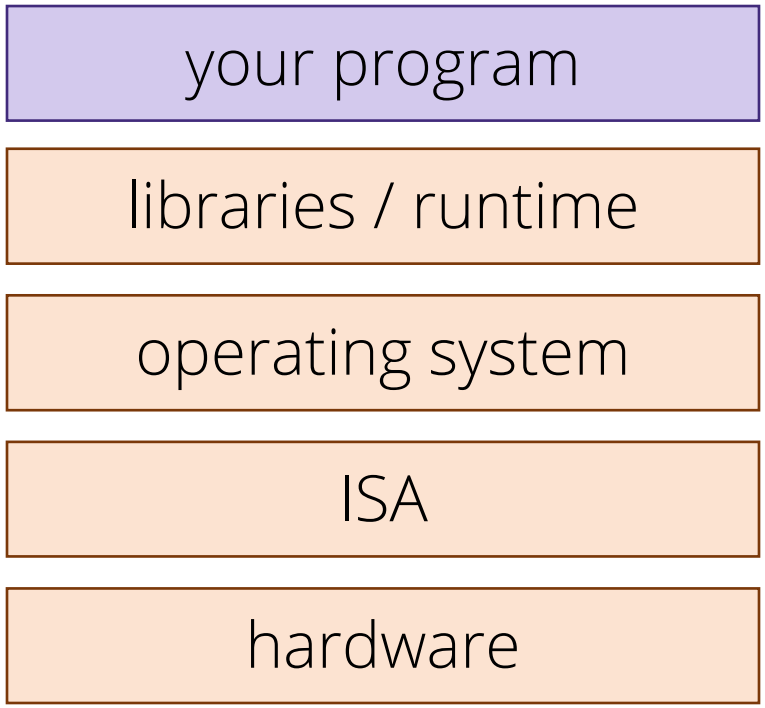
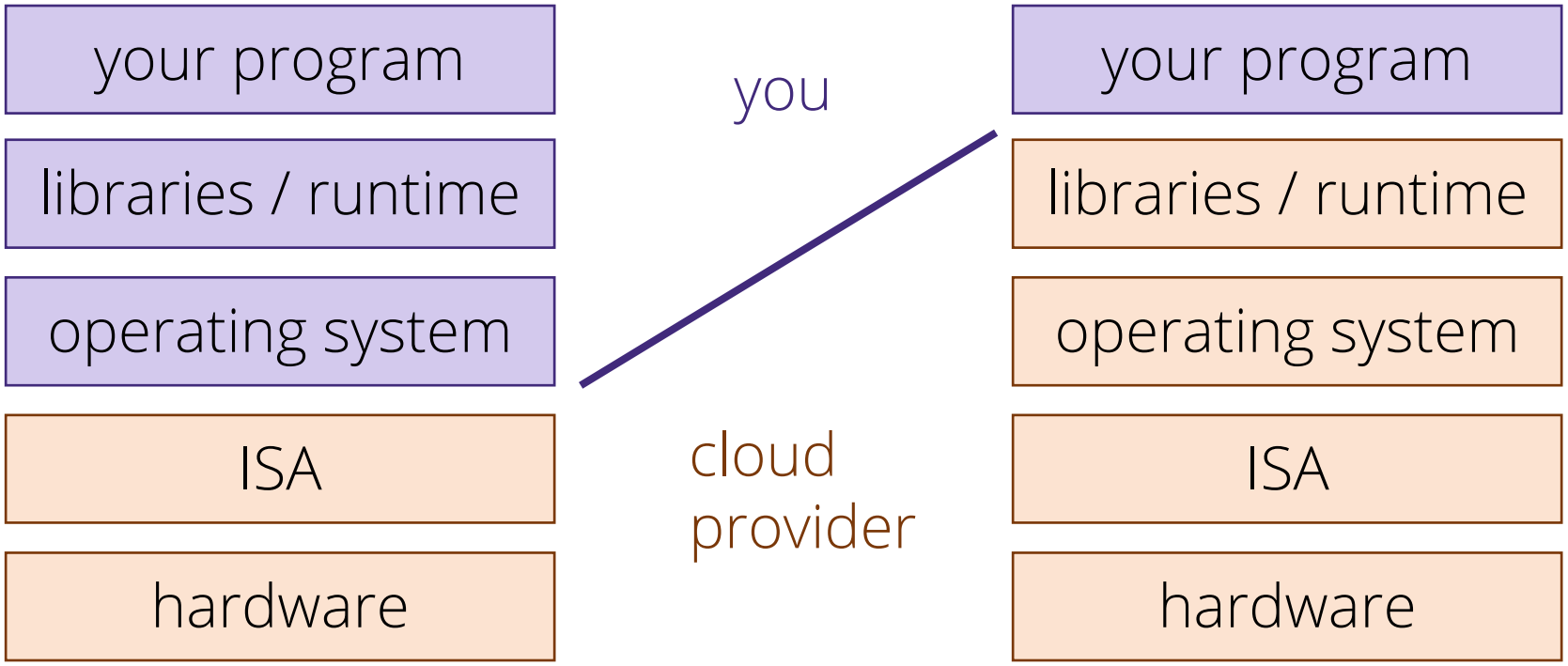
Vagrant

