CoHEsion

DK David Kent Consulting
ORACLE and Banner IT Services

Cloud Automation
In the Azure Cloud Using Terraform with DevOps

Hank Wojteczko - Cloud Practice Team Lead

Edwin Young - Cloud Engineer

ORACLE and Banner Experts

Real-world Practitioners.   Solutions You Trust.   Always..

My First Cloud Programming Assignment
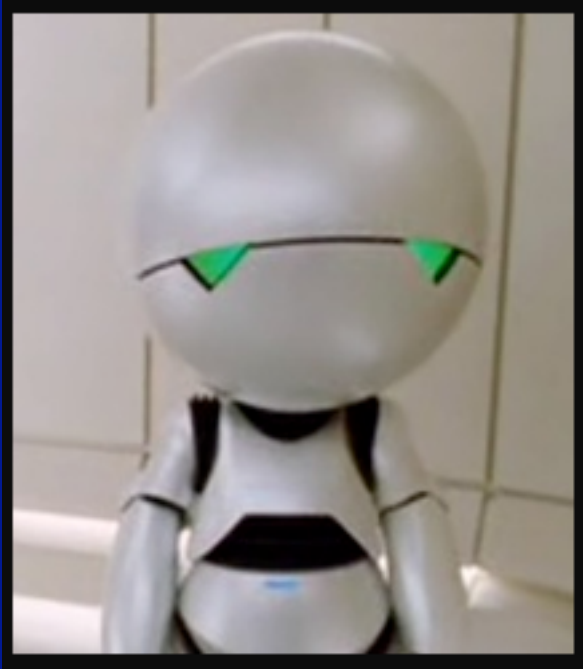
"Marvin, your code is so bad - you are depressing me"

David Kent Consulting
ORACLE and Banner IT Services

Terraform on Azure DevOPs

What Will I Learn Today

- A Brief Overview of the DevOps Model

- Azure Portal and Azure CLI Brief Overview

- Azure CLI, AZ PowerShell, & Terraform Pre-Installation Requirements on Fedora

- Terraform Basics & Language

- The azurerm Terraform Plugin Module

- Terraform CLI

- Setup a Terraform Pipeline in Azure DevOps

DK **David Kent Consulting**
ORACLE and Banner IT Services

What DevOps is Not

- Is not a one-size-fits-all standard

- Is not a product, nor is it a service

- Is not well describable in marketing hype, and of itself, will improve nothing

- Is not a replacement for ITIL, Agile, or other frameworks

- Is not a replacement for IT Operations, Security, Change Enablement, etc.

DK **David Kent Consulting**
ORACLE and Banner IT Services

What is DevOps

DevOps could be defined as:

