Introduction to Azure Infrastructure as Code

Carey Payette
@careypayette
Github: codingbandit

> Infrastructure as Code, also known as Programmable Infrastructure, is a process for managing computing and networking infrastructure using software development methodologies. These methodologies include version control, testing, continuous integration, and other practices.

Raka Mahesa – Author – Packt

Benefits of Infrastructure as Code (IaC)

- Full SDLC tooling (Source control, CI/CD)
- Infrastructure documentation and creation
- Correct environmental drift
- Deploy identical environments (dev/test/qa/prod)
- Environment consistency
- Speed
- History/Accountability
- Lower infrastructure management cost

What is the truth?

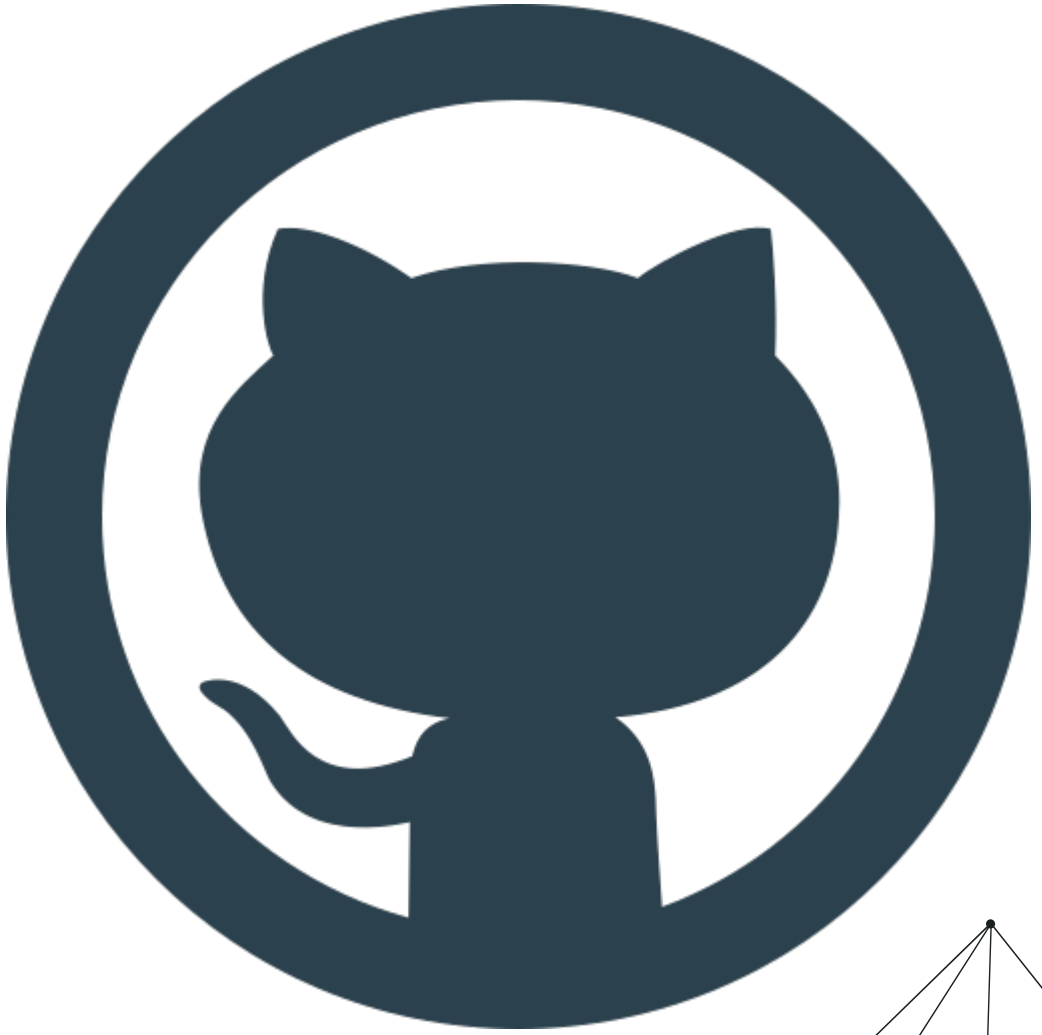
You CAN handle the truth!

