

Terraform

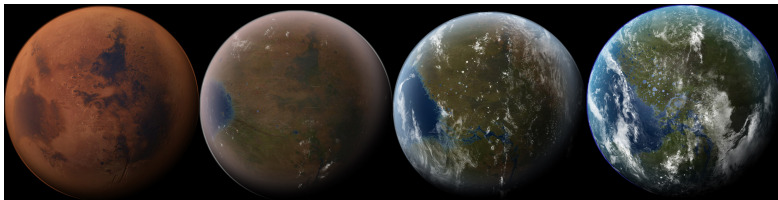
Terraform - A more versatile tool than you thought



Dr. Sebastian Oehlke

2019-10-16

We need n-resources ...now. They have to meet following criteria ...



<https://en.wikipedia.org/wiki/Image:MarsTransitionV.jpg>

- ▶ Infrastructure-as-code
- ▶ Hashicorp (Vagrant, Vault, Packer)

- ▶ Codified API-calls in declarative configuration files
- ▶ Open-Source

Marketing promises

- ▶ Safely and predictable
- ▶ Create, change or improve infrastructure
- ▶ Combine different provider

General advantages of working with code

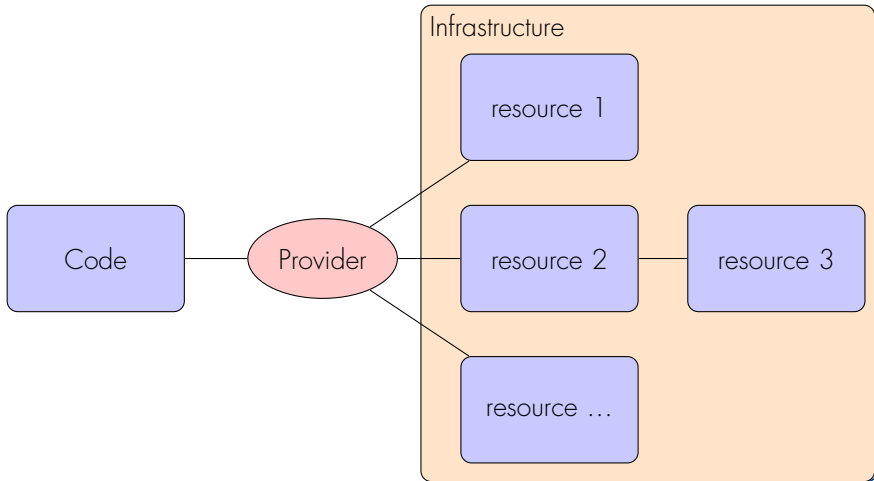
- ▶ Share code, version control and review possibilities
- ▶ Modular
- ▶ Easy to automate

Marketing promises

- ▶ Safely and predictable
- ▶ Create, change or improve infrastructure
- ▶ Combine different provider

General advantages of working with code

- ▶ Share code, version control and review possibilities
- ▶ Modular
- ▶ Easy to automate



Provider-groups:

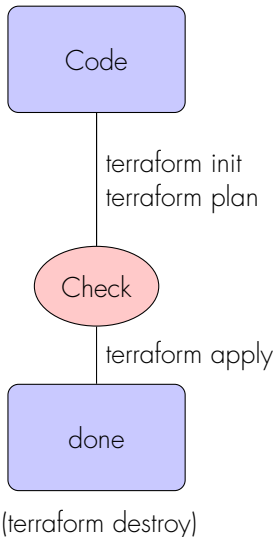
- ▶ Management of Cloud-Services
- ▶ Management of Container-Platforms
- ▶ Network Provider
- ▶ Version Control Systems
- ▶ Monitoring & System Management
- ▶ Databases

⇒ If there is an API, you can most likely manage it with Terraform.

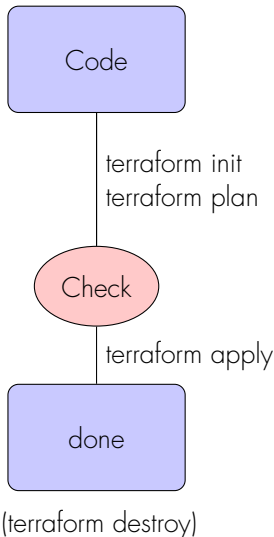
Provider-groups:

- ▶ Management of Cloud-Services
- ▶ Management of Container-Platforms
- ▶ Network Provider
- ▶ Version Control Systems
- ▶ Monitoring & System Management
- ▶ Databases

⇒ If there is an API, you can most likely manage it with Terraform.



- ▶ Syntax check and loading of required plugins
- ▶ Creation of an overview
 - ▶ What does exist already?
 - ▶ What will be changed?
 - ▶ How are the changes applied?
- ▶ Provider communication (API)
- ▶ (resource removal)



- ▶ Syntax check and loading of required plugins
- ▶ Creation of an overview
 - ▶ What does exist already?
 - ▶ What will be changed?
 - ▶ How are the changes applied?
- ▶ Provider communication (API)
- ▶ (resource removal)

variables.tf :

```
variable "region" {  
  description = "Region to be used"  
}
```

terraform.tfvars :

```
region = "eu-central-1"
```

Code :

```
region = var.region
```

⇒ Can be combined with modules

A module is a container for multiple resources that are used together.

```
tree minimal-module
.
|-- README.md
|-- main.tf
|-- variables.tf
|-- outputs.tf
```

A module is a container for multiple resources that are used together.

```
tree minimal-module
.
|-- README.md
|-- main.tf
|-- variables.tf
|-- outputs.tf
```