> "The combination of cultural philosophies, practices, and tools that increases an organization's ability to deliver applications and services at high velocity: evolving and improving products at a faster pace than organizations using traditional software development and infrastructure management processes. "
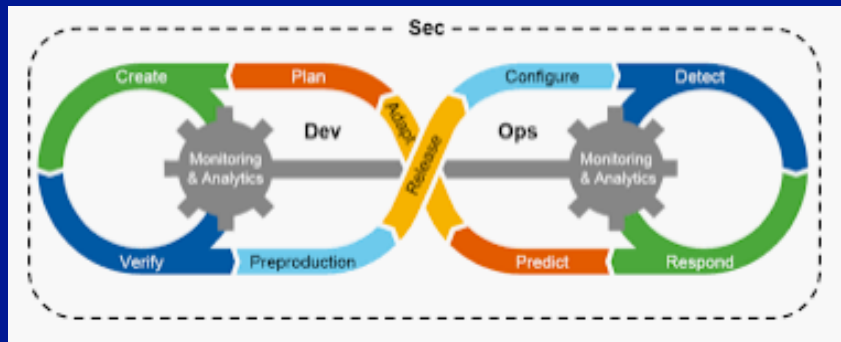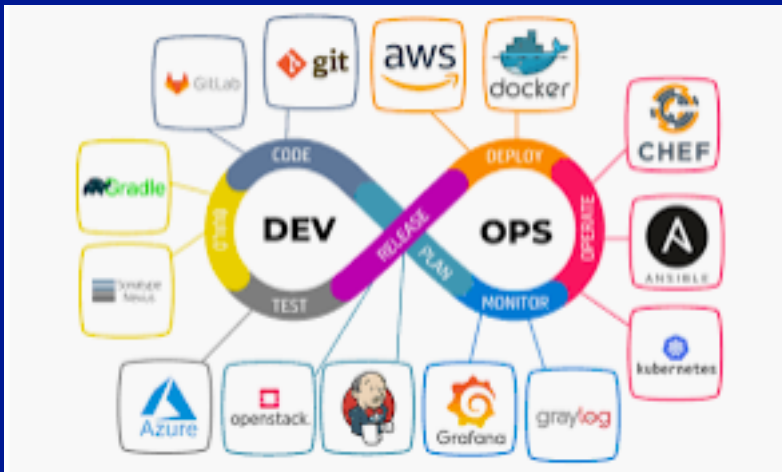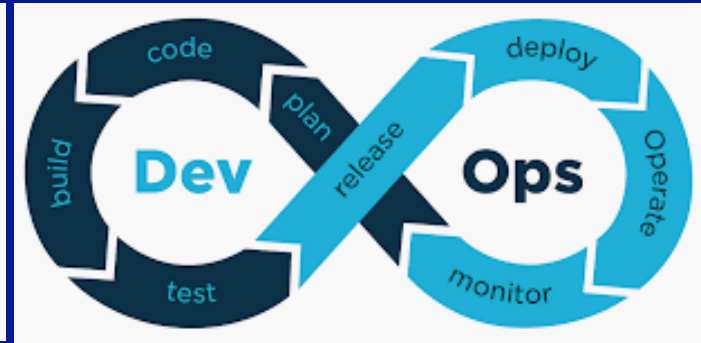
Per AWS at https://aws.amazon.com/devops/what-is-devops/

DK **David Kent Consulting**
ORACLE and Banner IT Services

What is DevOps to my Organization?

David Kent Consulting
ORACLE and Banner IT Services

The DevOps Value Proposition

"The commonly encountered 'continuous-centric' application of DevOps improves the speed of delivery and the resilience of IT services, resulting in faster ROI in functionality, fewer costly disruptions to business operations, and a better customer experience that reduces the risk of reduced enrollment and stagnant tuition. A broader application of DevOps principles, involving the business, further increases the speed of delivery and therefore ROI."

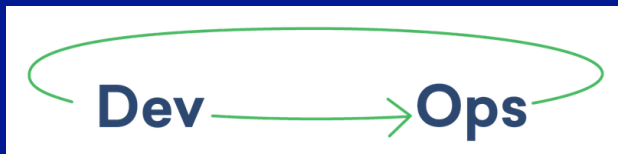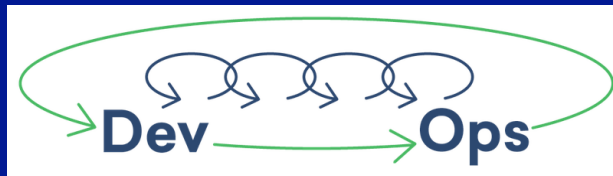Mark Smalley - The Business Value of DevOps, DASA 2017

DK **David Kent Consulting**
ORACLE and Banner IT Services

DevOps - The 3 Ways

- Organizations that Promote a Culture of Life-Long Learning and Experimentation as a Cornerstone

- Organizations that Structure to Permit Feedback

- Organizations that Successfully Implement Workflow

DevOps - Where Do I Begin

- Start by learning from some of the best sources, like "The Phoenix Project," "The DevOps Handbook," "High-Velocity IT"

- The culture of DevOps is more important than the tools

- Get your colleagues involved, both inside and outside of your organization. Cultural change does not happen in a vacuum!

