

Nickel + Terraform =



Better Infrastructure
Configuration for Less



Terraform

- Provision infrastructure declaratively
- Document it just like code
- Review it just like code
- Version it just like code

Sounds Great! But...

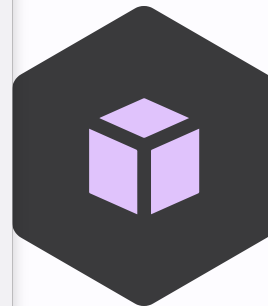
- Can you keep individual machines declarative?
- HCL vs. YAML vs. JSON vs. ...
- Can you move a deployment from one cloud to another?

I want

- Single source of truth
- Schemas that follow *my* requirements
- Extensibility and Reusability

nickel-lang.org

```
let AppSchema = {  
  name | Str,  
  version | SemanticVersion,  
  description | Str,  
} in  
{  
  name = "nickel",  
  version = "1.0.0",  
  description = "Our cool configuration language",  
} | AppSchema
```



Nickel

- JSON with functions
- and types
- but only when they help

Crash Course on Nickel

```
{  
  field1 = "value",  
  field2 = 5,  
  interpolation = "%{field1} is %{string.from field2}",  
  array = [ field1, field2, field3 ],  
}
```

Crash Course on Nickel

```
{  
  val = 5,  
  plus-one = val + 1  
}
```


Crash Course on Nickel

```
{  
  val = 5,  
  plus-one = val + 1  
} & {  
  val = 7  
}
```

Crash Course on Nickel

```
{
  val = 5,
  plus-one = val + 1
} & {
  val = 7
}

==> {
  val = 7,
  plus-one = 8
}
```

Crash Course on Nickel

```
let Contract = {  
  name | Str,  
  age | Num,  
}  
in {  
  name = "Viktor",  
  age = 31,  
} | Contract
```

Configuration

```
{
  vkleen = {
    extra-teams = [ "nickel" ],
  },
  yannham = {
    is-admin = true,
    extra-teams = [ "nickel" ],
  },
} | {_: User}
```

Contract

```
User = {
  is-admin
  | Bool
  | default = false,
  extra-teams
  | Array Str
  | default = [ ],
}
```

Making Terraform Understand

- For each user:
 - GitHub membership resource
 - Member property on all relevant team resources
- For each team:
 - Team resource
 - Team membership resource

Demo!



Summary

- Structure your configuration how *you* want
- Enforce custom properties early
- Reuse with small changes using recursive overriding

Try it!

```
nix flake init -t github:tweag/tf-ncl/intro-docs#hello-tf
nix run .#terraform -- hello-tf.ncl init
nix run .#terraform -- hello-tf.ncl apply
```

And pester me with issues!

- [#nickel-lang:matrix.org](https://matrix.org)
- <https://github.com/tweag/nickel>
- <https://github.com/tweag/tf-ncl>