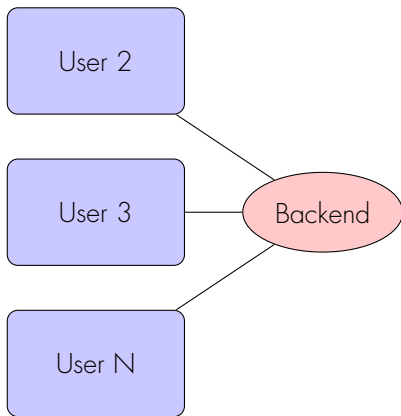
```
module "name" {  
    source = "/path/to/folder"  
    variable1 = "foo"  
    variable2 = "bar"  
}
```

```
module "frontend" {  
    source = "git::git@github.com:user/module.git/m-name?ref=v0.1"  
  
    variable1 = 8  
    variable2 = 20  
}
```



```
module "name" {  
    source = "/path/to/folder"  
    variable1 = "foo"  
    variable2 = "bar"  
}
```

```
module "frontend" {  
    source = "git::git@github.com:user/module.git/m-name?ref=v0.1"  
  
    variable1 = 8  
    variable2 = 20  
}
```



Just create a VM with a static NGINX

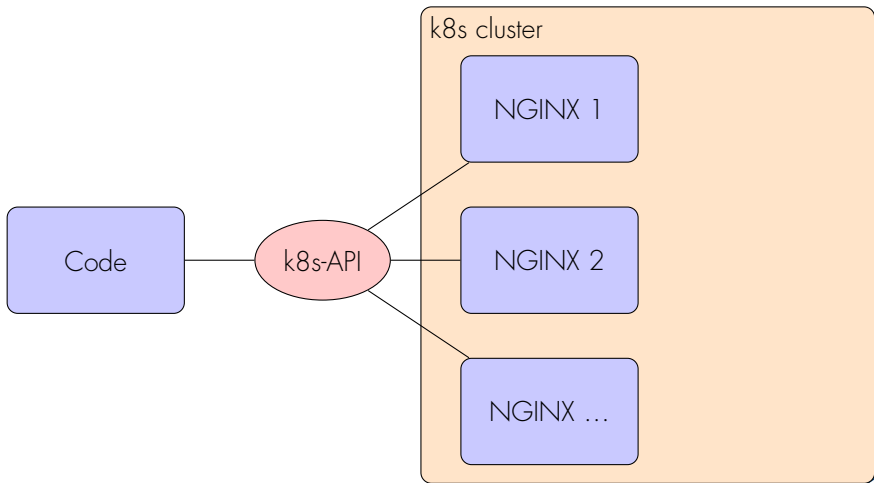
Provider-groups:

- ▶ ~~Management of Cloud-Services~~
- ▶ **Management of Container-Platforms**
- ▶ Network Provider
- ▶ Version Control Systems
- ▶ Monitoring & System Management
- ▶ Databases

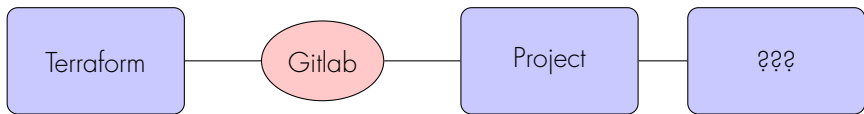
Provider-groups:

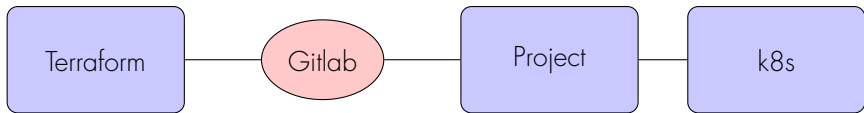
- ▶ Management of Cloud-Services
- ▶ Management of Container-Platforms
- ▶ Network Provider
- ▶ Version Control Systems
- ▶ Monitoring & System Management
- ▶ Databases











Terraform - It is worth a try!



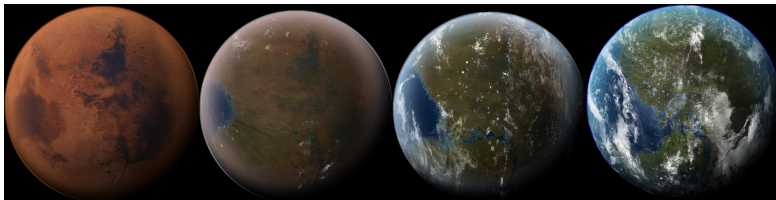
Start with something familiar

Terraform - It is worth a try!

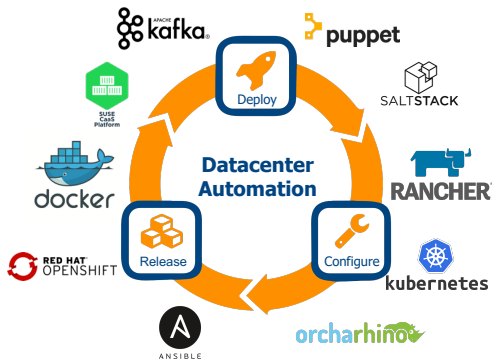
We need n-resources ...now. They have to meet following criteria ...

Terraform - It is worth a try!

~~We need n-resources ...now. They have to meet following criteria-~~



<https://en.wikipedia.org/wiki/Image:MarsTransitionV.jpg>



 @ATIXDE
 @ATIXAG
 ATIX-AG
 ATIX-AG
 oehlke@atix.de