codecentric

Infrastructure-as-Code with Pulumi

Better than all the others (like Ansible)?

Jonas Hecht | Senior IT-Nerd | github.com/jonashackt

@jonashackt

https://jonashackt.io

codecentric

jonashackt.io

Javamag

PRIVACY BY DESIG

Mit Netz und doppeltem Bod

iX

Microservices sta

Monolithen

Freie NAS-Distributionen

iX

Notebooks fürs

Kubernetes aus der

shortcuts

Vue.js für alle

Wissenswertes für Einsteiger und Experten

Jonas Hecht, Stephan Rauh, Alexander Schwartz, Marc Teufel

4+1

IoThüringen

FHE FACHHOCHSCHU ERFURT UNIVERS OF APPLIED SCIEN

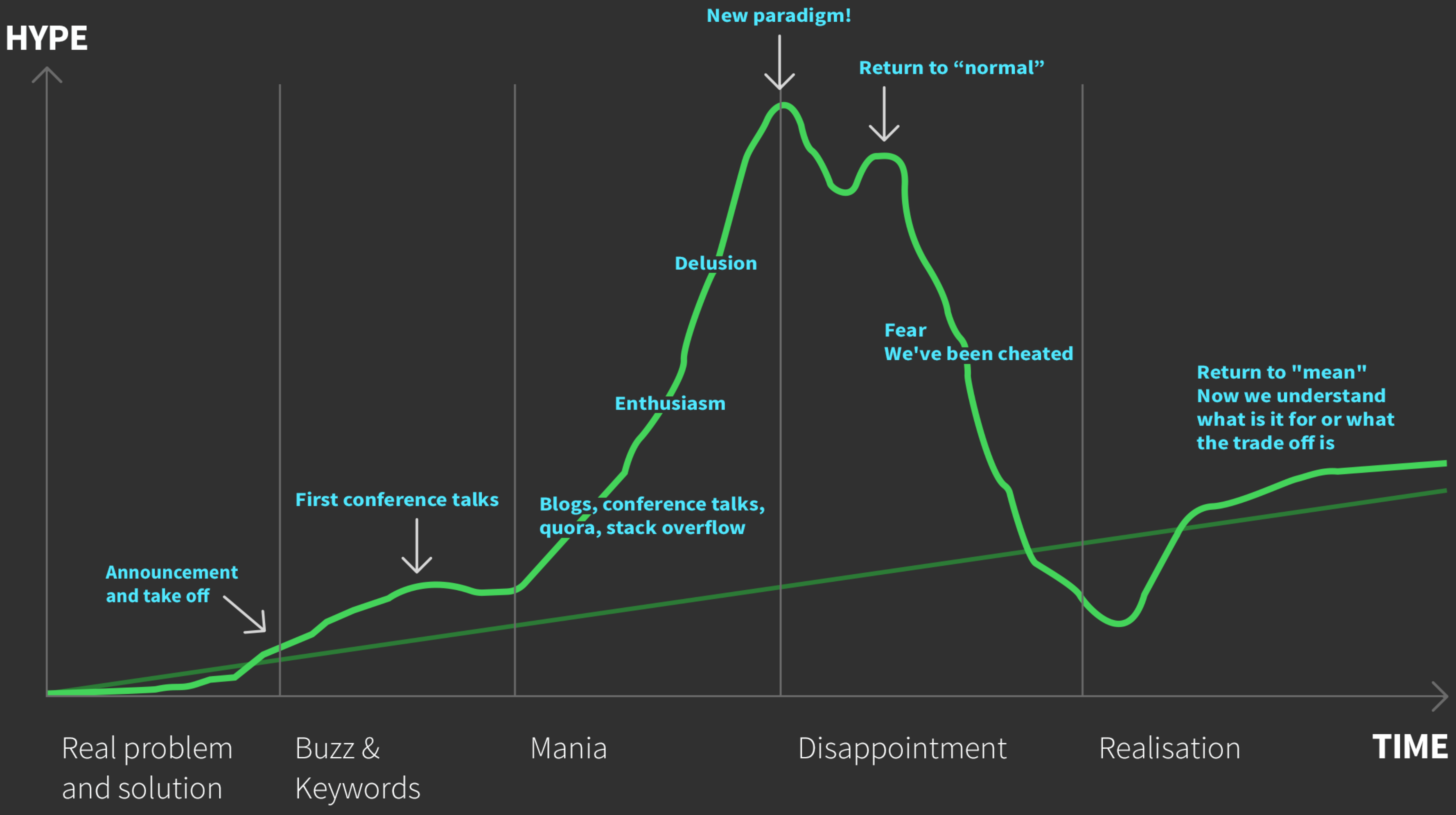
Angewandte Informatik

Bauhaus-Universität Weimar

Dev Ops code plan release deploy operate monitor test build

Thüringen

Our industry is like…



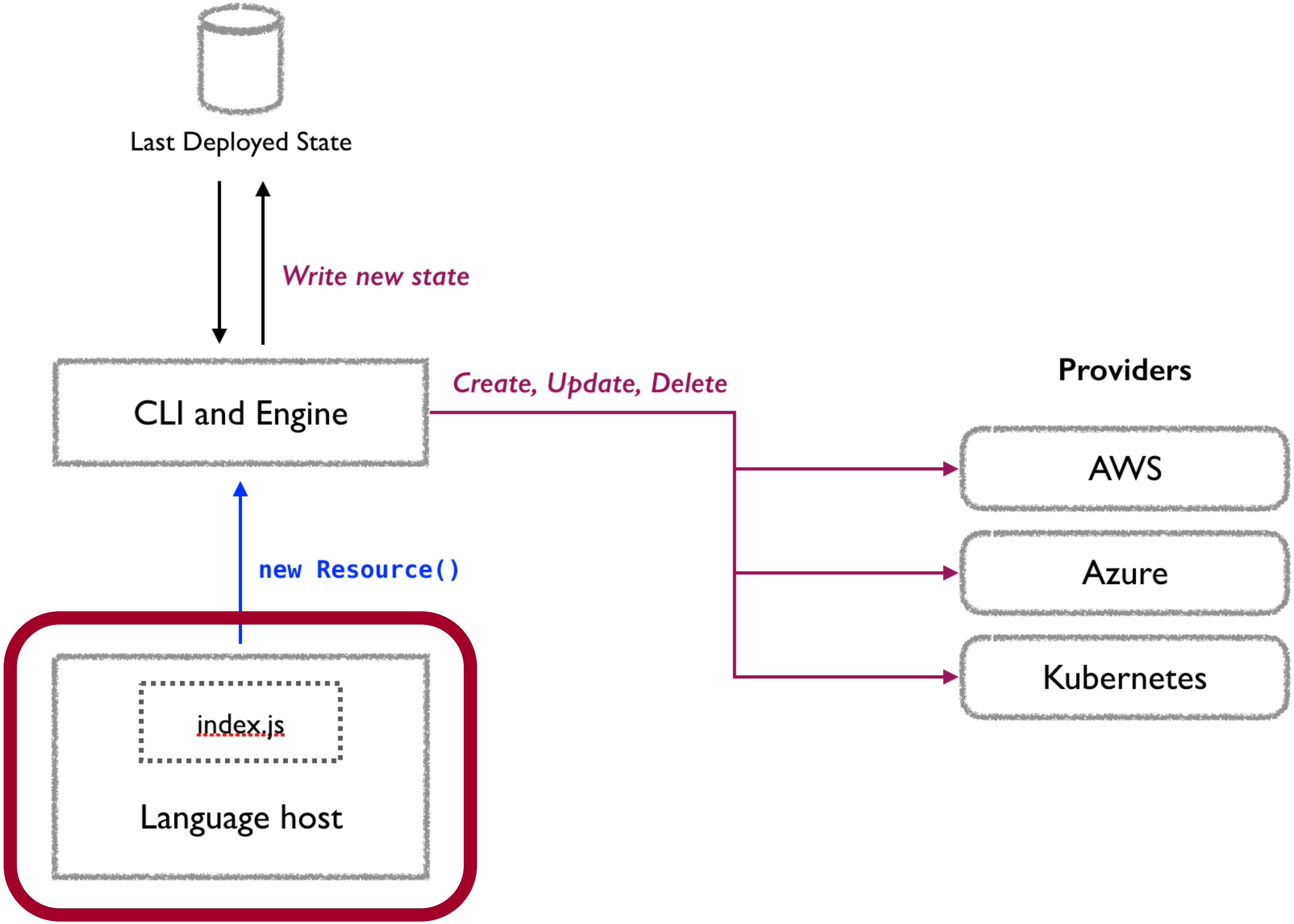
blog.daftcode.pl/hype-driven-development-3469fc2e9b22

Pulumi?

pulumi = YOUNG!

