



**Terraform is dead.**  
**Long live Pulumi.**

Putting the **Code** Back Into Infrastructure  
As **Code** (IaC)

# This Talk Isn't



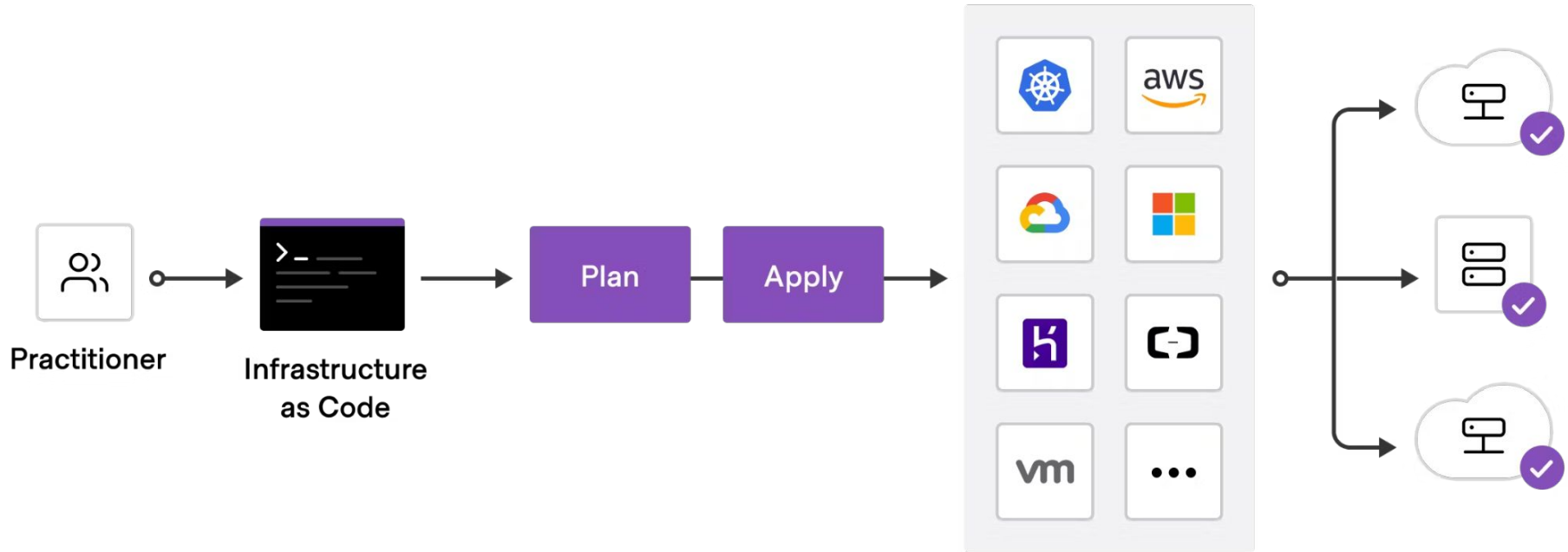
- Terraform == bad
- `drop(TF) || Pulumi.replace(TF)`
- `self.learn(DevOps) || demo(Pulumi)`