- Limit your acquisitions of potentially expensive tools, start small, plan well, be accurate, then go fast

DK **David Kent Consulting**
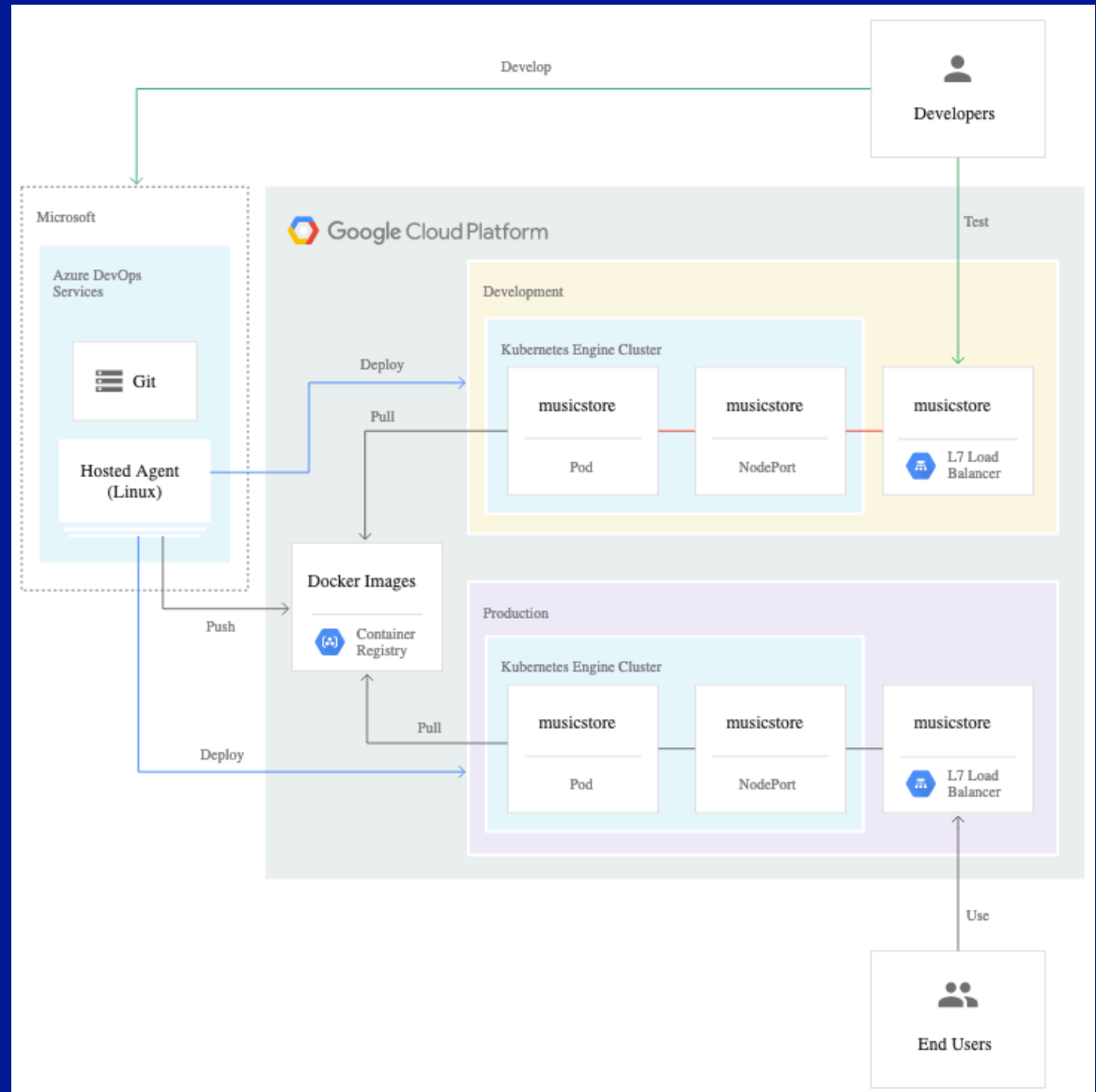ORACLE and Banner IT Services

What is Azure DevOps

- Azure DevOps is a tool that, if properly used, can be an enabler of adopting DevOps in your organization

- Applies the concepts of CD/CI within generally accepted practices of the DevOps model

- Provides versioning, flow control, reporting, project management, automated builds

- Integrates with ITSM tools like ServiceNow

- Integrates with Azure, AWS, Google

Azure DevOps Potential Use Case

Potential use case of integrating Azure DevOps with GitHub and the Google Cloud - Source: https://cloud.google.com/solutions/creating-cicd-pipeline-vsts-kubernetes-engine

David Kent Consulting
ORACLE and Banner IT Services

Why Infrastructure as Code

- To get the best value out of cloud, you must implement the cloud as infrastructure as code

