

# Terraform: Getting Started

Introducing IaC and Terraform



**Ned Bellavance**

HashiCorp Certified Instructor

[nedinthecloud.com](http://nedinthecloud.com)

**Provisioning infrastructure  
through software to  
achieve consistent and  
predictable environments.**



# Core Concepts of IaC

Defined in code

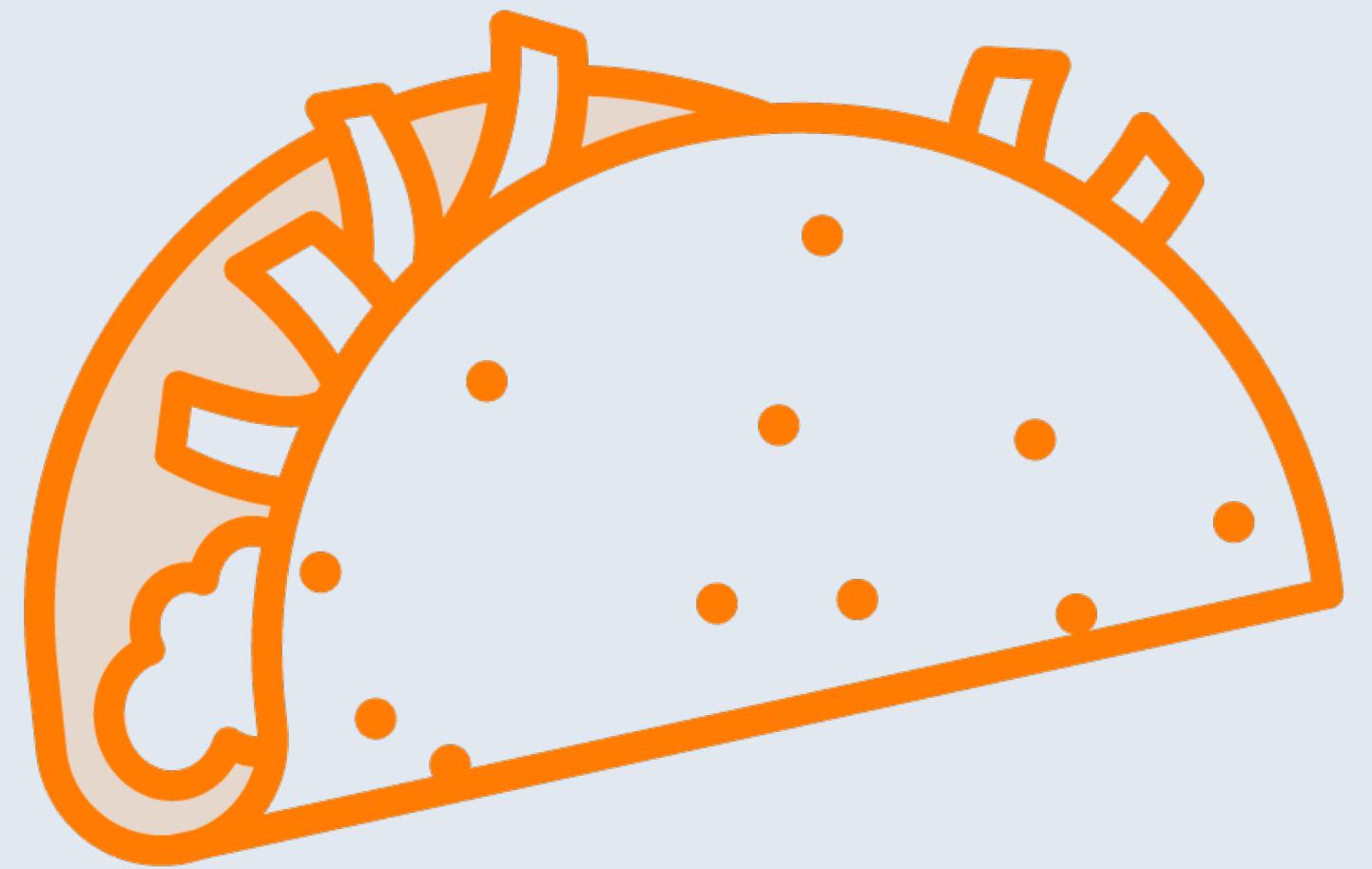
Stored in source control

Declarative vs. imperative



