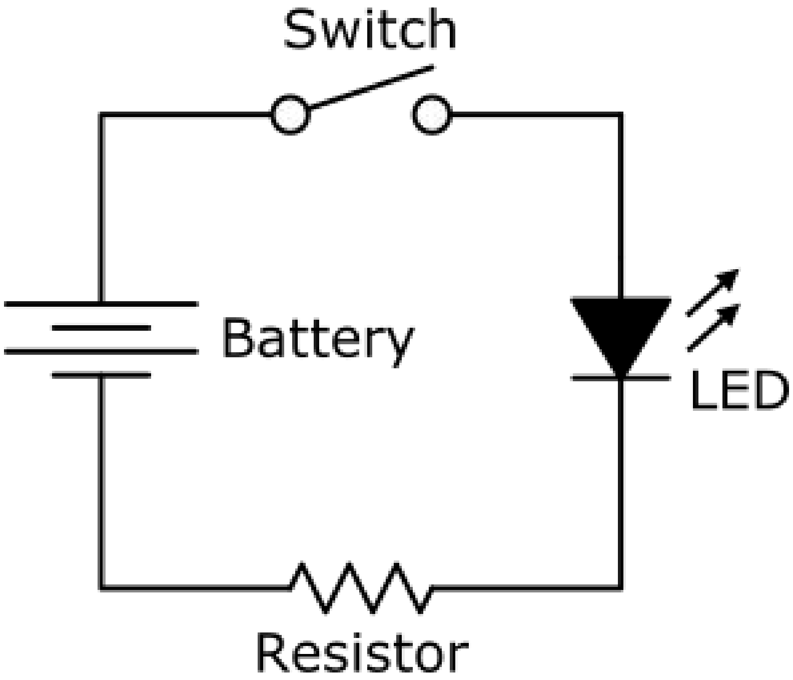
CANONICAL

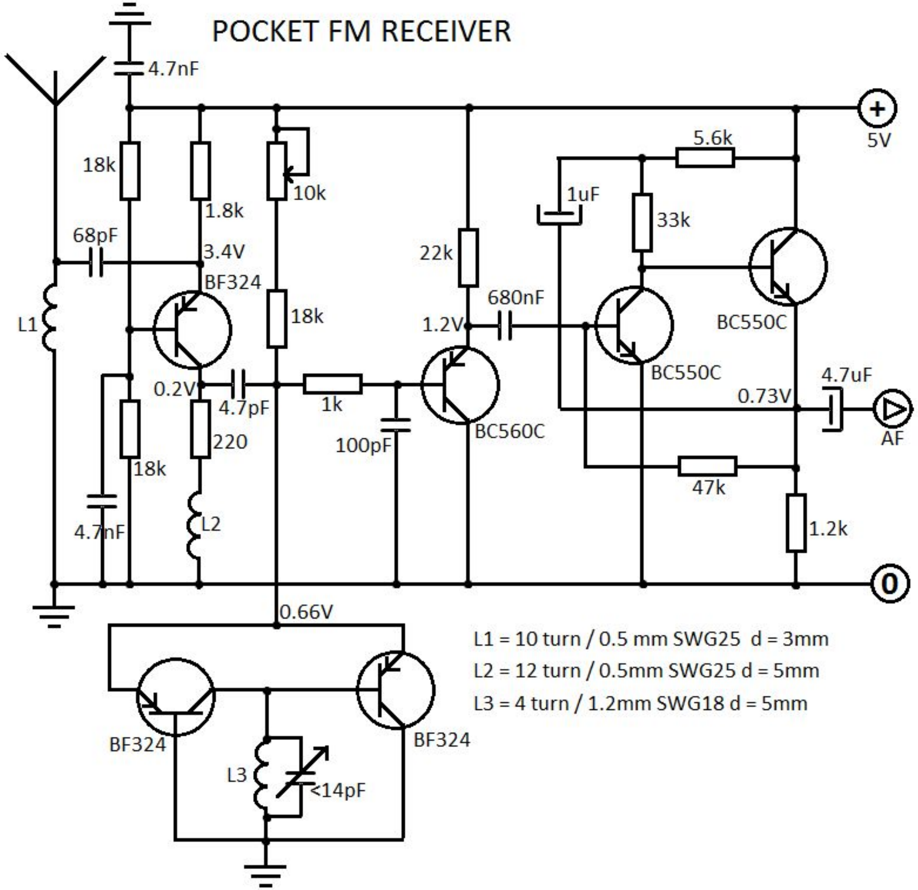
The magic of application modeling

Introduction to Juju

scarcity to abundance

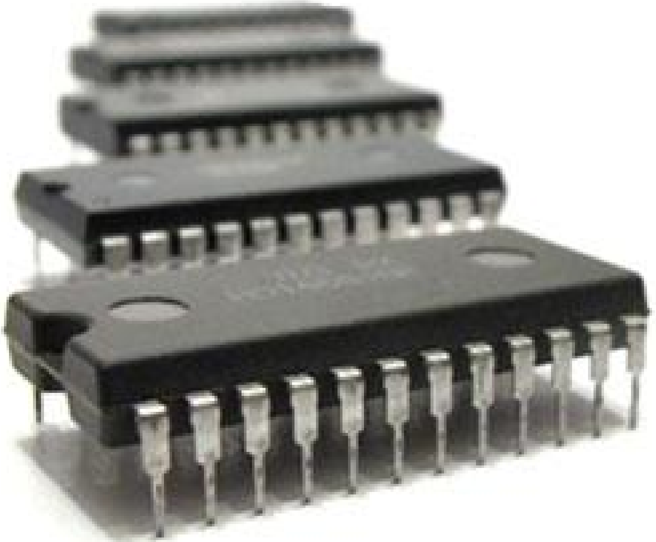
How did electrical engineers create models?

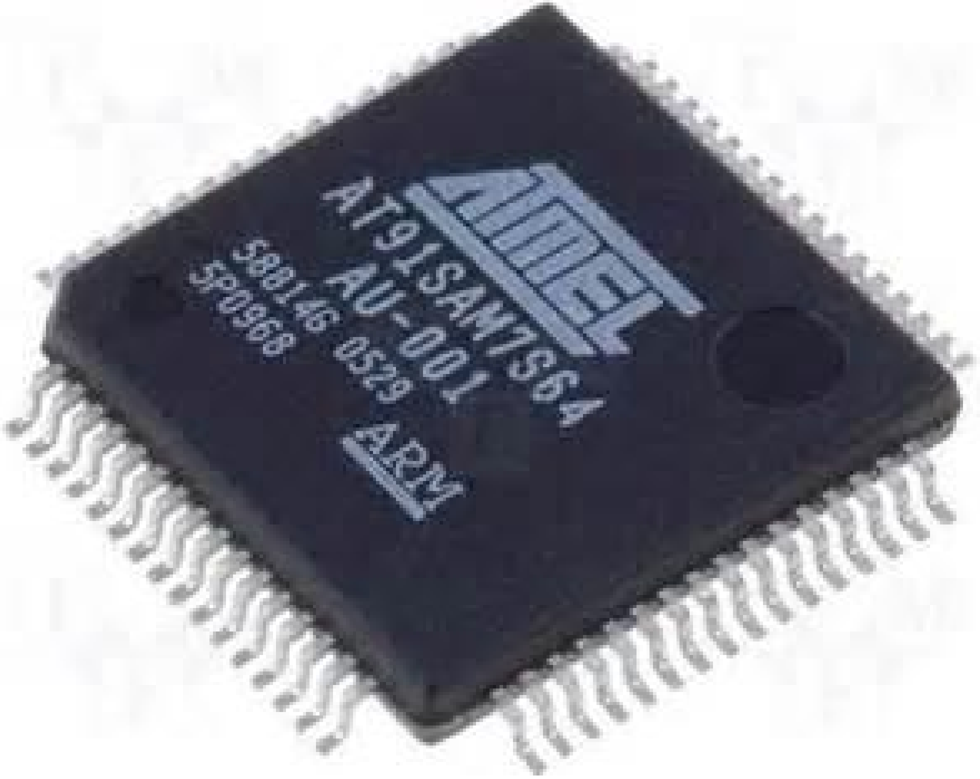




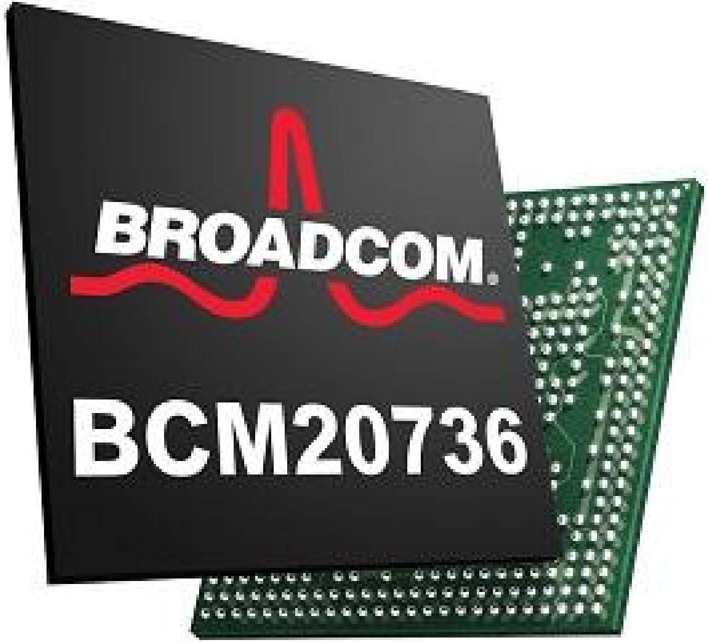
L1 = 10 turn / 0.5 mm SWG25 d = 3mm
L2 = 12 turn / 0.5mm SWG25 d = 5mm
L3 = 4 turn / 1.2mm SWG18 d = 5mm