- Do not treat the cloud like VmWare

- The cloud becomes highly elastic and scalable with infrastructure as code

- Infrastructure as code makes the cloud manageable, efficient, cost effective, and portable (new Banner instance in 30 minutes)

- Infrastructure as Code makes the cloud more secure

DK **David Kent Consulting**
ORACLE and Banner IT Services

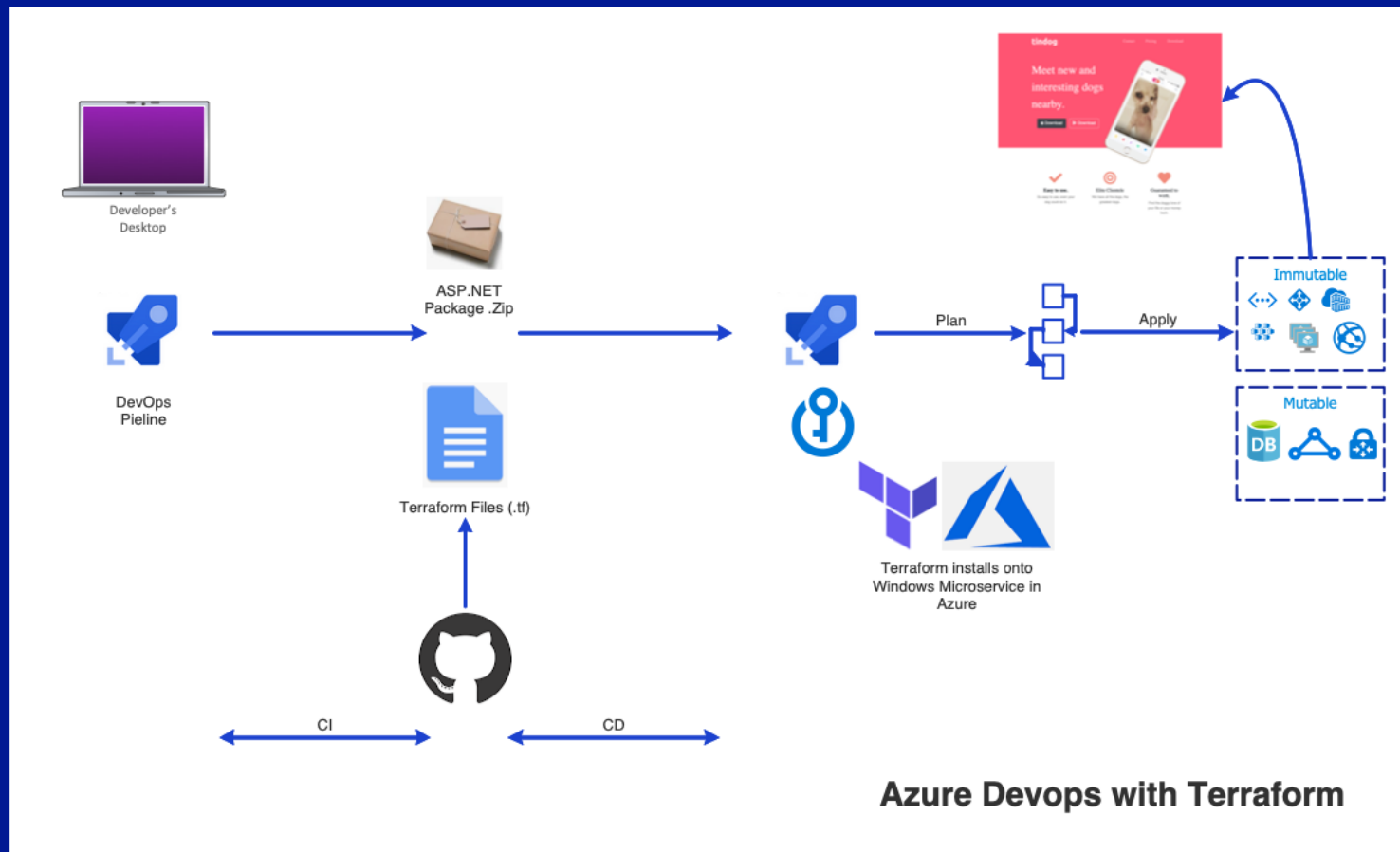# IT's Changing Role in the DevOps Organization
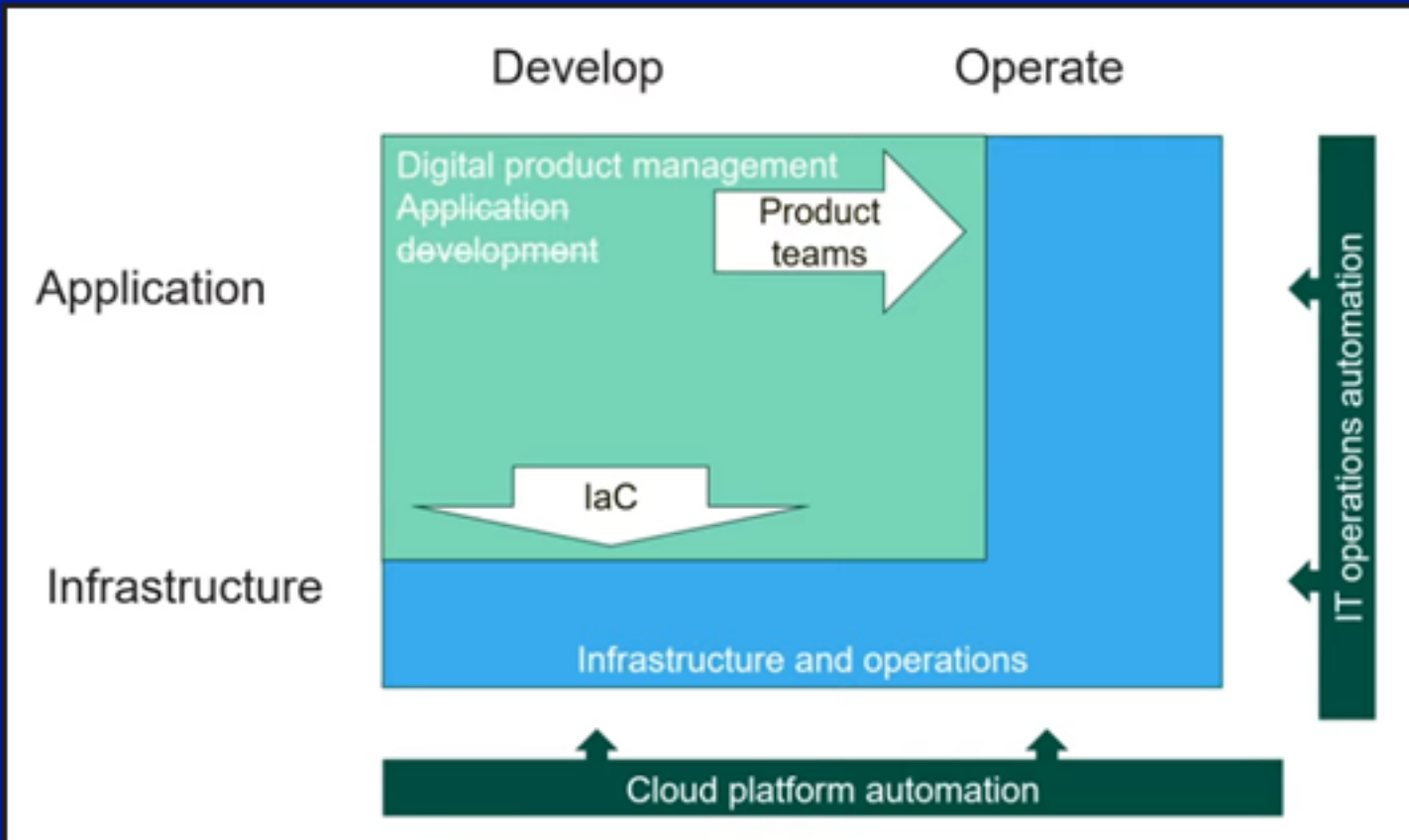
What is Terraform CLI