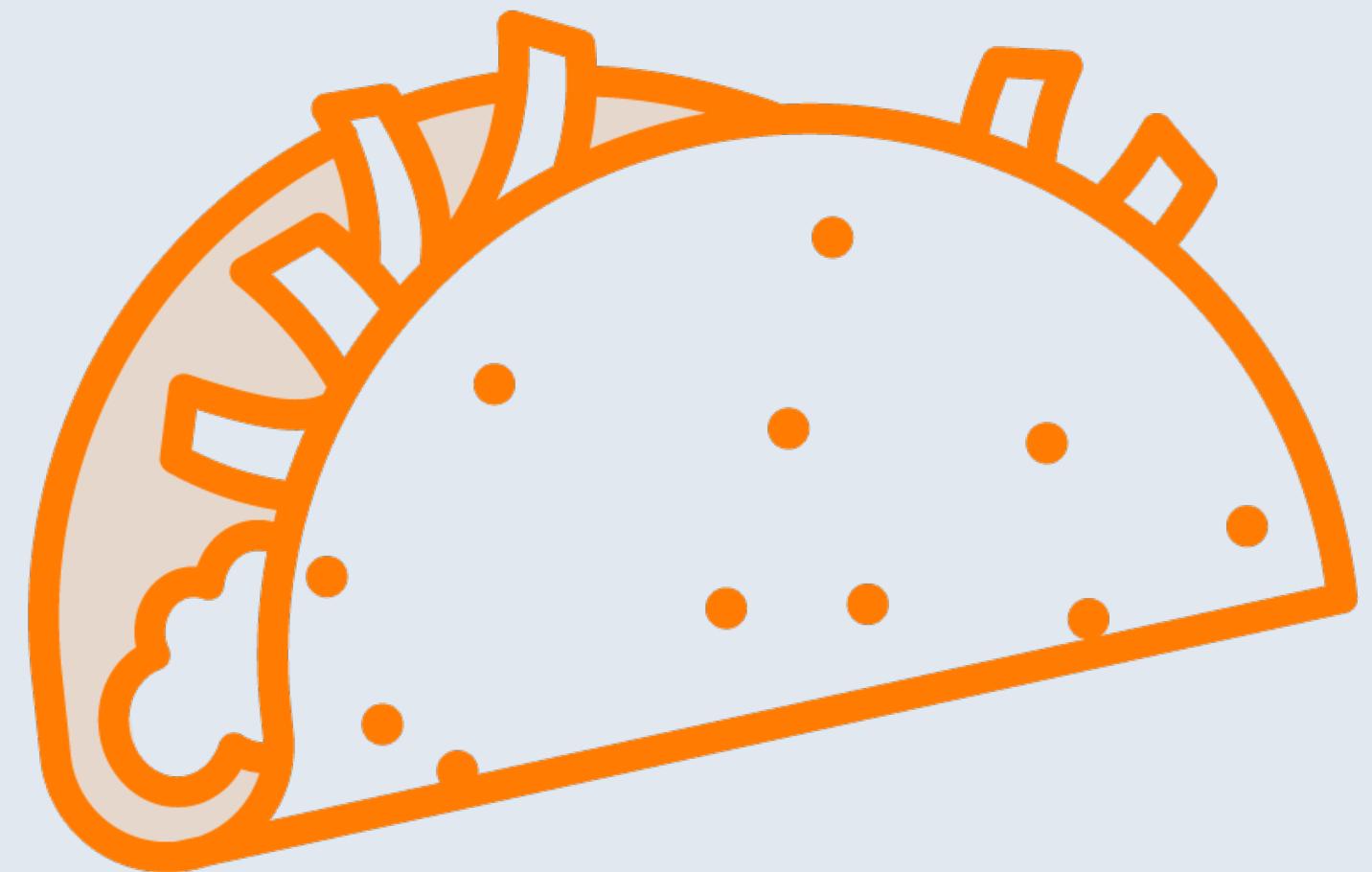
# Declarative or Imperative

```
# Make me a taco  
get shell  
get beans  
get cheese  
get lettuce  
get salsa  
  
put beans in shell  
put cheese on beans  
put lettuce on cheese  
put salsa on lettuce
```



# Declarative or Imperative

```
# Make me a taco
food "taco" "bean_taco" {
  ingredients = [
    "beans", "cheese",
    "lettuce", "salsa"
  ]
}
```



