DATASTAX

Terraform IT: Infrastructure as a Code

Aleks Volochnev & Eric Zietlow | 30.11.2020 | DataStax Monday Learning

DATASTAX

DataStax Monday Learning

Upgrade yourself, unlock new skills

- Most Important Topics
- From Engineers to Engineers
- Absolutely Free

DATASTAX

DataStax Monday Learning

Upgrade yourself, unlock new skills

- This Week: 30th of November
  - **Infrastructure as a Code**
- Next Time: 21th of December 2020
  - **Ceph: high-capacity distributed storage**

Hi, I'm Aleks! Welcome to the Workshop!

Aleks Volochnev **@HadesArchitect**

- Professional Cloud Architect
- Developer Advocate at DataStax
- "Monday Learning" Programme Editor

After many years in software development as a developer, techlead, lead DevOps engineer, and architect, Aleks focused himself on distributed applications and cloud architecture. Professional Cloud Architect and Developer Advocate at DataStax, he is happy to share his knowledge and expertise in the field of serverless and disaster tolerant systems.

Subscribe! linkedin.com/in/aleks-volochnev

Hi ! .. and I am Cedrick Lunven

CÉDRICK LUNVEN

Director Of Developer Advocacy At Datastax

- Creator and maintainer of FF4j
- ff4j.org

- Contributor JHIPSTER
- jhipster.tech

DATASTAX

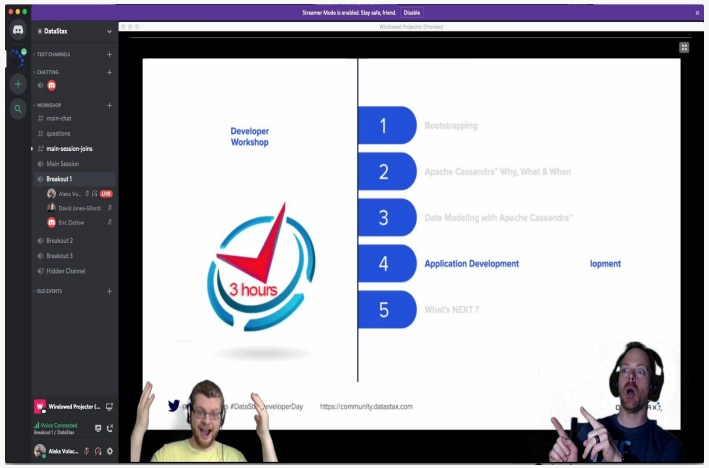
Hi I am Eric Zietlow

Developer Advocate at DataStax

- Long time member of the Cassandra community
- Technology enthusiast
- OSS contributor to multiple projects