COMS10012 Software Tools

Technology stack



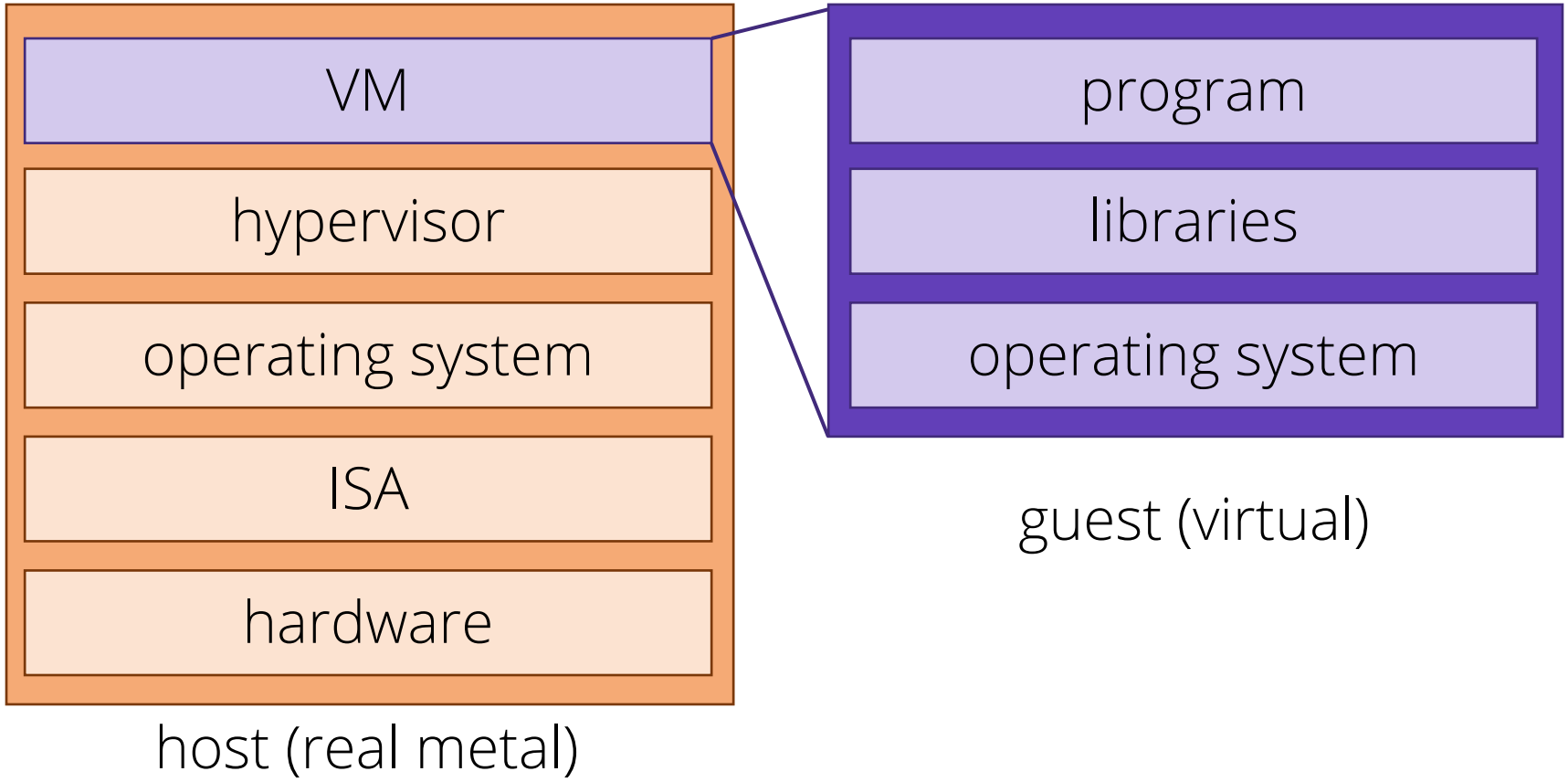
Preview: cloud



IaaS (Infrastructure as a Service)

PaaS (Platform ...)

