

UK VMUG UserCon - 13 December 2018



# Terraform, Terrawhat, TerraARGH, TerraYAY!

<https://www.virtualisedfruit.co.uk/terraform,-terrawhat,-terraargh,-terrayay!/> for full res videos

# Terraform, Terrawhat, TerraARGH, TerraYAY!

## Gareth Edwards

- Infrastructure Engineer
- Ecotricity
- 3 year vExpert and vExpert Pro



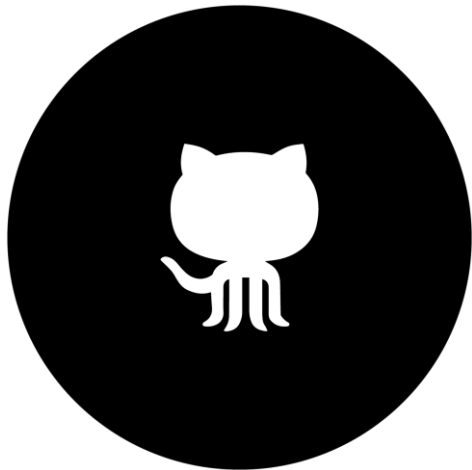
Before you learn the basics, you need to know the future



***SERVER***  ***LESS***

# Keeping things organized

- Store your code somewhere immutable or at least change controlled
- At least use version numbers!

