



Session Scope



Session Scope



Provisioning Cloud Infrastructure:

- time consuming
- hard to repeat
- becomes inconsistent between environments



Infrastructure as Code (IaC) minimize this.



Reduces the need for multiple scripting language.



Infrastructure as Code (IaC)



Infrastructure as Code (IaC)



Automate



Version, Share, and Reusability



Configure Script (blueprint)



Popular IaC tools:













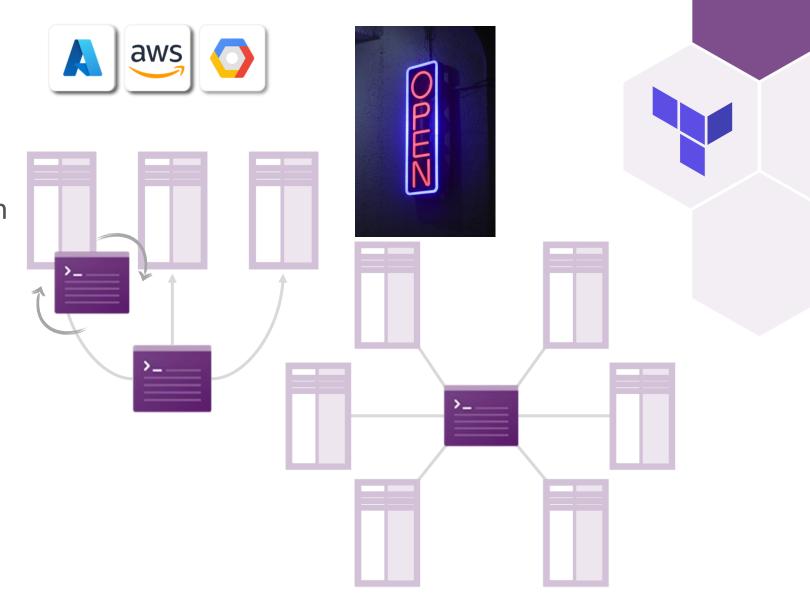


HashiCorp Terraform



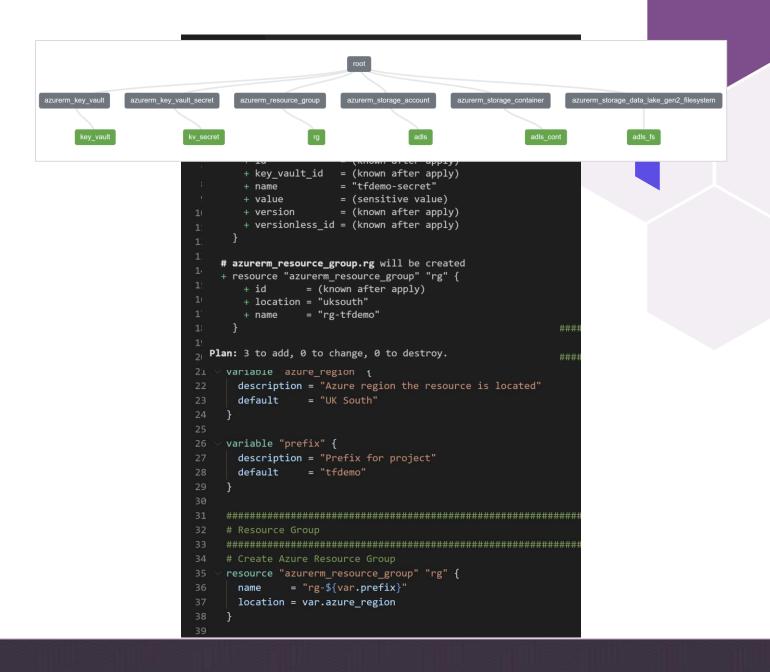
HashiCorp Terraform

- Cloud Agnostic
- Open Source
- Provisions, Manages and Version
- Consistent
- Reusable



Terraform Features

- Resource Graph
- Execution Plan
- Declarative Configure File
- HCL Terraform Language
- Supports JSON