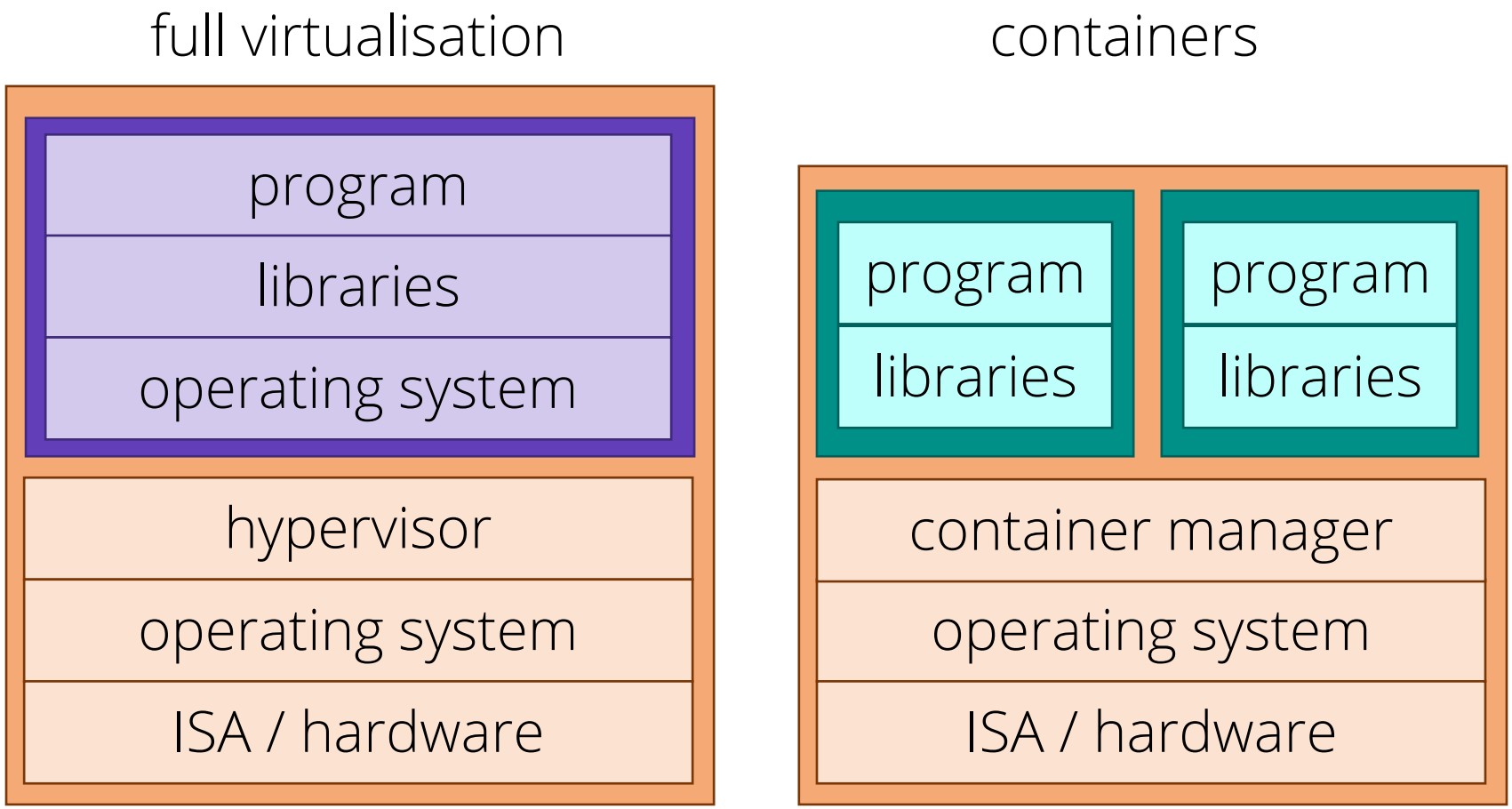
Virtualisation



Why virtualise?

- emulate a different stack
- reproducible build environment
- cost / scalability

Containers


full virtualisation
program
libraries
operating system
hypervisor
operating system
ISA / hardware
containers
program
libraries
program
libraries
container manager
operating system
ISA / hardware

Levels of virtualisation

1. Environments, e.g. python's **venv**
2. Containers (linux OS feature)
3. Full virtualisation

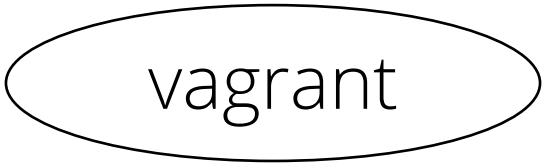
Software

Virtualisation:

VMware,
VirtualBox (Oracle)

Containers:

Docker
Kubernetes



bochs, qemu,
DOSbox, ...

OpenStack, rkt, ...

Installing vagrant

Lab machines (but not snowy, seis): installed

From the web: www.vagrantup.com/download

Linux: vagrant recommends *not* using your system's package manager.

Windows: read www.vagrantup.com/docs/installation, you may need to disable Hyper-V.

Vagrant

Host: folder with Vagrantfile (ruby)

Different providers

ssh access to guest

can share folders between host/guest

Vagrantfile

```
Vagrant.configure("2") do |config|
    config.vm.box = "generic/alpine310"
end
```

box repository:
https://app.vagrantup.com/boxes/search

Start the machine

```
$ vagrant up
Bringing machine 'default' up with
'virtualbox' provider...
==> default: Importing base box
'generic/alpine310'...
...
==> default: Machine booted and ready!
```


Commands

`vagrant up`	start machine
`vagrant ssh`	log in
`vagrant halt`	stop machine
`vagrant destroy`	delete machine

All commands require a Vagrantfile in the current directory.