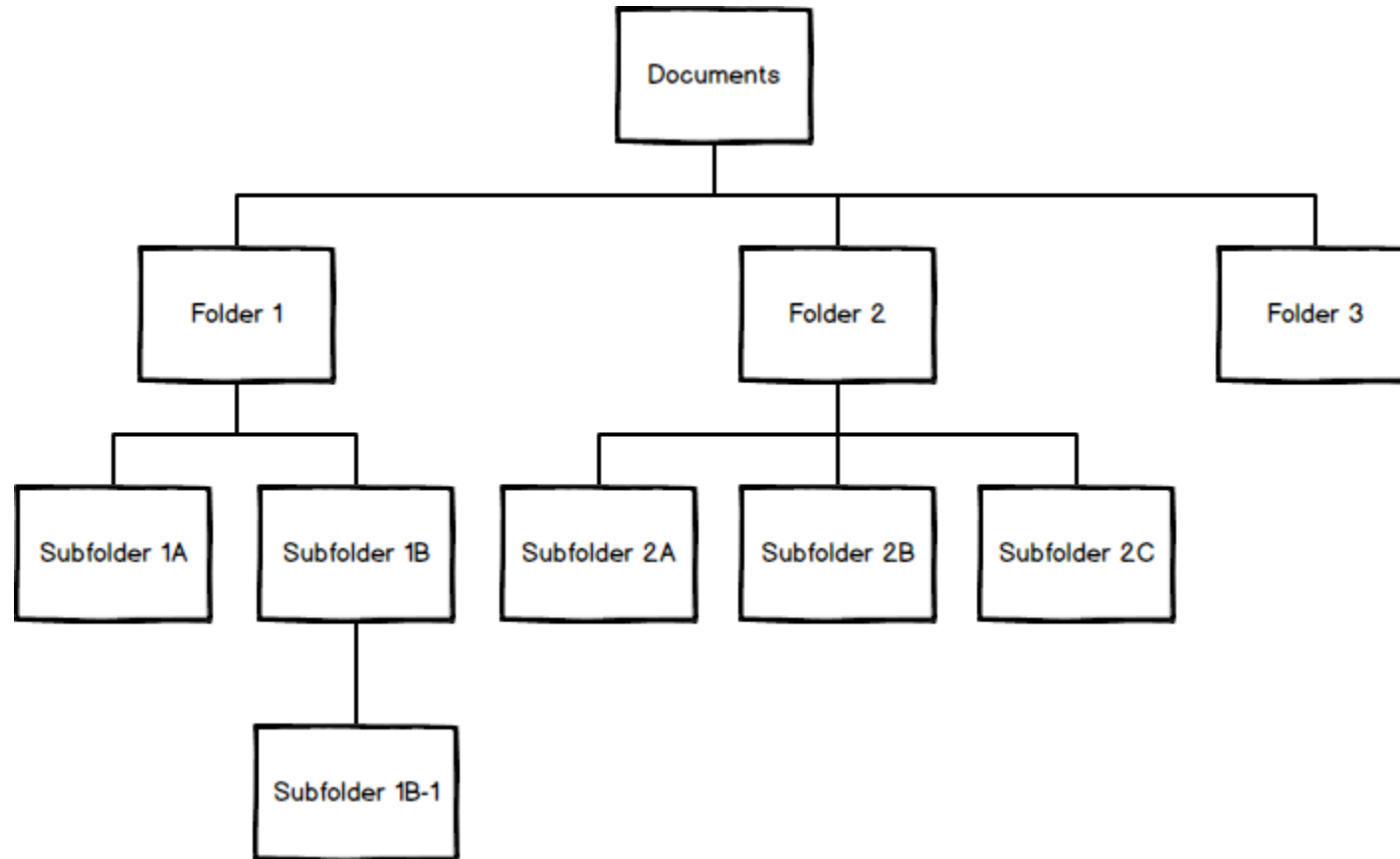


# Bitbucket



Visual Studio  
Team Services

# All about structure



# Setting things straight

## **Definitely**

Create a repeatable infrastructure across system sets

## **Can**

Configure system basics

## **Wont**

Perform software updates and management



# Remember Security – TFVARs

## **Pros**

Keeps accounts safe

No unexpected spend

Track Changes and who did what

## **Cons**

Something else to learn

Remember not to commit

Harder to debug if someone includes a new variable

# A PICTURE IS worth A THOUSAND WORDS



doubleplusunlucky:

The look of intensity on the monkey's face is what elevates this photograph to something truly magical.