- **Code** is the source of truth
- Not all infrastructure needs to be managed by code
- Code is the truth – **NOT** the deployment

Benefits of version control

For IaC code

- Version control / history
- Unit testing
- Code scans
- Policy checks/enforcement
- CI/CD
- Cost checks



Continuous Integration / Continuous Deployment

Build, test, and deployment automation

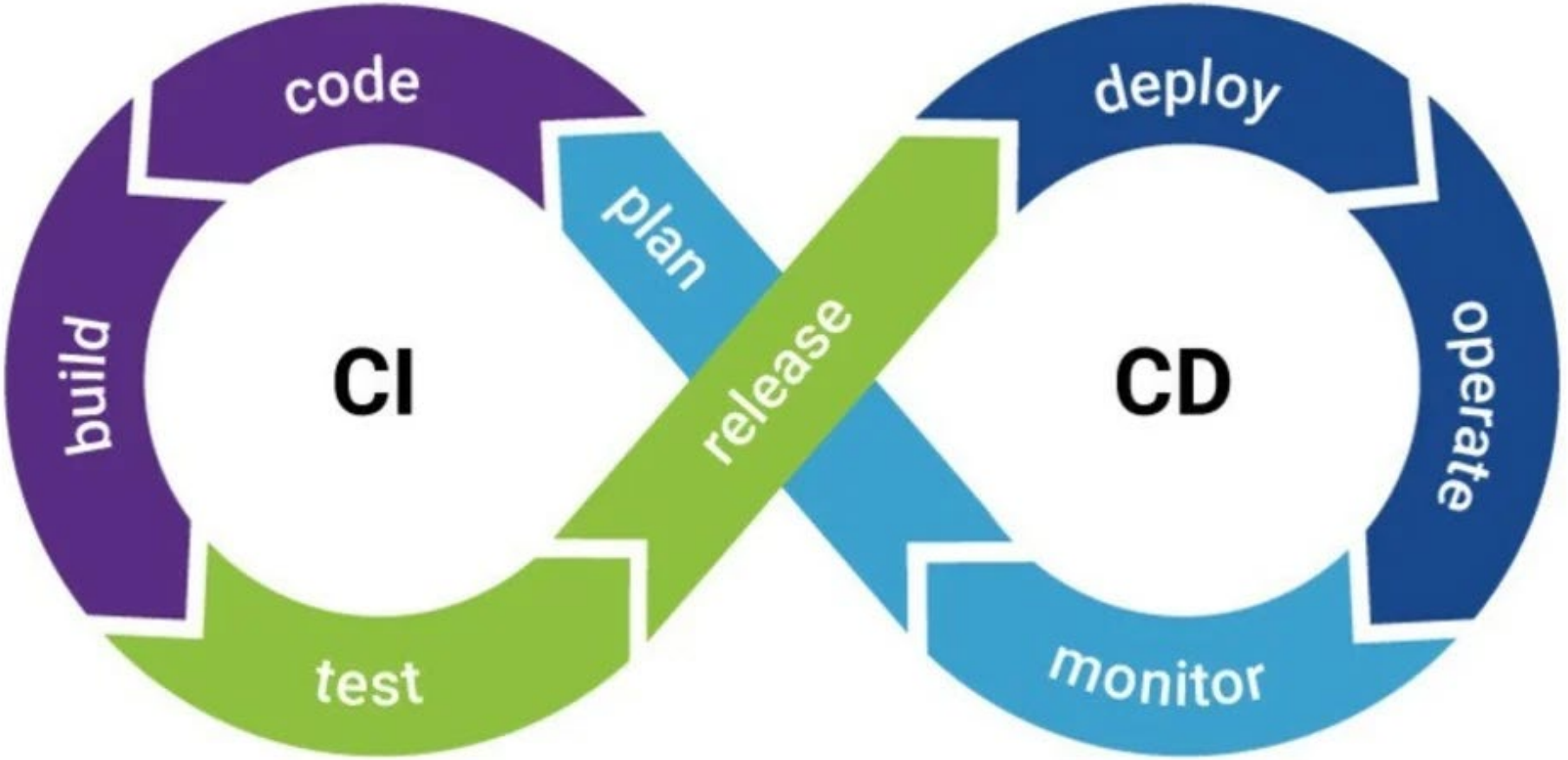


Image credit: https://dev.to/aws-builders/continuous-integration-and-delivery-ci-cd-using-aws-cdk-pipelines-with-bitbucket-4hc3

Azure CI/CD pipeline options

Sampling of automation products

- Azure Pipelines
- GitHub Actions
- Jenkins

Azure deployments

How do they work?