published June 2018

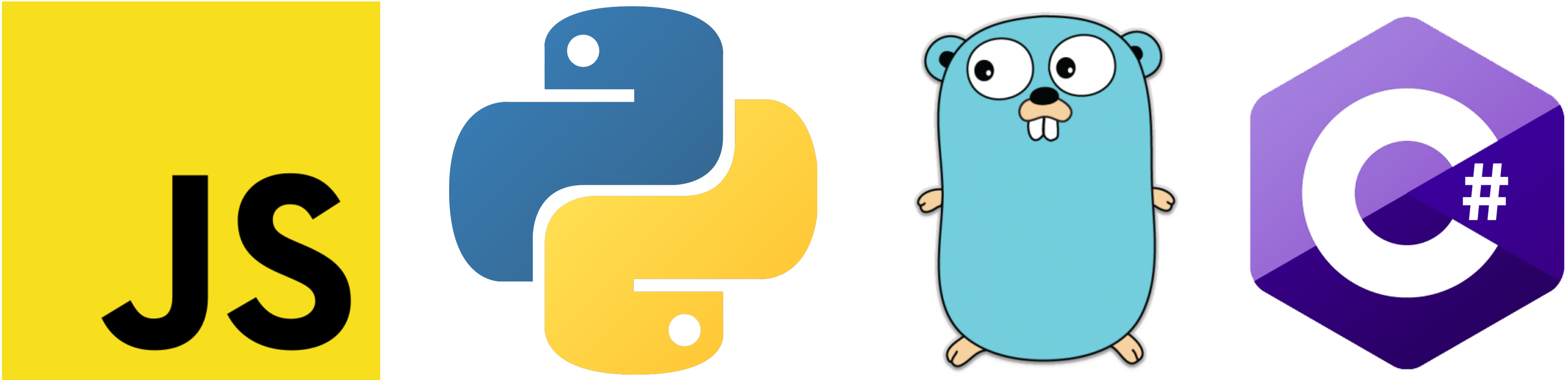
1.0 September 2019



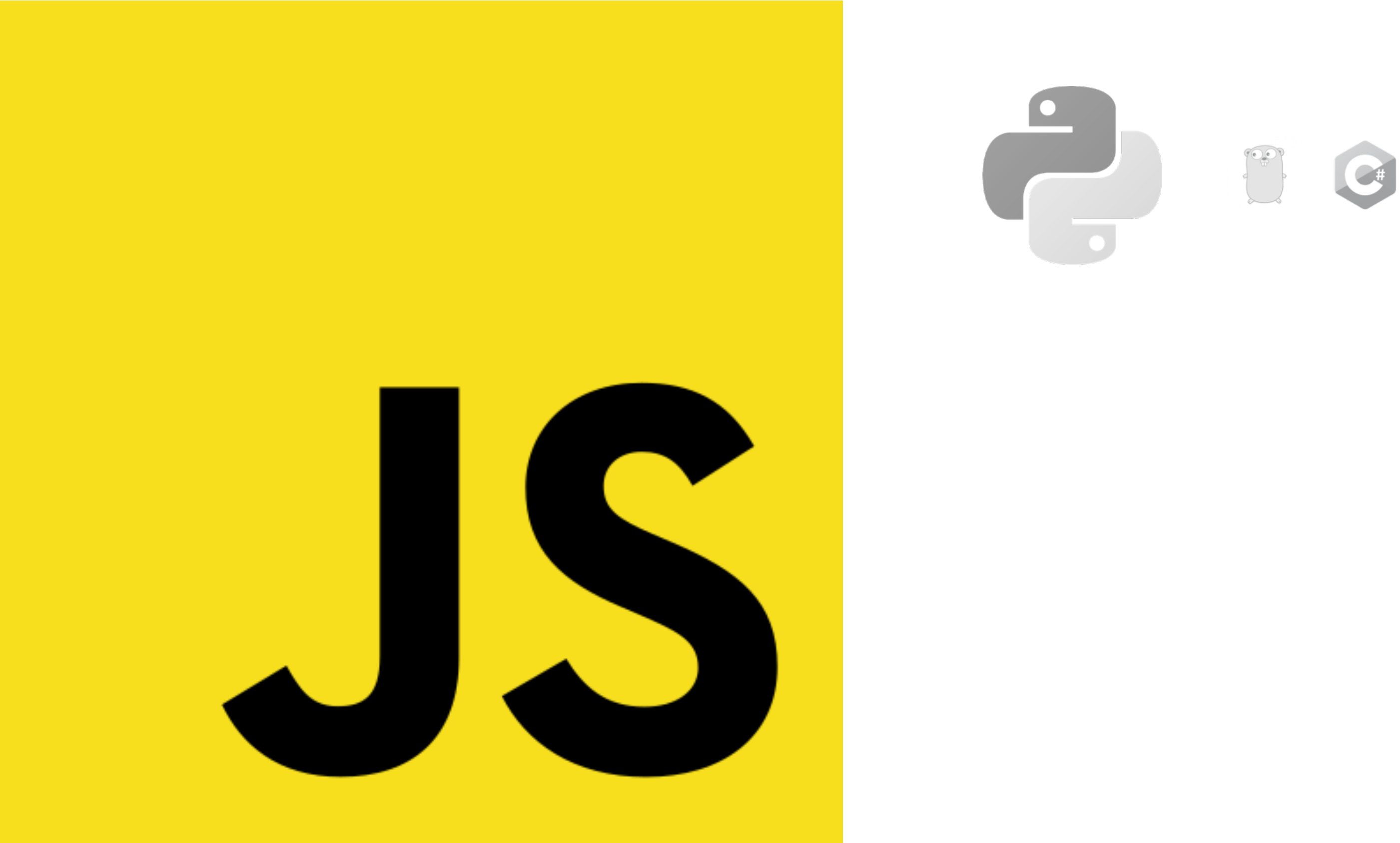
pulumi.com/docs/intro/concepts/how-pulumi-works/

> Pulumi is language agnostic to support multiple programming languages at the same time

pulumi.com/docs/intro/languages

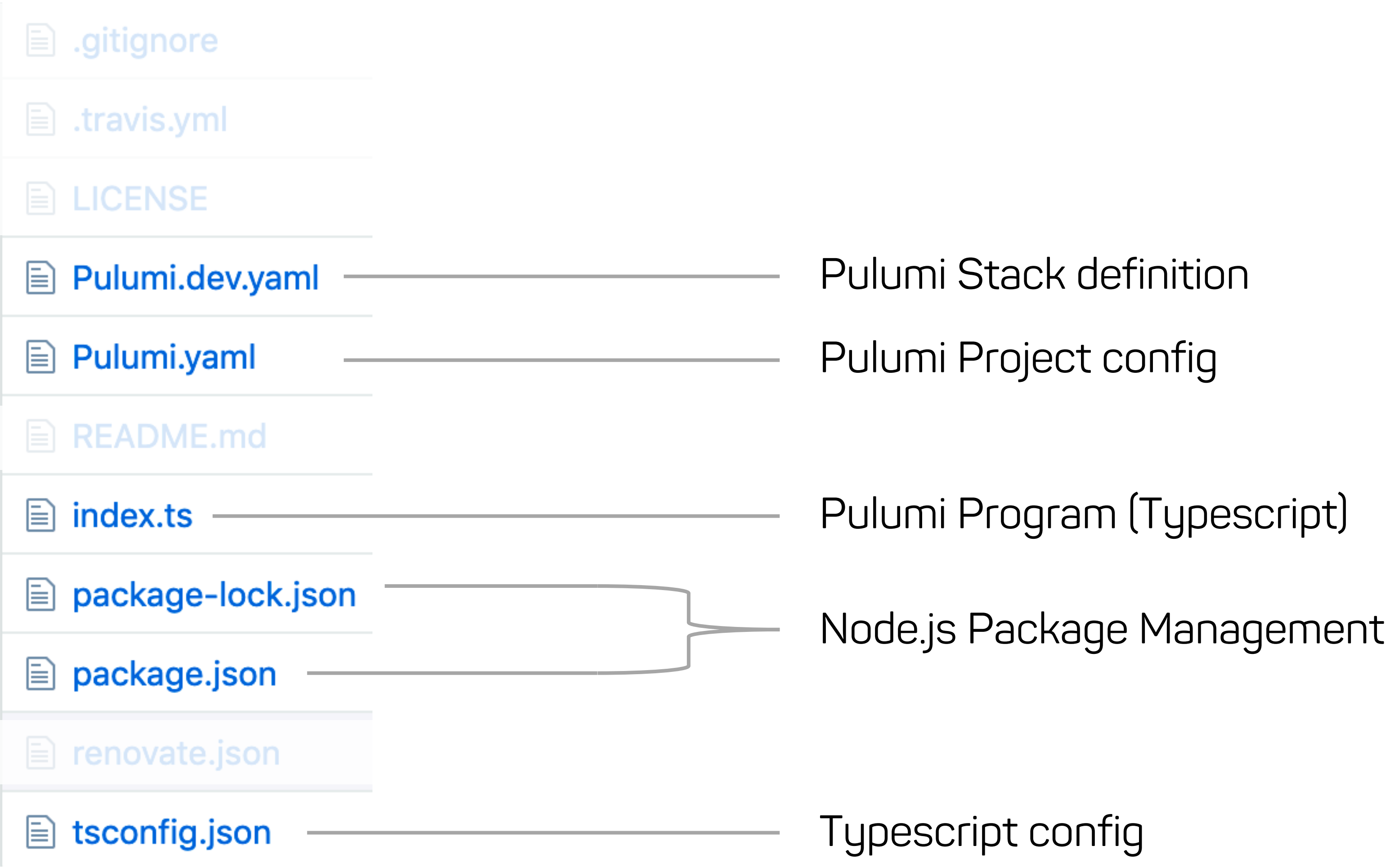


pulumi.com/docs/intro/languages



API reference: pulumi.com/docs/reference/pkg

Examples: github.com/pulumi/examples



.gitignore

.travis.yml

LICENSE

Pulumi.dev.yaml

Pulumi Stack definition

Pulumi.yaml

Pulumi Project config

README.md

index.ts

Pulumi Program (Typescript)