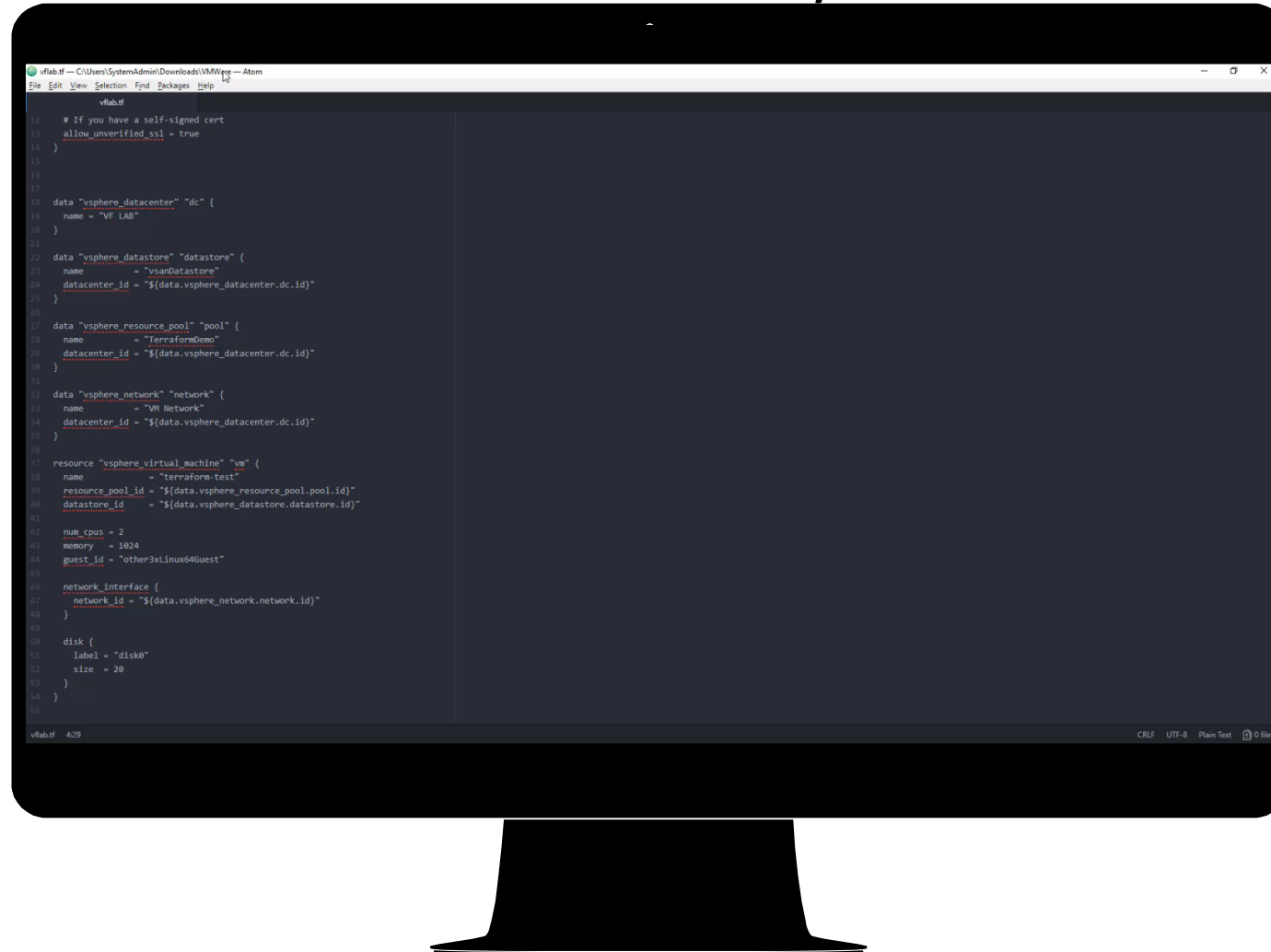
181,860 notes



- You will not get it right the first time
- Things will break
- Go away and come back
- Keep things consistent