Next came Integrated Circuits (IC)



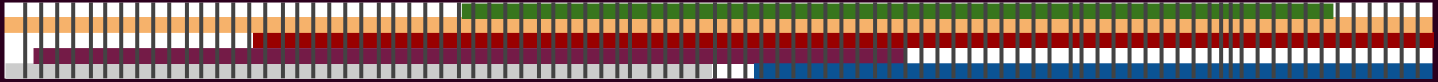


Then System on Chip (SoC)



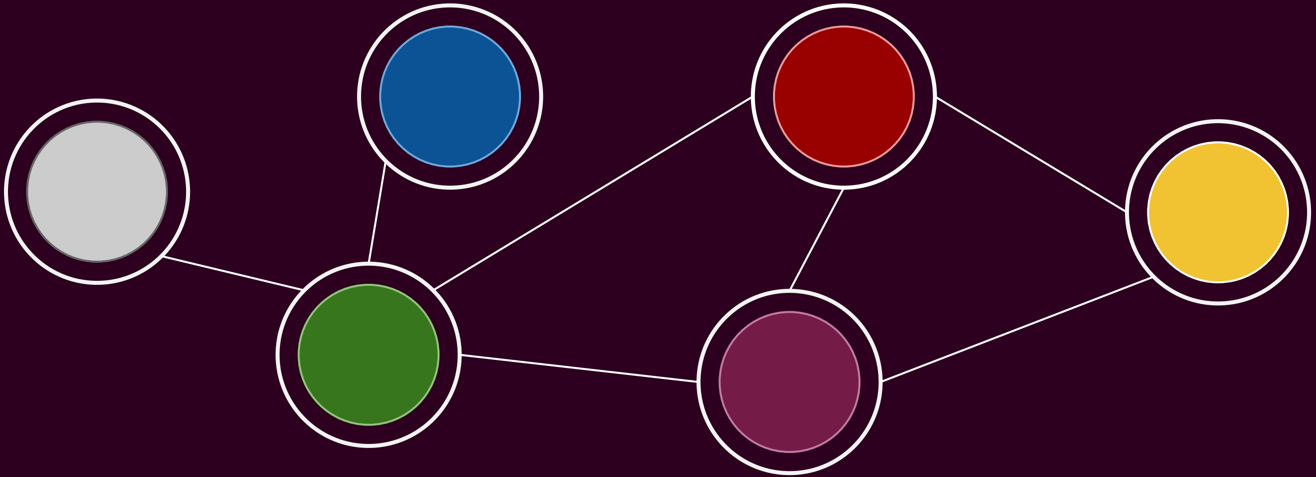
The same thing that happened in hardware

is happening in **software**



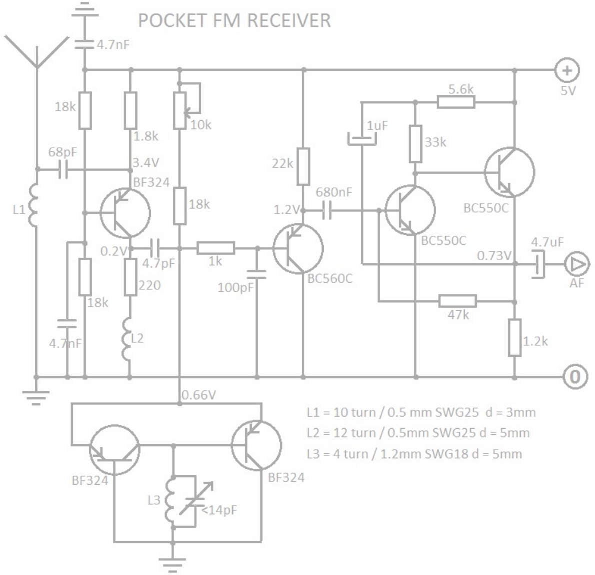
the phase change of modern software

scale, topology, momentum



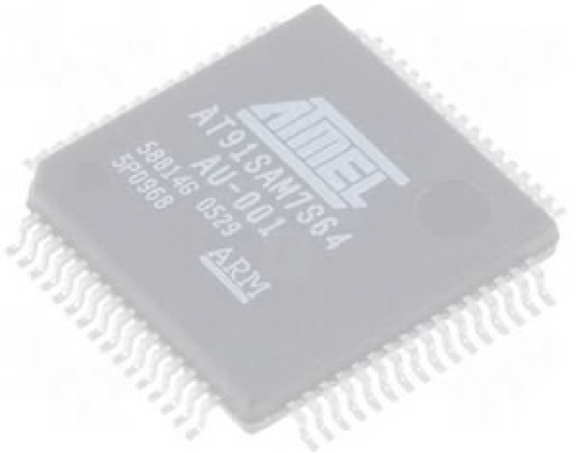
software is abundant

Making writing software less necessary



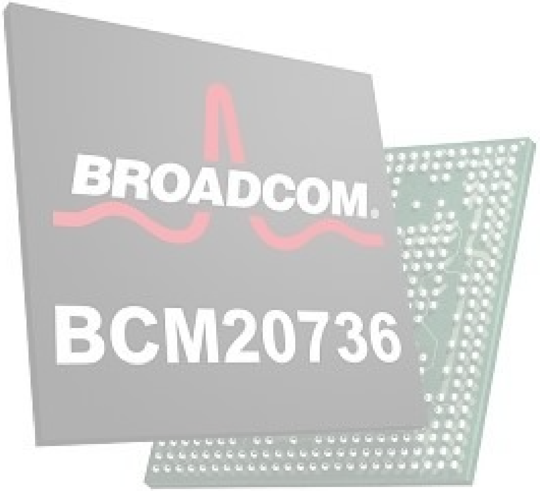
vi main.c

We used to write our programs from scratch



./configure; make; sudo make install

Install someone else’s project



sudo apt-get install

Install packaged software

the new software is too big for apt-get

Install software on the multiple machines in the cloud

What does that well?

the operations team

uses automation to install things in the cloud scarcity has shifted from code to operations

Build a solution from other pieces of software

But everyone is doing it differently

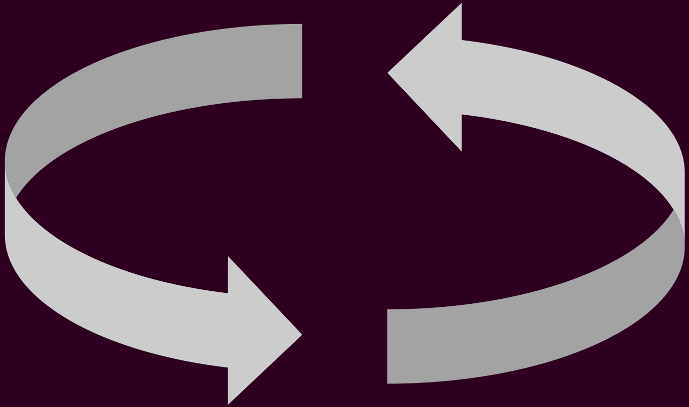
Industrial Duct Tape

Everyone pull in dependencies and grab your adhesive

To reach the next level we must

stop building “snowflakes”!

