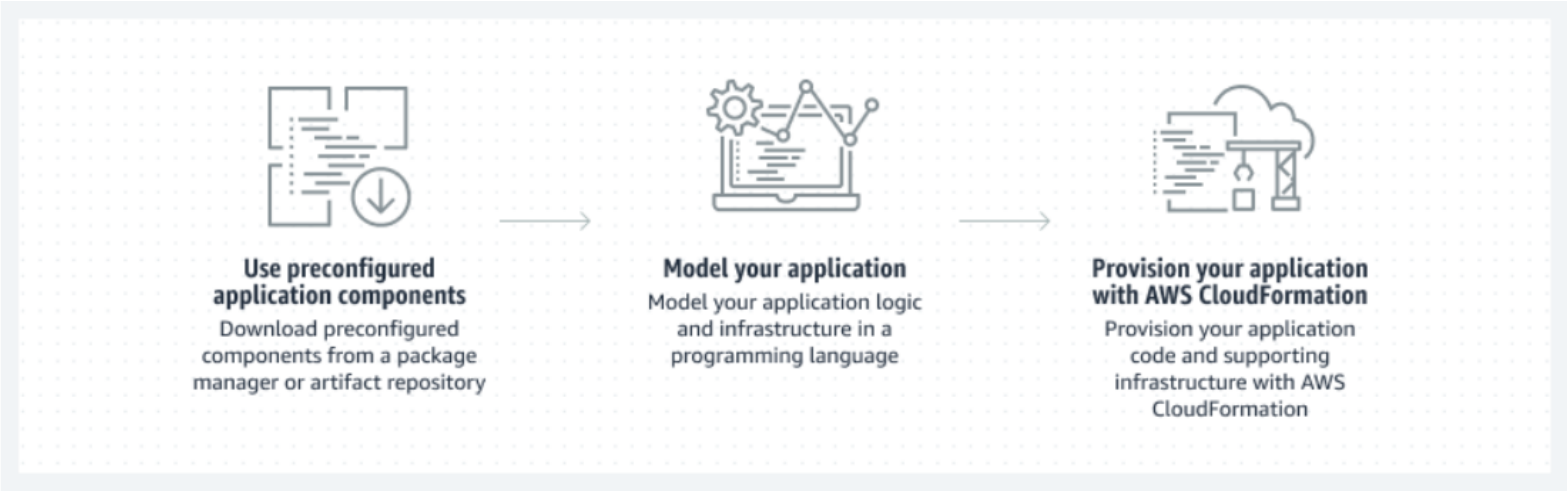
DEMO/LAB:

Azure - App Service Using ARM Template

Execute the “Hands-On” lab available at https://github.com/KernelGamut32/aws_azure_academy_2024_public/tree/main/week02/labs/lab02

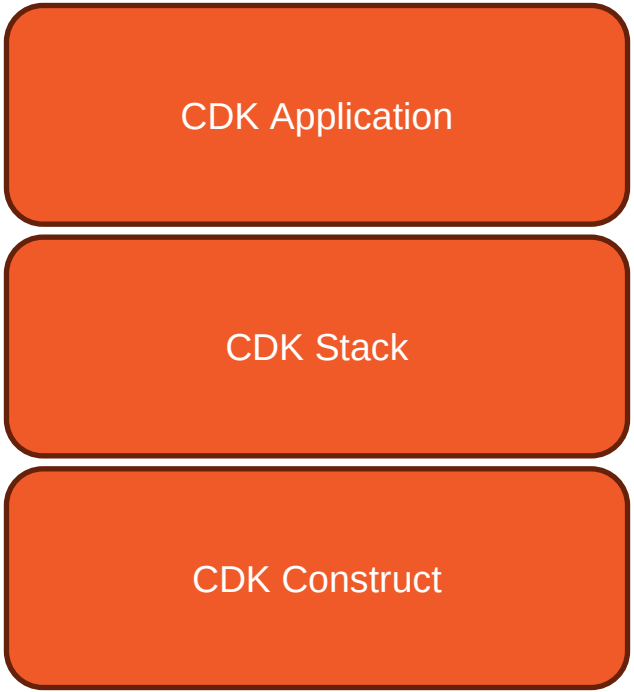
AWS Cloud Development Kit (CDK)