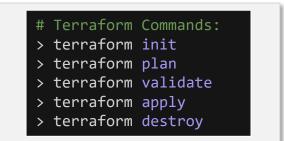


```
■ aa-repos > AppliedInfraAsCode > 
▼ main.tf > ...

   ## Configure the Azure Provider
   terraform {
    required_providers {
      azurerm = {
       source = "hashicorp/azurerm"
       version = "~> 2.90.0"
   provider "azurerm" {
    features {}
   # Variables
   variable "azure_region" {
    description = "Azure region the resource is located"
     default = "UK South"
   variable "prefix" {
     description = "Prefix for project"
     default = "tfdemo"
   # Resource Group
   # Create Azure Resource Group
   resource "azurerm_resource_group" "rg" {
          = "rg-${var.prefix}"
    location = var.azure_region
```



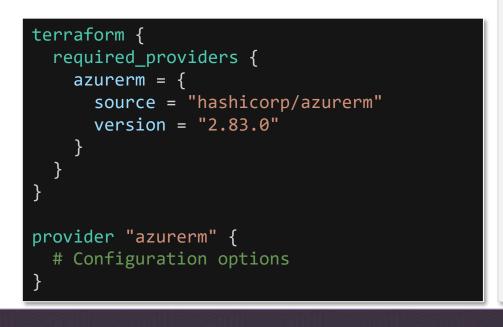


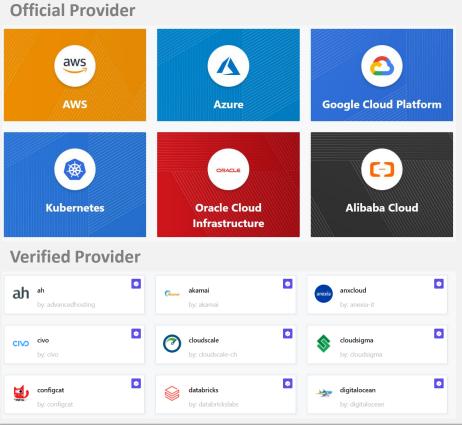




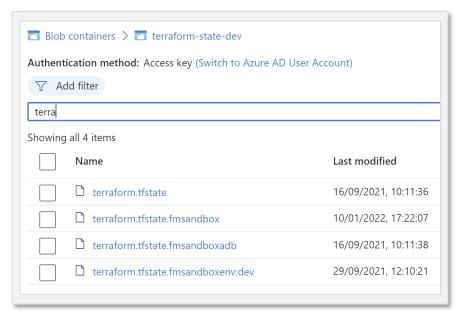


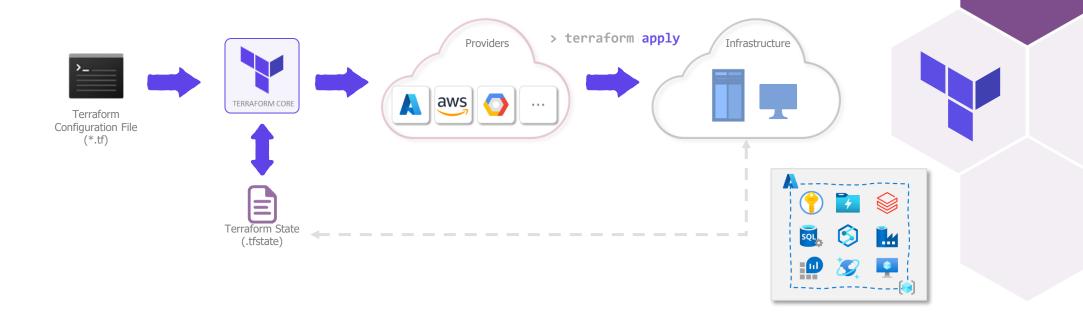




















Prepare