the solution is reusable, open source operations features



re-use

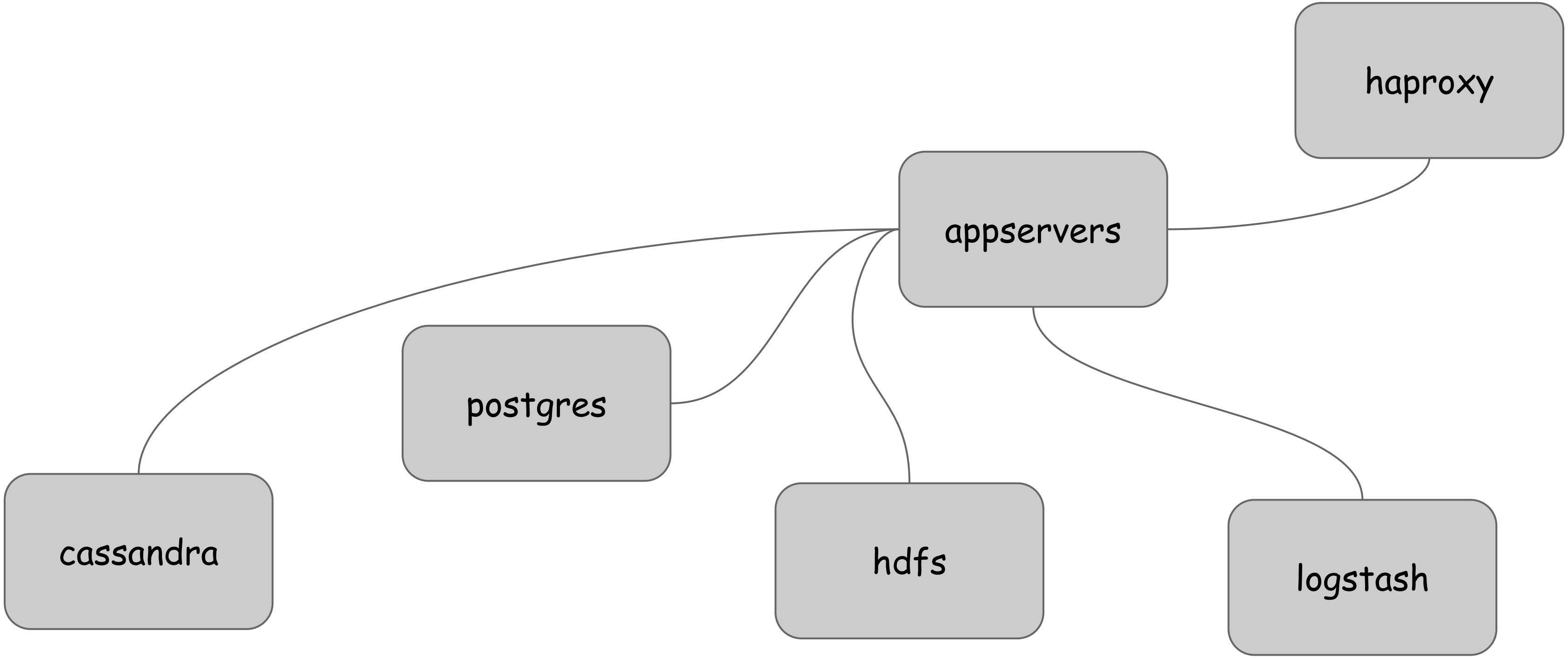
quality

cost

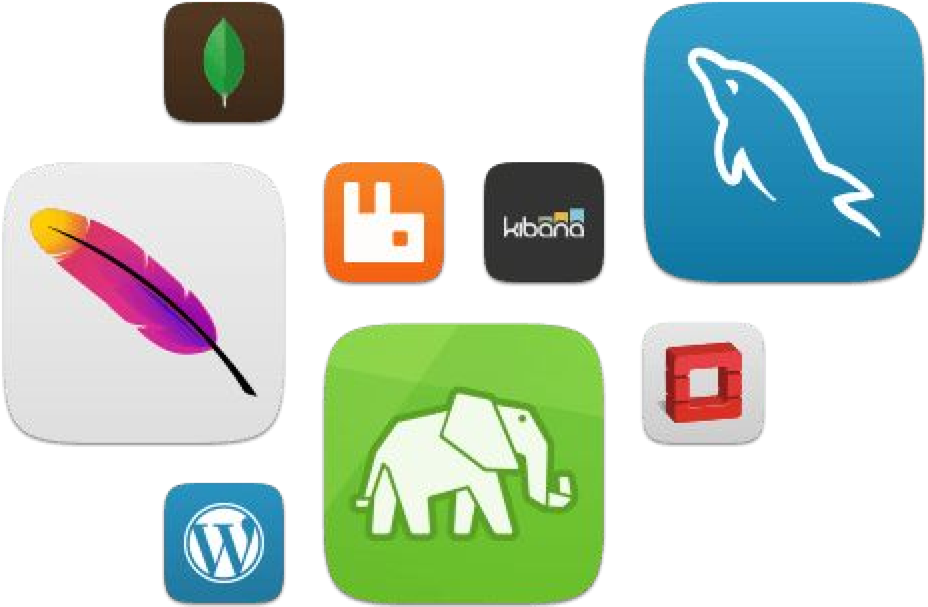
reuse requires encapsulation

e.g. deb, rpm

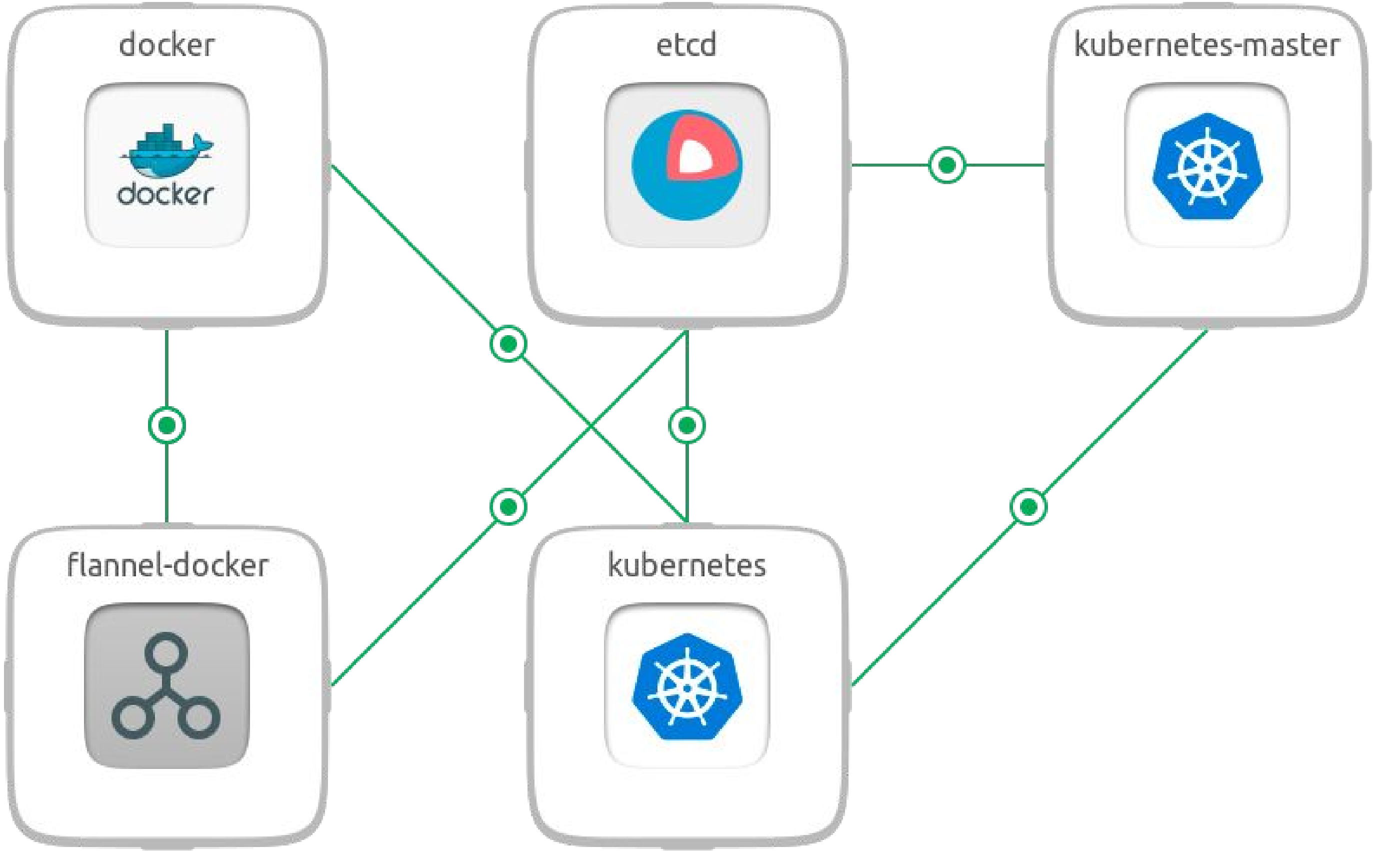
Encapsulation requires a model…
“Stuff connected to other stuff.”