CDK – Key Concepts



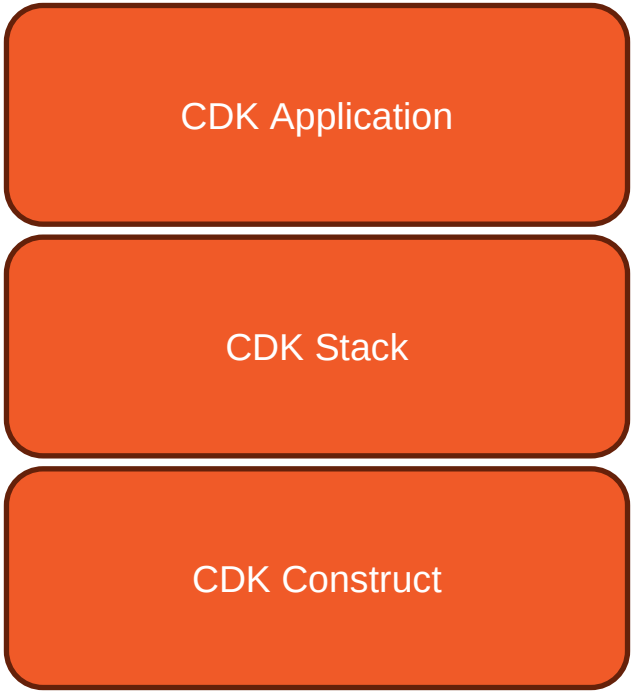
Source: https://aws.amazon.com/cdk/

CDK – Key Concepts



CDK – Key Concepts

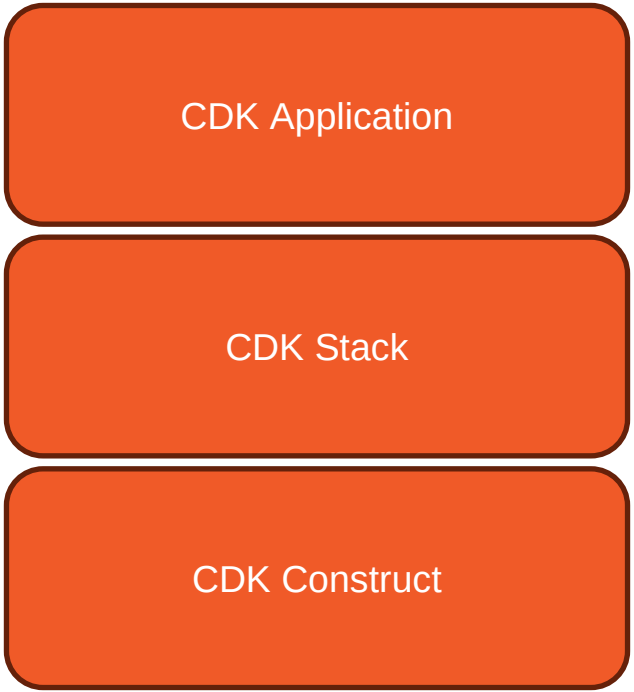
Describes the infrastructure to be built using a programming language



Built using TypeScript/JavaScript, Python, Java, C#, or Go

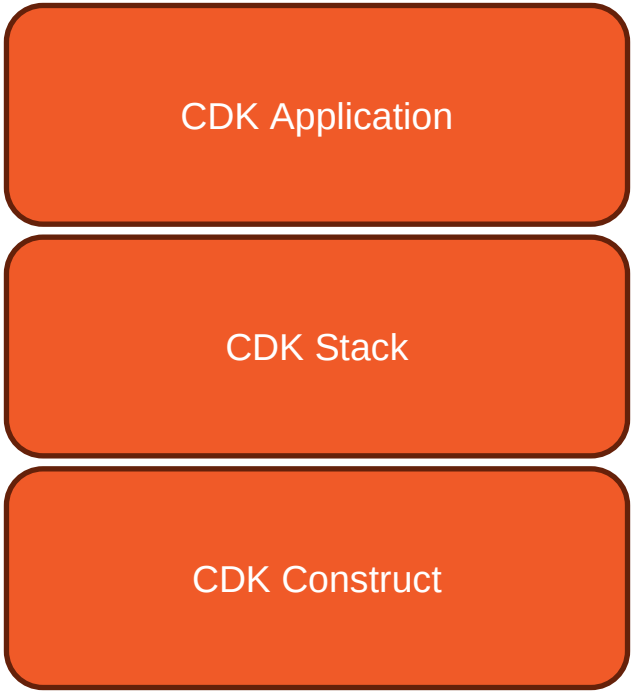
CDK – Key Concepts

Equivalent to a CloudFormation stack – a collection of related resources to be deployed



