Code your Azure Deployments using Terraform

think tecture

Thorsten Hans

@ThorstenHans

Consultant @ Thinktecture AG

think tecture

Thorsten Hans

The guy who's talkin'

Consultant @ Thinktecture AG

thorsten.hans@thinktecture.com
thinktecture.com

thorsten.hans@gmail.com
thorsten-hans.com
https://thns.io/slides

@ThorstenHans

Talking Points

1. What is HashiCorp Terraform
2. Let’s create a Azure Deployment, live!
3. References
4. Hands On Labs

What is

HashiCorp Terraform

Terraform

Write, Plan, and Create

1

Code your Infrastructure

2

Preview Changes before applying

3

Build reproducible infrastructure

Terraform - Write

Code your Infrastructure

- Can be stored in a version control system (git)
    - Change History / Change Tracking
    - Collaboration
- Better documentation
    - Removes tribal knowledge (SPoF)
- Predictable deployments
- TDD deployments are possible

Terraform - Plan

Preview Changes before applying

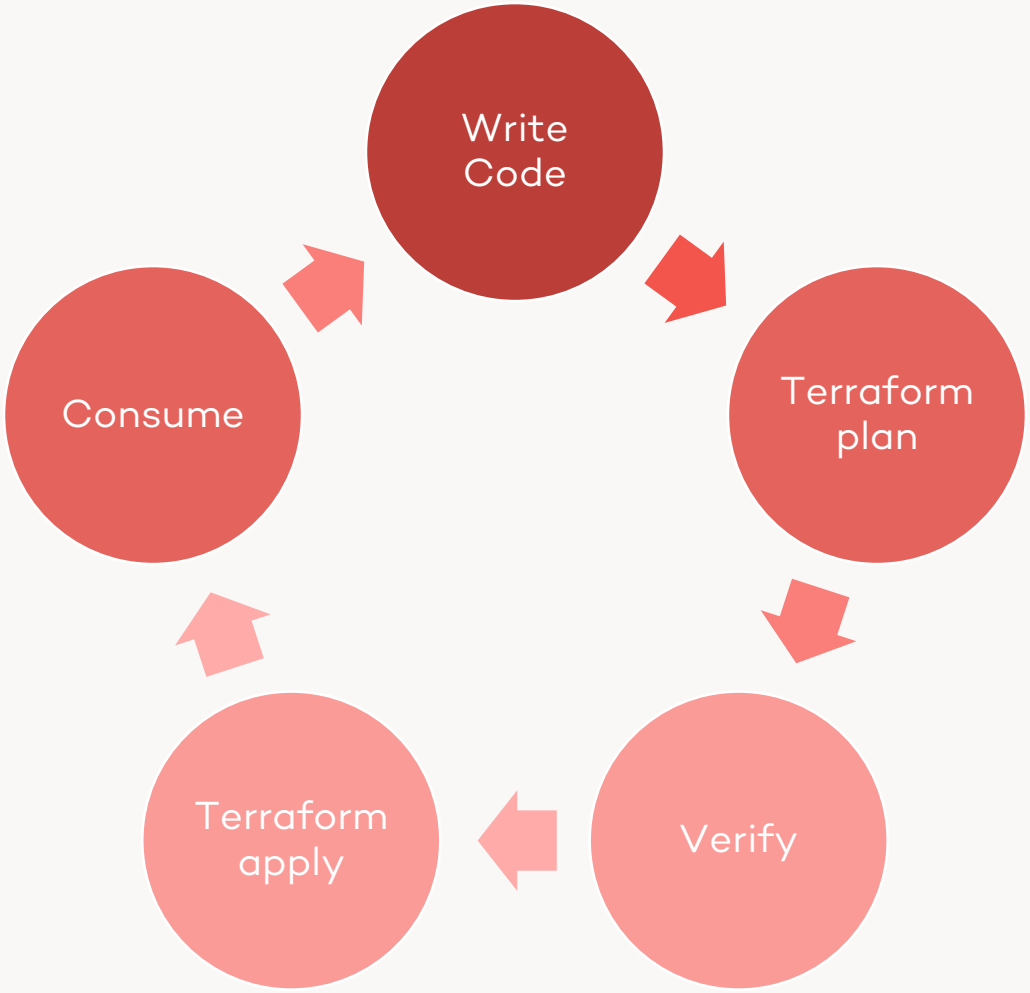
- Execution plan will tell you which parts of the environment will be
  - Created
  - Modified
  - Deleted
- Before making any changes to a environment
- Think of `git status`

Terraform - Create

Build reproducible infrastructure

- One Terraform script can be applied million times
- Configurable environments
    - Variables
    - Interpolation
    - Built-in functions

The Terraform Lifecycle



Terraform Destroy

Meet the HCL

HashiCorp Configuration Language

- Declarative language used to code your infrastructure
- Based on JSON but more focused
- Simple but powerful type system
    - Strings, Maps, Lists, Booleans*
- One language used for all targets
    - Azure, Azure Stack, VMWare, GCP, AWS, …

Let‘s create a Azure Deployment

Live

References

Good stuff to dive deeper

Terraform Features

Essential Features

- Modules
  - https://thns.io/2R73oIQ
- Workspaces
  - https://thns.io/2OZMs4Y
- Terraform State
  - https://thns.io/2AftGS2
- Backends
  - https://thns.io/2DFyNOy

Terraform Resources

Online and offline resources

- Terraform Best Practices
  - https://thns.io/2P0mb6z
- HashiCorp Youtube Channel
  - https://thns.io/2TBycmB
- The Terraform Book
  - https://thns.io/2TDMCCE
- Terraform: Up and Running
  - https://thns.io/2QdJSNz

Hands On Labs

Getting started with Terraform

Hands On Labs

Install required components

- Azure CLI 2.0
  - https://thns.io/install-az
- Terraform
  - https://thns.io/install-tf
- Visual Studio Code
  - https://thns.io/install-code
- Terraform Extension for Visual Studio Code

Hands On Labs

Tasks

- Clone the Repo
    - https://thns.io/hol-start
- Login in Azure CLI and set Subscription
- Deploy to your Azure Subscription
- Set yourself as author (see `variables.tf`)

Hands On Labs

Tasks

- Create an Azure Application Insights Resource
- Set Instrumentation key on App Service as App Setting

Q&A

think tecture

If you've further questions later on:

shoot me a mail at

thorsten.hans@thinktecture.com / thorsten.hans@gmail.com

or tweet at

@ThorstenHans

slides at

https://thns.io/slides