linkedin.com/in/ericzietlow/

STREAMS



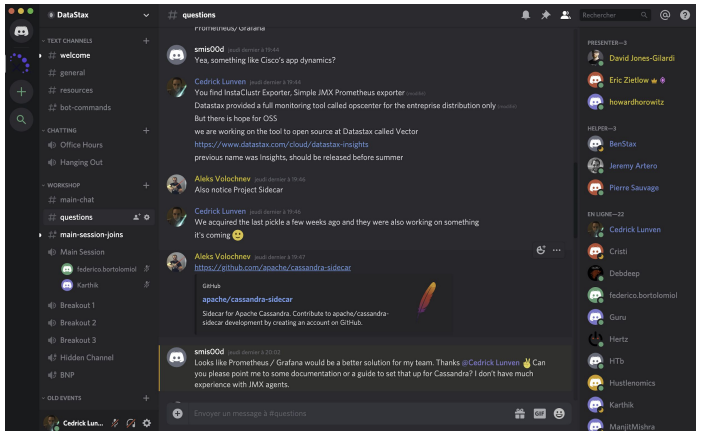
YouTube

Twitch

RUNTIME

HashiCorp Terraform

QUESTIONS



Discord

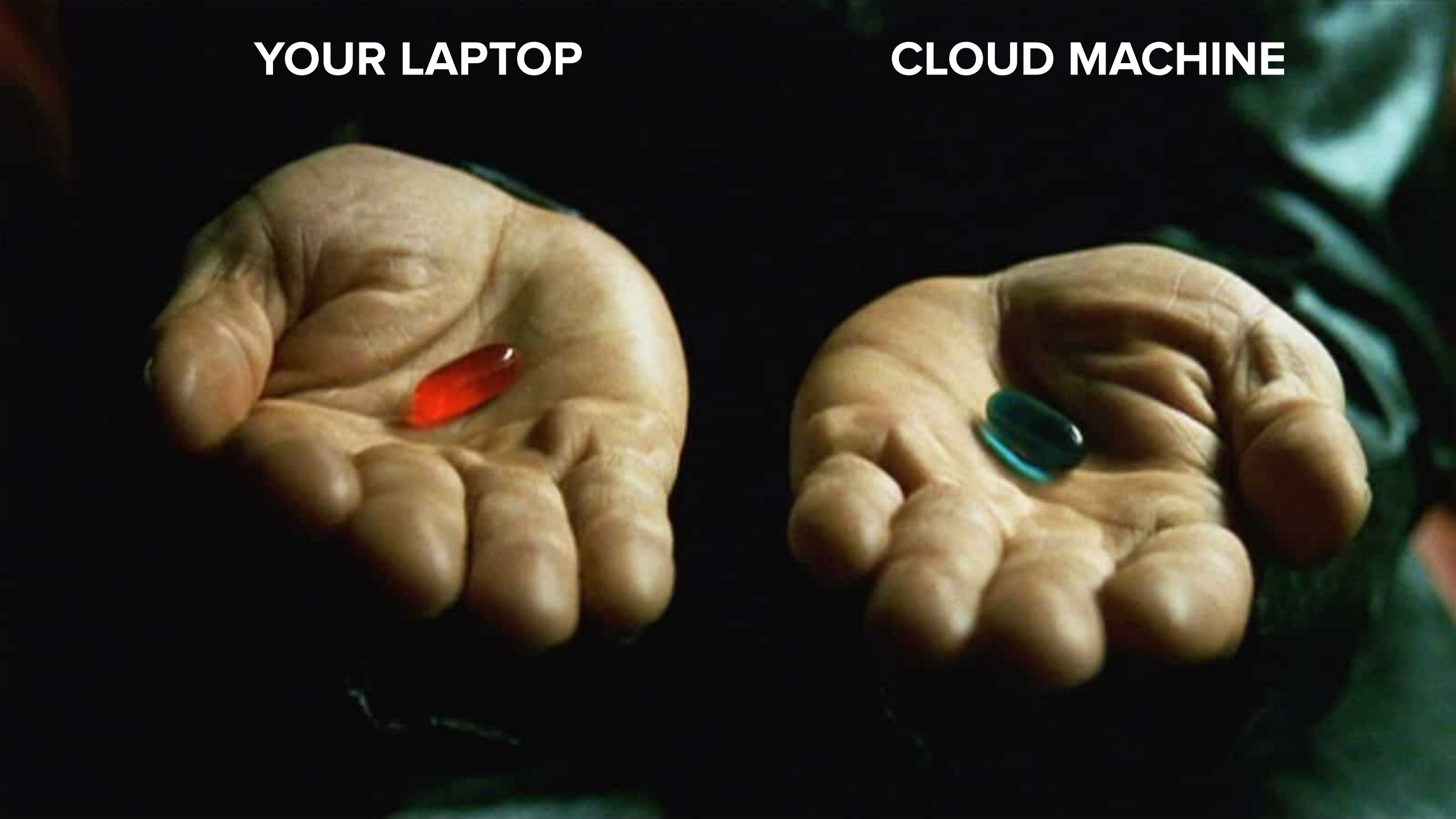
MATERIALS

Slides

Notebooks

Terraform Files

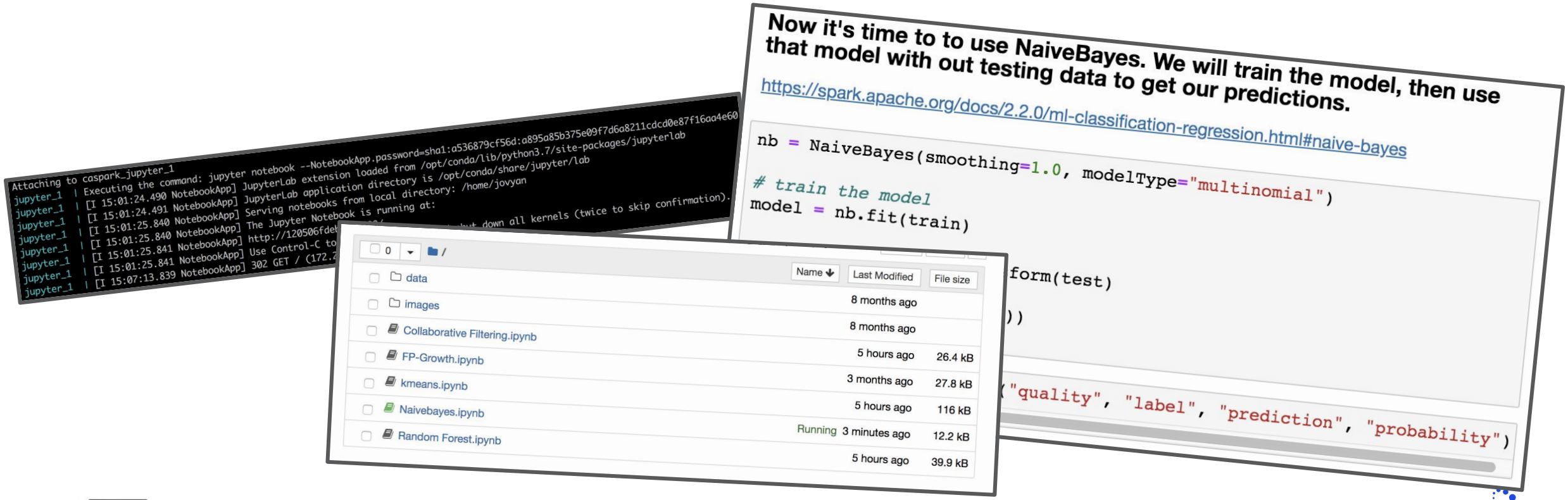
Github


YOUR LAPTOP
CLOUD MACHINE