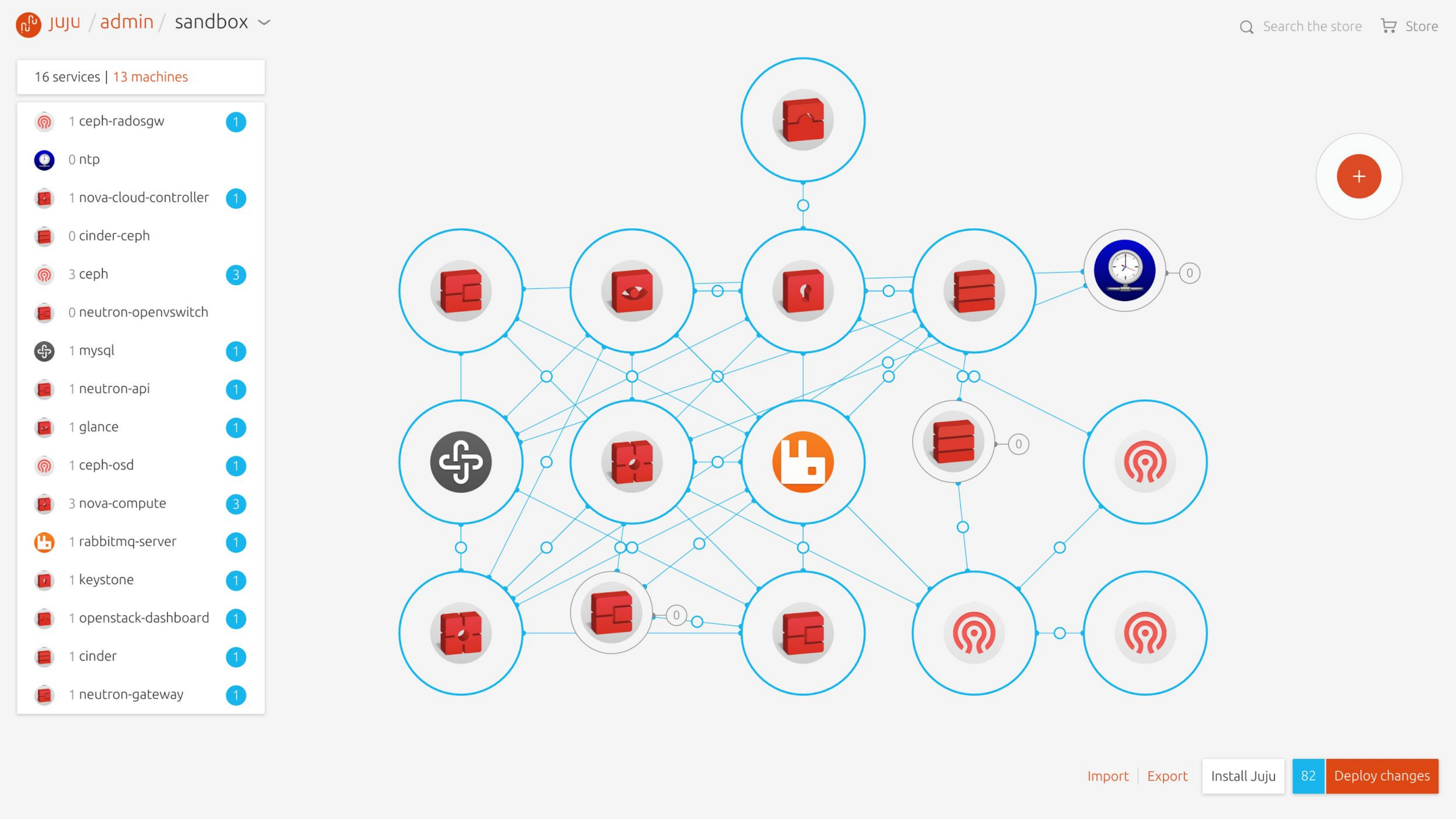


Charms are the building blocks of Juju models



to build application models





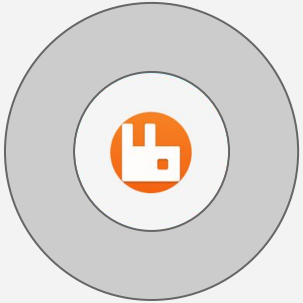
16 services | 13 machines

1	ceph-radosgw	1
0	ntp	
1	nova-cloud-controller	1
0	cinder-ceph	
3	ceph	3
0	neutron-openvswitch	
1	mysql	1
1	neutron-api	1
1	glance	1
1	ceph-osd	1
3	nova-compute	3
1	rabbitmq-server	1
1	keystone	1
1	openstack-dashboard	1
1	cinder	1
1	neutron-gateway	1

A charm contains more than operations

Raw materials

- Puppet, Chef, Bash scripts
- Tarballs, Zipfiles, JARs
- PyPI, NPM, Gems
- Golang builds
- Docker images
- KVM or Hyper-V images



Operations

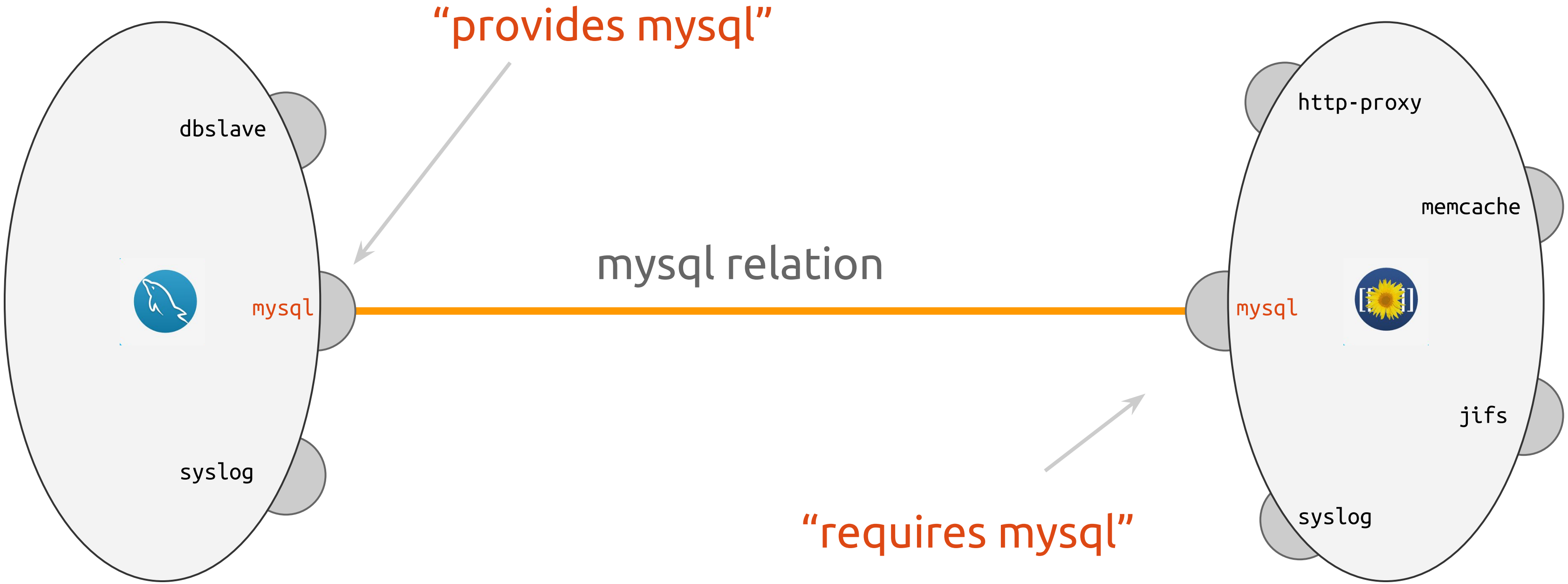
- Install & remove software
- Replication, HA, Scale-out
- Status changes & messages

- Integration
- Firewall management

Charms encapsulate application operations

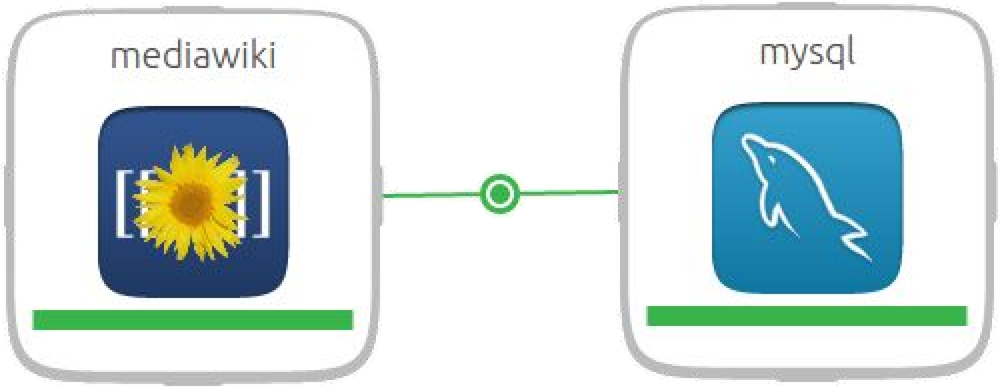
- → installation
- → configuration
- → connections
- → upgrades and updates
- → scale-out and scale-back
- → health checks
- → operational actions
- → benchmarks

Charms declare “interfaces”



