

# Why I Have 12 Virtual Labs

12 Labs. Yes, that is 12 (twelve) as in 1,2,3,4,5,6,7,8,9,10,11...12 labs. Why would someone have 12 labs you may ask? Well, this is a great question. In fact, if I really did subdivide by guest platform, there are much more, but for all intents and purposes, each lab is a different hosting environment for testing resources.

**Update: I actually did exclude Microsoft Azure. Had an inexplicable loss of account there. Long story. That's another post unto itself ☐**

## What do I mean by virtual lab?

These are nested virtualization labs that run a variety of different products, servers, hypervisors, and can be comprised of many different components all the way from something as simple as a virtual Windows desktop for using Microsoft Office, right up to running a cross-boundary OpenDaylight implementation, and everything in between.

The virtual lab resources can also range from ongoing, long-term resources to one-and-done deployments just to do a quick install test or shorter term proof-of-concept tests. There really is no set of rules about what it is that you can use your virtual lab resources for.

## Where are my virtual labs?

Ok, hold on for the list. I actually have two more locations pending, but those are also special projects and have to stay under wraps for a bit longer ☐

1. Mac Mini (permanent home lab)
2. VMware Workstation on Windows (lab on the go)
3. VMware Fusion on OSX (light use lab)
4. Vagrant/VirtualBox on Windows
5. Vagrant/VirtualBox on OSX
6. AWS
7. Digital Ocean
8. vCHS
9. baremetalcloud.com
10. Office isolated lab
11. Office vCloud lab
12. VMware Hands-on-Labs

Rather than having one large lab, my tactic has been to have more lab locations that are stored in different locations.

## Why so many labs?

I sum up what labs are for as having three specific use-cases:

1. Trying new things
2. Learning more about new things
3. Testing new things on existing things

It really is just that simple if you break it down. There are some labs that are used for doing product specific testing inside applications. I also test hypervisor features, high-availability options, cross-platform deployments, automation and orchestration.

## How do you use all these labs?

Great question! And I'll answer it with a very simple answer: WIP(3)

What that means is that I have a Work In Progress limit of 3 items. For those fans of Kanban, you will already know that I mean. I use Kanban for task management and my product of choice is LeanKit. Here is a sample board I have, which to many looks like a lot of stuff. The key thing to note is the highlighted section:



As you can see, there are tasks at various points in the backlog, but I have a strict limit of 3 items in my Work In Progress swim lane.

The trick with keeping things moving is that I have extremely short sprints (usually 2-3 days). I have a personal board for around the house stuff, but for my technical writing, learning, and testing, this is all held here.

## Back to the Labs!

Here are some quick notes about stuff that I do in each of the labs to give you an idea of what you have available with relatively low overhead and simple builds.

### Mac Mini (permanent home lab)

I'm using the William Lam vSphere 5.5 Mac Mini post to build a vSphere host to use for nested virtualization (<http://www.virtuallyghetto.com/2013/09/running-esxi-55-on-apple-mac-mini.html>). Massive thanks go to William for sharing this information.

The Mac Mini is running with 12 GB of RAM, and my plan is to grow this to a 3 Mac Mini cluster down the road.



It's a little constrained when I spin up lots of resources, but luckily I have multiple nested labs inside. I have lots of beta content (can't screen shot that stuff unfortunately) and I also have OpenStack, Hyper-V, vSphere, and vCloud nodes with various levels of interconnection inside.

### VMware Workstation on Windows (lab on the go)

On my laptop (HP Elitebook Core i5 16GB RAM, 180GB SSD) I have VMware Workstation 10 running to be able to have portable lab resources that come back and forth from the office with me.

This is more for the temporary resources, or for content that I am building that will be migrated to the Mac Mini as I find it becomes more permanent. I also build machines to upload to my office vSphere environment because it is easier to work offline sometimes for doing heavy testing without having to be VPN connected all the time.