package-lock.json

package.json

Node.js Package Management

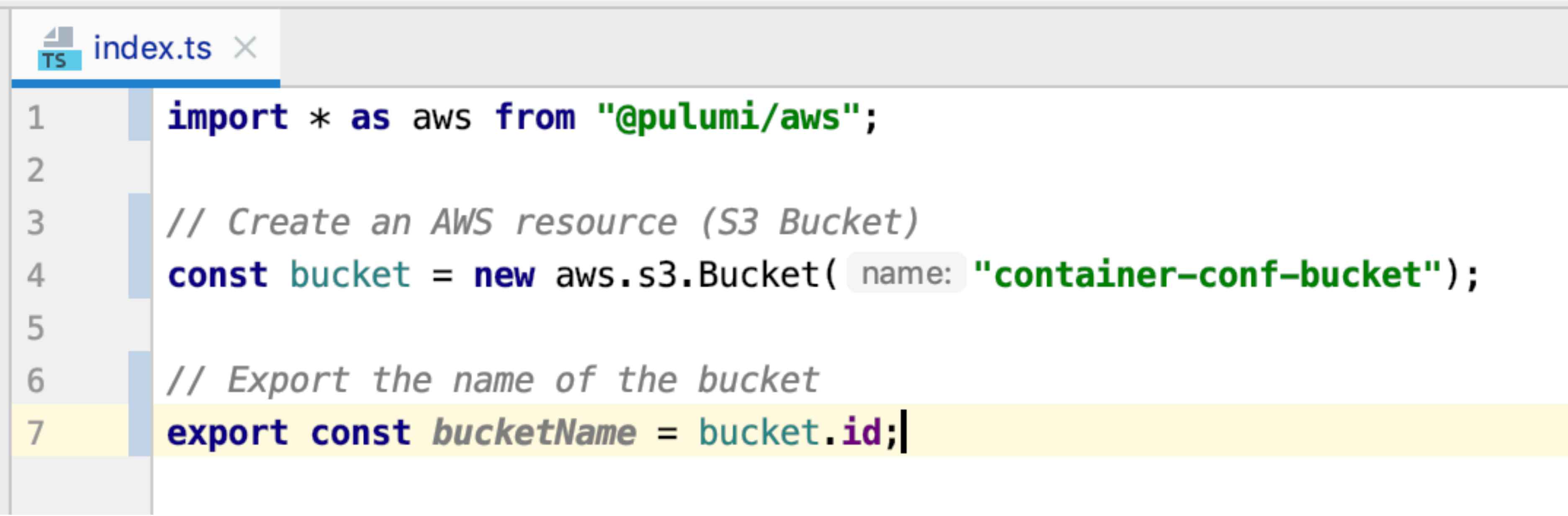
renovate.json

tsconfig.json

Typescript config

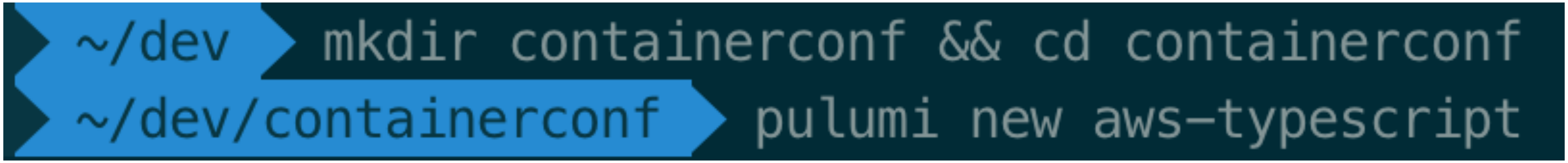
github.com/jonashackt/pulumi-typescript-aws-fargate

Pulumi program example



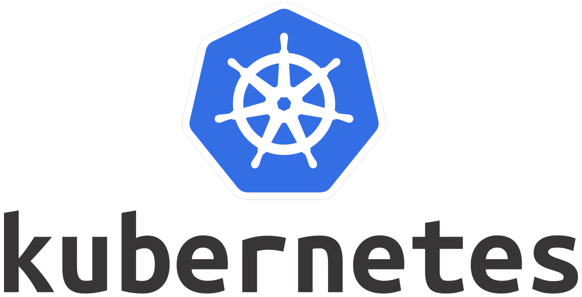
```typescript
import * as aws from "@pulumi/aws";

// Create an AWS resource (S3 Bucket)
const bucket = new aws.s3.Bucket( name: "container-conf-bucket");

// Export the name of the bucket
export const bucketName = bucket.id;
```

Create projects 🎬



```
~/dev  mkdir containerconf && cd containerconf
~/dev/containerconf  pulumi new aws-typescript
```

Multicloud?



pulumi.com/docs/intro/cloud-providers

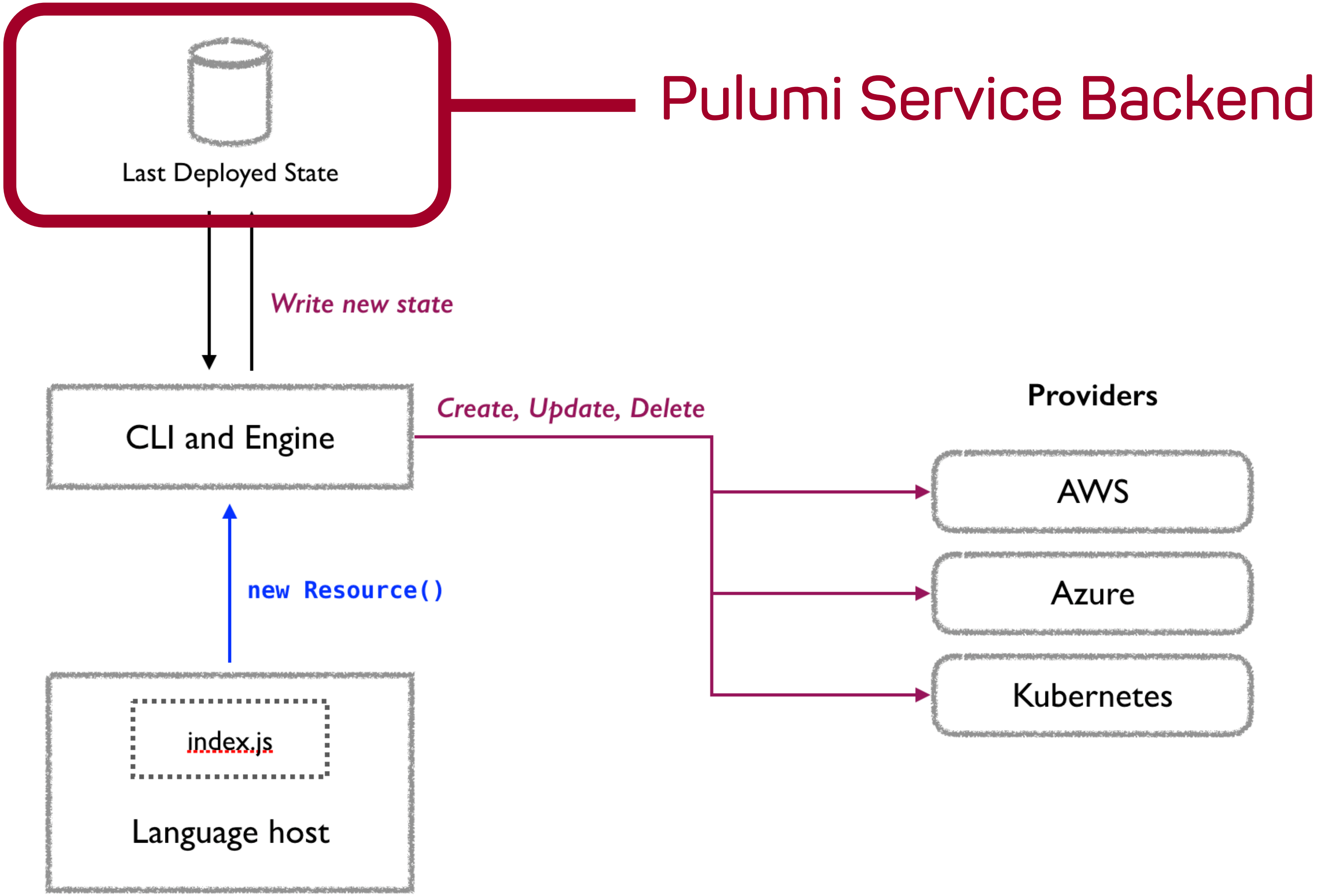
github.com/pulumi/examples

State

State

is stored in the Pulumi

Service Backend



pulumi.com/docs/intro/concepts/how-pulumi-works

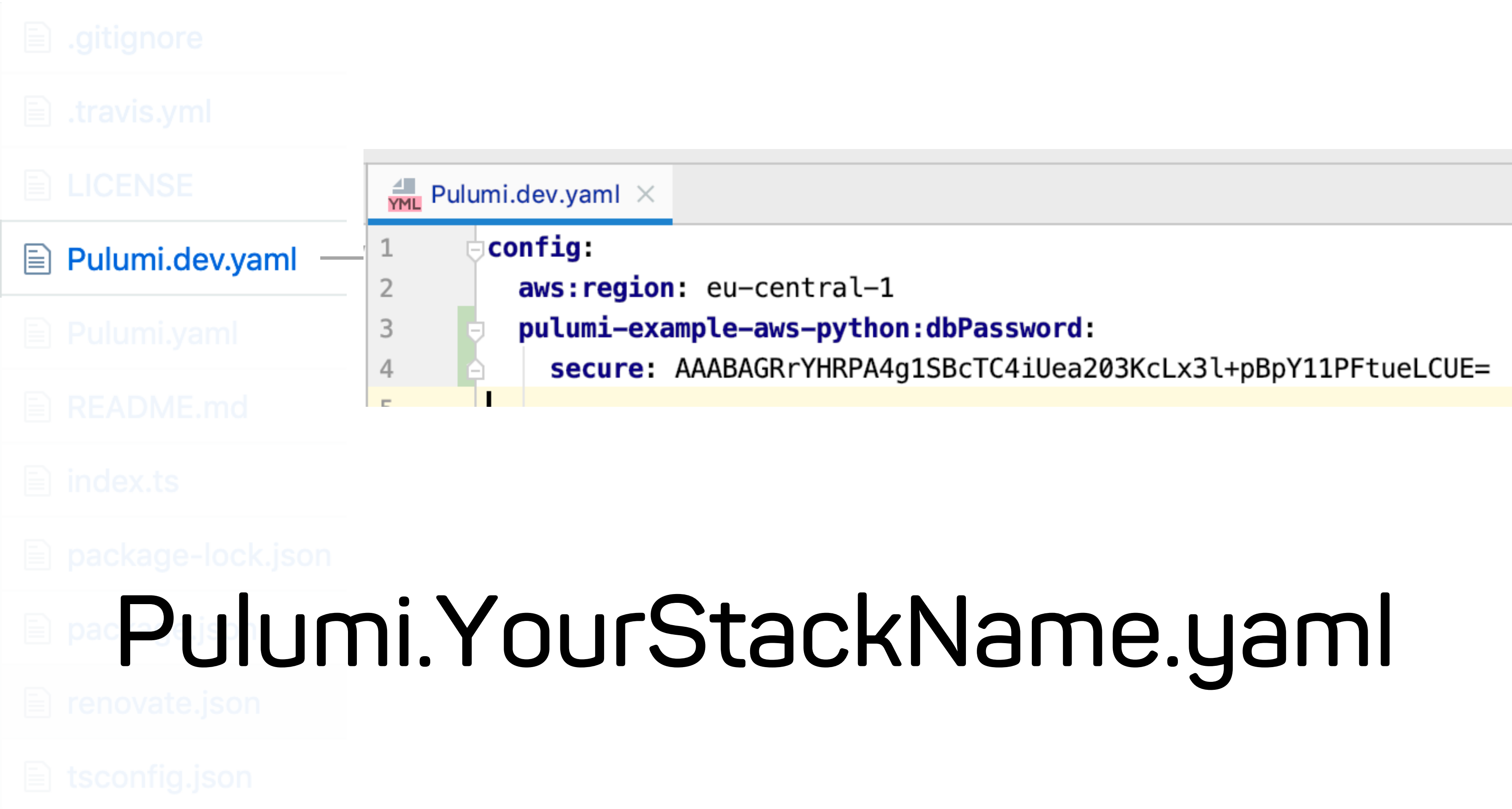
Service Backend options

- app.pulumi.com
- Self-managed backend
- on-premise app.pulumi.com
(Enterprise version only)