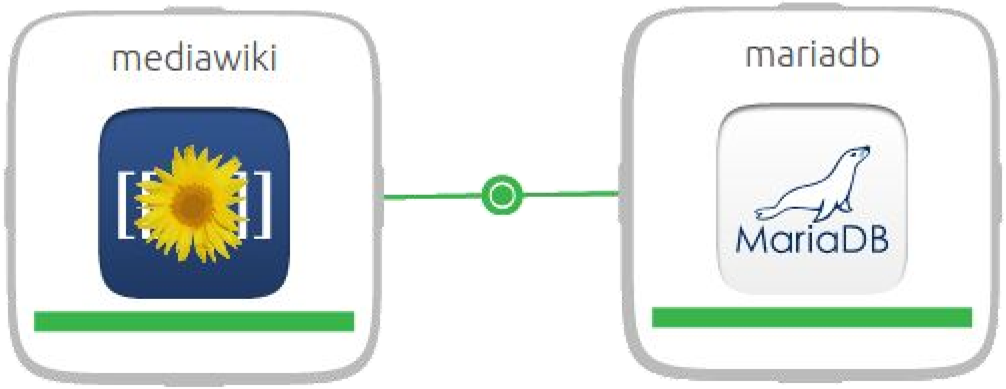
Charms are connectable

mediawiki **requires** a database, mysql **provides** a database



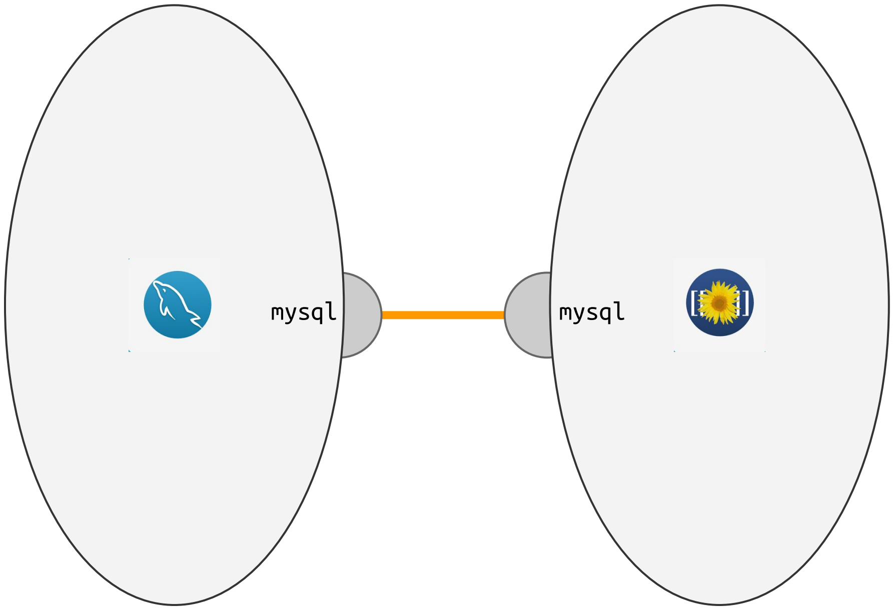
Charms are interchangeable

mariadb also **provides** a database



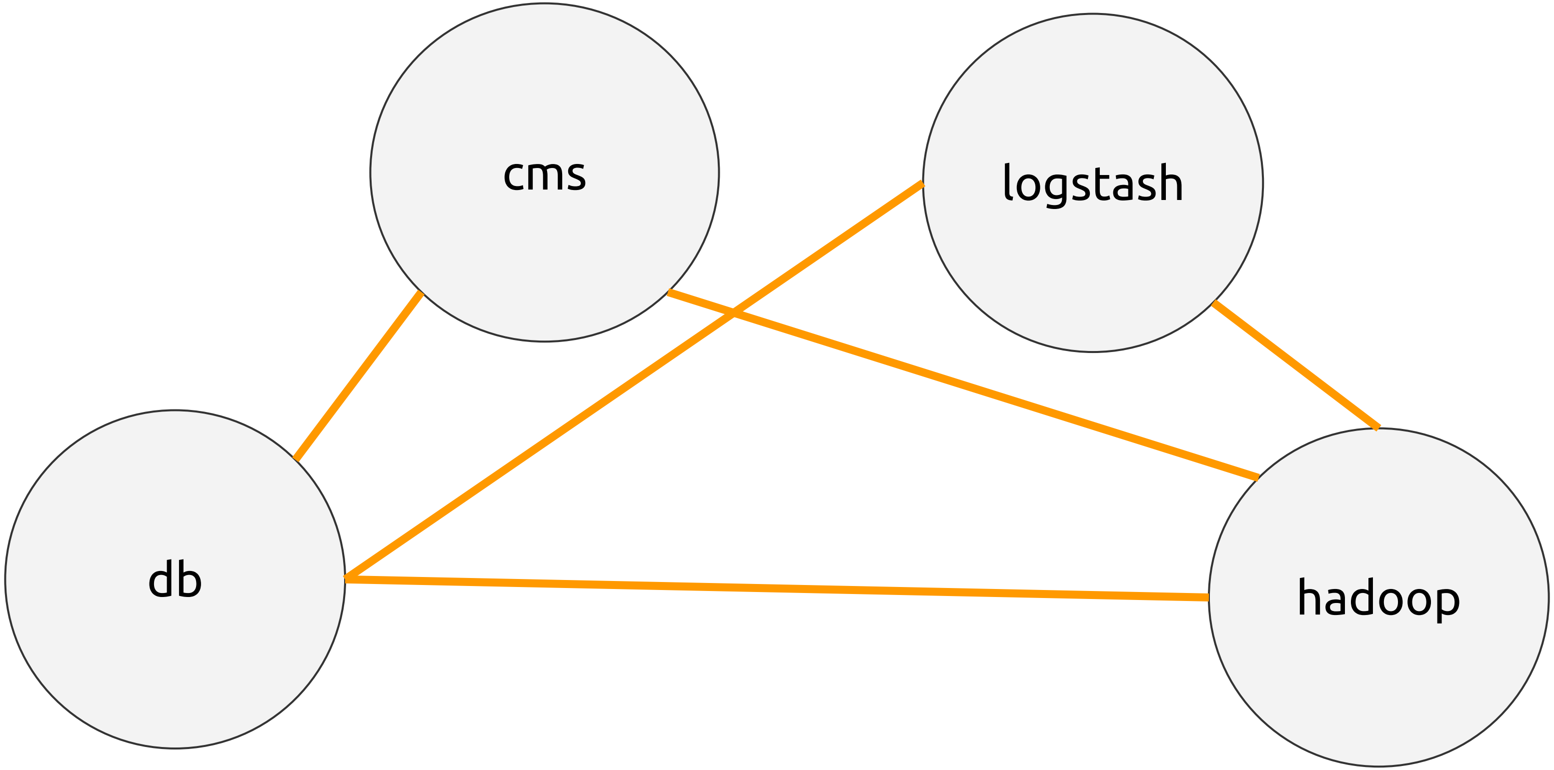
Event handling is done in “hooks”

build
install
upgrade
health
syslog-relation-join
syslog-relation-leave
mysql-relation-join
mysql-relation-leave
slave-relation-join
slave-relation-leave
scale
stop

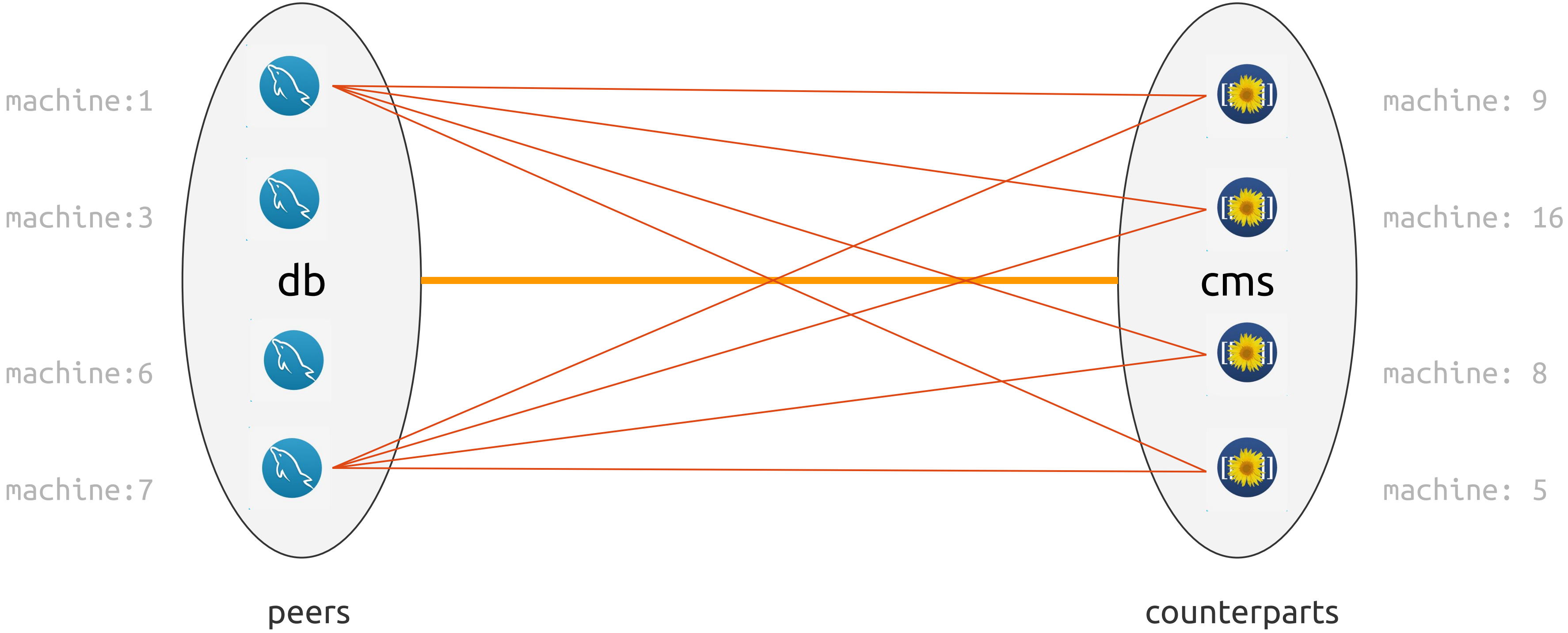


build
install
upgrade
health
http-relation-join
http-relation-leave
mysql-relation-join
mysql-relation-leave
memcache-relation-join
memcache-relation-leave
jifs-relation-join
jifs-relation-leave
scale
stop

