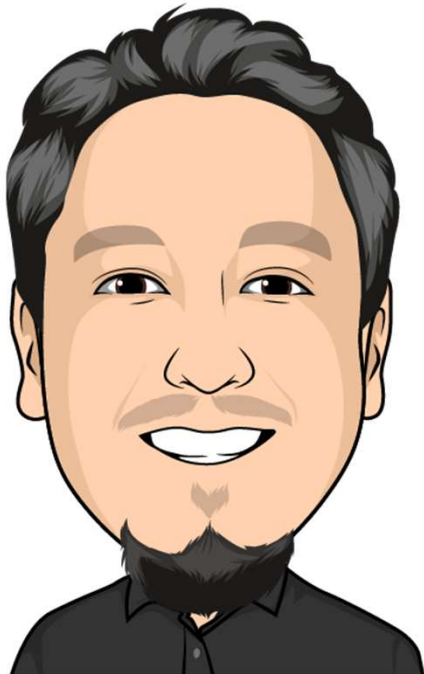


HashiCorp

Terraform

**ARM, Bicep, knees
and toes!**

**Infrastructure as
code for beginners.**



Originally from Mexico.

Tech Lead at Geneca.

Spend time with family, movies, videogames, soccer.



@thesoccerdev



drkclw



samueljgomez

Thank you to our sponsors!



Agenda

- Who is this talk for?
- Infrastructure as Code
- Tool overview
 - ARM templates
 - Bicep
 - Terraform



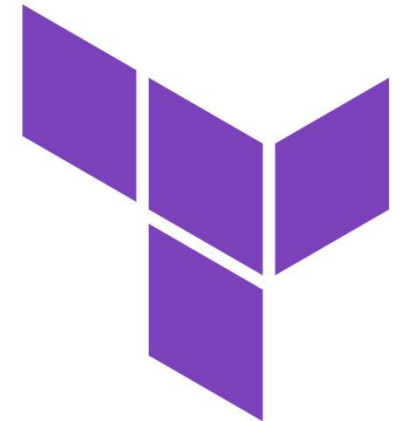
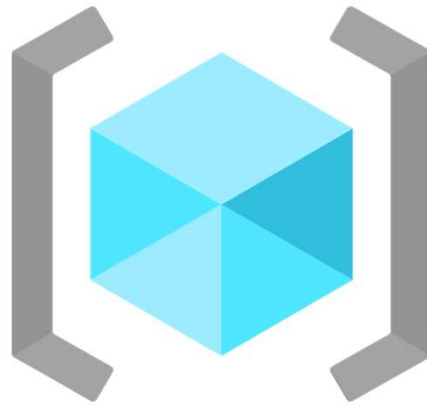
Who is this talk for?

- People completely new to IaC.
- Little experience with ARM templates.
- New to Bicep.
- New to Terraform.



Infrastructure as Code

- Deploy infrastructure in an **automated**, **consistent** and **repeatable** manner.



HashiCorp
Terraform





ARM templates

ARM templates

- JSON format with declarative syntax.
- Defines infrastructure and configuration.



ARM Template format

```
JSON Copy  
  
{  
  "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json",  
  "contentVersion": "",  
  "apiProfile": "",  
  "parameters": { },  
  "variables": { },  
  "functions": [ ],  
  "resources": [ ],  
  "outputs": { }  
}
```





ARM DEMO

“ Let this code compile,

In not an hour, but a little while.

Allow this application to run with speed and zest,

as this will keep users glued to my test.

If my code blows up in my face,

do it with style and grace.

And last but not least, when all is said and run,

let thy requirements have been correctly done.

<https://www.rambli.com/2016/06/the-prayer-of-the-demo-gods/>





Bicep

Bicep

- Domain specific language that aims to drastically simplify authoring experience.
- Transparent abstraction over ARM templates.



Bicep format

Bicep

Copy

```
targetScope = '<scope>'

@<decorator>(<argument>)
param <parameter-name> <parameter-data-type> = <default-value>

var <variable-name> = <variable-value>

resource <resource-symbolic-name> '<resource-type>@<api-version>' = {
  <resource-properties>
}

module <module-symbolic-name> '<path-to-file>' = {
  name: '<linked-deployment-name>'
  params: {
    <parameter-names-and-values>
  }
}

output <output-name> <output-data-type> = <output-value>
```



Modules

- Allow you to use reference other bicep AND ARM template files.
- Import modules from private or public registries.