- Azure Resource Manager
- Azure Resource Providers

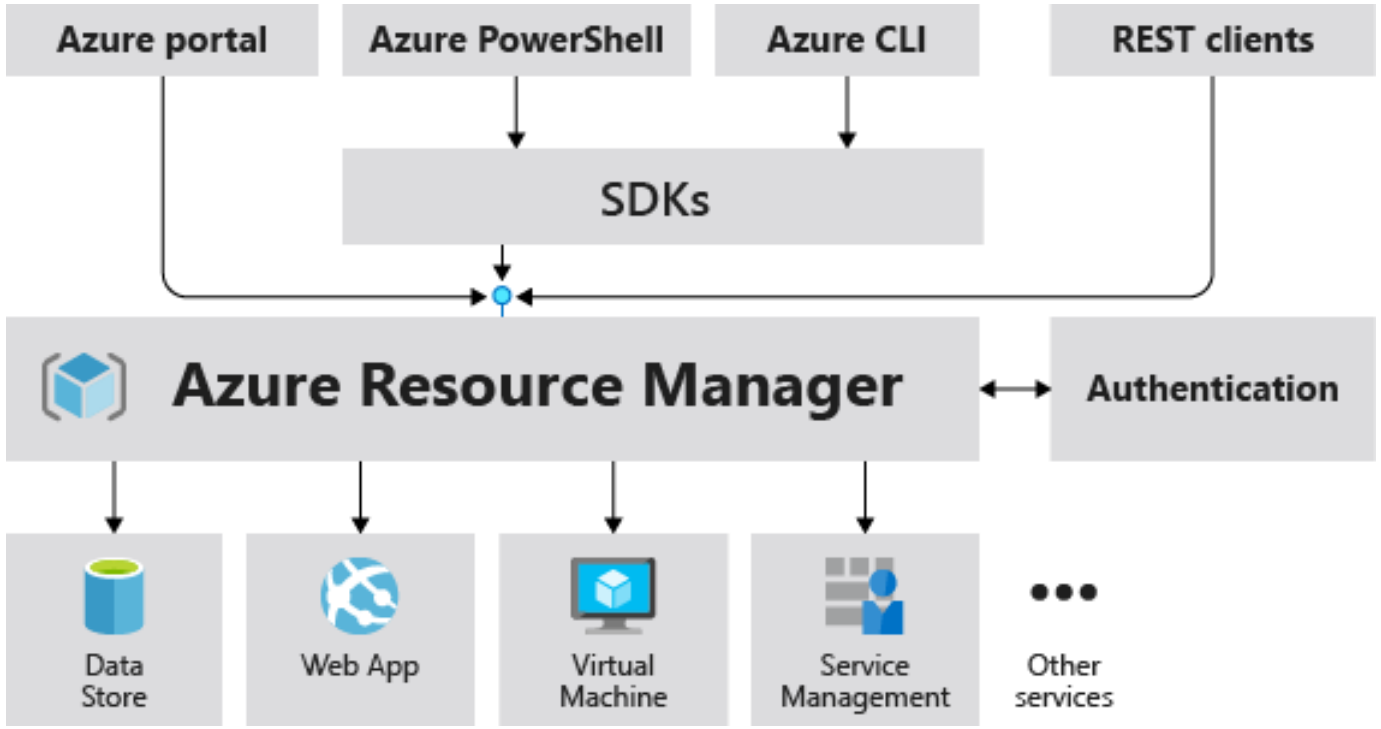


Image credit: https://learn.microsoft.com/en-us/azure/azure-resource-manager/management/overview

End state and Indempotency

Effect of applying and re-applying changes

- New resources
- Remove resources
- Change resources
- Dependencies

Overall guidance for getting started

Plan for success

- Start small
- Develop standards
- Reusability
- Adopt what is best for your organizational goals and skillsets