pulumi.com/docs/intro/concepts/state/

A Stack defines the State of a Pulumi project

Example: app.pulumi.com/jonashackt



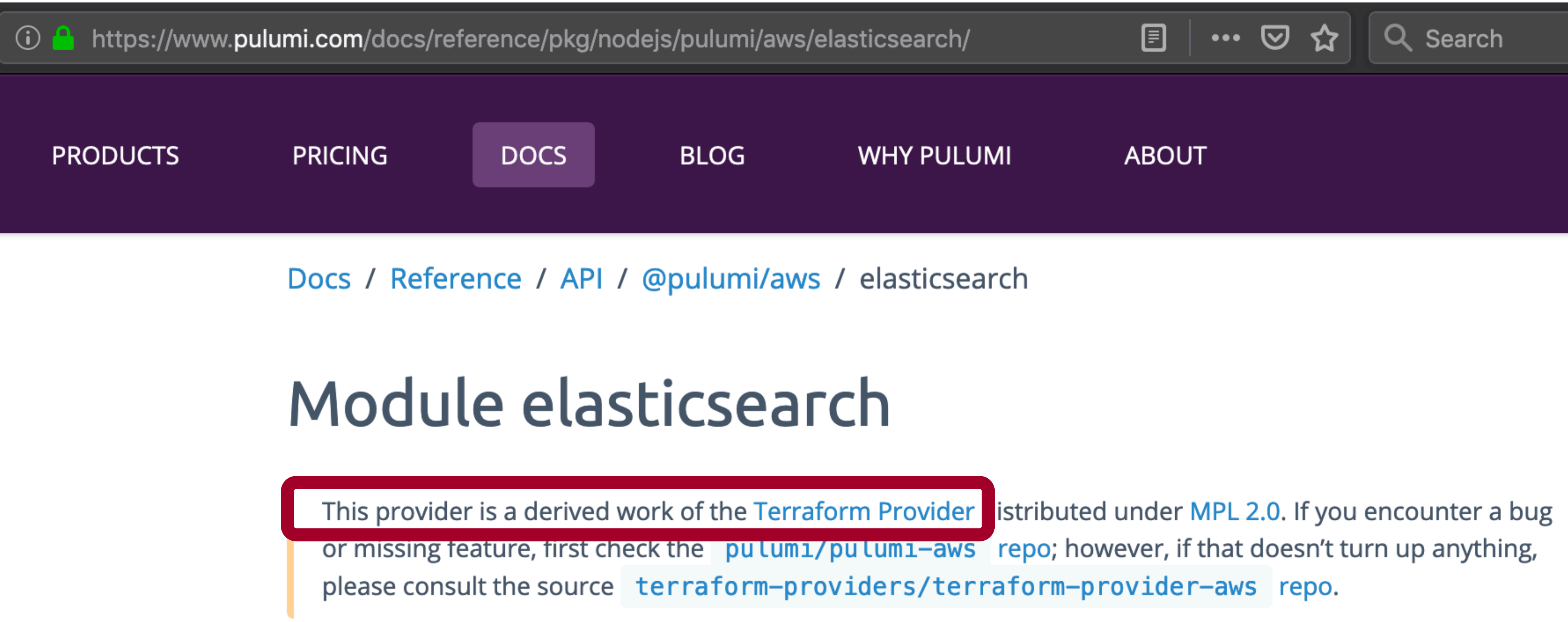
```yaml
config:
  aws:region: eu-central-1
  pulumi-example-aws-python:dbPassword:
    secure: AAABAGRrYHRPA4g1SBcTC4iUea203KcLx3l+pBpY11PFtueLCUE=
```

Pulumi.YourStackName.yaml

Resource Providers

aka

„Terraform Wrappers“

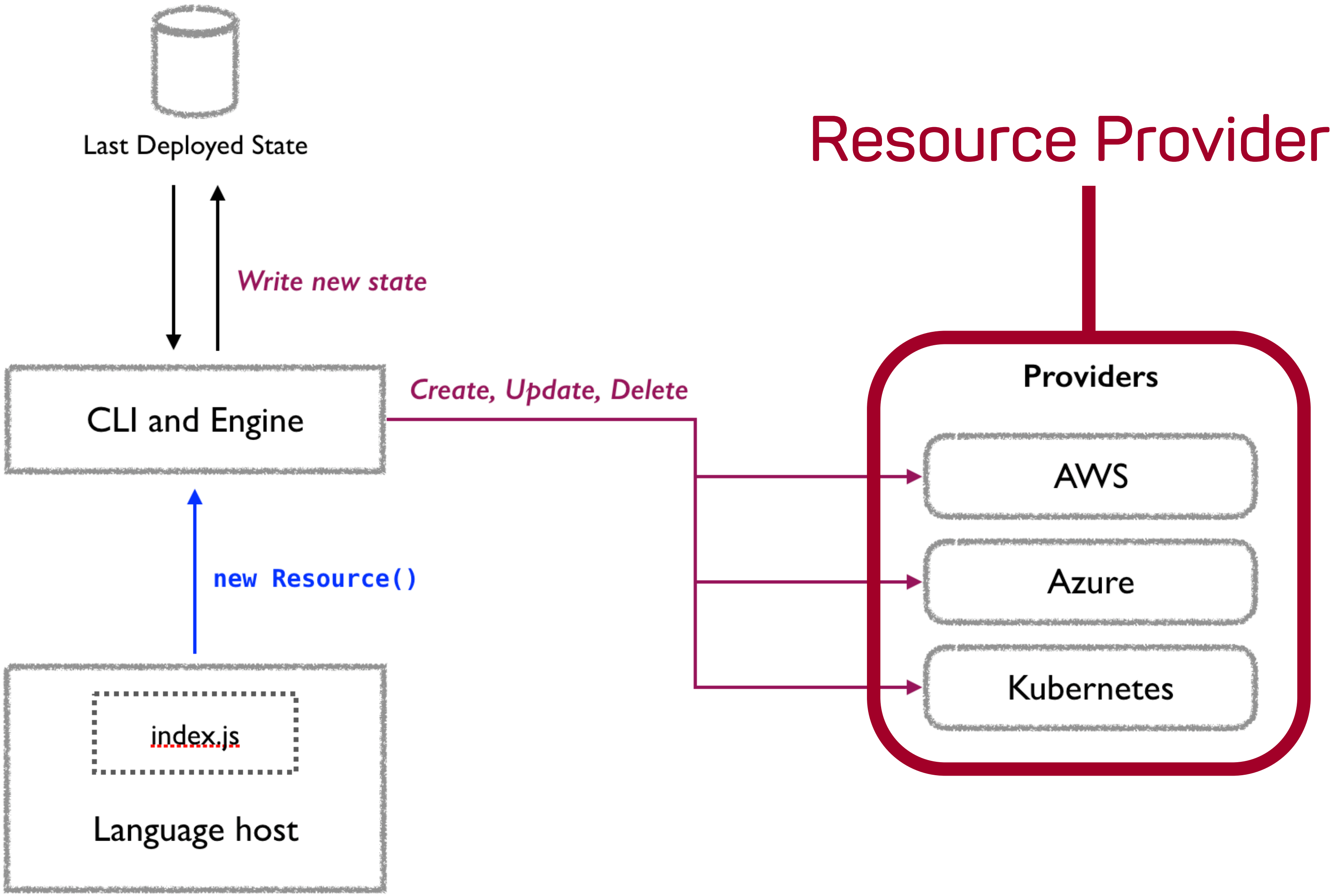

https://www.pulumi.com/docs/reference/pkg/nodejs/pulumi/aws/elasticsearch/
Search
PRODUCTS
PRICING
DOCS
BLOG
WHY PULUMI
ABOUT
Docs / Reference / API / @pulumi/aws / elasticsearch

Module elasticsearch

This provider is a derived work of the Terraform Provider istributed under MPL 2.0. If you encounter a bug or missing feature, first check the `pulumi/pulumi-aws` repo; however, if that doesn't turn up anything, please consult the source `terraform-providers/terraform-provider-aws` repo.