- Terraform is an open-source infrastructure as code tool developed by HashiCorp

- Can be classified as a low code wrapper platform

- Integrates with "provider" and custom modules to orchestrate public, private, hybrid, and/or multicloud infrastructure as code

- Can be used stand-alone, as a tool within another tool, or within Terraform Enterprise

- Is **_NOT_** suitable for configuration management for mutable cloud resources

DK **David Kent Consulting**
ORACLE and Banner IT Services

Azure DevOps Architecture With Terraform

Azure Devops with Terraform

Time in the Cloud Lab

- The Azure Portal
- Azure AZ Basics
- Break
- Terraform Language
- Terraform CLI
- Terraform State

David Kent Consulting
ORACLE and Banner IT Services

Time in the Cloud Lab

- The Azure Portal
- Azure AZ Basics
- Break
- Terraform Language
- Terraform CLI
- Terraform State


```
AZ CLI #_
AZ CLI #_




                              az login
```

Time in the Cloud Lab

- The Azure Portal
- Azure AZ Basics
- Break (15 minutes)
- Terraform Language
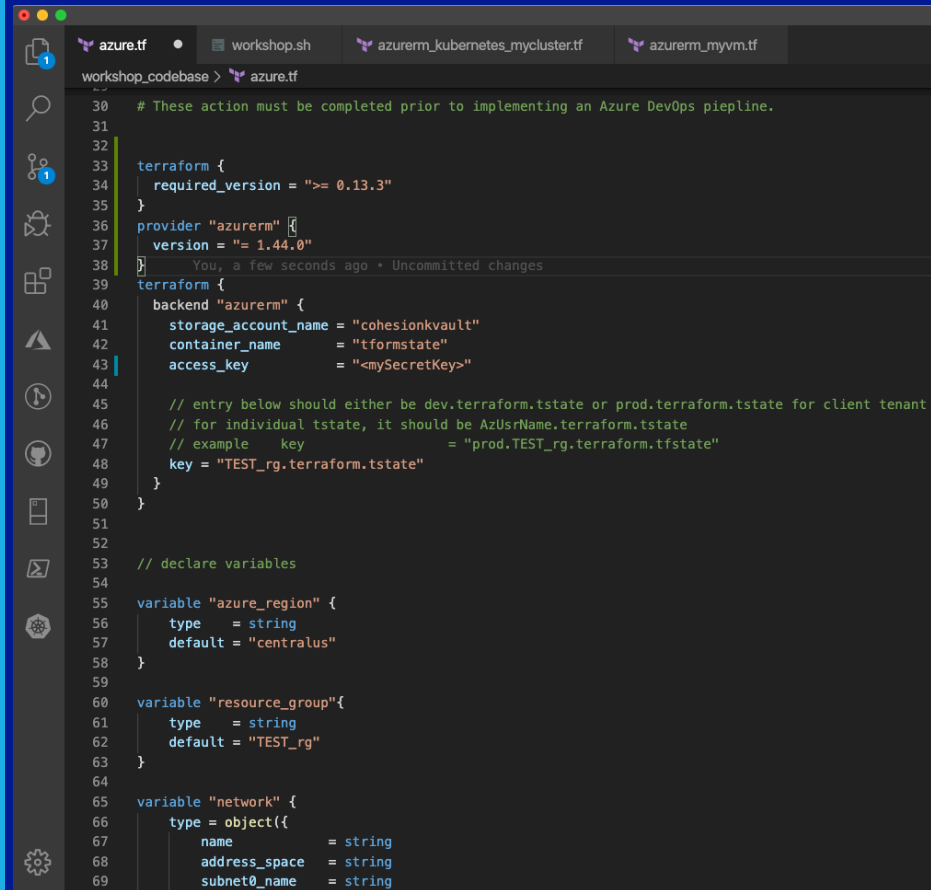- Terraform CLI
- Terraform State

DK David Kent Consulting
ORACLE and Banner IT Services

Time in the Cloud Lab

- The Azure Portal
- Azure AZ Basics
- Break (15 minutes)
- Terraform Language
- Terraform CLI
- Terraform State


```
30    # These action must be completed prior to implementing an Azure DevOps piepline.
31
32
33    terraform {
34      required_version = ">= 0.13.3"
35    }
36    provider "azurerm" {
37      version = "= 1.44.0"
38    }       You, a few seconds ago • Uncommitted changes
39    terraform {
40      backend "azurerm" {
41        storage_account_name = "cohesionkvault"
42        container_name       = "tformstate"
43        access_key           = "<mySecretKey>"
44
45        // entry below should either be dev.terraform.tstate or prod.terraform.tstate for client tenant
46        // for individual tstate, it should be AzUsrName.terraform.tstate
47        // example    key                = "prod.TEST_rg.terraform.tfstate"
48        key = "TEST_rg.terraform.tstate"
49      }
50    }
51
52
53    // declare variables
54
55    variable "azure_region" {
56        type    = string
57        default = "centralus"
58    }
59
60    variable "resource_group"{
61        type    = string
62        default = "TEST_rg"
63    }
64
65    variable "network" {
66        type = object({
67            name            = string
68            address_space   = string
69            subnet0_name    = string
```

