

Infrastructure as Code with Terraform

Giovanni Toraldo

Lead developer @ClouDesire

Twitter: @gionn

#coderstug #apericoder



TERRAFORM

[INTRO](#) [DOCS](#) [COMMUNITY](#)

[↓ DOWNLOAD](#)

[GITHUB](#)



Build, Combine, and Launch Infrastructure_

What is Terraform?

Like vagrant, but for production

Features

- Build, change, destroy resources:
 - IaaS: AWS, DigitalOcean, GCE, OpenStack
 - PaaS: Heroku, CloudFoundry
 - SaaS: Atlas, DNSimple, CloudFlare
- Current state is versioned
- Generates an execution plan to transition from current state to the desired state

Use cases

- Multi-tier applications (easy scale up/down)
- Self-service clusters (production-like)
- Demos (vagrant on clouds)
- Disposable environments (staging/QA)
- Multi-cloud environments

Demo time!

<https://gist.github.com/gionnn/787a3fc379eeac0ce01f>

How we use it in ClouDesire

- To manage infrastructure and DNS for on-demand staging and productions:
 - Before: everything by hand, using GUI
 - Now: everything with terraform, state versioned on git
- Custom script to generate ssh_config:
<https://gist.github.com/gionnn/fabbd0f6d6ad897d0338>
Used by fabric to knows on which nodes chef-solo should be launched

More?

- Modules can be created to encapsulate resources in a reusable way
- State push/pull to remote: S3, HTTP, Atlas
- Atlas integration: SaaS control room for teams, managed by HashiCorp

A deep purple spiral galaxy is centered in the frame against a black background. The galaxy's core is a bright, glowing purple oval, from which several spiral arms emerge, composed of smaller, dimmer purple points of light. The word "Cheers!" is written in a bold, white, sans-serif font, centered over the bright core of the galaxy.

Cheers!