Local Setup [optional]

You will need *docker* and *docker-compose*

- git clone https://github.com/DataStax-Academy/machine-learning-workshop-online
- cd machine-learning-workshop-online
- docker-compose pull

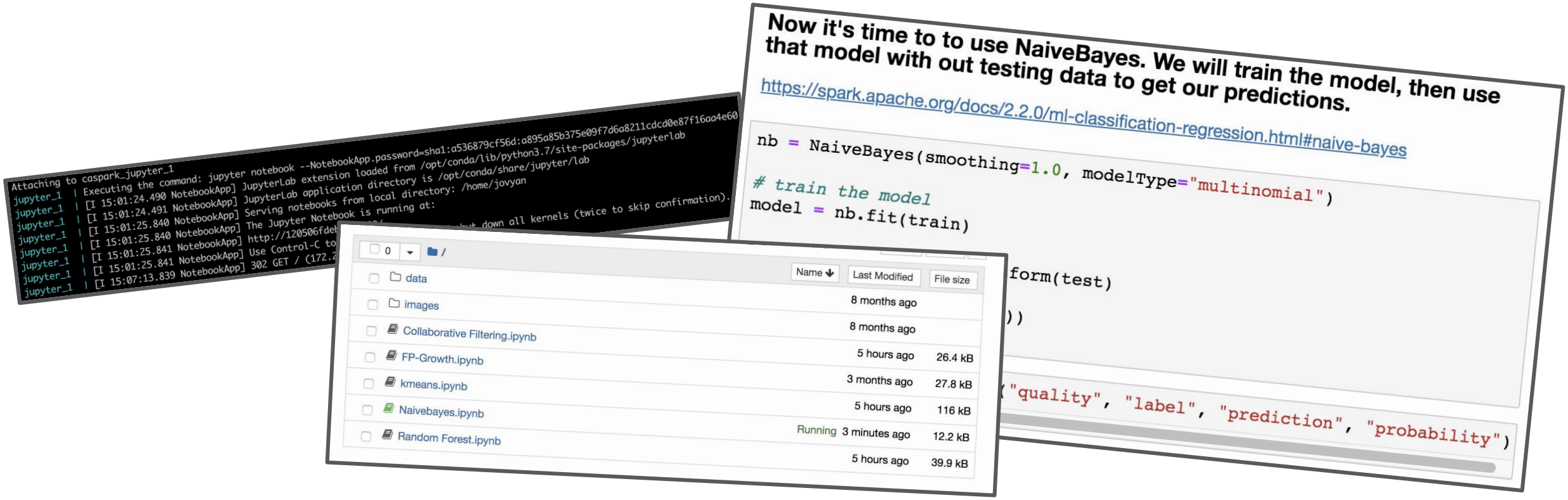


bit.ly/caspark-webinar

DATASTAX

Cloud Setup [optional]