# Core Concepts of IaC

**Defined in code**

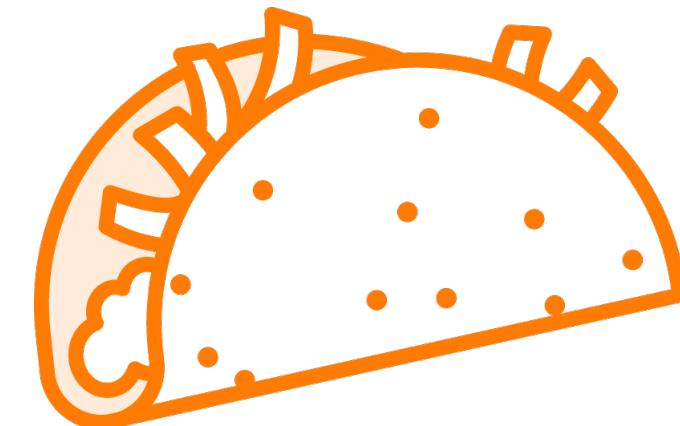
**Stored in source control**

**Declarative vs. imperative**

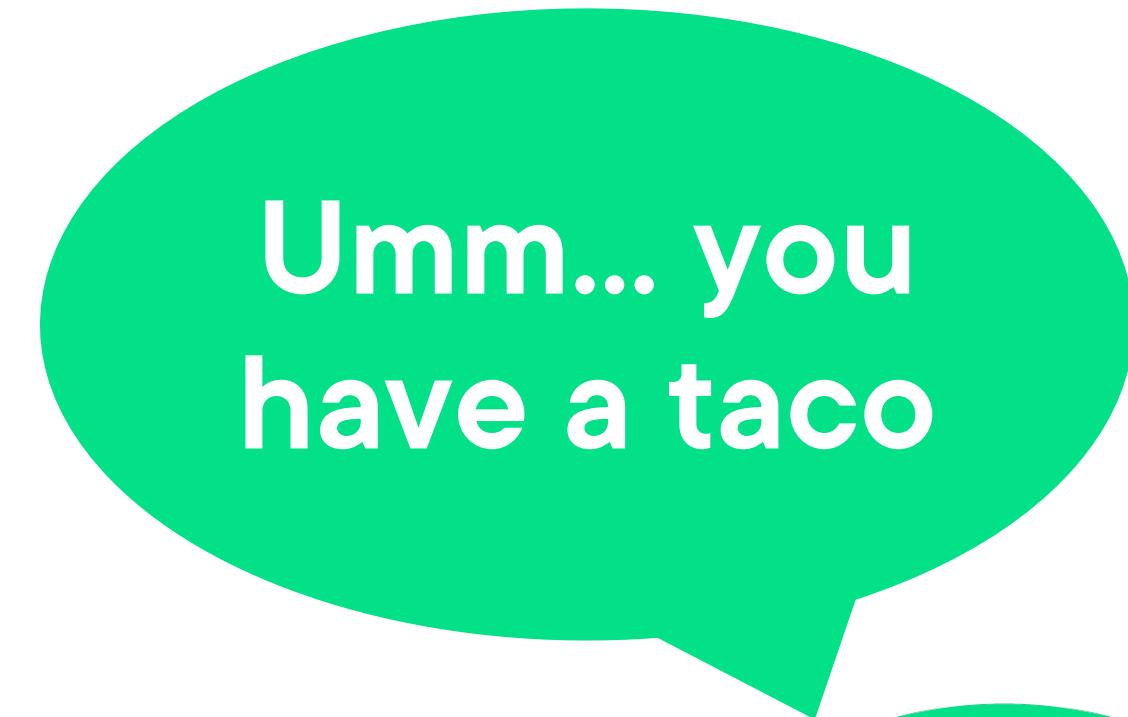
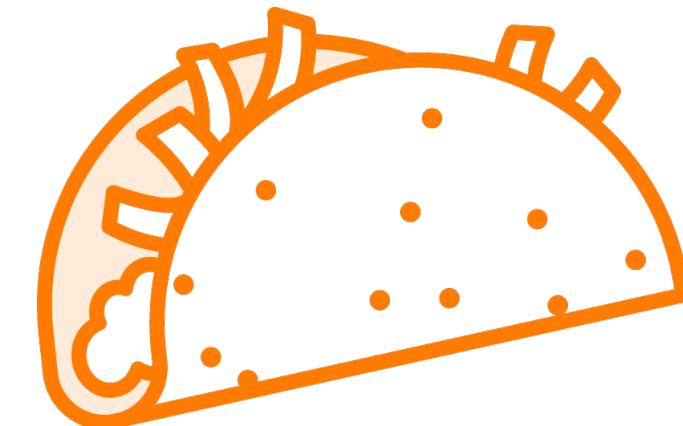
**Idempotent and consistent**

