IasC

opentofu and terragrunt

Opentofu/Terraform

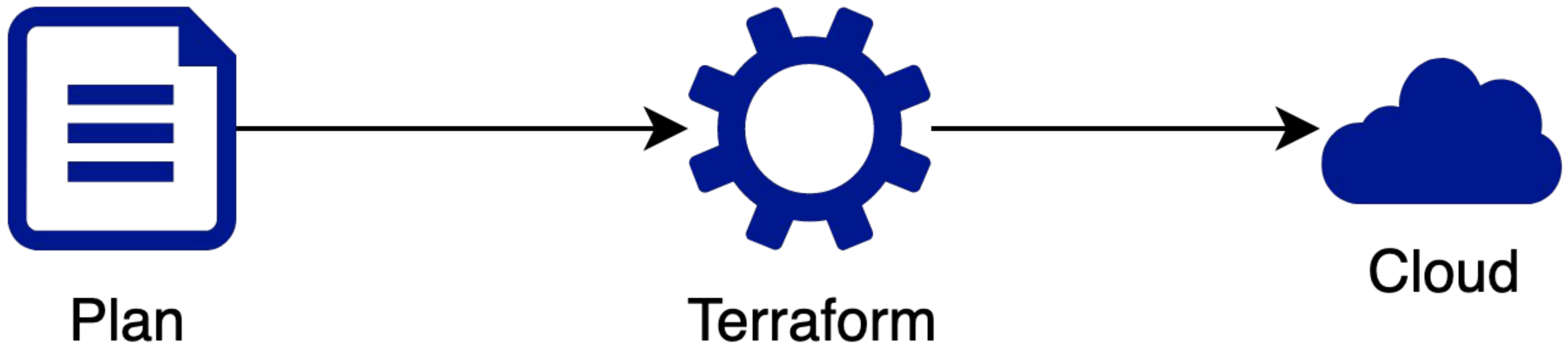
- Install
- K8s application provisioning
- Terragrunt

Opentofu/Terraform

- IaC tool for declarative infrastructure
- OpenTofu — open-source fork of Terraform
- Same functionality for all clouds
- Use same plugins
- Core files: main.tf, variables.tf, outputs.tf

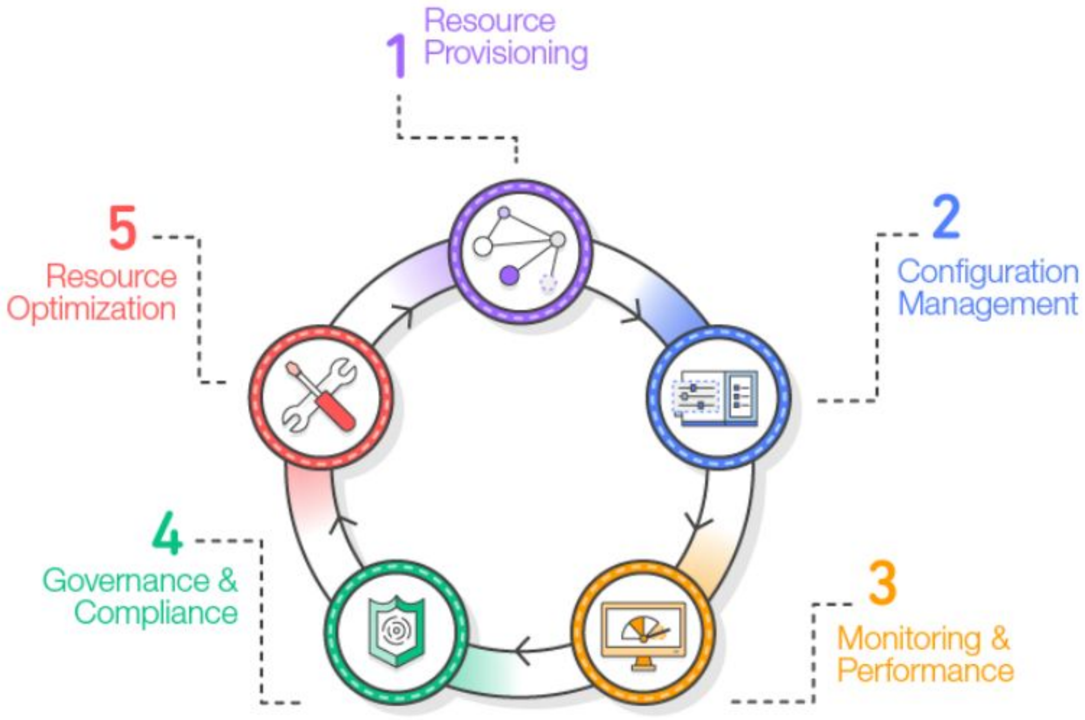
Opentofu/Terraform

- Define infrastructure as code to manage
  - building,
  - changing,
  - versioning