Contact Aleks via aleksandr.volochnev@datastax.com or linkedin.com/in/volochnev/ to get your cloud instance.



bit.ly/caspark-webinar

DATASTAX

DATASTAX®

Infrastructure as a Code

"Infrastructure as a Code is the process of managing and provisioning computer data centers through machine-readable definition files, rather than physical hardware configuration or interactive configuration tools"

Infrastructure Management: Yesterday

Write a mail? Call your hosting provider?

Basically, you needed to have your own datacenter, and people installing and configuring hardware and pulling wires.

Another option would be to buy or rent servers from hosting companies like GoDaddy or Hetzner.

In any case, process and slow and complex.

Here comes the Cloud Computing

- On-demand Self Service
- Rapid Elasticity
- Resource Pooling (Shared Resources)
- API programmable access
- Pay as you go

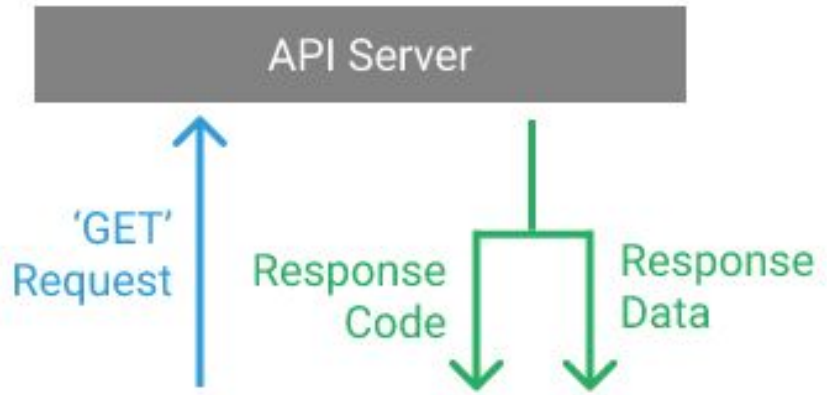
DATASTAX

Infrastructure Management: Today

Infrastructure management today more and more often involves API: instead manually pushing a server into a rack, we make an HTTP call to get a server we need.

Meanwhile Software Defined Networks simplified networking, making us pulling less wires.

In short, today we still use web interfaces, but also API calls to get what we need.