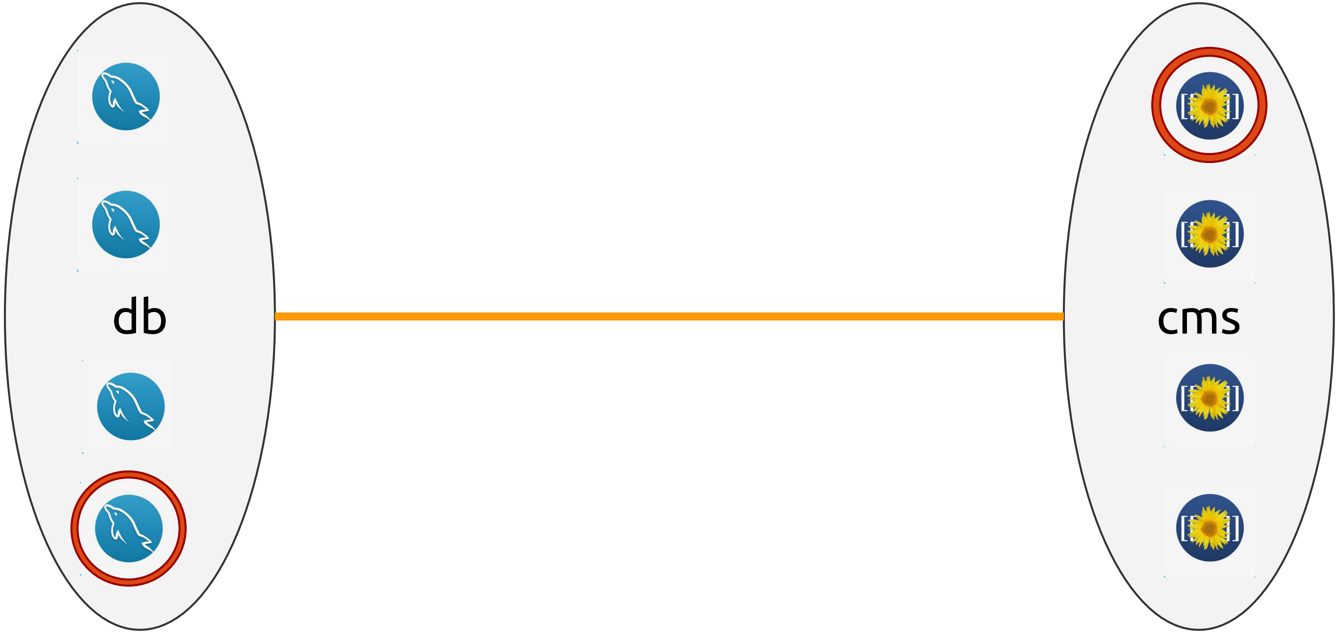
Model simply describes complex topologies



Apps can be scaled across many “units”



Juju designates “leader” units for each app



bundles are the model

bundles are a static representation of the model

bundles are:

- A collection of charms
- A list of relations
- The constraints of the systems
- The configuration of the charms

A bundle can be deployed in a single step.

Bundles allow easy repeatability, and a way to share complex models with other people.

to the Cloud!

c c vm c c c c
c c vm c c c c
c c vm c c c c
c c vm c c c c
c c vm c c c c
c c vm c c c c
c c vm c c c c
c c vm c c c c
c c vm c c c c
c c vm c c c c
c c vm c c c c
c c vm c c c c
c c vm c c c c
c c vm c c c c
c c vm c c c c
c c vm c c c c
c c vm c c c c
c c vm c c c c
c c vm c c c c
c c vm c c c c

WHAT IF I TOLD YOU THERE IS NO CLOUD
IT'S JUST SOMEONE ELSES COMPUTER

cloud computing

resources have gone from scarcity to abundance

with Juju the model is portable

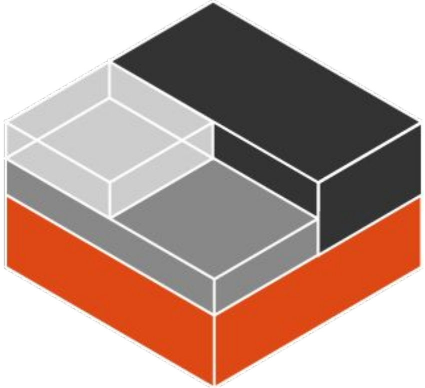
move your application to another cloud

Public Clouds

What if we do not use public clouds?

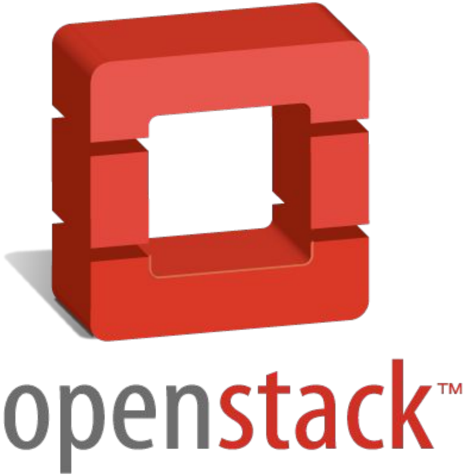
Got servers? You can deploy to the servers that you already have

Juju works with bare metal too



Linux Containers (LXC)

Metal As A Service (MAAS)



Juju works with **bare metal** too

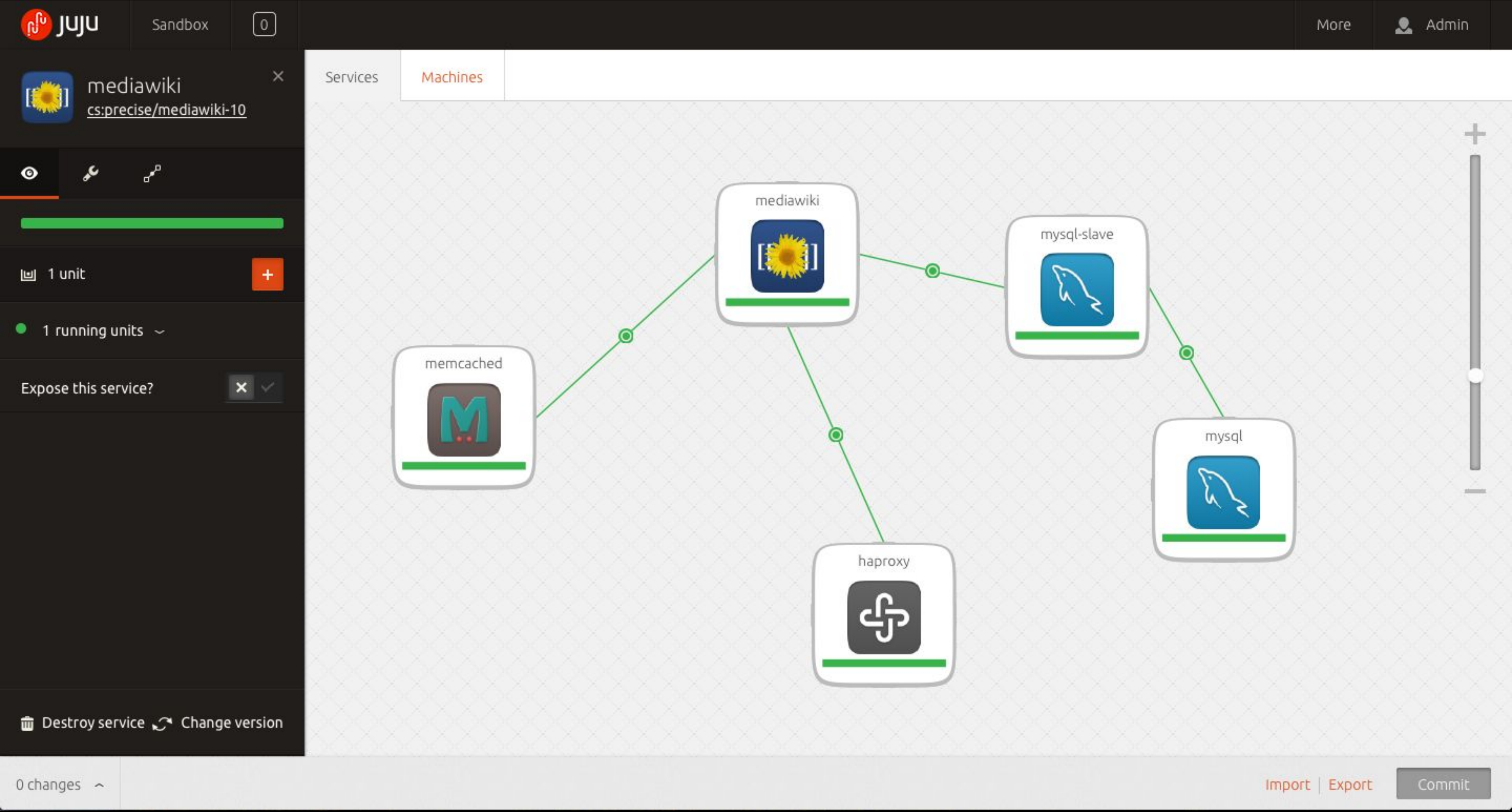
Linux Containers (LXC)