# Outcome Of This Talk



- Handling opposition
- Highlight features
- Adopting

# Infrastructure as Code (IaC) 101



# My Experience



## Terraform

5 years, 2018

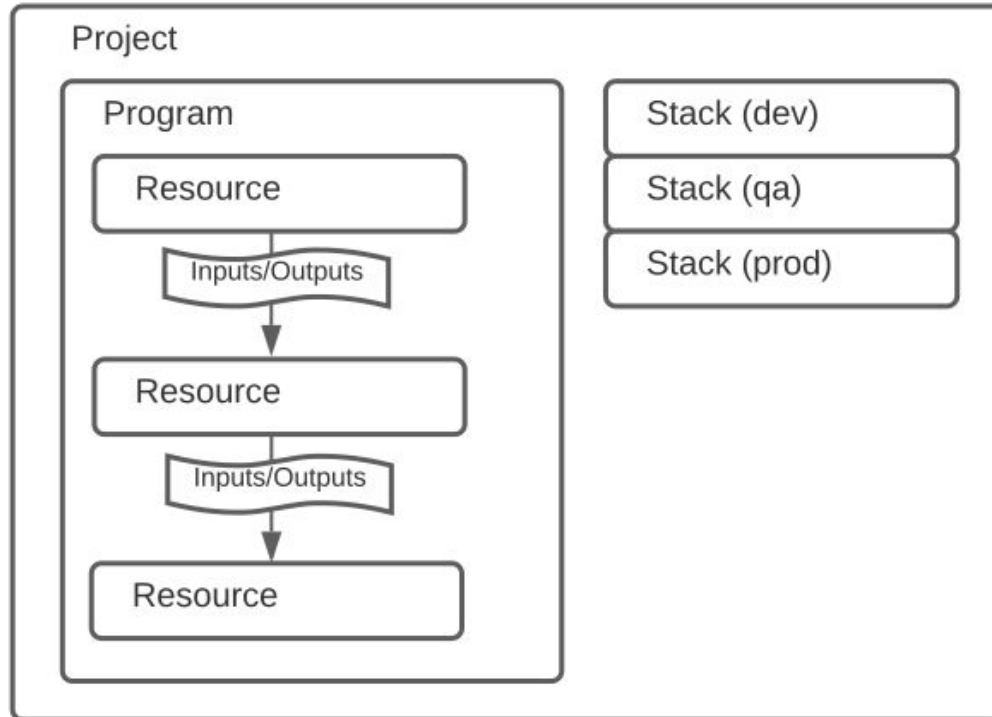
AWS, GCP

## Pulumi

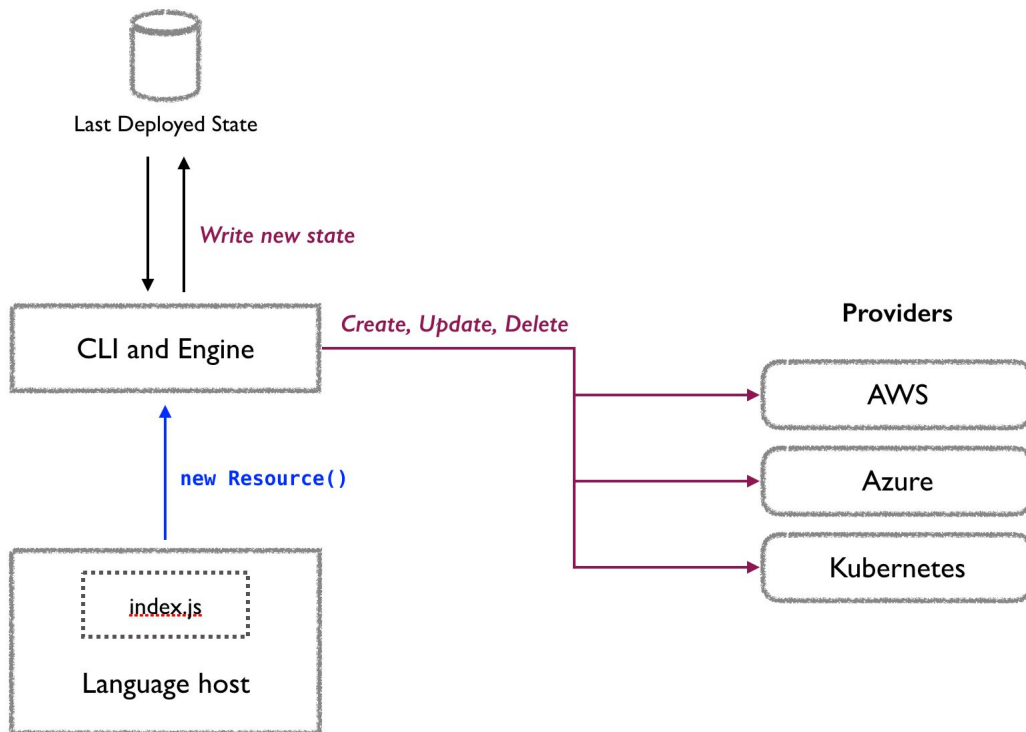
+1 years, 2022

AWS, Civo,  
Scaleway, Proxmox

# Cover A Bit Of Pulumi



# The Flow



# No Domain Specific Language (DSL)



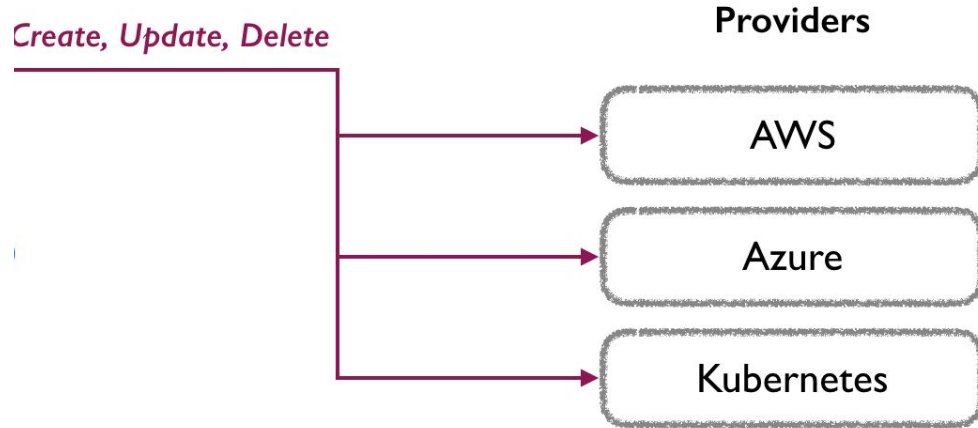
Uses General Purpose Languages (GPL)



# Quick Pulumi Typescript Look



# Dynamic Providers



Write One



Tf Pain



# Delivery Time: Impacts



- Development tooling, gone
- Package management, gone
- Normal autocomplete/doc, gone
- No cross pollination with HCL<>Code

# Missing local functions



Have a common function?

```
corp_format(project, tag, blah)
```

Pulumi: We can write it.

# What About CDKTF?



Cloud Development Kit for Terraform lets you use familiar programming languages to define and provision infrastructure.

# Lag on provider changes

## Add tags argument into aws lb data source #12265



leoskyrocker opened this issue on Mar 4, 2020 · 15 comments · Fixed by [#6458](#)



**ewbankkit** mentioned this issue on Jul 23, 2021

**Find aws\_lb by tags in data source #6458**



**ewbankkit** closed this as completed in [#6458](#) on Jul 23, 2021

# Pulumi: Generate from API docs

# Secrets in state



```
> cat terraform.tfstate

{
  "version": 4,
  "terraform_version": "0.12.4",
  "serial": 13,
  "lineage": "6395ed5e-49d9-d469-2bc4-75c44303c8b7",
  "outputs": {},
  "resources": [
    {
      "mode": "managed",
      "type": "aws_route53_health_check",
      "name": "LouisJohnBichardRoute53HealthCheck",
      "provider": "provider.aws",
      "instances": [
        {
          "schema_version": 0,