# My First VM

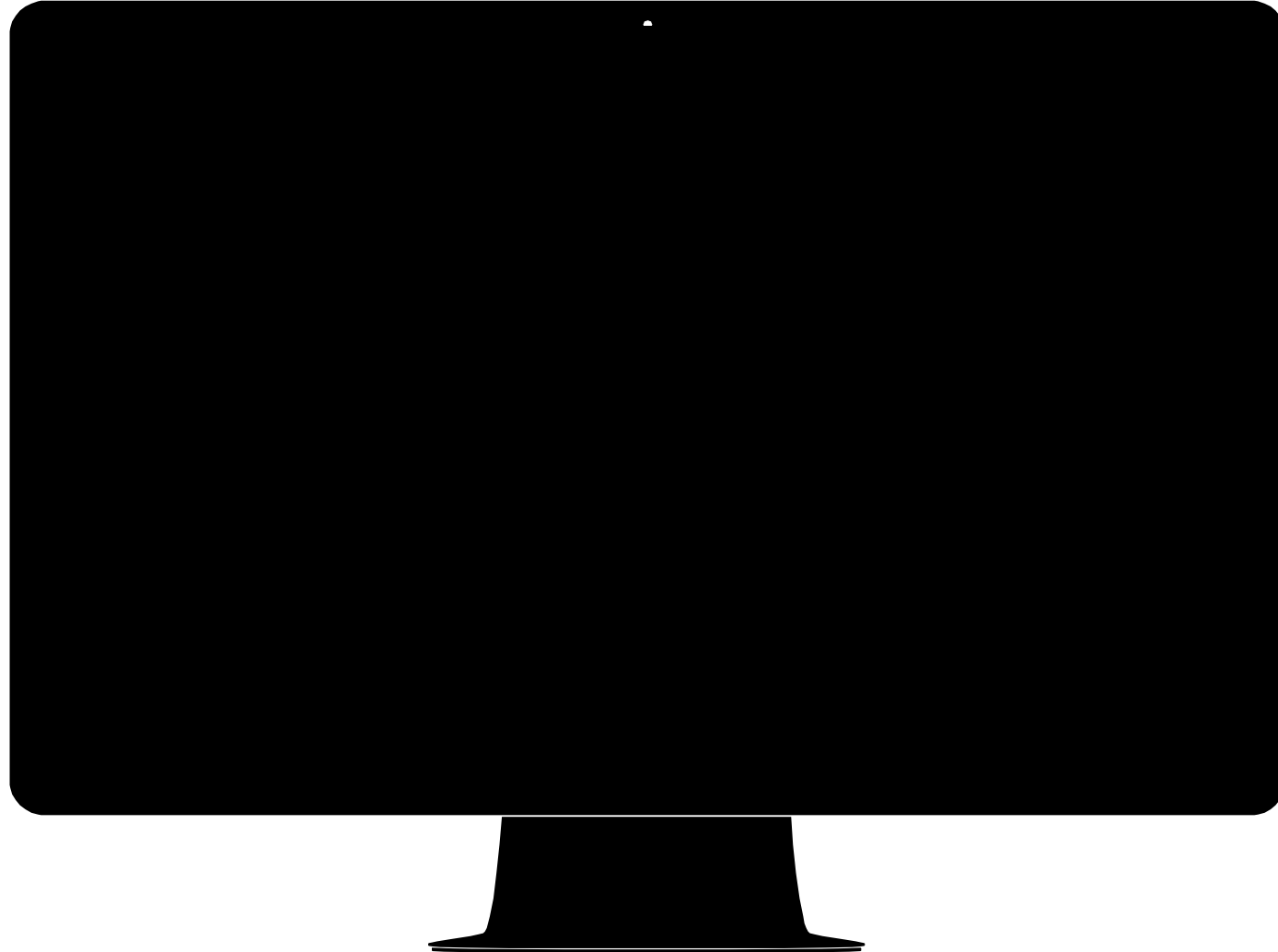
How to get started in the most basic way



```
12 # If you have a self-signed cert
13 allow_unverified_ssl = true
14 }
15
16
17
18
19 data "vsphere_datacenter" "dc" {
20   name = "VF LAB"
21 }
22
23 data "vsphere_datastore" "datastore" {
24   name           = "vsanDatastore"
25   datacenter_id = "${data.vsphere_datacenter.dc.id}"
26 }
27
28 data "vsphere_resource_pool" "pool" {
29   name           = "TerraformDemo"
30   datacenter_id = "${data.vsphere_datacenter.dc.id}"
31 }
32
33 data "vsphere_network" "network" {
34   name           = "VM Network"
35   datacenter_id = "${data.vsphere_datacenter.dc.id}"
36 }
37
38 resource "vsphere_virtual_machine" "vm" {
39   name           = "terraform-test"
40   resource_pool_id = "${data.vsphere_resource_pool.pool.id}"
41   datastore_id   = "${data.vsphere_datastore.datastore.id}"
42
43   num_cpus = 2
44   memory   = 1024
45   guest_id = "other3xLinux64Guest"
46
47   network_interface {
48     network_id = "${data.vsphere_network.network.id}"
49   }
50
51   disk {
52     label = "disk0"
53     size  = 20
54   }
55 }
56
```