DK **David Kent Consulting**
ORACLE and Banner IT Services

20

Time in the Cloud Lab

- The Azure Portal

- Azure AZ Basics

- Break (15 minutes)

- Terraform Language

- Terraform CLI

- Terraform State

https://www.terraform.io/docs/commands/index.html

HashiCorp   Learn how Terraform fits into the **HashiCorp Stack** >

Terraform   Intro   Tutorials   Docs ⌄   Community   Enterprise   ⬇ Download   GitHub   Sign In   Create Account

Terraform CLI
EXPAND ALL | FILTER

> Configuration Language
⌄ Commands (CLI)
   · CLI Config File
   · Environment Variables
   · apply
   · console
   · destroy
   · env
   · fmt
   · force-unlock
   · get
   · graph
   · import
   · init
   · login
   · logout
   · output
   · plan
   · providers

Terraform Commands (CLI)

For a hands-on tutorial, try the Get Started track on HashiCorp Learn.

Terraform is controlled via a very easy to use command-line interface (CLI). Terraform is only a single command-line application: terraform. This application then takes a subcommand such as "apply" or "plan". The complete list of subcommands is in the navigation to the left.

The terraform CLI is a well-behaved command line application. In erroneous cases, a non-zero exit status will be returned. It also responds to -h and --help as you'd most likely expect.

To view a list of the available commands at any time, just run terraform with no arguments:

```
$ terraform
Usage: terraform [-version] [-help] <command> [args]

The available commands for execution are listed below.
The most common, useful commands are shown first, followed by
less common or more advanced commands. If you're just getting
started with Terraform, stick with the common commands. For the
other commands, please read the help and docs before usage.

Common commands:
    apply              Builds or changes infrastructure
    console            Interactive console for Terraform interpolations
    destroy            Destroy Terraform-managed infrastructure
    env                Workspace management
    fmt                Rewrites config files to canonical format
    get                Download and install modules for the configuration
```

DK **David Kent Consulting**
ORACLE and Banner IT Services

21

Time in the Cloud Lab

- The Azure Portal
- Azure AZ Basics
- Break (15 minutes)
- Terraform Language
- Terraform CLI
- Terraform State

```
{
  "version": 4,
  "terraform_version": "0.13.3",
  "serial": 135,
  "lineage": "xxxxxxxx-xxxx-xxxx-01a6-a90d7cdddddb",
  "outputs": {},
  "resources": [
    {
      "mode": "managed",
      "type": "azurerm_kubernetes_cluster",
      "name": "my_cluster",
      "provider": "provider[\"registry.terraform.io/hashicorp/azurerm\"]",
      "instances": [
        {
          "schema_version": 0,
          "attributes": {
            "addon_profile": [
              {
                "aci_connector_linux": [
                  {
                    "enabled": false,
                    "subnet_name": ""
                  }
                ],
                "azure_policy": [
                  {
                    "enabled": false
                  }
                ],
                "http_application_routing": [
                  {
                    "enabled": false,
                    "http_application_routing_zone_name": ""
                  }
                ],
                "kube_dashboard": [
                  {
                    "enabled": false
                  }
                ],
                "oms_agent": [
                  {
                    "enabled": false,
                    "log_analytics_workspace_id": ""
                  }
                ]
              }
            ]
```