🤔

Resource Providers



Last Deployed State

Write new state

CLI and Engine

Create, Update, Delete

`new Resource()`

index.js

Language host

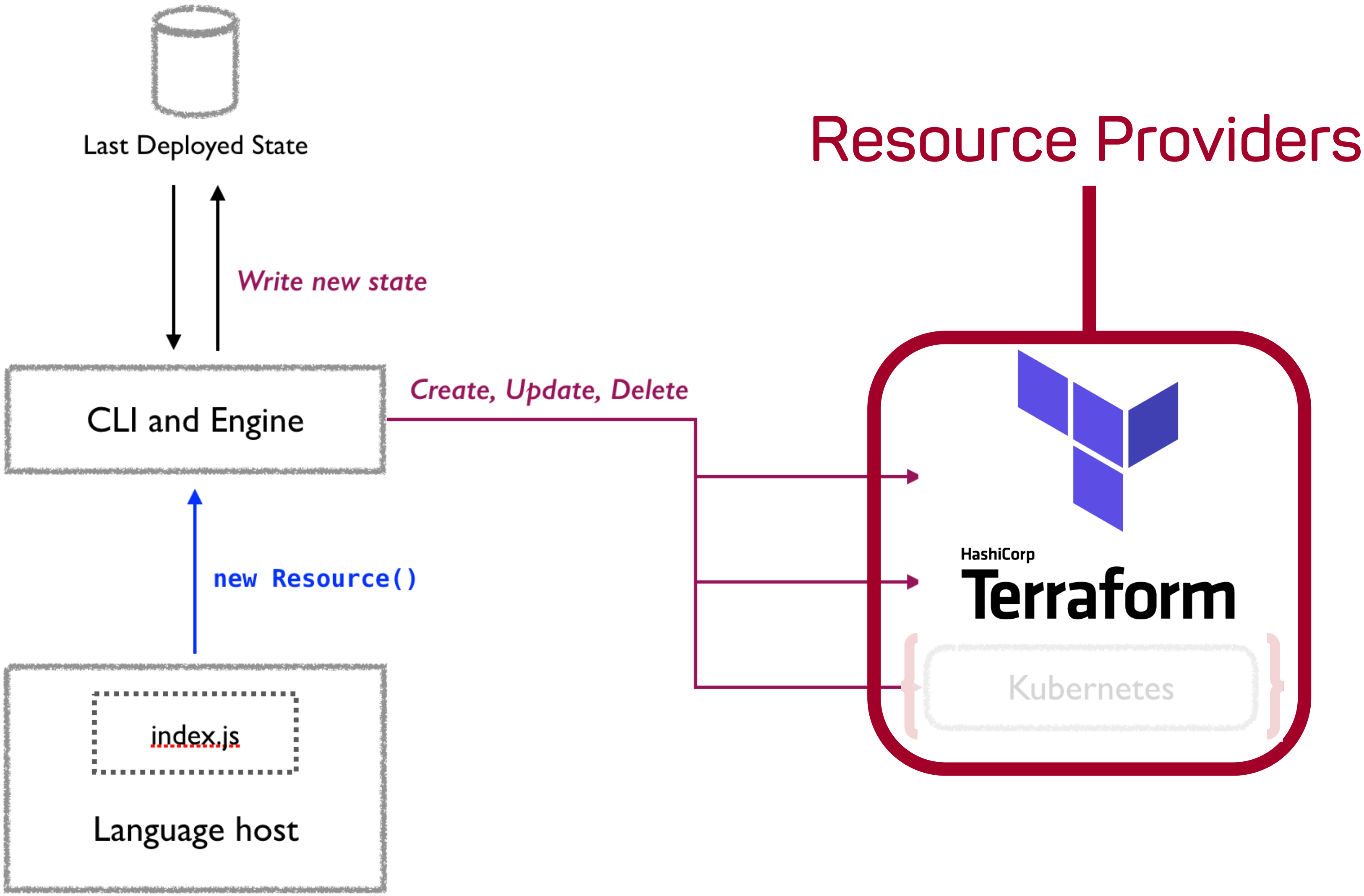
Providers

AWS

Azure

Kubernetes

pulumi.com/docs/intro/concepts/how-pulumi-works

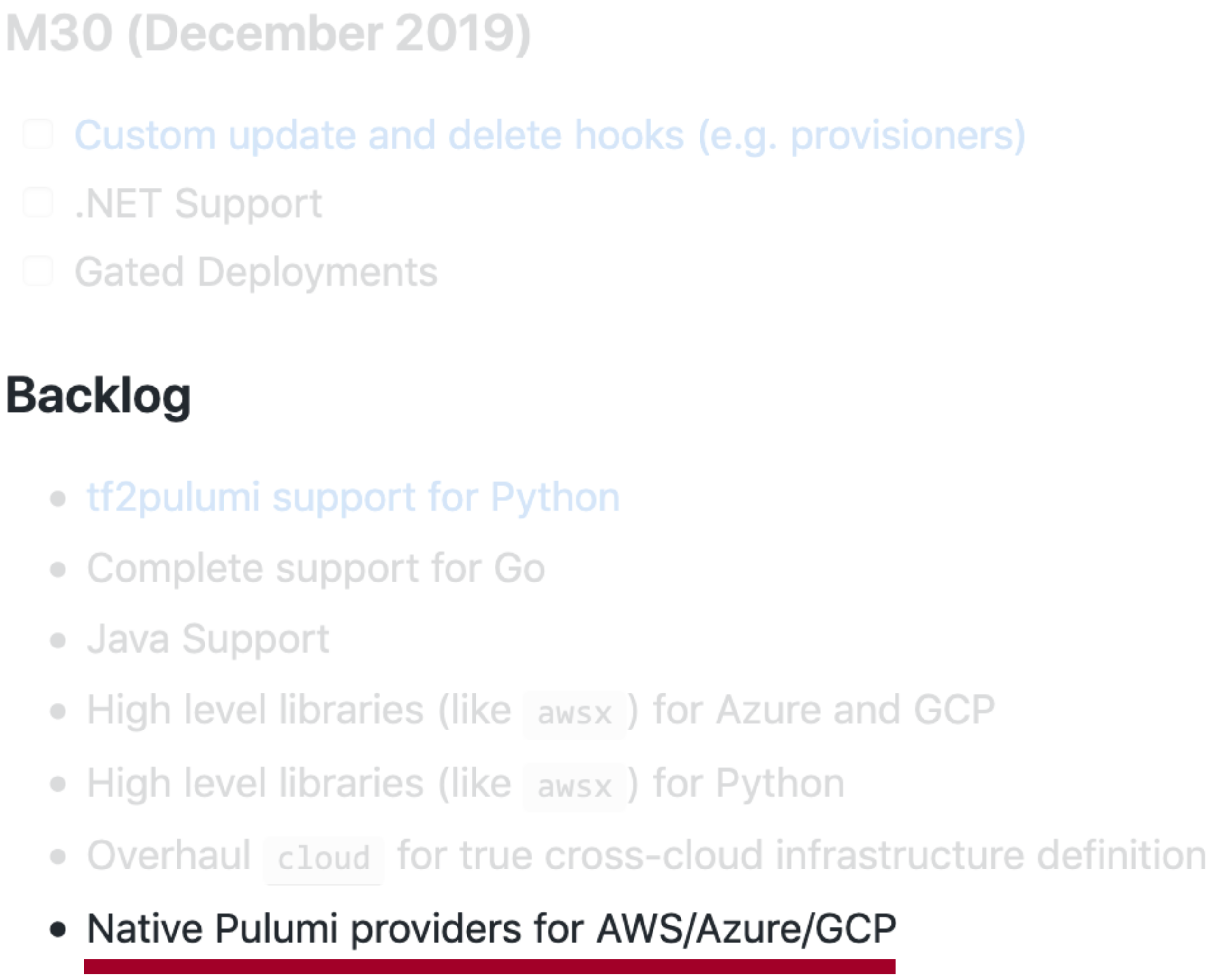


pulumi.com/docs/intro/concepts/how-pulumi-works

pulumi = „Terraform Wrapper“

2019! (except Kubernetes provider)

pulumi.com/docs/intro/vs/terraform/#converting-from-terraform



M30 (December 2019)

- □ Custom update and delete hooks (e.g. provisioners)
- □ .NET Support
- □ Gated Deployments

Backlog

- tf2pulumi support for Python
- Complete support for Go
- Java Support
- High level libraries (like `awsx`) for Azure and GCP
- High level libraries (like `awsx`) for Python
- Overhaul `cloud` for true cross-cloud infrastructure definition
- Native Pulumi providers for AWS/Azure/GCP

github.com/pulumi/pulumi/wiki/Roadmap

Pulumi vs. X

pulumi

CHEF

SALTSTACK

?

ANSIBLE

HashiCorp
Terraform

puppet

CloudFormation

pulumi

vs.

ANSIBLE

1. Academic comparison
2. Handle configuration drift
3. Current state of my infrastructure?
4. Tools shouldn‘t suck! (no master!, no agents!)
5. Able to do the job! (all major Clouds & on-premise)

1. Academic comparison

Configuration Management

puppet

CHEF

SALTSTACK

ANSIBLE

Provisioning

HashiCorp Terraform

CloudFormation

pulumi

ANSIBLE

docs.ansible.com/ansible/latest/modules/list_of_cloud_modules.html

> Pulumi is fundamentally different than these tools and works great alongside them

pulumi.com/docs/intro/vs/

Configuration Management

Provisioning

ANSIBLE + pulumi

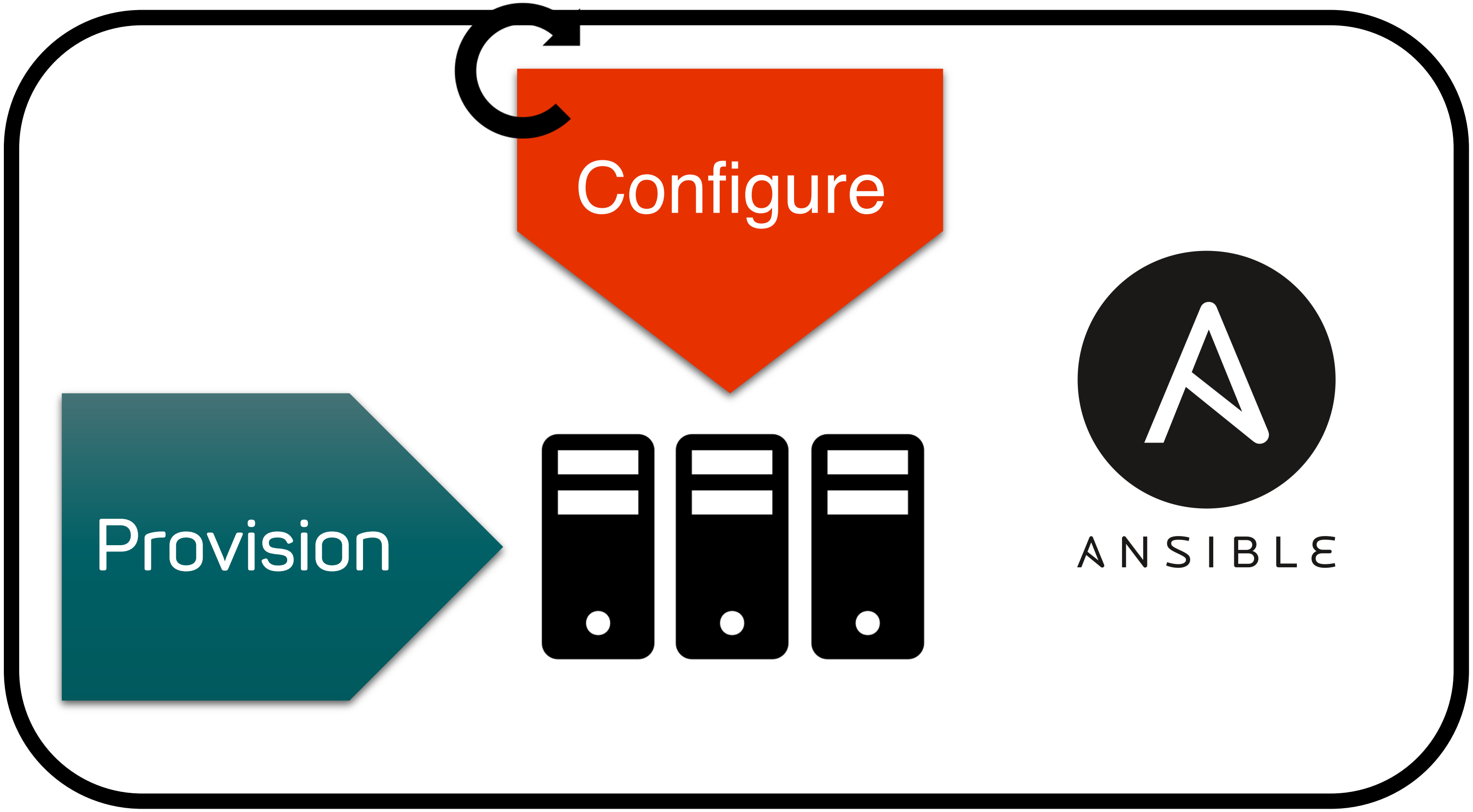
😳

2. How do they handle configuration drift?

> the state of the machine drifts from the baseline due to manual changes and updates