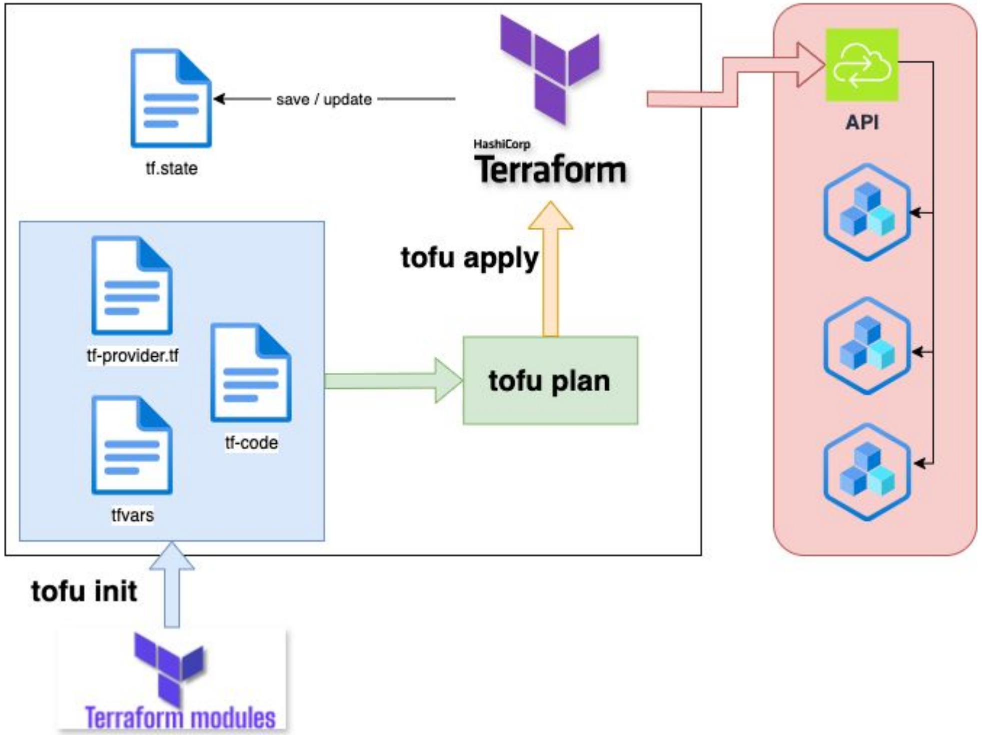
Opentofu/Terraform

-



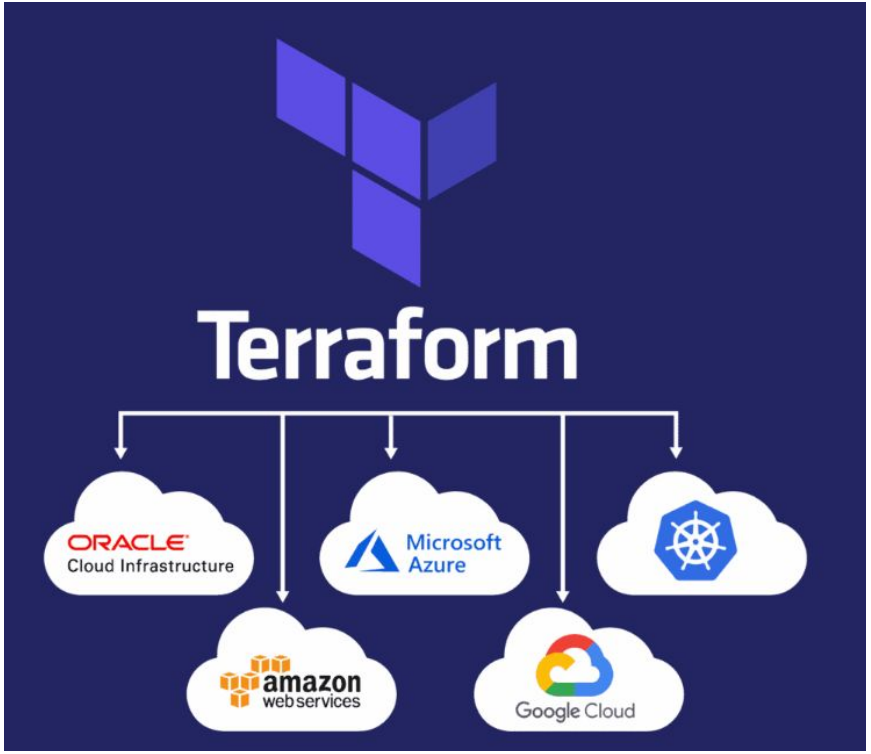
Opentofu/Terraform. How to

-



Opentofu/Terraform

- Providers
  - https://registry.terraform.io/browse/providers
- Modules repository
  - https://registry.terraform.io/browse/modules