A process called “synthesis” is used to convert CDK stacks to CloudFormation templates

CDK – Key Concepts

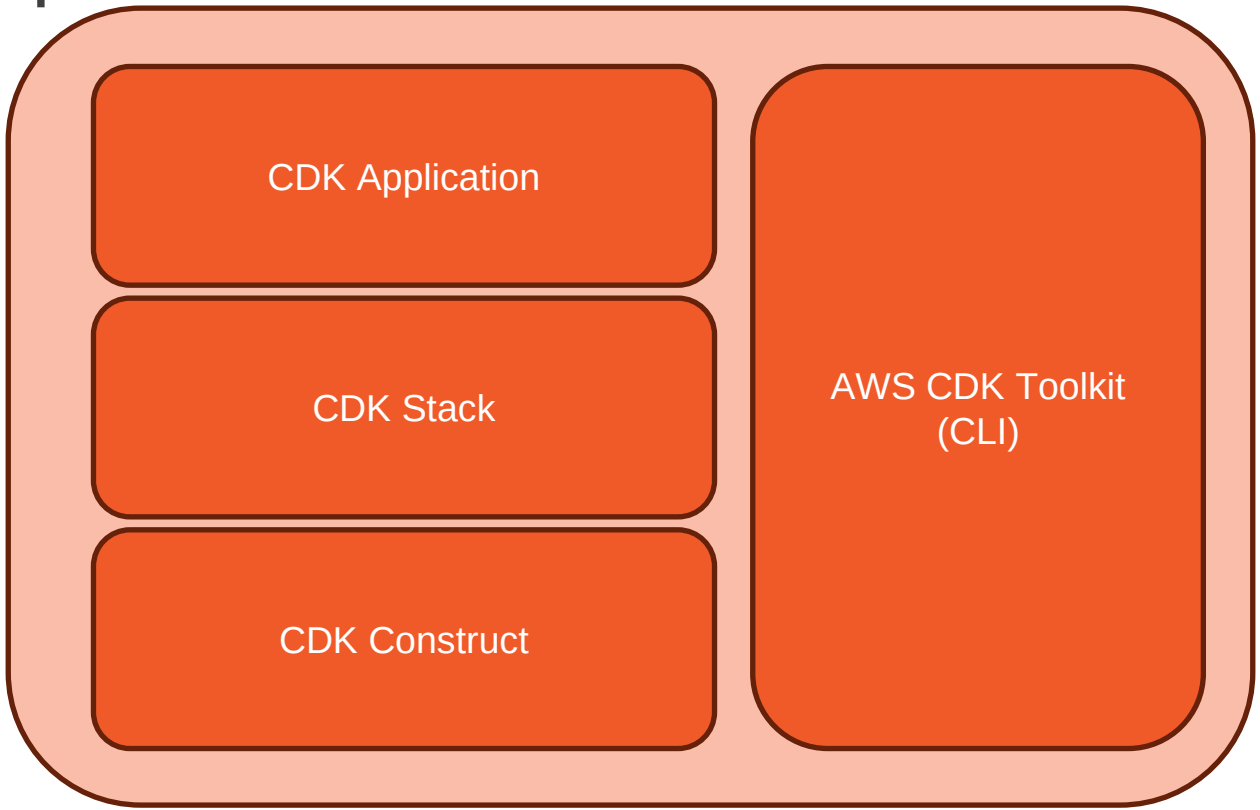


Representation of one or more Cloud resources categorized into multiple “levels”