shadow-soft.com/ansible-idempotency-configuration-drift

Mutable Infrastructure



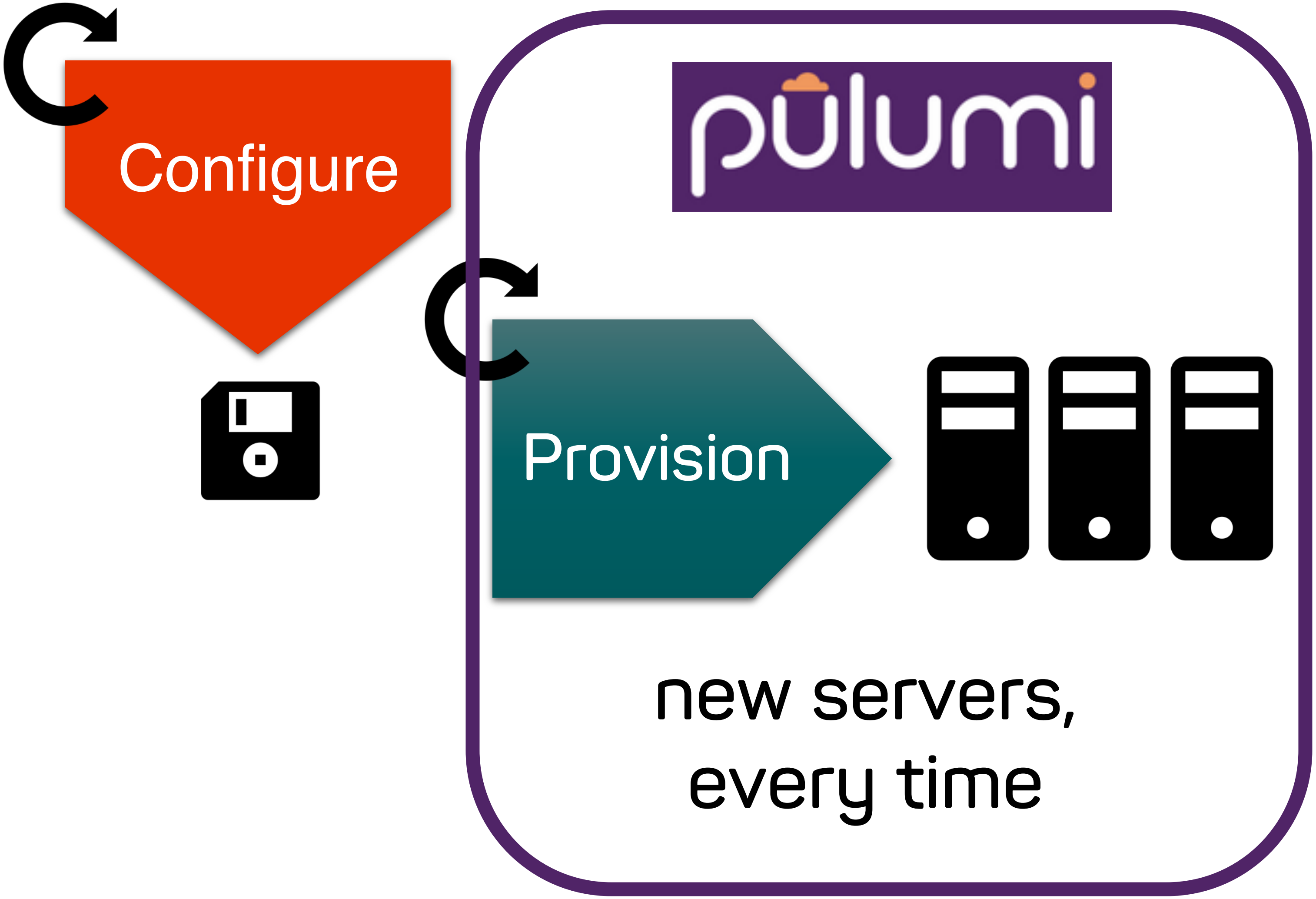
Configure

Provision

ANSIBLE

same servers,
changed every time

Immutable Infrastructure



Configure

pulumi

Provision

new servers,
every time

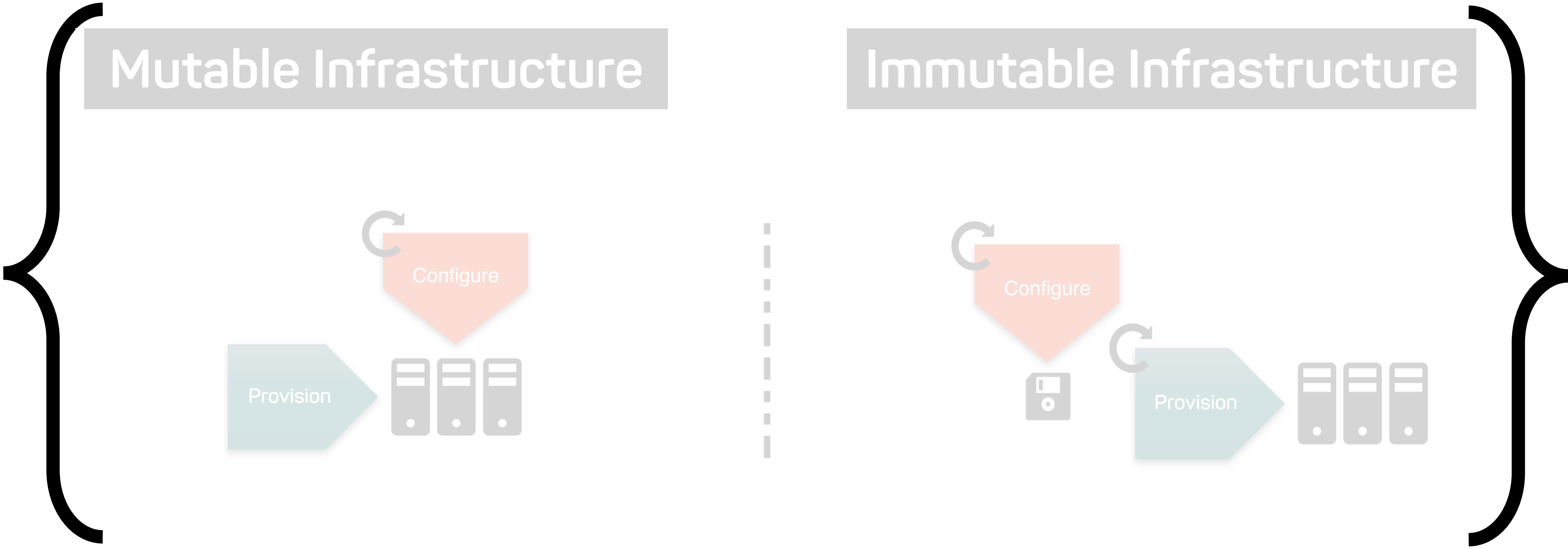
What if we don‘t allow „manual changes“?

Use software engineering practices

like

Continuous Integration

for your infrastructure code!