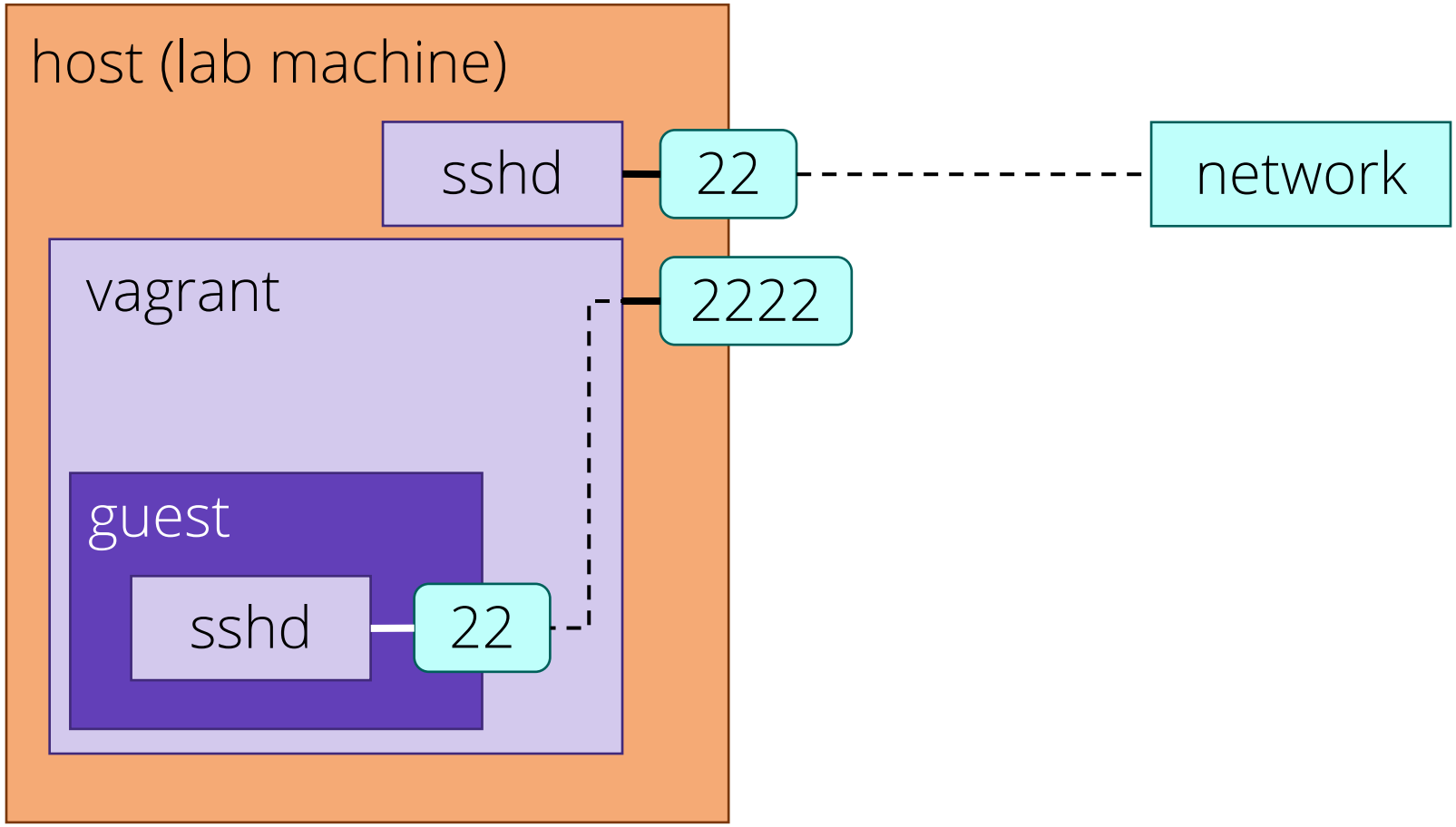
Log in

```
$ vagrant ssh
alpine310:~$
alpine310:~$ whoami
vagrant
alpine310:~$ exit
logout
Connection to 127.0.0.1 closed.
$
```

ssh

```
$ vagrant up
...
==> default: Forwarding ports...
    default: 22 (guest) => 2222 (host)
...
    default: SSH address: 127.0.0.1:2222
    default: SSH username: vagrant
    default: SSH auth method: private key
    default: Vagrant insecure key detected. Vagrant
will automatically replace this with a newly generated
keypair for better security.
    default: Inserting generated public key within
guest...
```


ssh and port forwarding



keys

Remember: if you have a *secret key*, you can ssh in to a machine that has the matching *public key*.

Vagrant box (in repository) has a default public/secret key pair.

When you provision (**`vagrant up`**) a box, it creates a new key pair – this is more secure, and you can use it with **`vagrant ssh`**.

Storage

Normal use: virtual machines stored in

- Linux: **`~/.vagrant.d`**
- Windows: **`C:\Users\NAME\.vagrant.d`**

Some configuration goes in the **`.vagrant`** folder in the folder with the Vagrantfile.

Storage – lab machines

VMs are stored in `/tmp` and may not survive host reboots!

Also, they are not on NFS, so not visible from other lab machines.

- This is by design.
- **Treat VMs on lab machines as disposable – back up your data somewhere else!**

Alpine linux

Minimal installation – no gcc, git etc. by default (we'll learn to install it ourselves)

strong on security – good choice in production

small: container can fit in 8MB

- musl-libc + busybox
- OpenRC