David Kent Consulting
ORACLE and Banner IT Services

GitHub Repo for Lab Exercises

LUNCH

45 Minutes

BUILD IT
tindog - A Dating App for Dogs
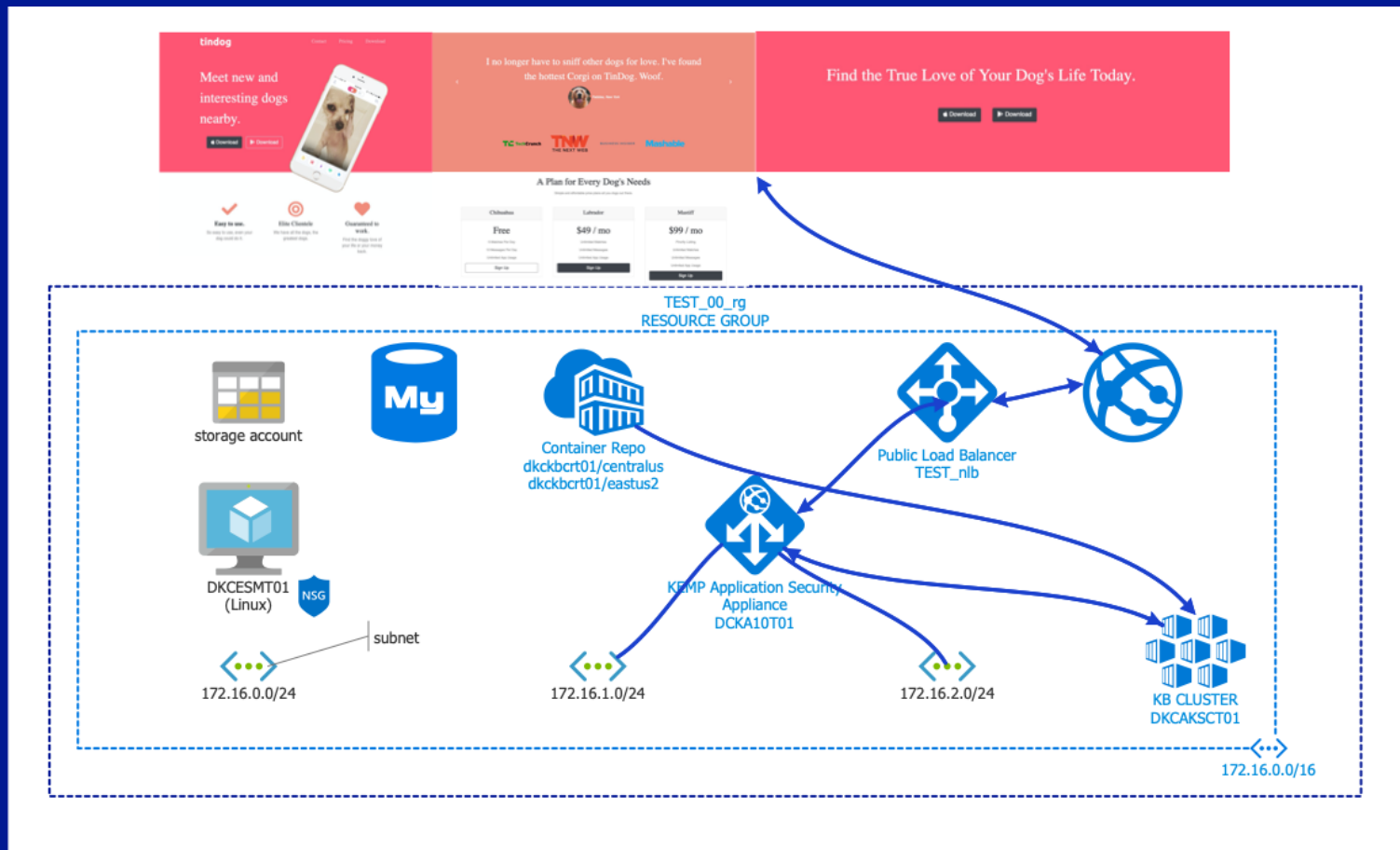
References

The DevOps Handbook - Gene Kim & Patrick Debois

The Phoenix Project - Kim, Behr, Spafford

Reflections on High-velocity IT - Mark Smalley

Crucial Conversations - Patterson

Kubernetes for Developers - Packt>

Azure DevOps Server 2019 Cookbook - Packt>

Questions?

Thank You!

Hank Wojteczko - Cloud Practice Team Lead
David Kent Consulting, Inc
hankwojteczko@davidkentconsulting.com
832.226.4432(m)

Edwin Young - Cloud Engineer
David Kent Consulting, Inc
edwinyoung@davidkentconsulting.com
832.823.3724(m)