Mutable Infrastructure

Configure

Provision

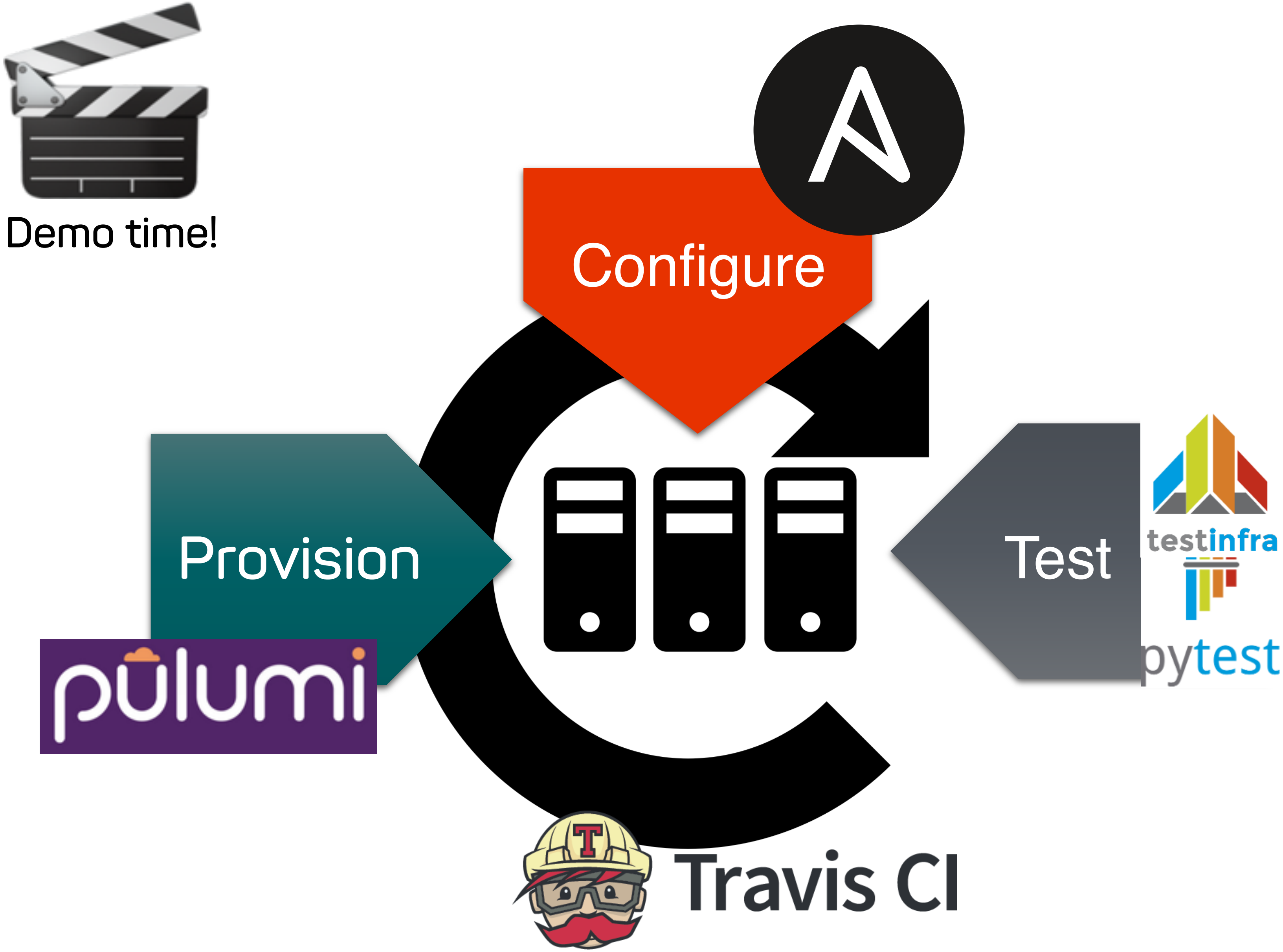
Immutable Infrastructure

Configure

Provision

Running our Infrastructure code in **CI/CD pipelines**, this comparison become less relevant!

Demo time!



Configure

Provision

Test

github.com/jonashackt/pulumi-python-aws-ansible

3. Current state of my infrastructure?

There‘s this 3rd comparison called Procedural vs. Declarative…

But remember the

<u>Continuous Integration</u>

thingy?

4. Tools shouldn‘t suck

(no master!, no agents!)

master

- central place to see status of your infrastructure
- continuously enforce configuration in the background

- extra infrastructure!
- needs to be maintained
- client-2-master & master-2-servers communication needs ports & security

no master

Remember the

Continuous Integration

Thingy again?

master

puppet

CHEF

SALTSTACK

pulumi

no master 🤩

HashiCorp Terraform

CloudFormation

ANSIBLE

Agent needs to be installed on the server

puppet

CHEF

SALTSTACK

No Agent needed 🤩

HashiCorp Terraform

ANSIBLE

pulumi

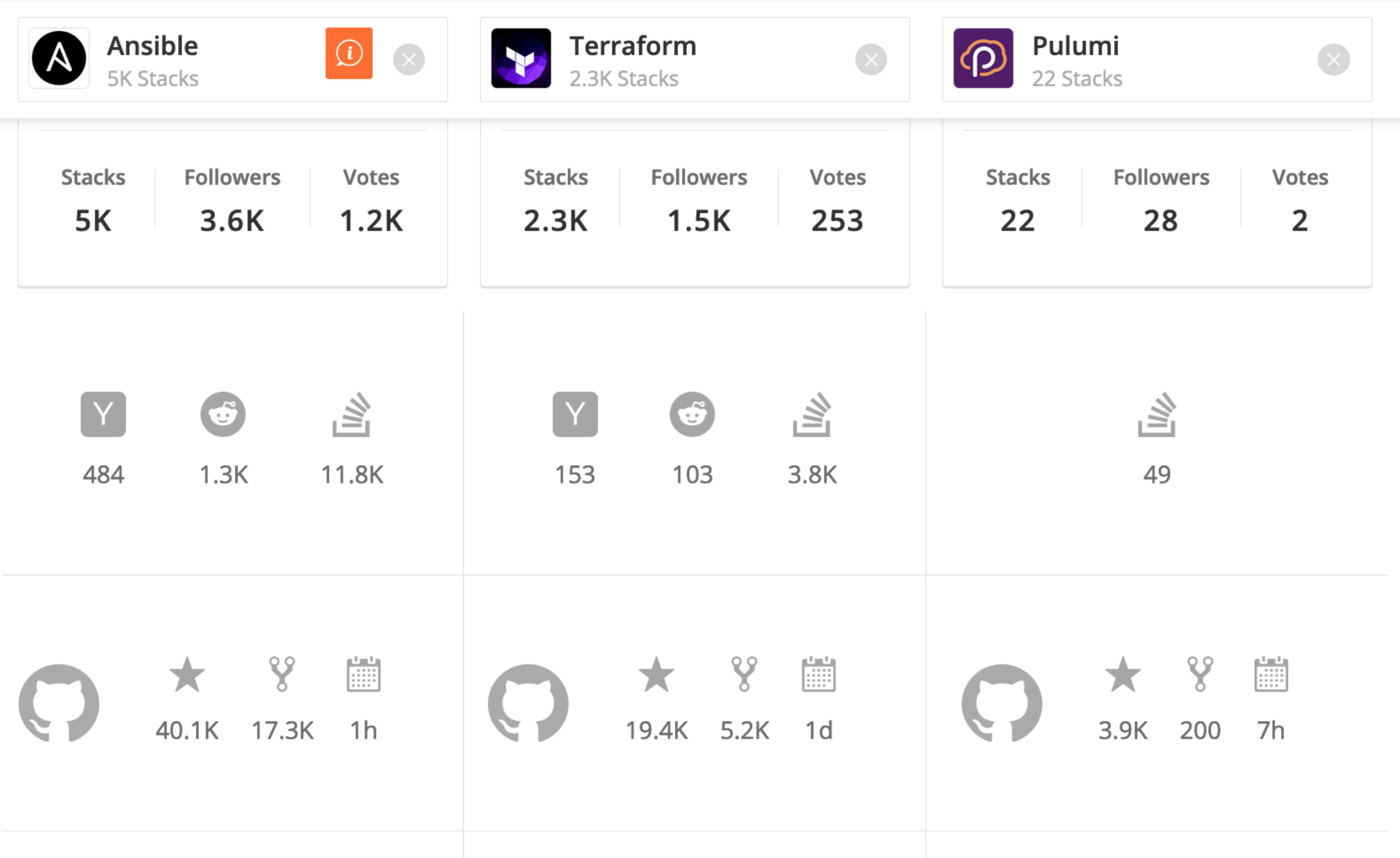
5. Able to do the job!

(all major Clouds & on-premise)

ANSIBLE

pulumi

🤔



Ansible
5K Stacks

Stacks	Followers	Votes
5K	3.6K	1.2K

Y 484 1.3K 11.8K

★ 40.1K 17.3K 1h

Terraform
2.3K Stacks

Stacks	Followers	Votes
2.3K	1.5K	253

Y 153 103 3.8K

★ 19.4K 5.2K 1d

Pulumi
22 Stacks

Stacks	Followers	Votes
22	28	2

49

★ 3.9K 200 7h

stackshare.io/stackups/ansible-vs-terraform-vs-pulumi

💪 google.com 🥺

💪 stackoverflow.com 🥺

💪 documentation 🥺

💪 blog & articles count 🥺

💪 community size 🥺

💪 on-premise also 🥺

💪 Both config mgt. & provisioning 🥺

48,130 commits

💪

4,738 contributors

4,046 commits

🥺

51 contributors

One more thing...

pulumi crosswalk

FOR AMAZON WEB SERVICES (AWS)

JS

”

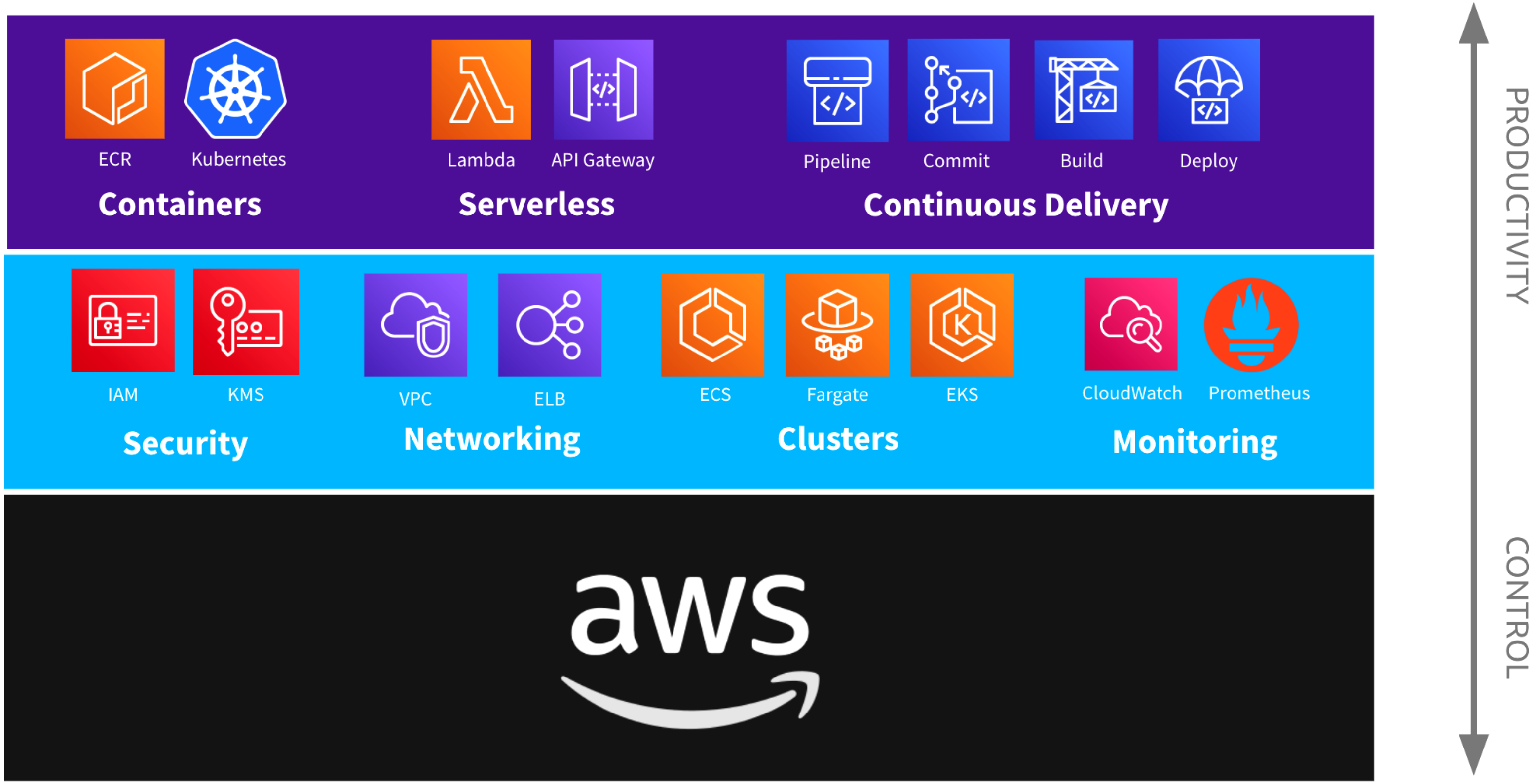
Crosswalk for AWS supports “day one” tasks, such as creating your initial container-based workloads using ECS/Fargate/EKS & serverless workloads (API Gateway / Lambda)

pulumi.com/docs/guides/crosswalk/aws

APPLICATION
DELIVERY
Go to production with containers and serverless.