Conditional (if)

- Allow you to deploy resources only when condition is met.

```
Bicep Copy  
  
param deployZone bool  
  
resource dnsZone 'Microsoft.Network/dnszones@2018-05-01' = if (deployZone) {  
  name: 'myZone'  
  location: 'global'  
}
```



Loops

- Define multiple copies of:
 - Resources
 - Modules
 - Variables
 - Properties
 - Outputs



Loop over index

Bicep


 Copy

```
[for <index> in range(<startIndex>, <numberOfElements>): {  
  ...  
}]
```



Loop over items

Bicep


 Copy

```
[for <item> in <collection>: {  
  ...  
}]
```



Loop over dictionary items

Bicep

 Copy

```
[for <item> in items(<object>): {  
  ...  
}]
```



Loop over integer index and items

Bicep

 Copy

```
[for (<item>, <index>) in <collection>: {  
  ...  
}]
```



Loop over items with conditional

Bicep

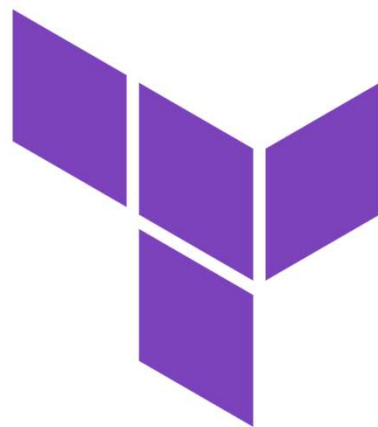
 Copy

```
[for <item> in <collection>: if(<condition>) {  
  ...  
}]
```





BICEP DEMO



HashiCorp

Terraform

Terraform

Terraform.

- Open source IaC tool for provisioning and managing infrastructure.
- HCL syntax.
- Can provision resources to multiple cloud providers.



Terraform providers.


- Plugins used to interact with cloud providers, SaaS providers and other APIs.
 - Azure
 - AWS
 - Google Cloud



Terraform syntax.

- Arguments.

```
image_id = "abc123"
```

Copy 

- Blocks.

```
resource "aws_instance" "example" {  
  ami = "abc123"  
  
  network_interface {  
    # ...  
  }  
}
```

Copy 



Terraform state.

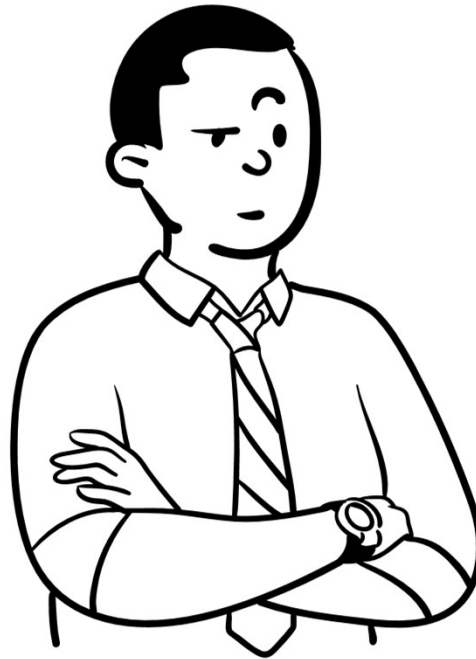
- Terraform stores infrastructure state in a separate file.





TERRAFORM DEMO

Which one should I use?



Terraform.

- Resources not in Azure (including on-prem).
- Mix of cloud providers.
- More accessible for developers.



Bicep.

- Resources in Azure only.
- Share templates via registries.
- More accessible to developers.



ARM.

- Resources in Azure only.
- Do not want to learn new language.
- Team has experience with ARM templates.





Questions?

Useful links.

- <https://learn.microsoft.com/en-us/azure/templates/>.
- <https://github.com/Azure/azure-quickstart-templates>
- <https://github.com/Azure/bicep>.
- <https://github.com/hashicorp/terraform-provider-azurerm/tree/main/examples>.



Useful links.

- <https://github.com/Azure/bicep-registry-modules/tree/main/modules>





@thesoccerdev



drkclw



samueljgomez