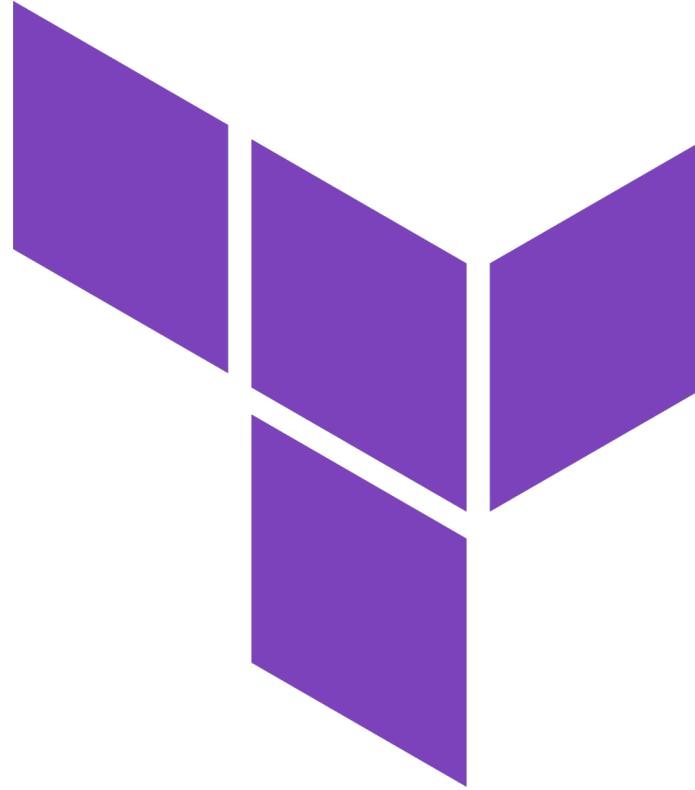


# Idempotent



# Idempotent





**HashiCorp**  
**Terraform**

[hashicorp.com/brand](https://hashicorp.com/brand)

**Infrastructure automation tool**

**Terraform Community**

**Platform agnostic**

**Single binary compiled from Go**

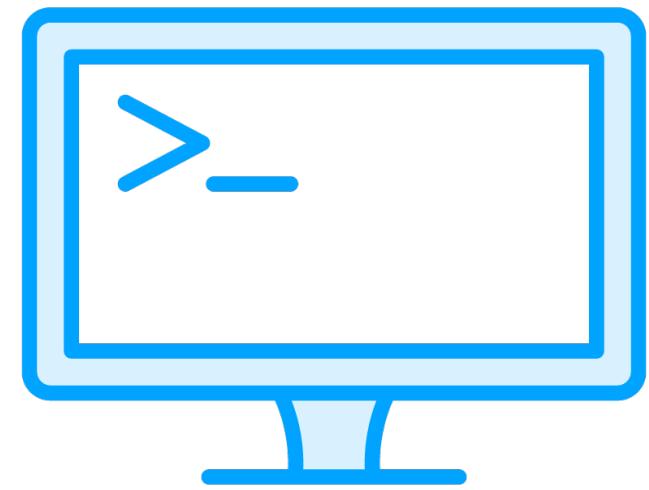
**Declarative syntax**

**HashiCorp Configuration Language (HCL) or  
JSON**

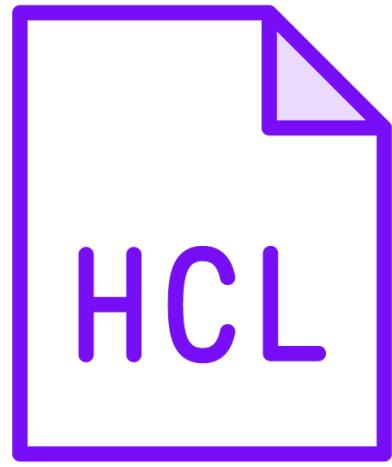
**No central server or agent**



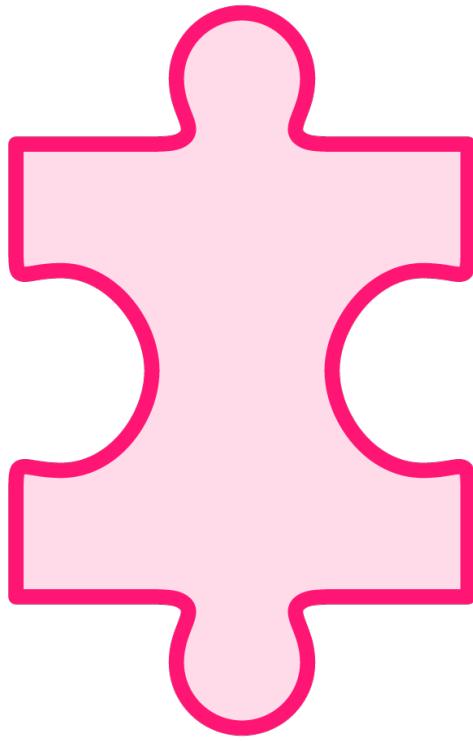
# Core Components



Terraform binary



Configuration files



Provider plugins



State data



# Terraform Workflow

## Write

- Author the code

## Plan

- Compare code against state

## Apply

- Apply changes and record results





## Installation

**Download the binary for your platform**

**Add to your PATH environment variable**

**Share and enjoy!**



# Installation and CLI Basics

Learn about installing Terraform and using the basic command line interface.

