

Using Modules for Common Configurations



Ned Bellavance

MICROSOFT MVP, CLOUD AND DATACENTER MANAGEMENT

@ned1313 www.nedinthecloud.com



Overview



Another day, another request

Modules make code reuse easier

Outputs and process flow

Putting it all together



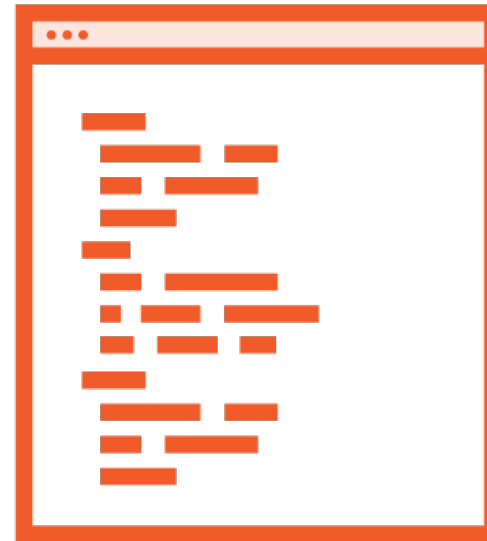
Automating Infrastructure Deployment



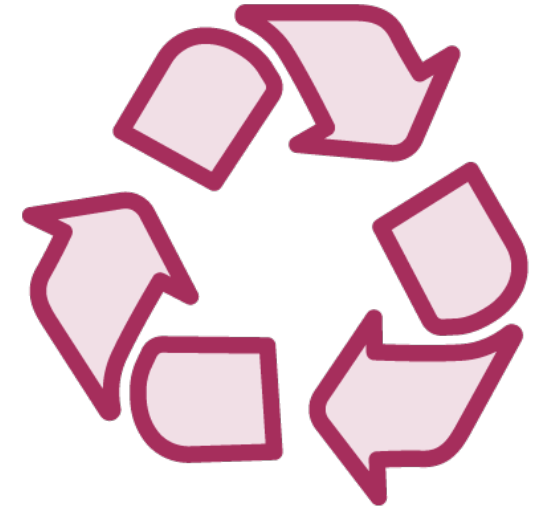
Provisioning
resources



Planning
updates

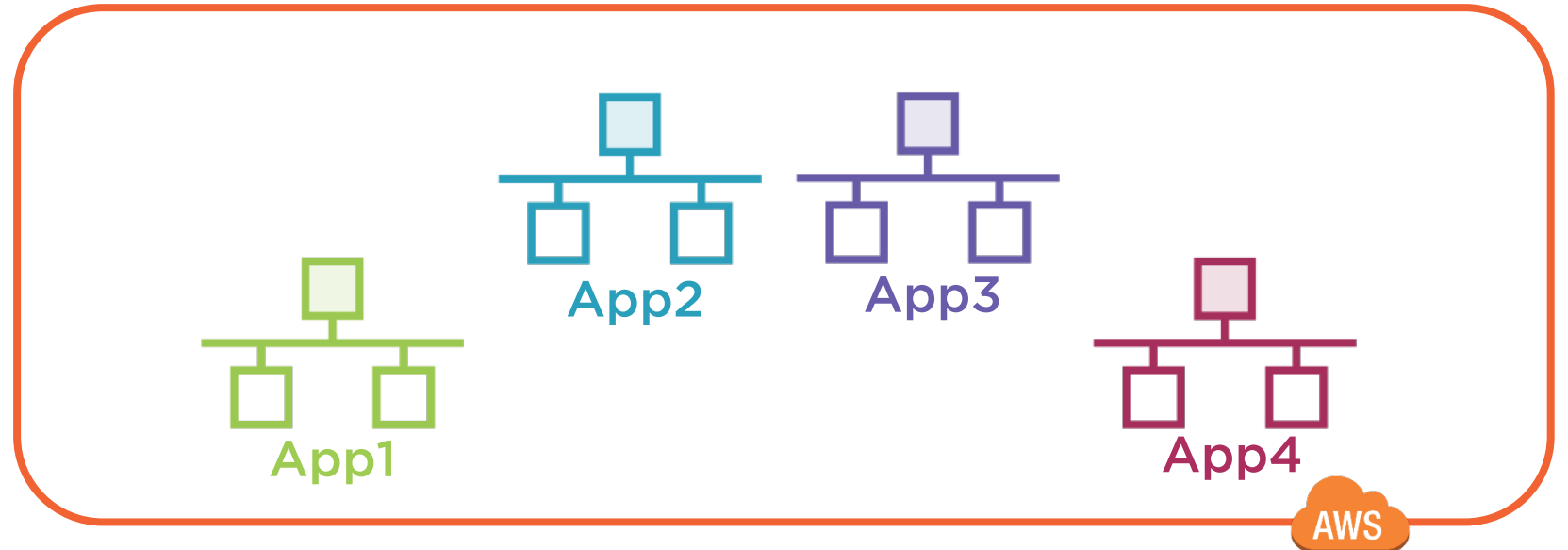


Using source
control

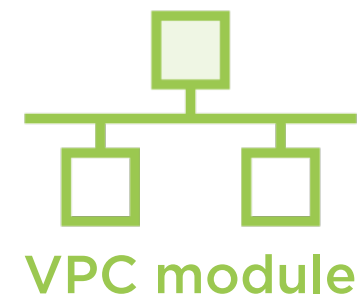


Reusing
templates

The Scenario



Subnets
Internet gateway
Routing
Security groups



Terraform Modules



Code reuse

Remote or local source

Terraform evaluation

Mini-Terraform configuration

Multiple instances (no count)

Module Components



Variables



Resources



Outputs

Terraform Module

```
variable "name" {}
```

```
resource "aws_s3_bucket" "bucket" {  
    name = "${var.name}"  
    [...]  
}
```

```
output "bucket_id" {  
    value = "${aws_s3_bucket.bucket.id}"  
}
```



Terraform Module

#Create module bucket

```
module "bucket" {  
    name = "MahBucket"  
    source = ".\\Modules\\s3"  
}
```

#Use MahBucket

```
resource "aws_s3_bucket_object" {  
    bucket = "${module.bucket.bucket_id}"  
    key = "/walrus/bucket.txt"  
    source = "./mahbucket.txt"  
}
```



Demo



Examine the Terraform file

Deploy the configuration

Review the results

Play along!

- AWS account
- Azure subscription
- DNS domain
- Terraform software (terraform.io)
- Demo files



Some of the resources
deployed in AWS and Azure
may cost **money**. You've
been warned.



Summary



Terraform modules

Code reuse

Abstraction

What's next?



Course Summary



Build infrastructure automagically

Ensure consistent repeatable deployment

Reuse existing configurations

Increase your productivity

Make your job better or find a better job!



Go Build Something Great!