Opentofu/Terraform. Code example

```
provider "aws" {
  region = "eu-central-1"
}

resource "aws_s3_bucket" "example" {
  bucket = "demo-bucket"
}
```

Opentofu/Terraform. Start

- terraform init
- terraform plan
- terraform apply
- ...
- terraform destroy

- tofu init
- tofu plan
- tofu apply
- ...
- tofu destroy

-auto-approve (to skip console confirmation)

Opentofu/Terraform. Variables

- Input
  - type
  - default values
  - description
- Output
  - can be constructed from output/state

Openforu/Terraform. Demo

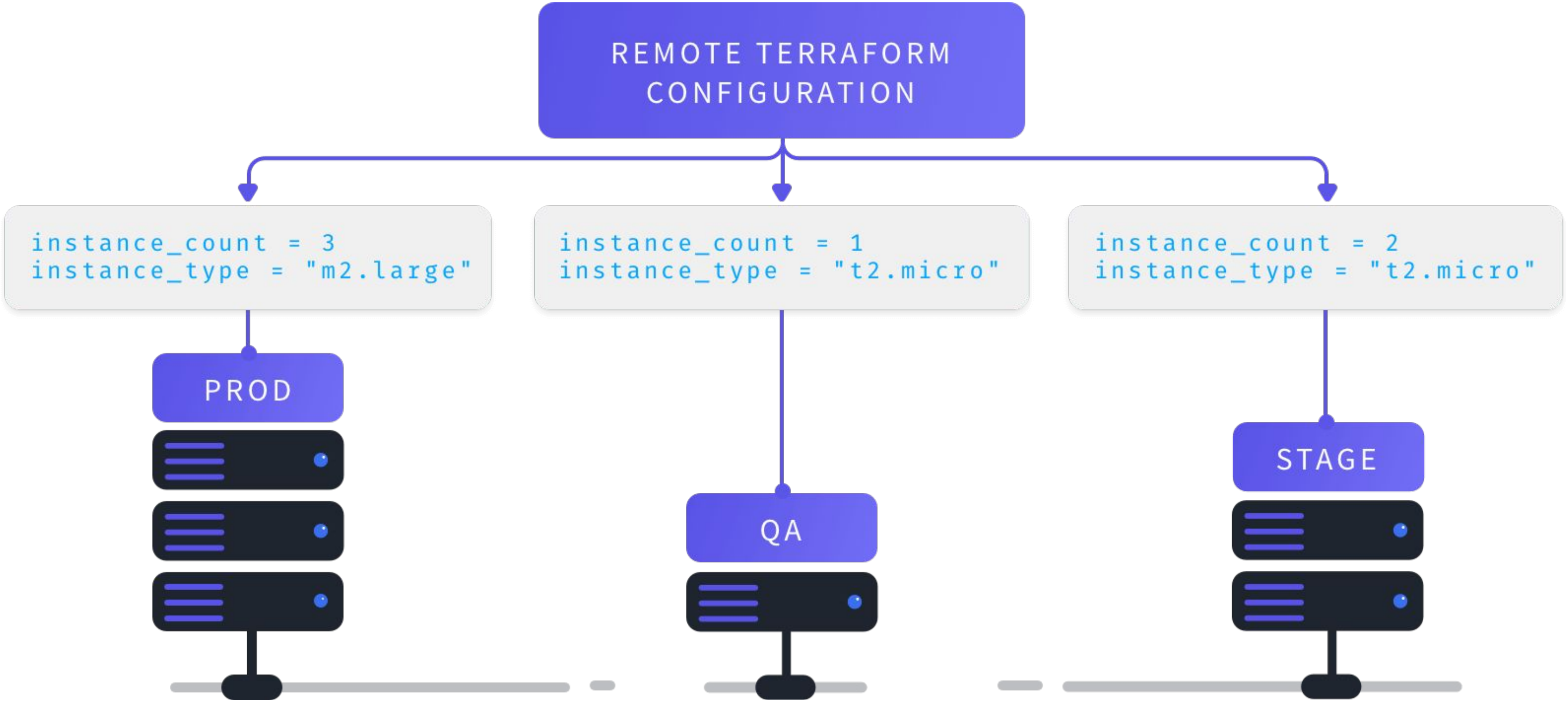
- https://registry.terraform.io/providers/hashicorp/kubernetes/latest/docs
  - Install opentofu: https://opentofu.org/docs/intro/install/deb/
  - Setup provider
  - Create application in k8s
  - Add input variables
  - Add output variables