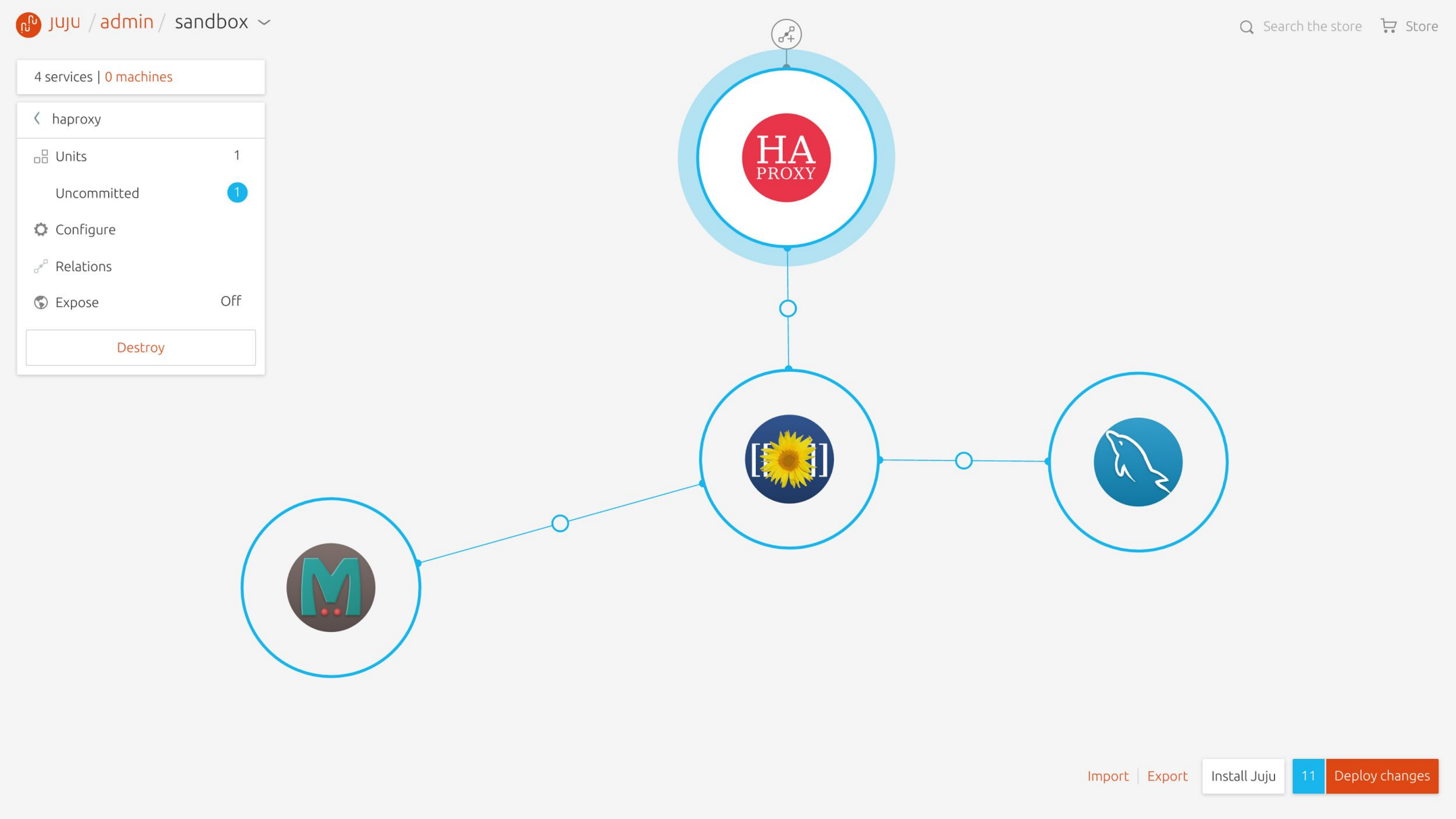
Metal As A Service (MAAS)

http://maas.io

openstack™

VAGRANT


juju
Sandbox
0
More
Admin
mediawiki
cs:precise/mediawiki-10
1 unit
+
1 running units
Expose this service?
Destroy service
Change version
Services
Machines
mediawiki
mysql-slave
memcached
mysql
haproxy
0 changes
Import
Export
Commit