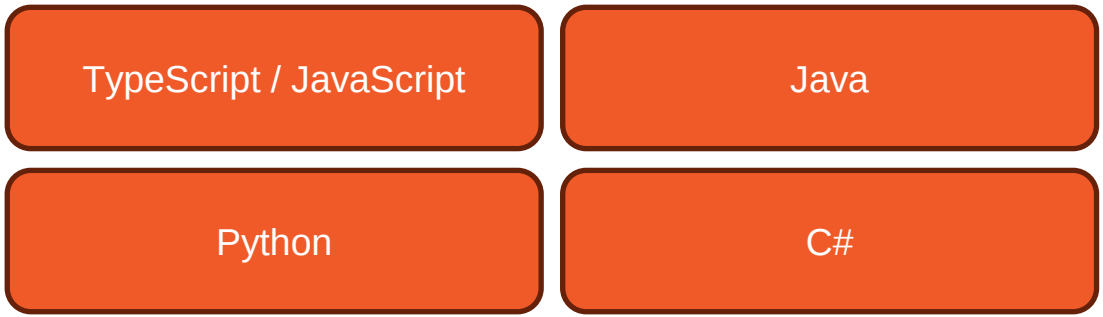
L1, L2, and L3

CDK – Key Concepts

Supports application bootstrapping, application synthesis, and deployment

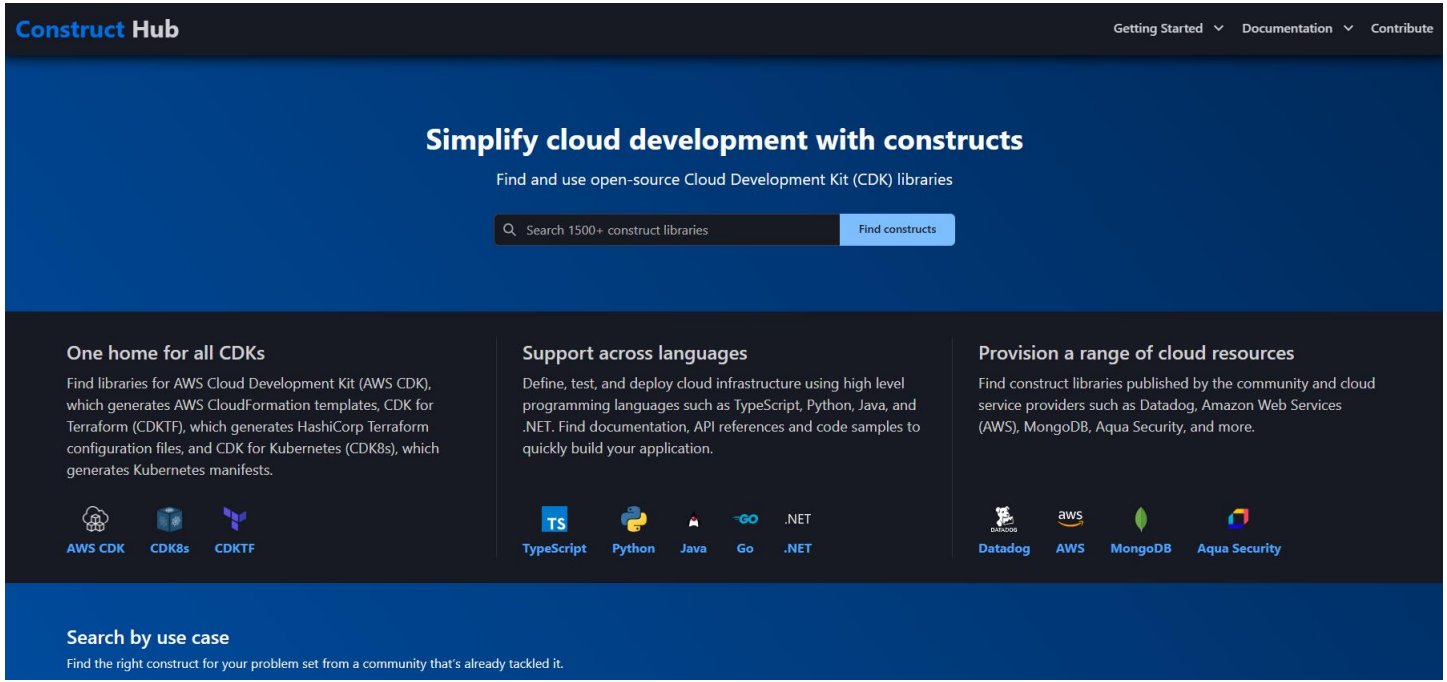


CDK – Languages Supported



See https://docs.aws.amazon.com/cdk/v2/guide/getting_started.html for more info

CDK – Construct Hub



See https://constructs.dev/ for more info

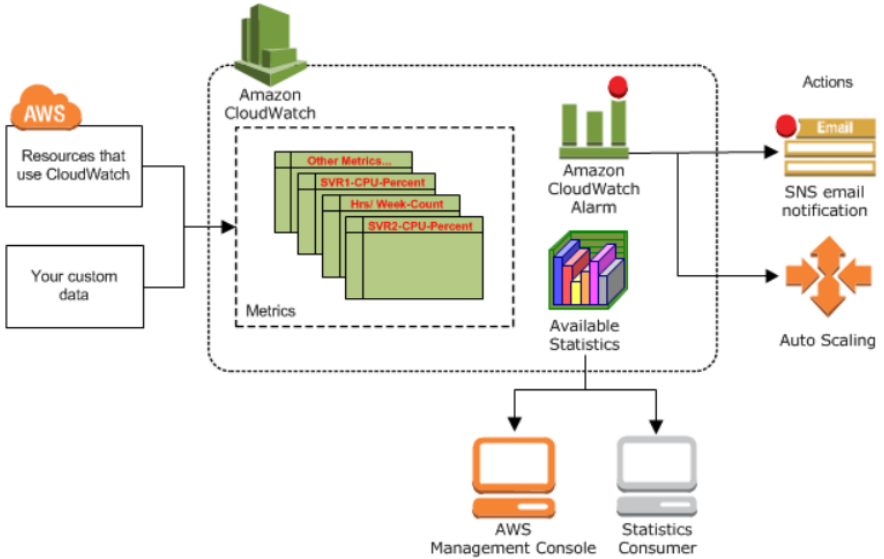
Operational Management in the Cloud