Terraform & Terragrunt

Terragrunt

- Wrapper for Terraform/OpenTofu, simplifying the management of complex infrastructure deployments by enabling shared modules, remote state…
- Features :
  - dynamic backend
  - Efficient variable management (no extra params are needed, all variables can be accumulated from files automatically)
  - Deployment all in one command
  - Before/after scripts

Terragrunt. Intro



```
instance_count = 3
instance_type = "m2.large"
```

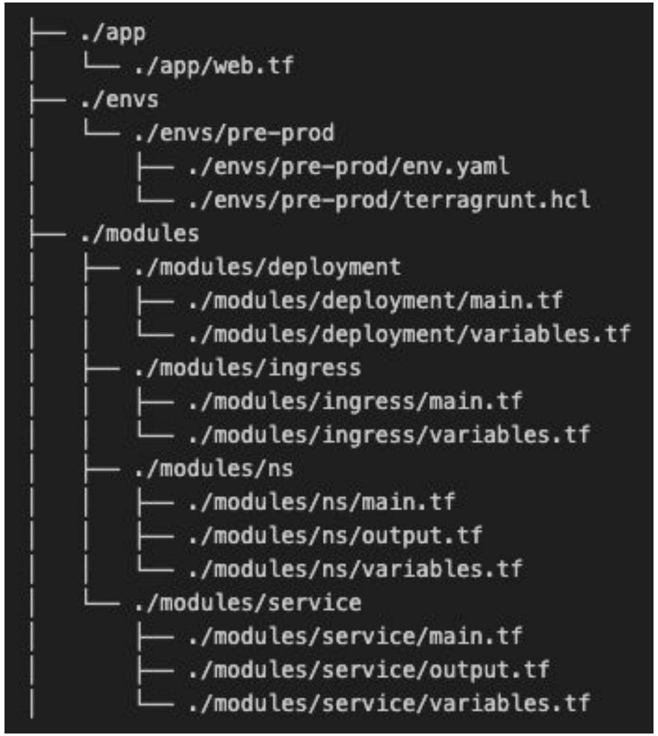
```
instance_count = 1
instance_type = "t2.micro"
```

```
instance_count = 2
instance_type = "t2.micro"
```

Terragrunt. hcl

- HCL
  - HashiCorp Configuration Language
- Blocks

```
locals {}
terraform {
  source = "path"
}
generate "provider" {}
inputs = {}
generate "providers" {}
remote_state {}
```



```
├── ./app
│   └── ./app/web.tf
├── ./envs
│   └── ./envs/pre-prod
│       ├── ./envs/pre-prod/env.yaml
│       └── ./envs/pre-prod/terragrunt.hcl
├── ./modules
│   ├── ./modules/deployment
│   │   ├── ./modules/deployment/main.tf
│   │   └── ./modules/deployment/variables.tf
│   ├── ./modules/ingress
│   │   ├── ./modules/ingress/main.tf
│   │   └── ./modules/ingress/variables.tf
│   ├── ./modules/ns
│   │   ├── ./modules/ns/main.tf
│   │   ├── ./modules/ns/output.tf
│   │   └── ./modules/ns/variables.tf
│   └── ./modules/service
│       ├── ./modules/service/main.tf
│       ├── ./modules/service/output.tf
│       └── ./modules/service/variables.tf
```

Terragrunt. Modules

- Typical reusable module layout:
    - module-name/
    - ├── main.tf   # resources
    - ├── variables.tf   # inputs
    - ├── outputs.tf   # exports
    - ├── versions.tf   # provider constraints
    - └── README.md

Terragrunt. Demo

- https://terragrunt.gruntwork.io/docs/#getting-started
  - Install: download binary https://github.com/gruntwork-io/terragrunt/releases and move to /usr/local/bin/
  - Add dynamic provider
  - Convert terraform to modules
  - Setup application module call
  - Deploy

Terragrunt. CI/CD

Common pipeline stages:

1. Validate → fmt → lint
2. terragrunt plan (preview infra changes)
3. Manual approval step
4. terragrunt apply

Best practices:

- Use ephemeral runners for security
- Mask sensitive outputs
- Upload plan files as artifacts

Opentofu/Terraform

- https://www.terraform.io/docs/providers/docker/r/service.html
- https://newcontext-oss.github.io/kitchen-terraform/tutorials/docker_provider.html
- https://medium.com/@Joachim8675309/docker-the-terraform-way-a7c16b5f59ed
- https://www.terraform.io/docs/configuration/index.html
- https://blog.gruntwork.io/terraform-tips-tricks-loops-if-statements-and-gotchas-f739bbae55f9
- https://opentofu.org/docs/intro/install/deb/
- https://everythingdevops.dev/kubernetes-with-opentofu-a-guide-to-being-fully-open-source/