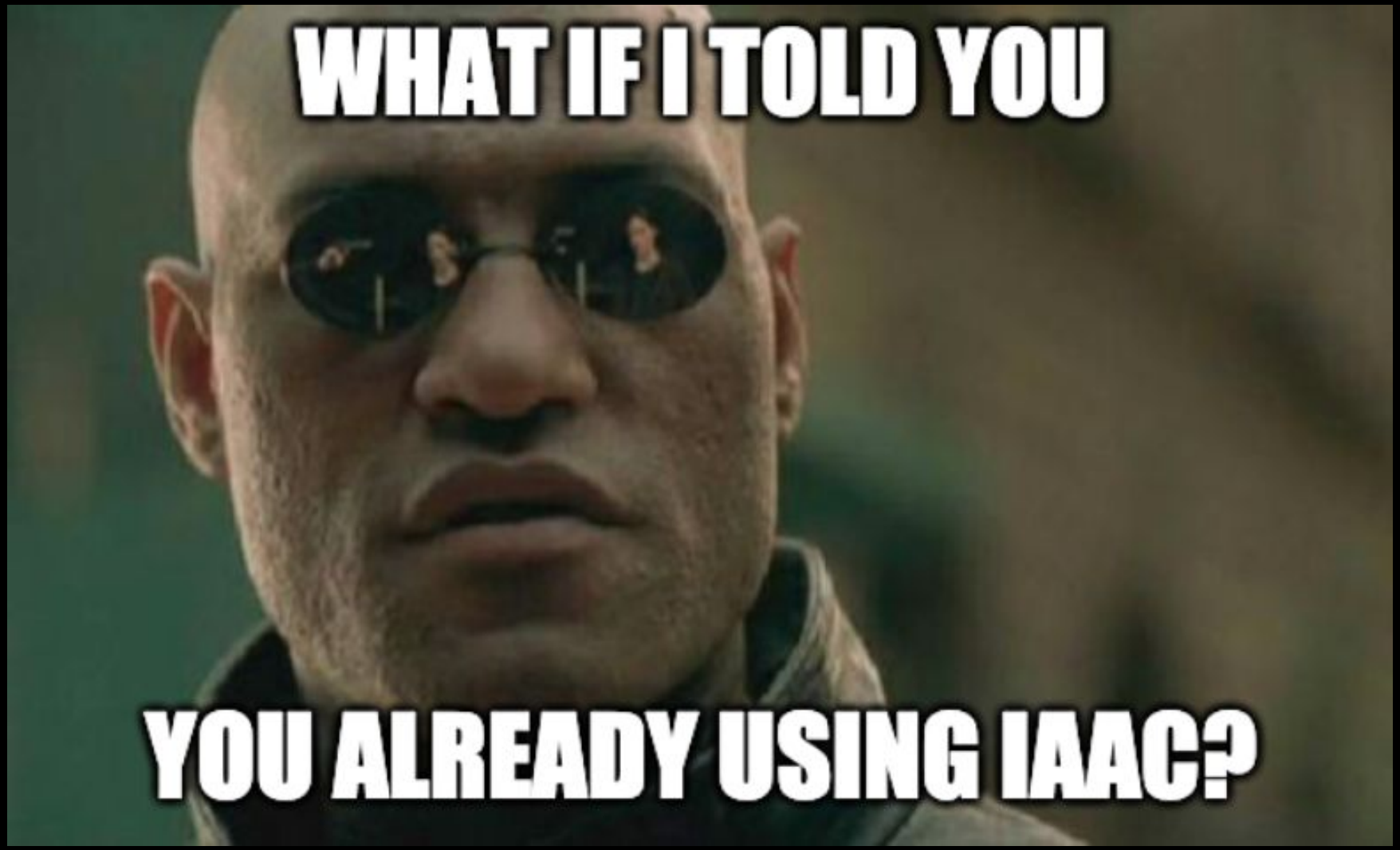


DATASTAX

“Aleks, what is Infrastructure as a Code?”

Have you ever used docker && docker-compose?

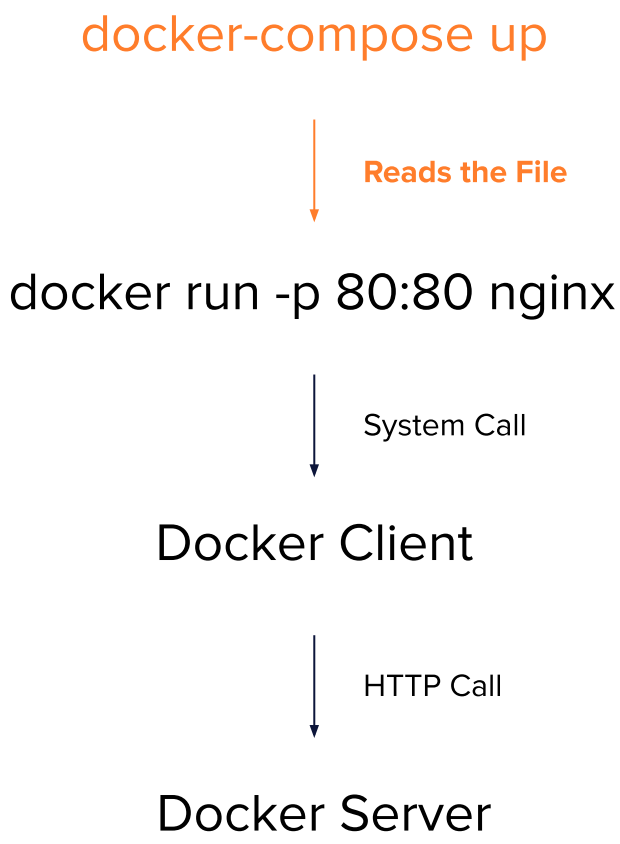
DATASTAX®


WHAT IF I TOLD YOU
YOU ALREADY USING IAAC?

Docker vs Docker-Compose

Docker Client calls API methods provided by Docker Daemon via HTTP calls.

Docker Compose reads configuration file and runs proper Docker commands.



DATASTAX

Infrastructure... as a File?

Basically, IaaC allows you to describe your Infrastructure in a file, exactly like you write your code, and then use a special tool to place API calls. We use docker-compose, docker swarm or kubernetes to do this kind of job when we operate containerised applications.