          "attributes": {
            "child_health_threshold": 0,
            "child_healthchecks": [],
            "cloudwatch_alarm_name": null,
            "cloudwatch_alarm_region": null,
```

Pulumi: State secrets are encrypted



## Missing Tf API Itself



```
tf.init() && tf.plan()
```

```
if(deploy) : tf.up()
```

```
If (destroy): tf.destory()
```

Pulumi: Automation API exists.



# Pulumi's Automation API



# Pulumi: Multi-language



Imagine:

- DevOps team can write X language
- Application team can write Y language

# Pulumi: As Rest API



Imagine:

- Ease of creating self service platform
- Just like normal HTTP services

# Pulumi: SSH Tunnel



Imagine:

- Spin up on-demand secure box
- Connect through & then run IaC



# Adopting



## Coexist: Pulumi + X



```
new tf.state.RemoteStateReference()
```

```
data "pulumi_stack_outputs" "stack_outputs" {}
```

[Pulumi Docs](#)

# Import



`pulumi import --from terraform` to adopt all resources from an existing `.tfstate` file.

[Pulumi Docs](#)



# Convert



Any\* Terraform HCL to Pulumi code using  
`pulumi convert --from terraform`

[Pulumi Docs](#)



# Closing



## Closing: Not All Rainbows & Sunshine



- Abstraction “hell” + Flexibility
- Two things to maintain & know
- Time migrating + onboarding

# Closing: Handling Opposition



- Butter knife vs steak knife
- HCL is only valuable for Terraform
- Fallacy that 2 things is always worse

# Closing: Highlight features



- Secrets are encrypted
- Normal dev IDE, autocomplete, packages
- Automation API

# Closing: Adopting & Transitioning



- Brand new project. Ideally Stateless.
- Pulumi TF bridge
- Leaving TF in spots is fine.

# Questions?



Slides: <https://shorturl.at/tuTX9>

Pulumi Examples: <https://shorturl.at/kosT5>