

# Virtualisation Technologies

Frank Walsh

# Virtualbox

- *"The World's Most Popular Open Source, Cross-Platform Workstation Virtualization.."*
- Available for all common platforms (Linux/Windows/Mac)
- Hosted (Type 2) hypervisor
- Labs will do installation and VM creation
- Worth checking out chapter 1 of manual for features

<https://www.virtualbox.org/manual/ch01.html>



# Vagrant

- Used to build and manage virtual machine environments
  - Typically development environments
- Initially built on Virtualbox
- Can create reproducible environments as similar as possible to production servers
- Reason we're using it:
  - Idea is to create identical environment irrespective of underlying OS/hardware
  - No more "well it works/doesn't work on my machine..." (hopefully! )



# Vagrant interaction

- Done through the **command-line interface(CLI)**.
- Vagrant interface is available using the **vagrant** command
  - vagrant command in turn has many subcommands, such as vagrant up,
- Run **vagrant** by itself to display all available subcommands.
- Run any Vagrant sub command with the -h flag to output help about that specific command.

```
frank@frank-Latitude-E5540 ~$ vagrant up -h
==> vagrant: A new version of vagrant is available: 2.1.2!
==> vagrant: To upgrade visit: https://www.vagrantup.com/downloads.html

Usage: vagrant up [options] [name|id]

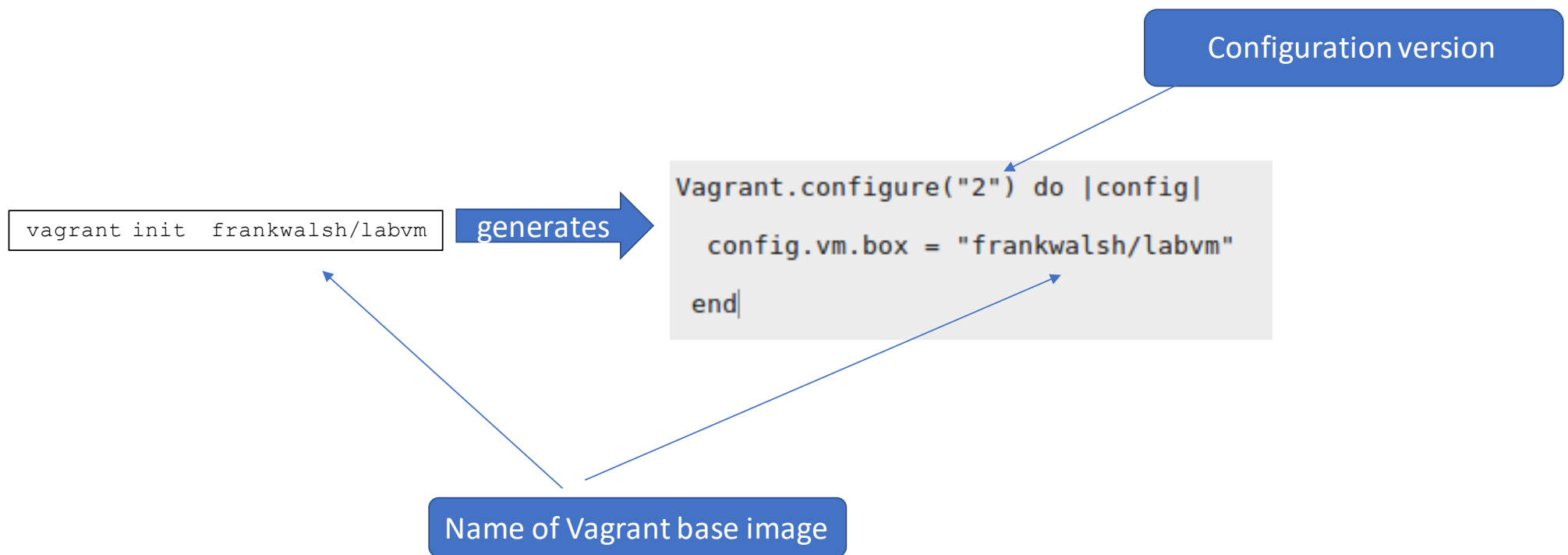
Options:
  --[no-]provision                Enable or disable provisioning
  --provision-with x,y,z          Enable only certain provisioners, by type or by name.
  --[no-]destroy-on-error         Destroy machine if any fatal error happens (default to true)
  --[no-]parallel                 Enable or disable parallelism if provider supports it
  --provider PROVIDER             Back the machine with a specific provider
  --[no-]install-provider         If possible, install the provider if it isn't installed
  -h, --help                      Print this help
```

# Vagrant Box

- Base image used to create a virtual machine
- Ready to go with installed packages/software
- Can use previously/curated boxes or you can make your own
- Use `vagrant box add <user_name/box_space>` to add a box to a project
  - e.g. `vagrant box add frankwalsh/labvm` will download this box from the Vagrant Cloud box catalog.

# Vagrantfile

- Run `vagrant init <box_name>` creates a basic Vagrantfile



# Vagrant Up

- To boot a vagrant environment:

```
vagrant up
```

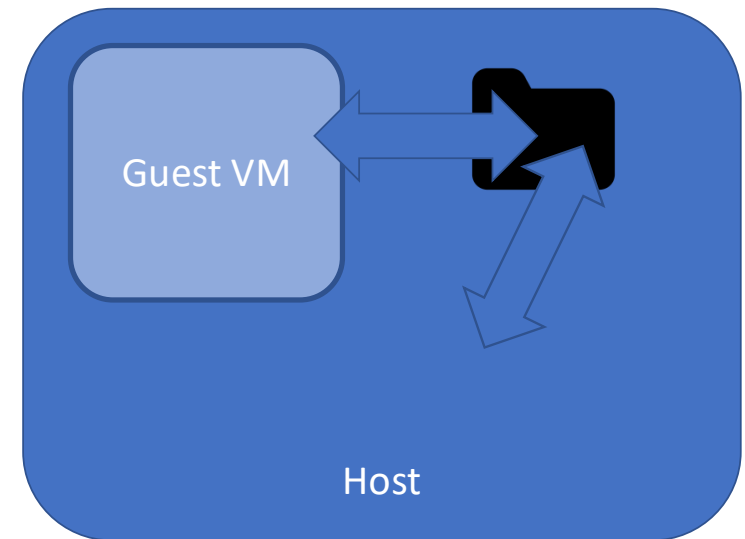
By default, Vagrant runs the VM without a UI

- To **connect** to a VM:

```
vagrant ssh
```

# Vagrant: Synced Folder

- Synchronise your files to and from the guest machine using a **Synced folder**
- By default, the project directory on the host machine( i.e. the one containing the Vagrantfile) is synced to the **/vagrant** directory in your guest machine.





# Vagrant Share

- Allows you to share your Vagrant environment with anyone in the world
  - Might use this with assignments
- **HTTP sharing:** creates URL that routes directly into your Vagrant environment
- **SSH sharing:** SSH access to your Vagrant environment by anyone by running `vagrant connect --ssh` on the remote side.
- **General sharing:** anyone can access exposed ports of your Vagrant environment. Useful for access to your Vagrant environment as if it were a computer on the LAN.