29 lines (29 sloc) | 1.42 KB

```
version: '3.3' # docker-compose config version
volumes:
    wordpress-data: {}
    mysql-data: {}
services: # let's define services
    wordpress: # First service is wordpress
        image: wordpress:5.5.1-php7.3 # image to use
        volumes: # attach volumes
            - wordpress-data:/var/www/html # Plugins and themes are managed by wordpres
        depends_on: # wait for DB to start
            - database
        ports: # publish ports
            - "8000:80"
        restart: on-failure # restart in case of failure
        environment: # set environment variables
            WORDPRESS_DB_HOST: database:3306
            WORDPRESS_DB_USER: wordpress
            WORDPRESS_DB_PASSWORD: wordpress # Don't like passwords in source code? We t
            WORDPRESS_DB_NAME: wordpress
    database: # wordpress needs a database
        image: mysql:5.7 # Let's stick to MySQL v5.7
        volumes: # Again volume to store data
            - mysql-data:/var/lib/mysql # Notice, it's a named volume, not a bind mount.
        restart: on-failure # Restart `always` will works as well but it's hard to stop
        environment: # Env vars do some configuration
            MYSQL_ROOT_PASSWORD: secretpassword
            MYSQL_DATABASE: wordpress
            MYSQL_USER: wordpress
            MYSQL_PASSWORD: wordpress
```


DATASTAX

API Calls

Target service must provide an API. As long as you have an API and a client to access it, it already sounds good. A good example for us will be an *aws cli* tool what's used to execute commands on AWS EC2 service.

Notice it's an HTTP interface so you could run the same commands with *curl* or another http-based client.

IaaC Tools

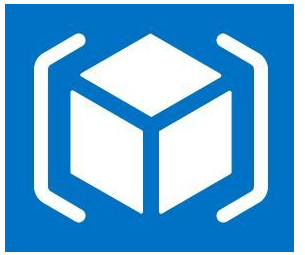
Multiple Cloud Support

Pulumi

Vendor Specific

AWS

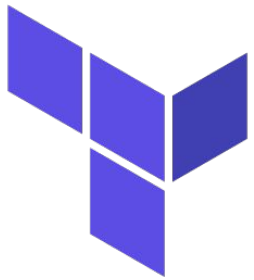
Azure



GCP

Terraform

“Terraform is an open-source Infrastructure as Code software tool created by HashiCorp. Users define and provision data center infrastructure using a declarative configuration language known as HashiCorp Configuration Language (HCL)”



HashiCorp Terraform

- Active since 2014
- Declarative
- Supports AWS, Azure, Google Cloud etc.

Sample Terraform File

This is a pretty simple example where we define two AWS security groups and an EC2 Instance.

Secgroups are to allow access to the node and the node itself runs a t2.micro instance