## **VMware Fusion on OSX (light use lab)**

This is actually my older lab. It is a Macbook aluminum (2008) running 16 GB of RAM that I use for multimedia work normally. Now this also runs a variety of Linux and vSphere machines, but my vSphere resources are slowly being migrated to the Mac Mini. I call this light use because I've literally burned out 3 hard drives on this system with all of the I/O, so I take it easy on the poor old beast now ☐

## **Vagrant/VirtualBox on OSX**

This is where the magic of the Couch to OpenStack work happened. The core scripts for the Couch to OpenStack lab environment are written for Vagrant and VirtualBox on OSX.

## **Vagrant/VirtualBox on Windows**

Because I run Windows on my work laptop, I was actually using Vagrant and VirtualBox on Windows which highlighted some issues with the deployment scripts for Couch to OpenStack. I built a separate branch with the Windows fixes for the lab that helped the non-Mac folks jump in and test things out.

## **AWS**

You can't do cloud without touching AWS. I use AWS for spinning up lots of on-demand resources and testing out orchestration and scale-up application designs. Having knowledge of the different components of the

Pro tip: Don't do performance testing here...trust me! I got a \$148.00 bill for one VM because I forgot to shut of a process that grew and shrank a database on a MySQL instance to test out some Galera clustering on an AWS node.

## **Digital Ocean**

Digital Ocean is wicked cheap, and very cool. (no...they aren't a sponsor). I also use the Vagrant provider for Digital Ocean (<https://github.com/smdahlen/vagrant-digitalocean>) which is really cool. I have a sample of how to use this from my previous post on deploying Jenkins on Digital Ocean here.

The good thing with Digital Ocean is that even if I forget the machines and leave them active, I use a lot of the 5\$/mth small instances which don't dent the pocket book too much.

## **vCHS**

I <3 VMware, and I've use the vCloud Hybrid Service as part of the vExpert early access. Watching this service grow has been exciting, but I have to warn that this is not a typical place for lab deployments because the pay-as-you-go option is not ready yet. That will be the real trigger to see more of my work move into there.

## **baremetalcloud.com**

As sponsors of Virtual Design Master Season 1, and as strong community supporters, the team at Bare Metal Cloud have always been very helpful. Having bare metal resources by the hour help me to test out a variety of builds and this was my first foray into creating a vCloud Hybrid service of my own last year ☐

## Office isolated lab

Some things aren't meant to share the sandbox with others. This is where I do wildly destructive testing and work on applications and environments that emulate production workloads, but because they are so similar, sometimes they need to be isolated.

## Office vCloud lab

This is the "eat your own dog food" lab that I run which is converting processes to VMware vCloud from the traditional vSphere/vCenter implementation that I have running in production. It allows for lots of set up/tear down and repetitive testing without cross-contamination of production resources.

At the same time, this is also production accessible so that it can be shared by the team for training and testing with more resources.

## VMware Hands-on-Labs

Last, but certainly not least is the VMware Hands-on-Labs environment. Once known to early adopters as Project Nee, this is a fully functional set of live labs which are the same ones used at the VMworld Hands-on-Lab environment.

What I love about the VMware labs more than anything is that they provide fully functional environments which allow you to test out some stuff that is "off-script" from the lab scenario. These are real nested lab builds that are full product deployments, so you can get actual hands-on experience with the real products that saves you a lot of time. Even quick lab set up on your own environment will not be as quick as being able to spin up a demo on VMware Hands-on-Labs (<http://labs.hol.vmware.com/>). Plus it is FREE!!!!

## How much does it cost?

This is the classic question. While VMware HOL is free along with many of my other resources, some do have both capital and operational costs.

**Mac Mini** - 1000\$ one-time cost including 16GB upgrade

**AWS** - Usually 25\$ per month average

**Digital Ocean** - For test work I spin up and tear down lots of resources here. The average run rate is about 20\$ per month because I destroy machines as soon as I'm done testing

**Bare Metal Cloud** - Average is about 20-40\$ per month when I'm active in here because I do build tests and then return the hosts to inventory pretty quickly

**Nakivo backup for my Hyper-V** - easy way to protect my content

## Hope this helps!

That is a little view into what I am doing now on my current set of public projects.

Stay tuned this week for the DiscoPosse in Production post that will show you how my web resources and permanent resources are laid out.