AWS Services

Amazon CloudWatch

Source: https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/WhatIsCloudWatch.html

Amazon CloudWatch – Architecture



Source: https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/cloudwatch_architecture.html

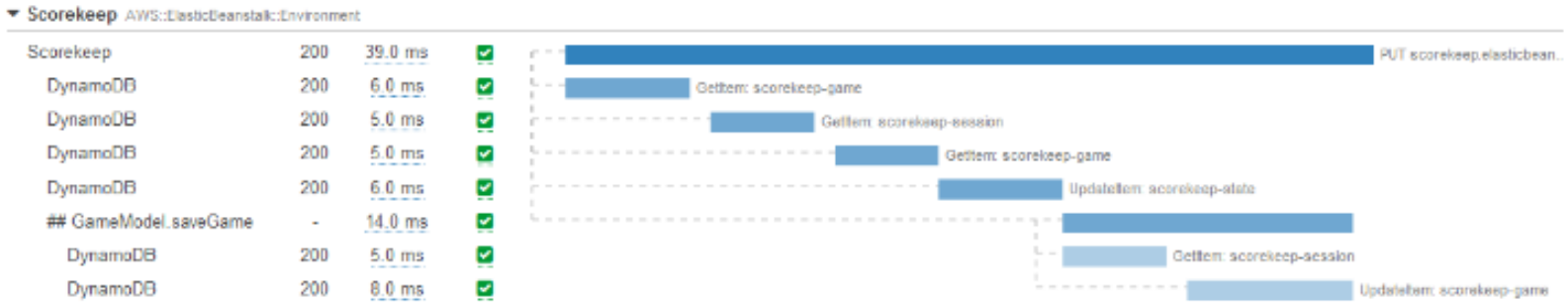
Amazon CloudWatch – Application Insights

Source: https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/appinsights-what-is.html