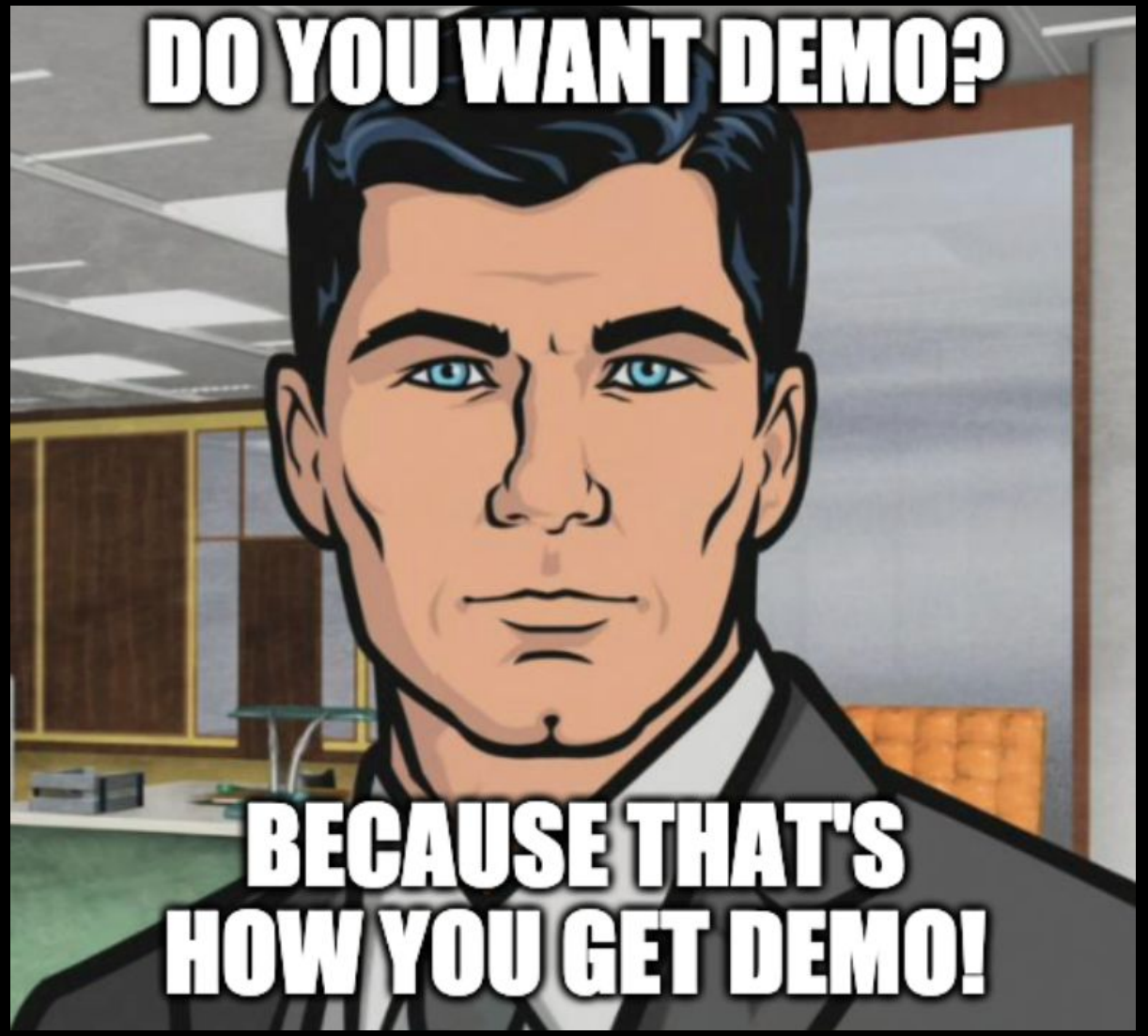
Edit file | Preview changes

```
resource "aws_security_group" "ssh" {
  name = "allow_ssh"
  ingress {
      from_port = 22
      to_port = 22
      protocol = "tcp"
      cidr_blocks = ["0.0.0.0/0"]
  }
}
resource "aws_security_group" "http" {
  name = "allow_http"
  ingress {
      from_port = 80
      to_port = 80
      protocol = "tcp"
      cidr_blocks = ["0.0.0.0/0"]
  }
}
resource "aws_instance" "voting" {
  ami           = "ami-cfca25a0"
  instance_type = "t2.micro"
  security_groups = ["${aws_security_group.ssh.name}", "${aws_security_group.http.name}"]
  key_name        = "${aws_key_pair.deployer.key_name}"
  connection {
    user = "core"
  }
  provisioner "remote-exec" {
    inline = [
      "docker run -dp 80:80 -e REDIS_HOST=${aws_instance.redis.private_ip} ditmc/voting",
    ]
  }
}
```

Light, Camera, Action!

- terraform init - to initialise a project
- terraform plan - to overview changes and see if you are satisfied
- terraform apply - to apply the changes
- terraform destroy - to clean up the workspace

DATASTAX


DO YOU WANT DEMO?
BECAUSE THAT'S
HOW YOU GET DEMO!

New Features

- Observability
- Traceability
- Fast Deployment
- Reproducibility
- Rollback Capability

“Forewarned is Forearmed”

- 6.2k forks, 24.6k stars, 1.5k contributors… still in development.
- “If you don’t understand how AWS works, Terraform will not make your life easier. Indeed, it might make it worse.” Henrique Barcelos
- Terraform definitions are cloud-specific
- State update is very slow on large infrastructures.

IaaC vs Configuration Management

IaaC is not the same as Configuration Management, and Terraform not to replace Ansible or Chef.

Terraform is not to provision servers (except of maybe very initial touch) but to deploy infrastructure. You still need Configuration Management tools to have your servers ready to serve.

DATASTAX®

Thank You!
Questions?