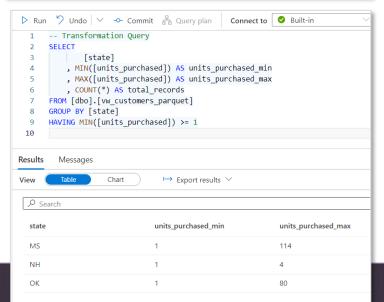




Prepare

Ingest











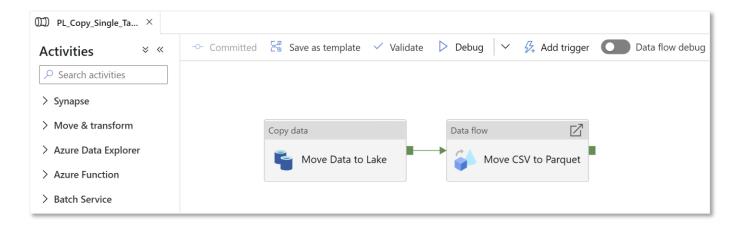




Ingest



Orchestrate







Unified Analytics Platform







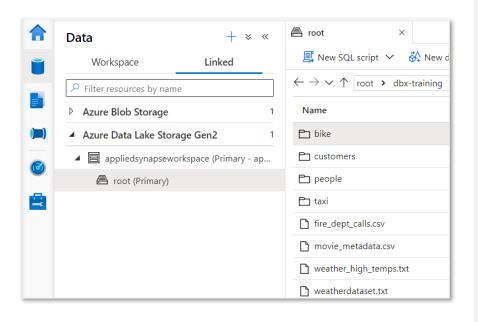
Ingest

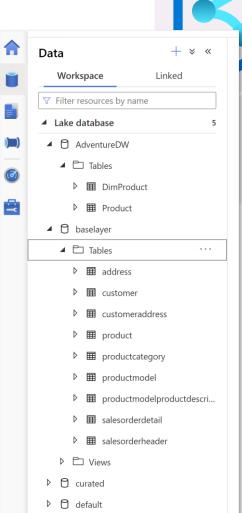






Explore







Unified Analytics Platform







Ingest



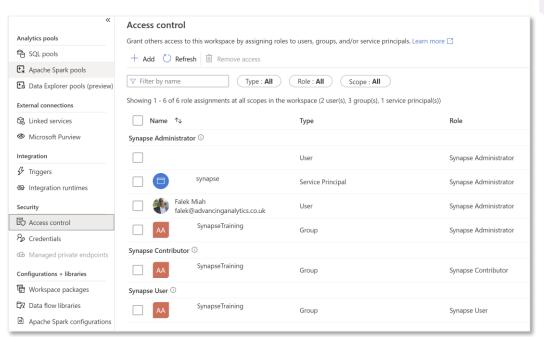
Orchestrate



Explore



Monitor and Control







Unified Analytics Platform







Ingest



Orchestrate

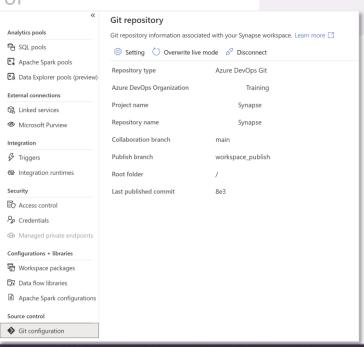


Explore

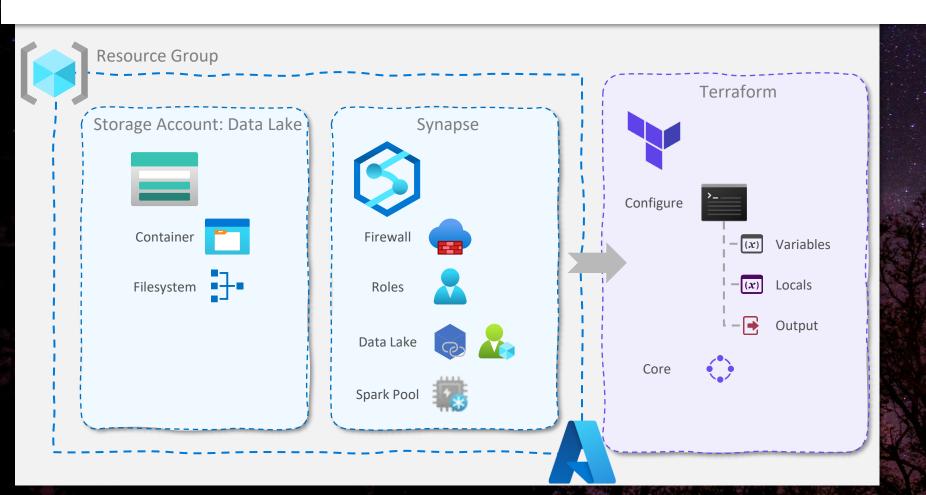


Monitor and Control





Demo



Using Terraform

- Azure Resource Group
- Storage Account
- Synapse Workspace

Execute Terraform

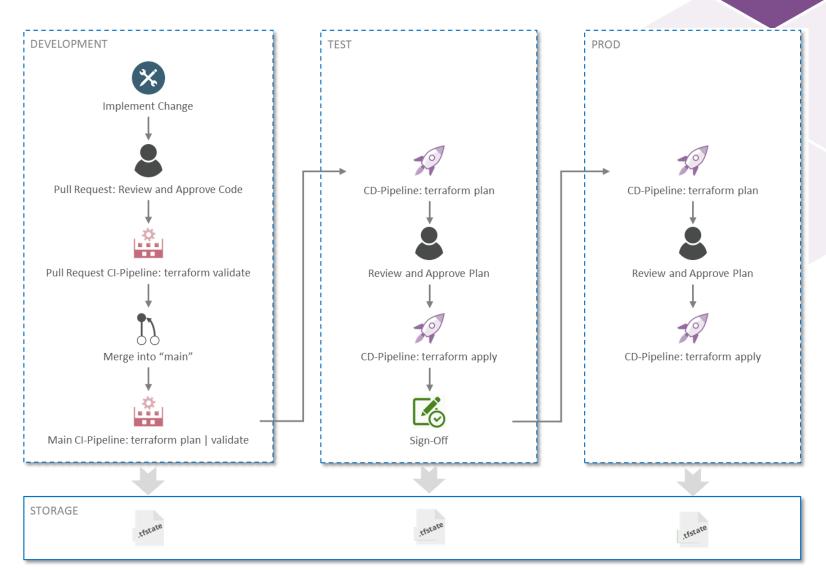


Deployment



Deployment

- Without manual intervention
- Without having explicit permissions
- CI/CD tool for release management
- Adopt GitOps approach