INFRASTRUCTURE
PATTERNS
Well-Architected infrastructure for modern applications.

FOUNDATION
ALL OF AWS
Unopinionated infrastructure as code for all AWS resources and features.



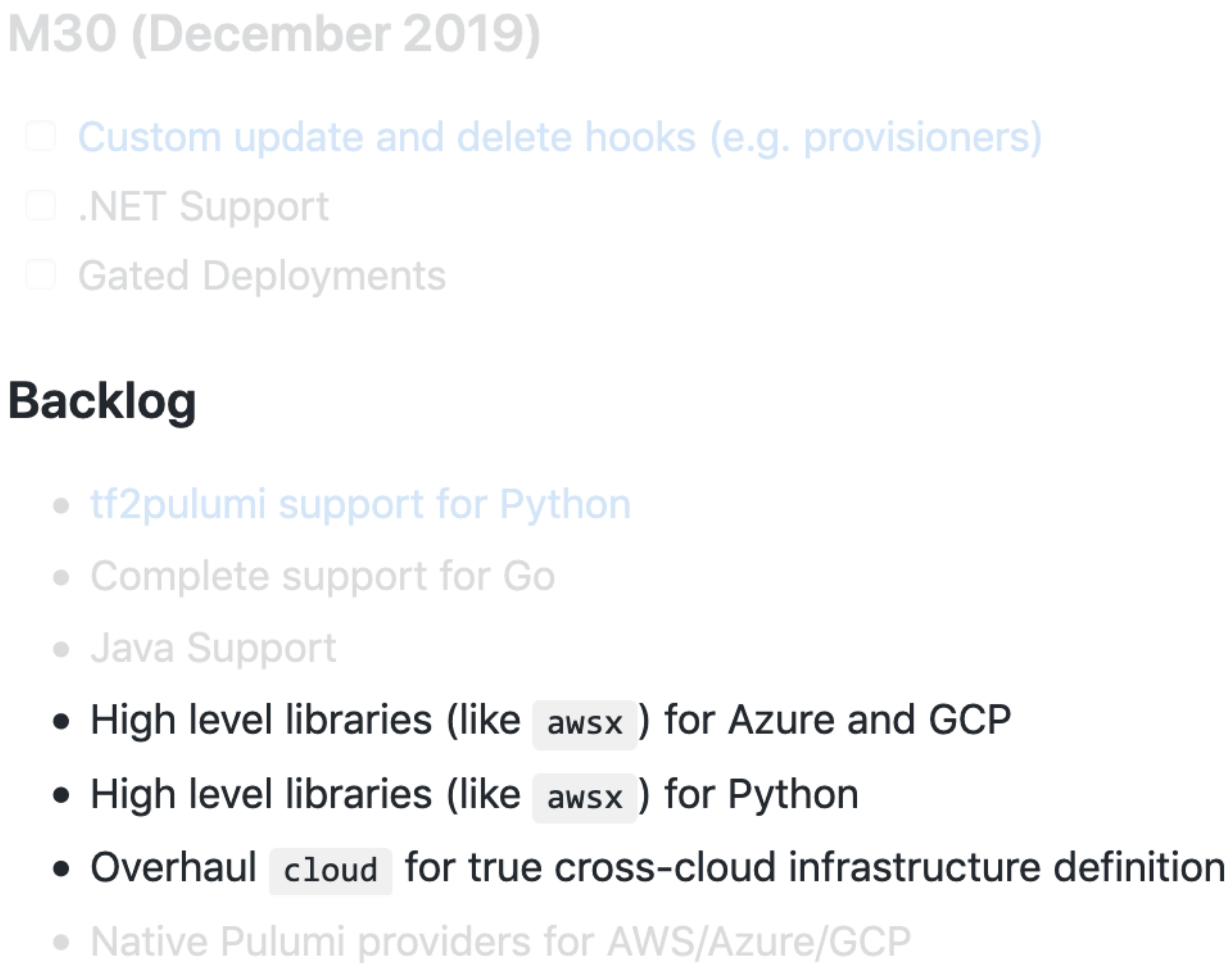
pulumi.com/docs/guides/crosswalk/aws

pulumi crosswalk
FOR AMAZON WEB SERVICES (AWS)

JS

github.com/jonashackt/pulumi-typescript-aws-fargate

App: github.com/jonashackt/spring-boot-vuejs



M30 (December 2019)

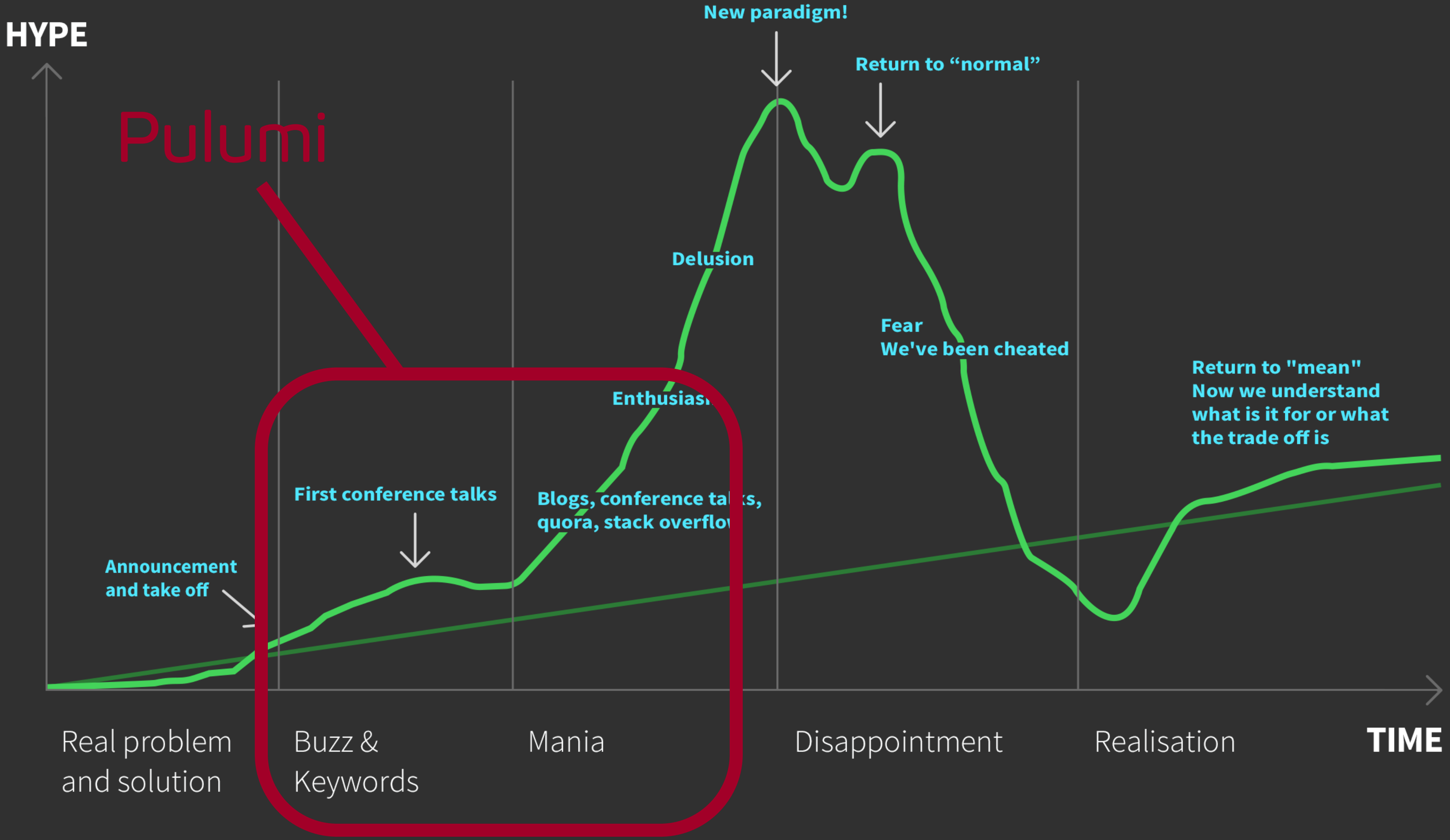
- □ Custom update and delete hooks (e.g. provisioners)
- □ .NET Support
- □ Gated Deployments

Backlog

- tf2pulumi support for Python
- Complete support for Go
- Java Support
- High level libraries (like `awsx`) for Azure and GCP
- High level libraries (like `awsx`) for Python
- Overhaul `cloud` for true cross-cloud infrastructure definition
- Native Pulumi providers for AWS/Azure/GCP

github.com/pulumi/pulumi/wiki/Roadmap

Summary



blog.daftcode.pl/hype-driven-development-3469fc2e9b22

pulumi
2019

aws 🤔

JS 🤔

HashiCorp
Terraform 🤔

pulumi
crosswalk 😍
FOR AMAZON WEB SERVICES (AWS)

No matter what IaC tool you use…

Treat your infrastructure code
AS CODE!

Use Testframeworks – or even do **TDD!**

Run Tests automatically - in your
Continuous Integration Pipeline!

Run your infrastructure code **frequently!**
(scheduled CI jobs)

Always aim for **reproducible builds** (aka
dependency management)

Automatically update dependencies
(e.g. renovatebot)

github.com/jonashackt/pulumi-talk

@jonashackt