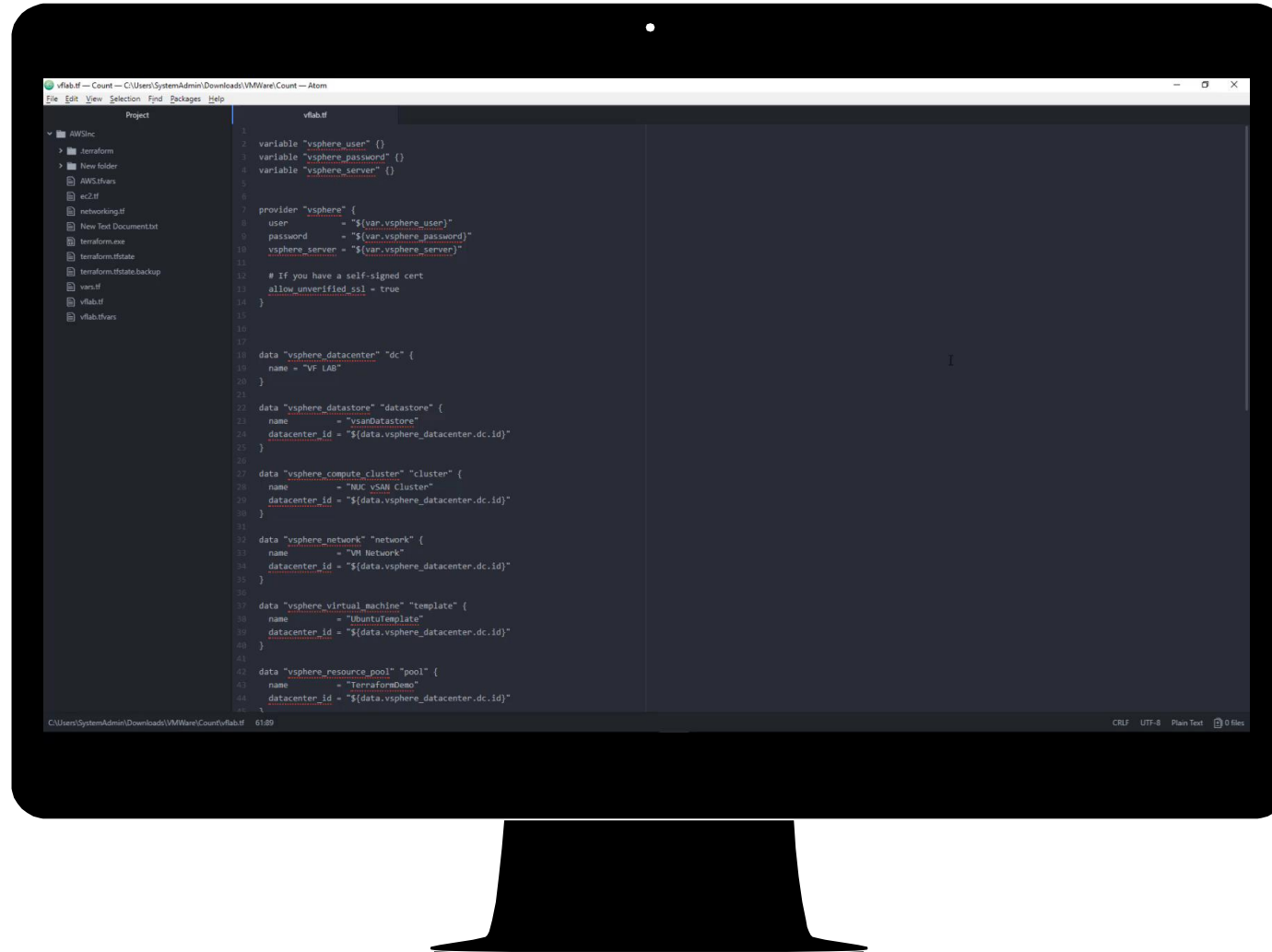
# Template Time

How to get started in the most basic way



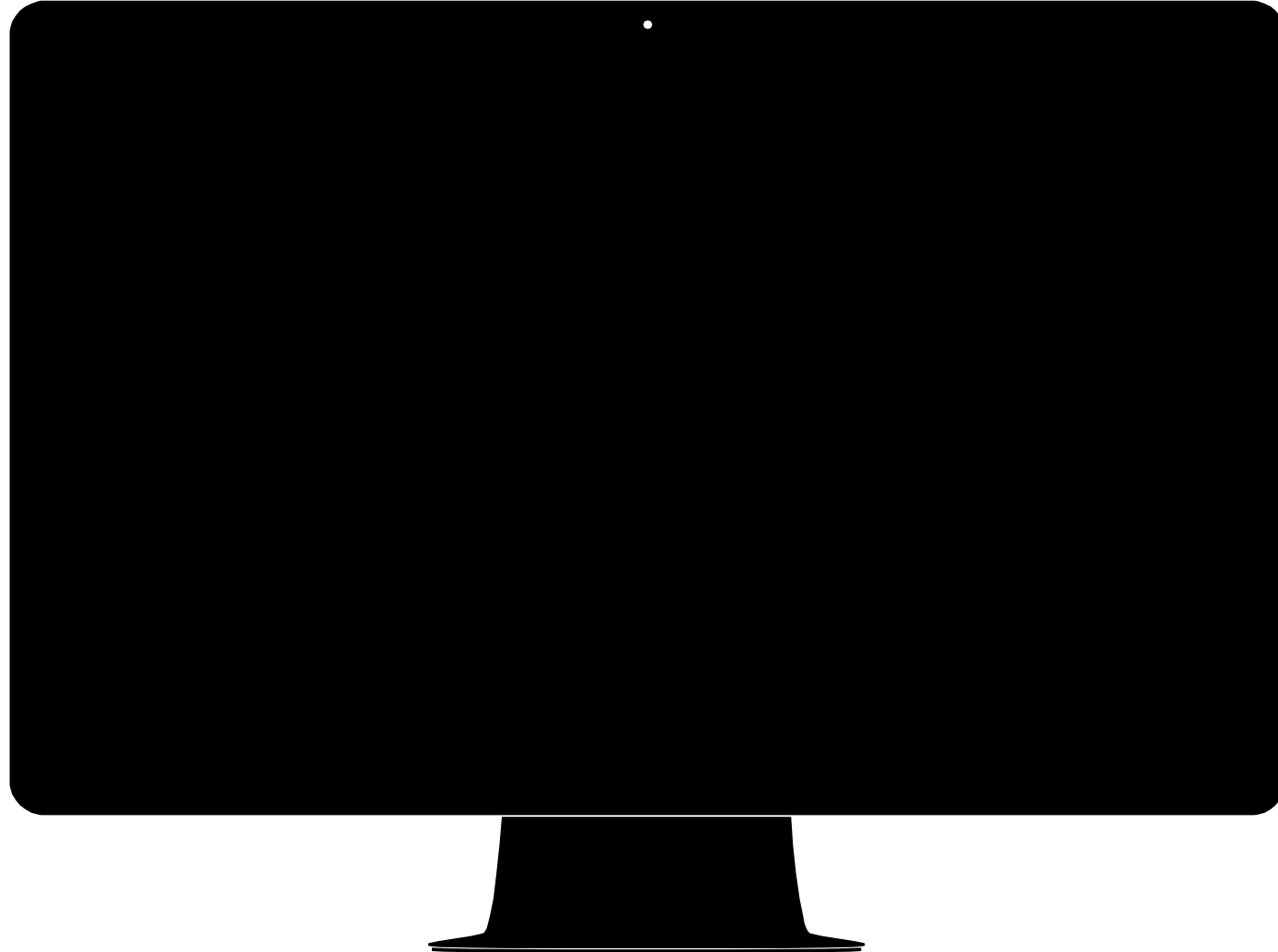
# Scale Time

Lets do it how we are used to



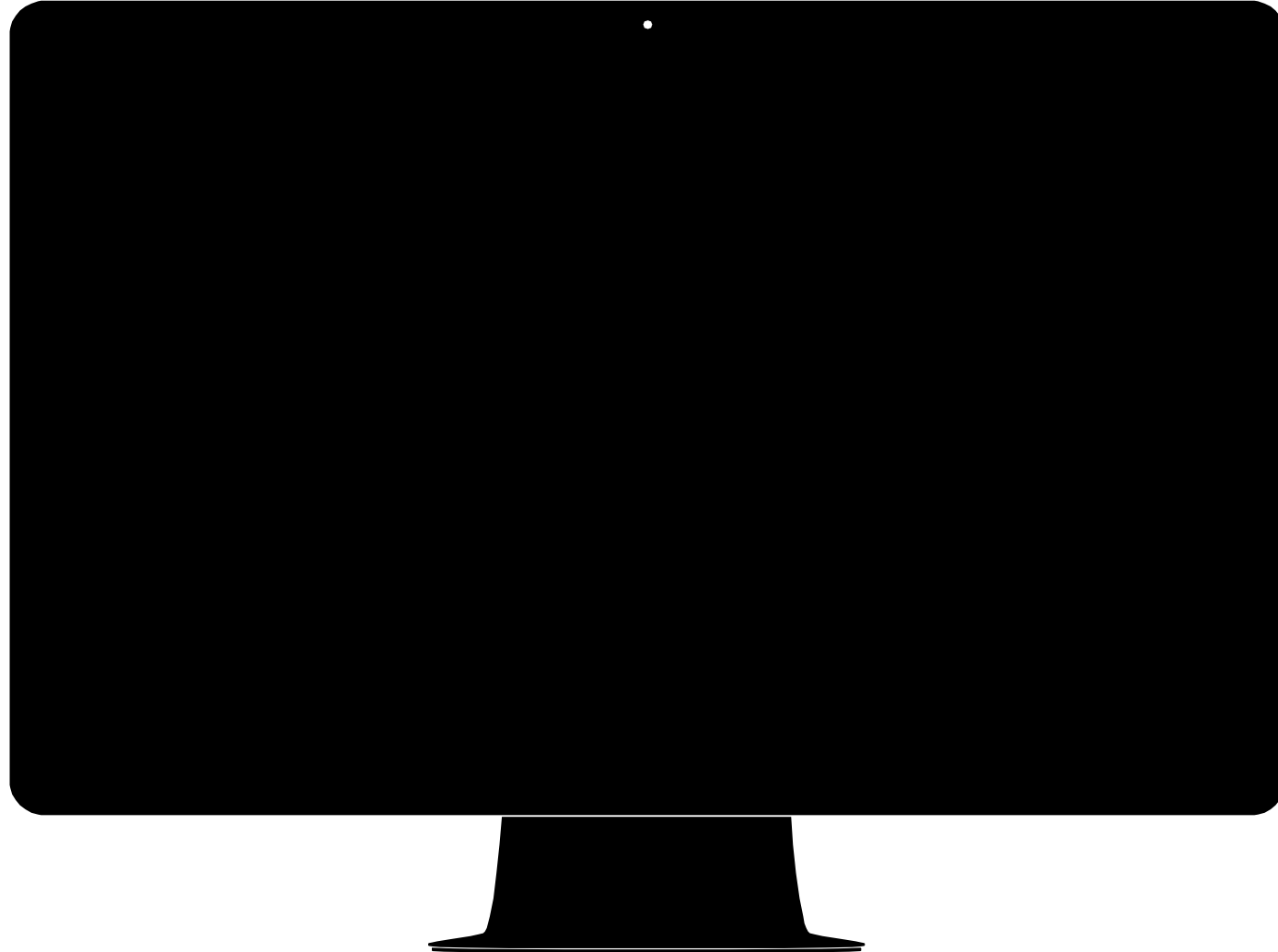
# Time to get AWS Involved

Lets get cloudy



# First giveth, then taketh away

Inspired by a rock song



# A few key tips



## Integrations

Work out any integrations with your systems or 3<sup>rd</sup> parties right away like networking for instance



## Lifecycle

Plan this out, are you only doing this to recover it all if it breaks using it as scaffolding



## Create Images

Using the latest images (AMI's) on Amazon is double edged. Saves patching but can cause discrepancies



## User Engagement

If you are changing something engage with your users. This may even be your team. How does this affect logins or changes?



## Keep Positive

This will feel up hill at first but keep going, the reward of having things replicated easy for testing is amazing!



## The big kaboom

If you change some things be prepared for things to go bang... such as changing the AMI on a DC

# Thanks! **ANY QUESTIONS?**

You can find me at:

@GarethEdwards86

Gareth@vexpert.pro



**VMUG** **USERCON**