Deployment Tools









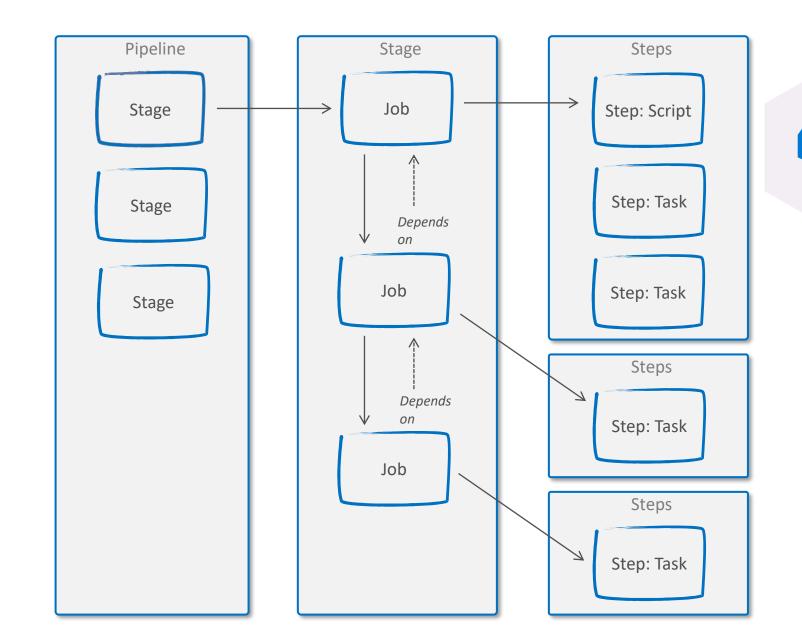




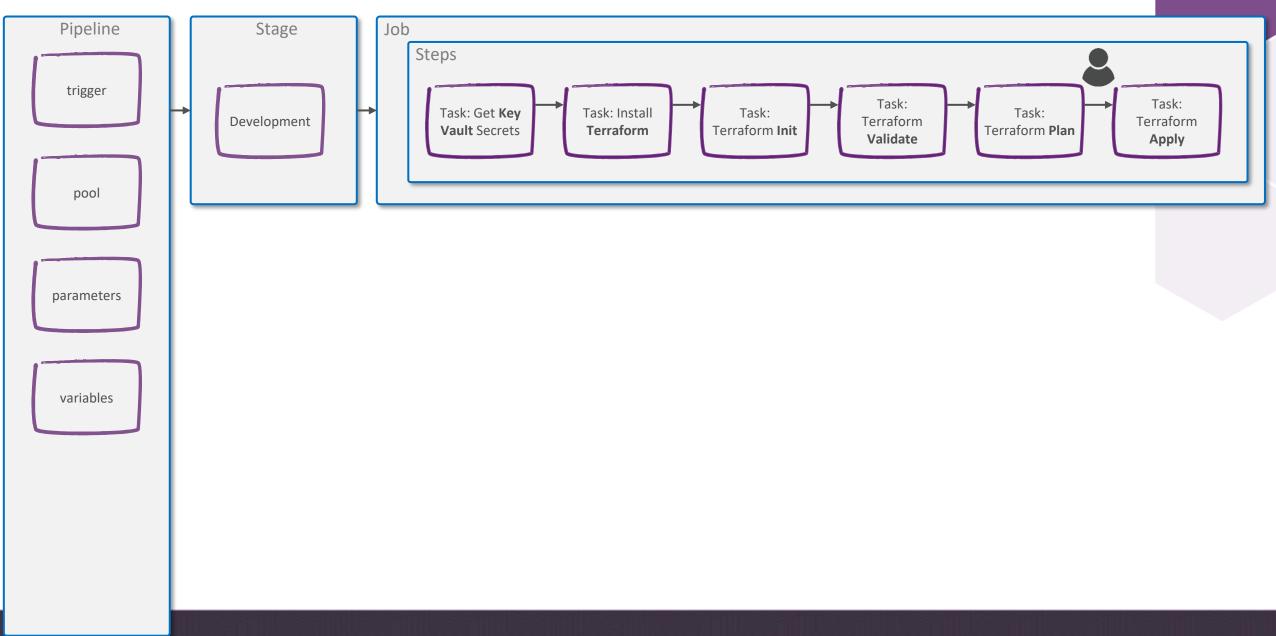
Azure DevOps

YML Pipelines consist of:

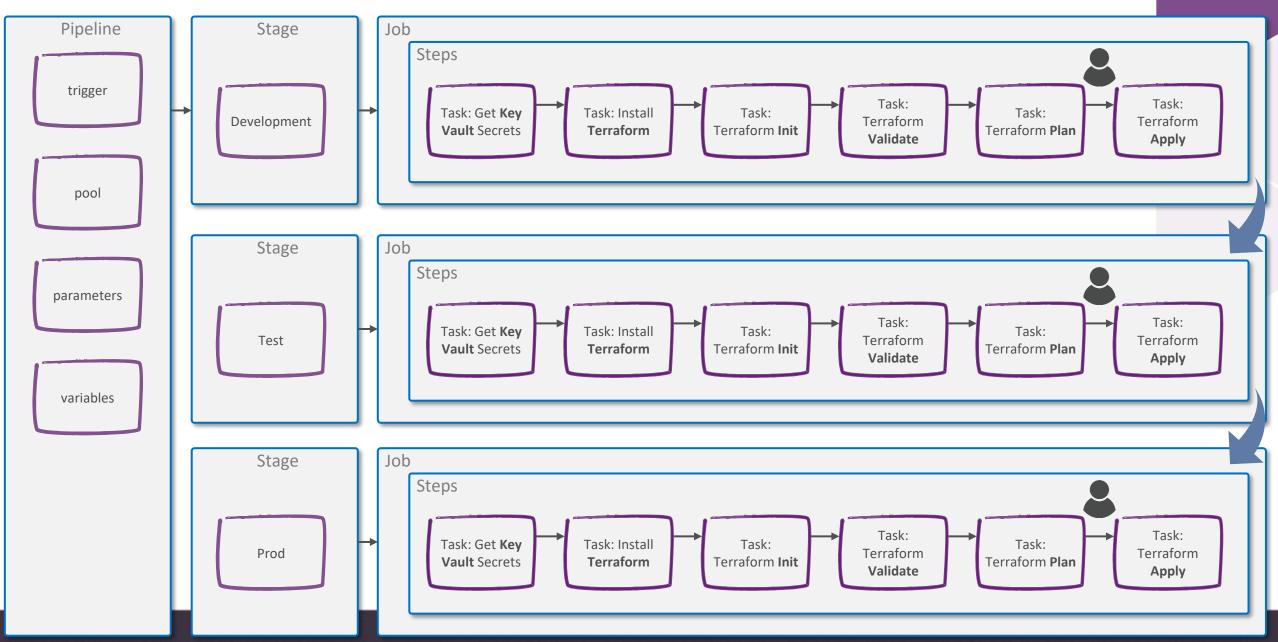
- Stages
- Jobs
- Steps



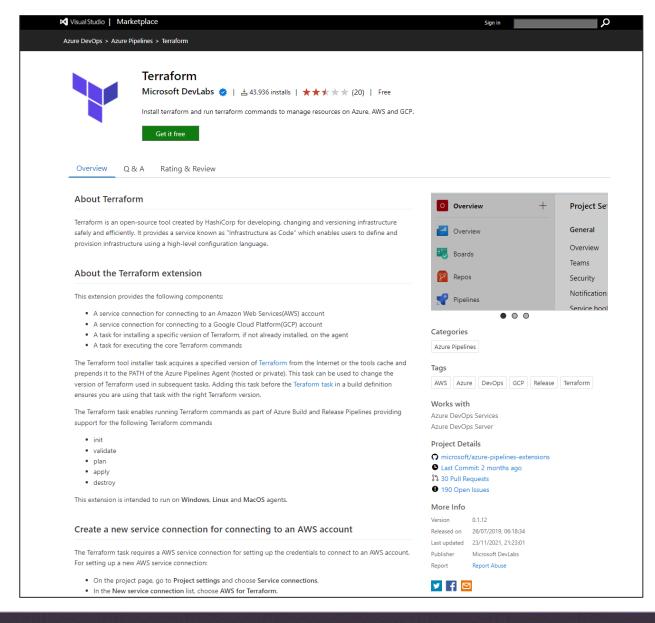
Azure DevOps Terraform Pipeline



Azure DevOps Terraform Pipeline

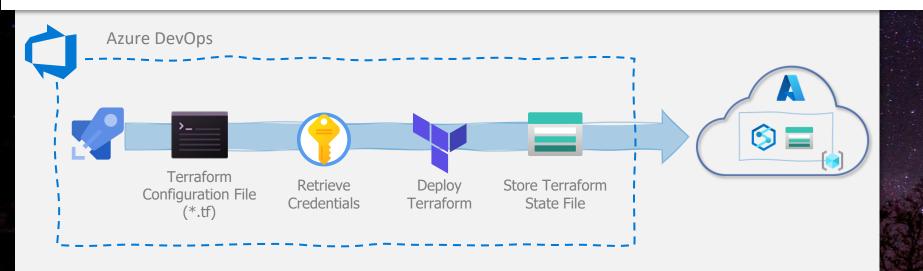


Azure DevOps Terraform Extension

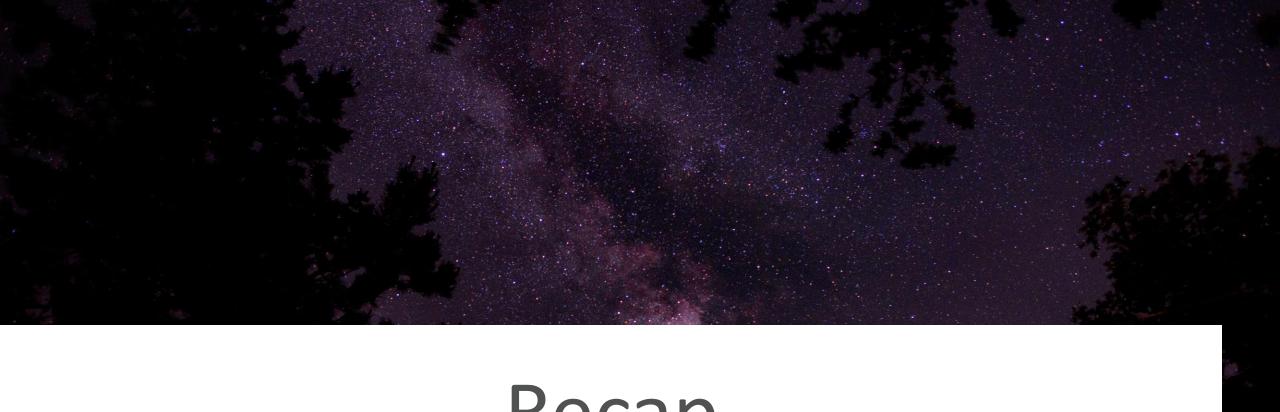




Demo



Run a basic DevOps pipeline



Recap



Recap



Quick and Effective



Easy to Read and Maintain



Repeatable



Consistent



Continuous Deployment



YML Pipelines



Governance and Controls







Azure Synapse



Azure DevOps





Thank You