Infrastructure as Code Options

Declarative

- ARM
- CloudFormation
- Ansible
- Terraform

Imperative/Procedural

- CLI
- Postman Collections (REST)
- SDK
- Pulumi*

ARM templates

Declarative

- JSON
- Native building blocks
- Always up-to-date

BICEP

Declarative

- Domain specific language (DSL) for Azure Infrastructure
- Azure Only
- New kid on the block

```
param name string = 'vanillastaticwebappbicep'
param location string = 'East US 2'
param sku string = 'Free'
param skucode string = 'Free'
param repositoryUrl string = 'https://github.com/codingbandit/iac-static-web-app.git'
param branch string = 'main'

@secure()
param repositoryToken string
param appLocation string = '/'
param apiLocation string = ''
param resourceTags object = {
  Environment: 'Development'
  Project: 'Testing SWA with Bicep'
  ApplicationName: 'vanillastaticwebapp'
}
param appSettings object = {
  MY_APP_SETTING1: 'value 1'
  MY_APP_SETTING2: 'value 2'
}

resource name_resource 'Microsoft.Web/staticSites@2021-01-15' = {
  name: name
  location: location
  tags: resourceTags
  properties: {
    repositoryUrl: repositoryUrl
    branch: branch
    repositoryToken: repositoryToken
    buildProperties: {
      appLocation: appLocation
      apiLocation: apiLocation
    }
  }
  sku: {
```

Ansible

Declarative

- YAML
- Playbooks
- Linux-based

```yaml
---
- name: Deploy Azure Web App from GitHub
  hosts: localhost
  connection: local
  gather_facts: no

  vars:
    resource_group_name: vanillawebappansible-rg
    app_name: vanillastaticwebappansible
    location: eastus2
    sku: S1
    is_linux: true
    github_repo: https://github.com/codingbandit/iac-static-web-app.git
    github_branch: main

  tasks:
    - name: Create resource group
      azure_rm_resourcegroup:
        name: "{{ resource_group_name }}"
        location: "{{ location }}"
      register: rg_result

    - name: Create Azure App Service Plan
      azure_rm_appserviceplan:
        resource_group: "{{ resource_group_name }}"
        name: "{{ app_name }}-plan"
        location: "{{ location }}"
        sku: "{{ sku }}"
        is_linux: "{{ is_linux }}"
      register: appserviceplan_result

    - name: Create Azure Web App
      azure_rm_webapp:
        resource_group: "{{ resource_group_name }}"
        name: "{{ app_name }}"
        location: "{{ location }}"
        plan: "{{ app_name }}-plan"
        deployment_source:
          url: "{{ github_repo }}"
          branch: "{{ github_branch }}"
        scm_type: "GitHub"
        startup_file: "index.html"
      register: webapp_result
```

Terraform

Declarative

- HCL (HashiCorp Configuration Language)
- Strong Community support
- Great documentation

```
resource "azurerm_resource_group" "tfrg" {
  location = var.location
  name     = var.resource_group_name
  tags     = var.tags
}

resource "azurerm_static_web_app" "tfstaticsite" {
  name                = "terraformexample"
  resource_group_name = azurerm_resource_group.tfrg.name
  location            = var.location
  sku_tier = "Free"
  sku_size = "Free"
  tags = {
    "value1" : "value1value"
  }
}

output "static_site_key" {
  value = azurerm_static_web_app.tfstaticsite.api_key
  sensitive = true
}
```

Pulumi

Procedural

- Multi-language support
  - Node.js (JavaScript, TypeScript)
  - Python
  - Go
  - .NET (C#, F#, VB.NET)
  - Java
  - YAML

```
using Pulumi.AzureNative.Resources;
using Pulumi.AzureNative.Web;
using Pulumi.AzureNative.Web.Inputs;

return await Pulumi.Deployment.RunAsync(() =>
{
    // Create an Azure Resource Group
    var resourceGroup = new ResourceGroup("vanillawebapppulumi-rg", new ResourceGroupArgs
                        {
                            Location = "EastUS2"
                        });

    // Create an Azure Static Web App
    var staticWebApp = new StaticSite("vanillawebapppulumi", new StaticSiteArgs{
        Branch = "main",
        RepositoryUrl = "https://github.com/codingbandit/iac-static-web-app",
        ResourceGroupName = resourceGroup.Name,
        Location = resourceGroup.Location,
        RepositoryToken = "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",
        BuildProperties = new StaticSiteBuildPropertiesArgs{
            AppLocation = "/",
            ApiLocation = "",
            AppArtifactLocation = ""
        },
        Sku = new SkuDescriptionArgs
            {
                Name = "Free",
                Tier = "Free",
            },
    });
});
```

DEMOS

Thank you!

Carey Payette
@careypayette
GitHub: codingbandit
cpayette@trilliuminnovations.com