AWS CloudTrail

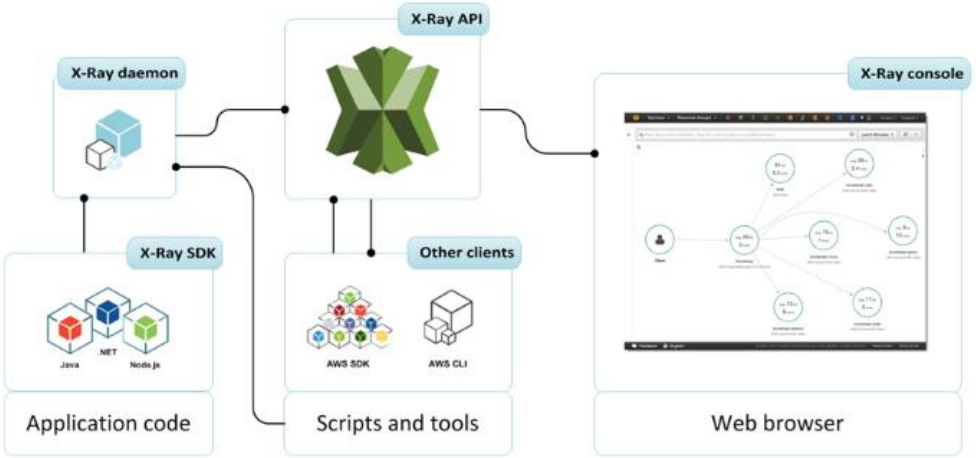
Source: https://docs.aws.amazon.com/awscloudtrail/latest/userguide/cloudtrail-user-guide.html

AWS XRay



Source: https://docs.aws.amazon.com/xray/latest/devguide/aws-xray.html

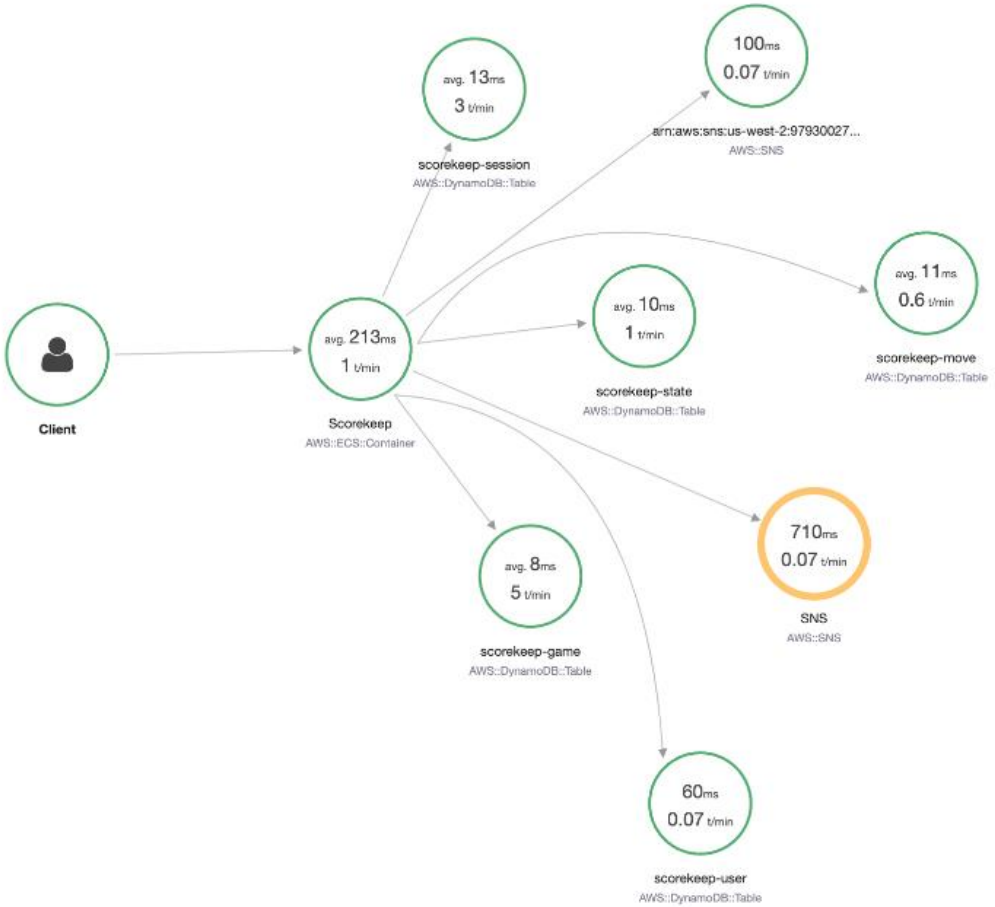
AWS XRay



Source: https://docs.aws.amazon.com/xray/latest/devguide/aws-xray.html

AWS XRay

Source:
https://docs.aws.amazon.com/xray/latest/devguide/aws-xray.html



DEMO/LAB:

AWS - VPC Flow Logs

Execute the “Hands-On” lab available at
https://github.com/KernelGamut32/aws_azure_academy_2024_public/tree/main/week02/labs/lab03

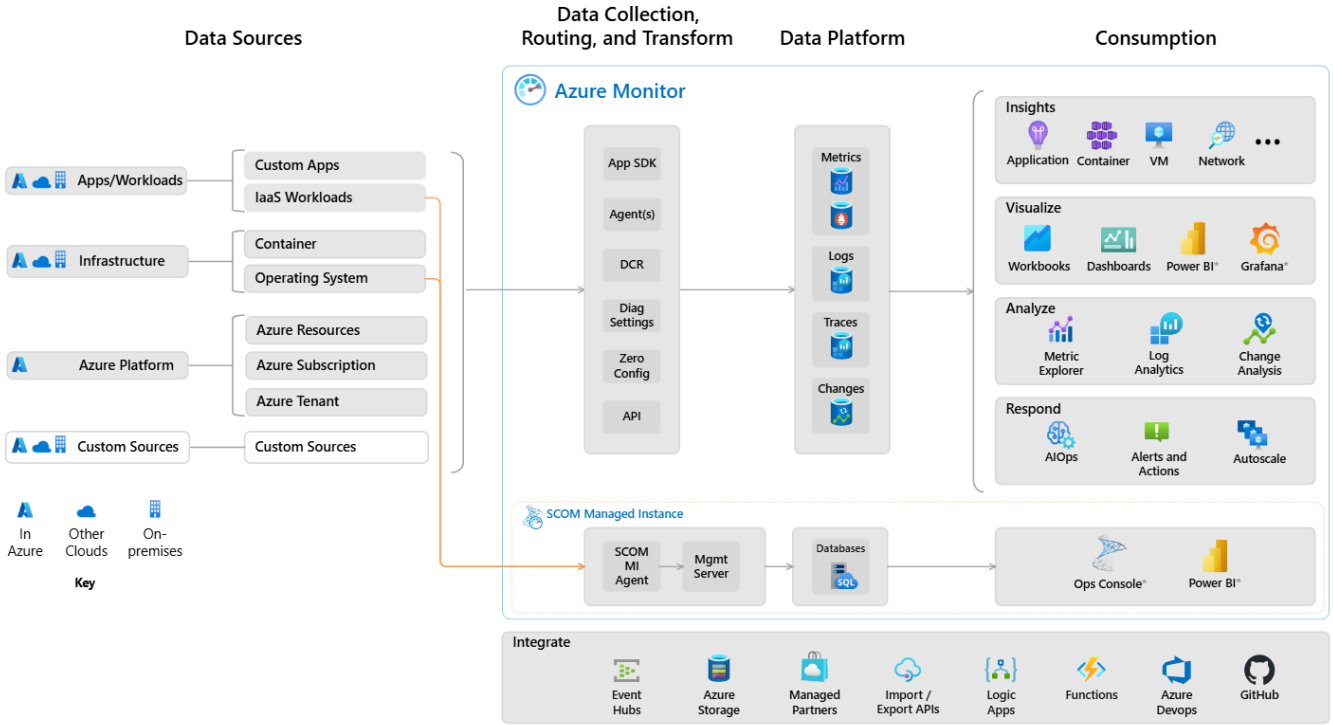
DEMO/LAB:

AWS - Troubleshooting Serverless

Execute the “Hands-On” lab available at https://github.com/KernelGamut32/aws_azure_academy_2024_public/tree/main/week02/labs/lab04

Azure Services

Azure Monitor



Source: https://learn.microsoft.com/en-us/azure/azure-monitor/overview

AWS CloudWatch Dashboard

DEMO/LAB:

AWS - CloudWatch Dashboards

Execute the “Hands-On” lab available at https://github.com/KernelGamut32/aws_azure_academy_2024_public/tree/main/week02/labs/lab05

Application Resiliency Patterns

Application Resiliency Patterns

https://aws.amazon.com/blogs/architecture/disaster-recovery-dr-architecture-on-aws-part-i-strategies-for-recovery-in-the-cloud/

https://aws.amazon.com/blogs/architecture/disaster-recovery-dr-architecture-on-aws-part-ii-backup-and-restore-with-rapid-recovery/

https://aws.amazon.com/blogs/architecture/disaster-recovery-dr-architecture-on-aws-part-iii-pilot-light-and-warm-standby/

https://aws.amazon.com/blogs/architecture/disaster-recovery-dr-architecture-on-aws-part-iv-multi-site-active-active/

https://aws.amazon.com/blogs/networking-and-content-delivery/creating-disaster-recovery-mechanisms-using-amazon-route-53/

https://aws.amazon.com/blogs/compute/using-the-circuit-breaker-pattern-with-aws-step-functions-and-amazon-dynamodb/

https://www.beabetterdev.com/2021/10/01/aws-api-gateway-request-throttling/

Knowledge Check

Question #1

Q: This hosting option provides a “function” hosted in the Cloud that can be used to expose functionality using multiple languages, can be triggered in multiple ways, and provides a consumption-based costing model. Your choice?

1. IaaS
2. PaaS
3. Serverless
4. SaaS

Enter your answer in the chat…

Question #2

Q: This hosting option requires a higher degree of operational management on the part of the organization that chooses to use it for hosting resources in the Cloud. Your choice?

1. IaaS
2. PaaS
3. Serverless
4. SaaS

Enter your answer in the chat…

Question #3

Q: Which of the following is one of the Compute options available to you in AWS?

1. Function App
2. EC2
3. VPC
4. DynamoDB

Enter your answer in the chat…

Question #4

Q: Which of the following is NOT one of the Database options available to you in Azure?

1. Azure Cosmos DB
2. SQL Databases
3. SQL managed instances
4. Storage accounts

Enter your answer in the chat…

Question #5

Q: Which of the following is an advantage of IaC (Infrastructure-as-Code)?

1. Testability
2. Repeatability
3. Auditability
4. All of the above

Enter your answer in the chat…

Question #6

Q: This IaC tool in AWS uses YAML or JSON to define a template for infrastructure definition/configuration that can be “pushed” to AWS to automatically create resources. Your choice?

1. ARM Templates
2. CloudFormation
3. CDK
4. Athena

Enter your answer in the chat…

Question #7

Q: This IaC tool in Azure uses JSON to define a template for infrastructure definition/configuration that can be “pushed” to Azure to automatically create resources. Your choice?

1. ARM Templates
2. CloudFormation
3. CDK
4. Athena

Enter your answer in the chat…

Question #8

Q: This IaC tool in AWS allows you to use a “higher order” language (like Python or TypeScript) define a template for infrastructure definition/configuration that can be “pushed” to AWS to automatically create resources. Your choice?

1. ARM Templates
2. CloudFormation
3. CDK
4. Athena

Enter your answer in the chat…

Question #9

Q: Which of the following Application Resiliency patterns reviewed boasts lowest RPO/RTO but incurs largest cost?

1. Backup & Restore
2. Pilot Light
3. Warm Standby
4. Multi-Site Active/Active

Enter your answer in the chat…

Question #10

Q: Which of the following options can be used to log information about network traffic passing through your VPC including the ability to “pipe” that data to an S3 bucket for analytics or to a CloudWatch log for metrics, monitoring, and alarming?

1. Athena
2. IaC Generator
3. Flow Logs
4. Security Group

Enter your answer in the chat…

Question #11

Q: Which of the following AWS service offerings can be used to support distributed tracing of activity flowing through your Cloud components and their integrations (e.g., Lambda to DynamoDB table)?

1. Flow Logs
2. X-Ray Tracing
3. Application Insights
4. IaC Generator

Enter your answer in the chat…

Thank you!

If you have additional questions, please reach out to me at:
asanders@gamuttechnologysvcs.com

PLURALSIGHT