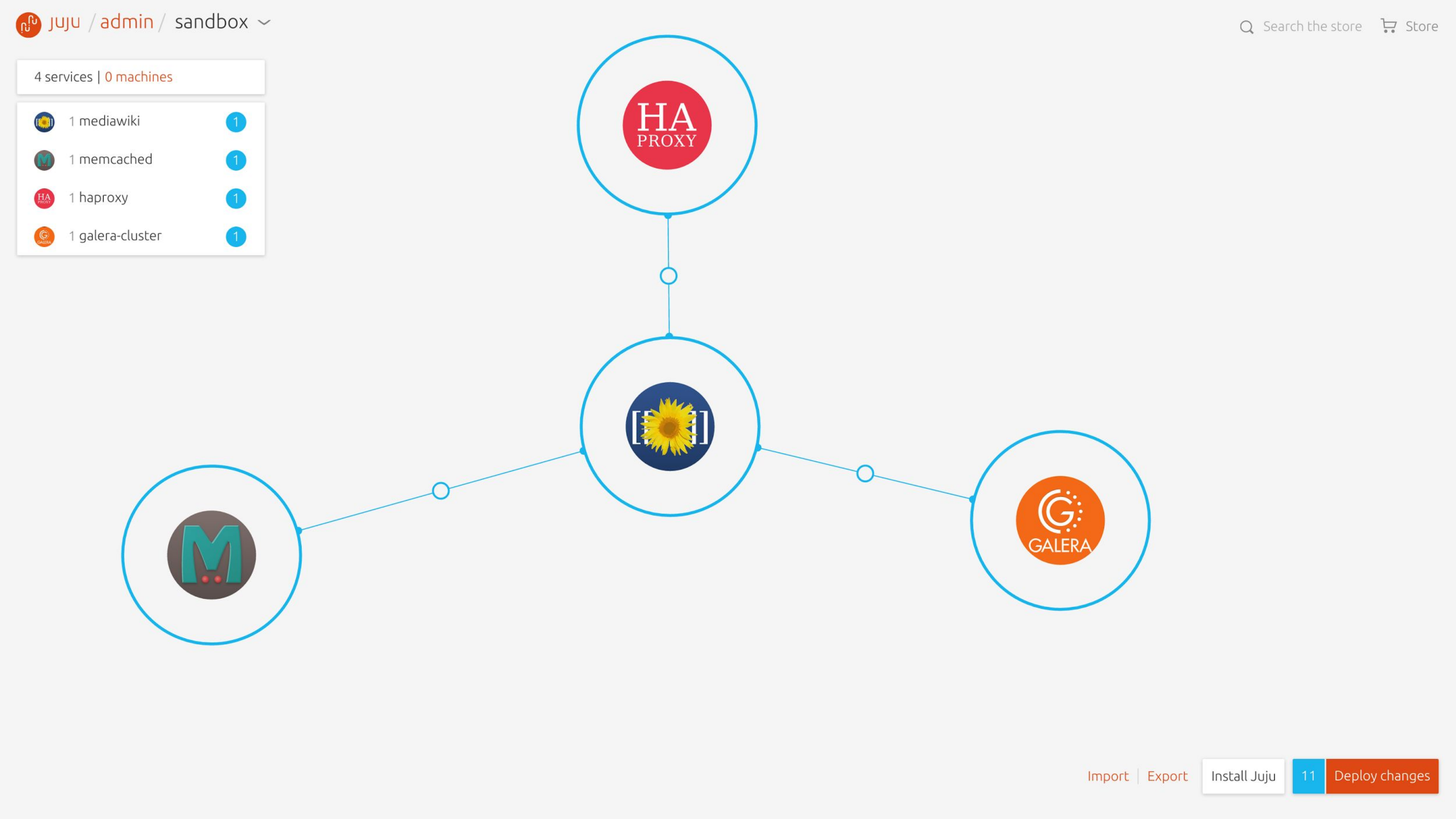


4 services | 0 machines

Units 1

Uncommitted 1

Expose Off

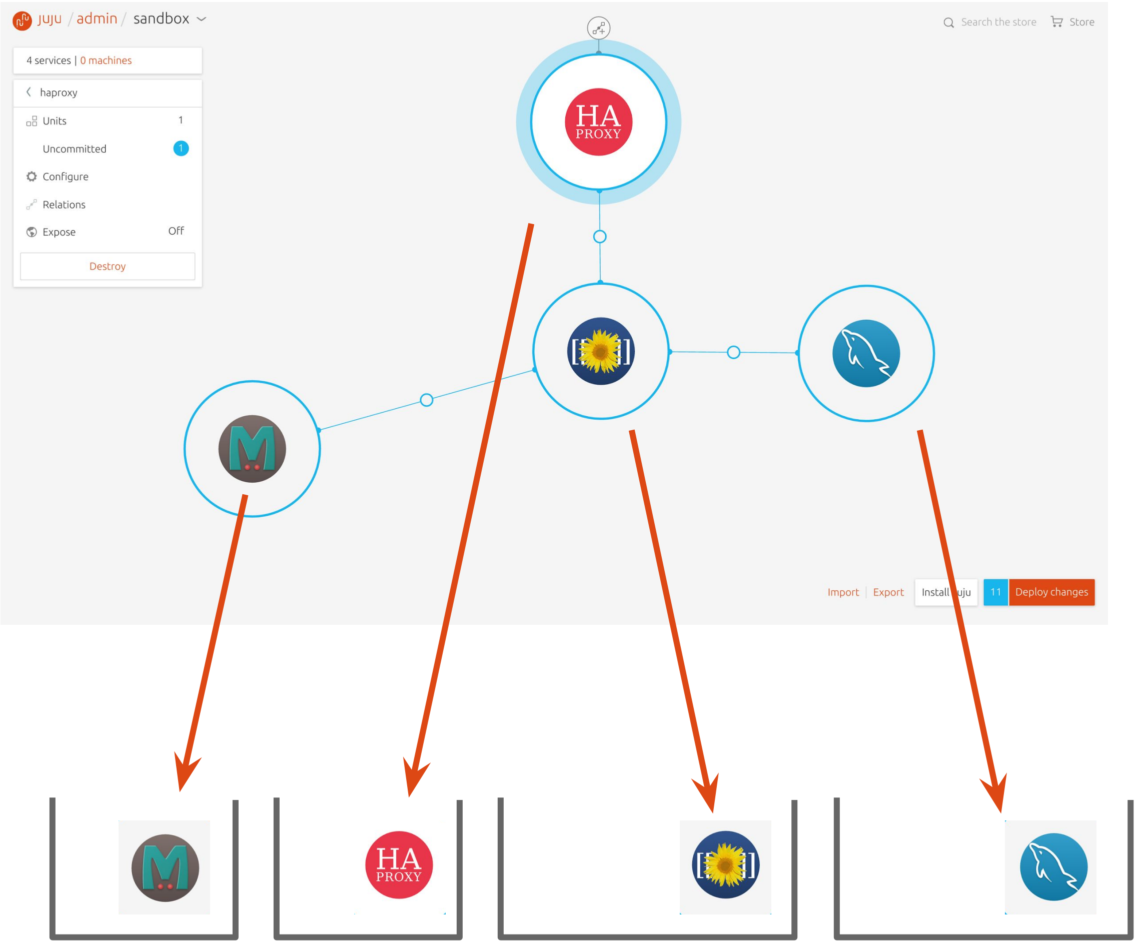

juju / admin / sandbox
Search the store
Store
4 services | 0 machines
1 mediawiki 1
1 memcached 1
1 haproxy 1
1 galera-cluster 1
HA PROXY
GALERA
Import
Export
Install Juju
11
Deploy changes

The modeling language for applications

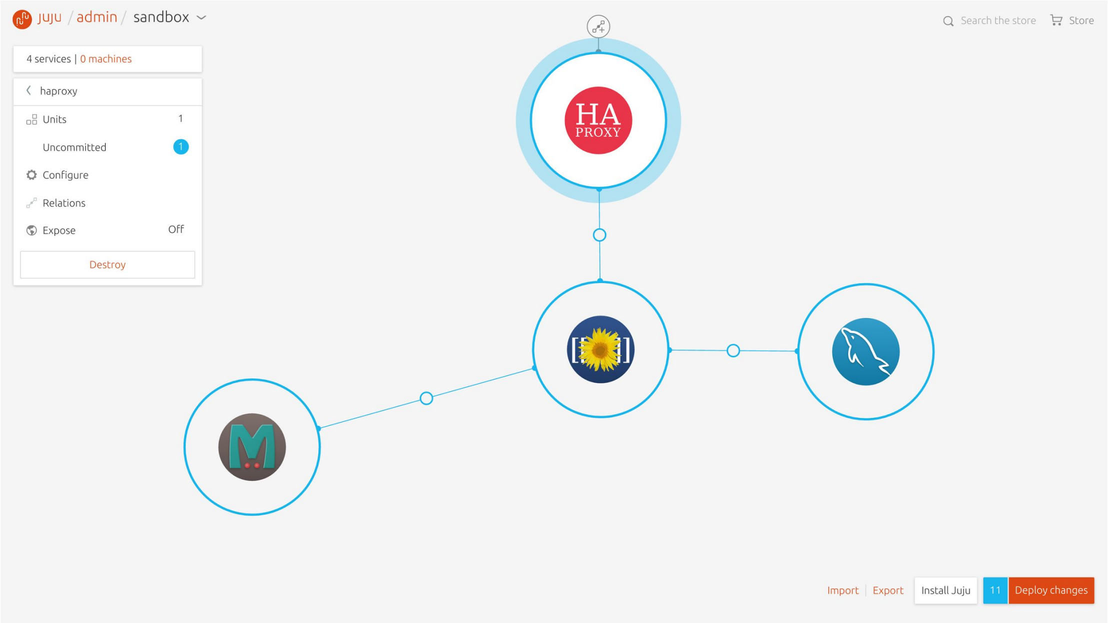
And an abstraction for network, storage and compute resources

application modeling

with reusable software components



m1.small m1.small m1.large m1.large



4 services | 0 machines

haproxy

Units 1

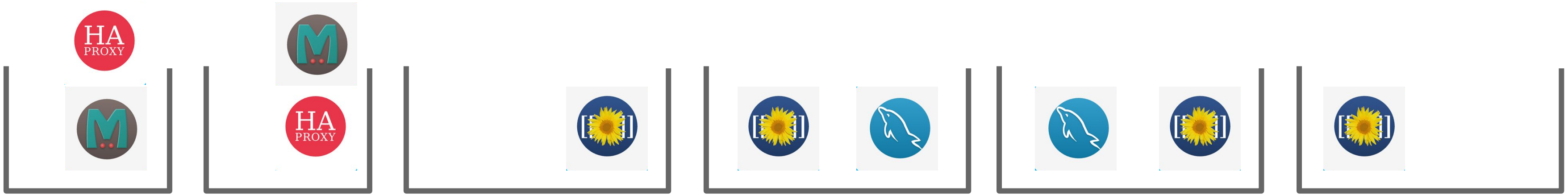
Uncommitted 1

Configure

Relations

Expose Off

Destroy



m1.small m1.small m1.large m1.large m1.large m1.large

Charms are not only Ubuntu!

Ubuntu workloads

CentOS workloads

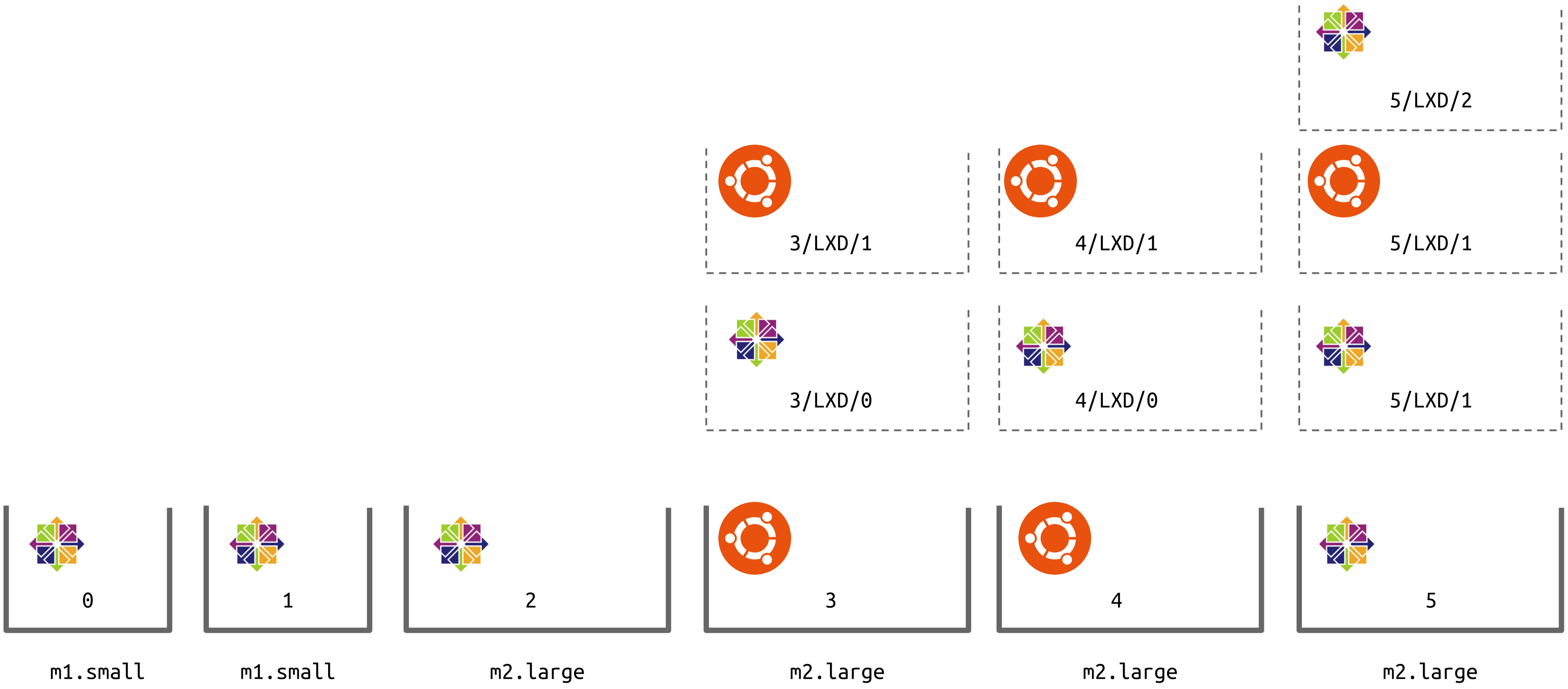
RHEL workloads

Windows workloads

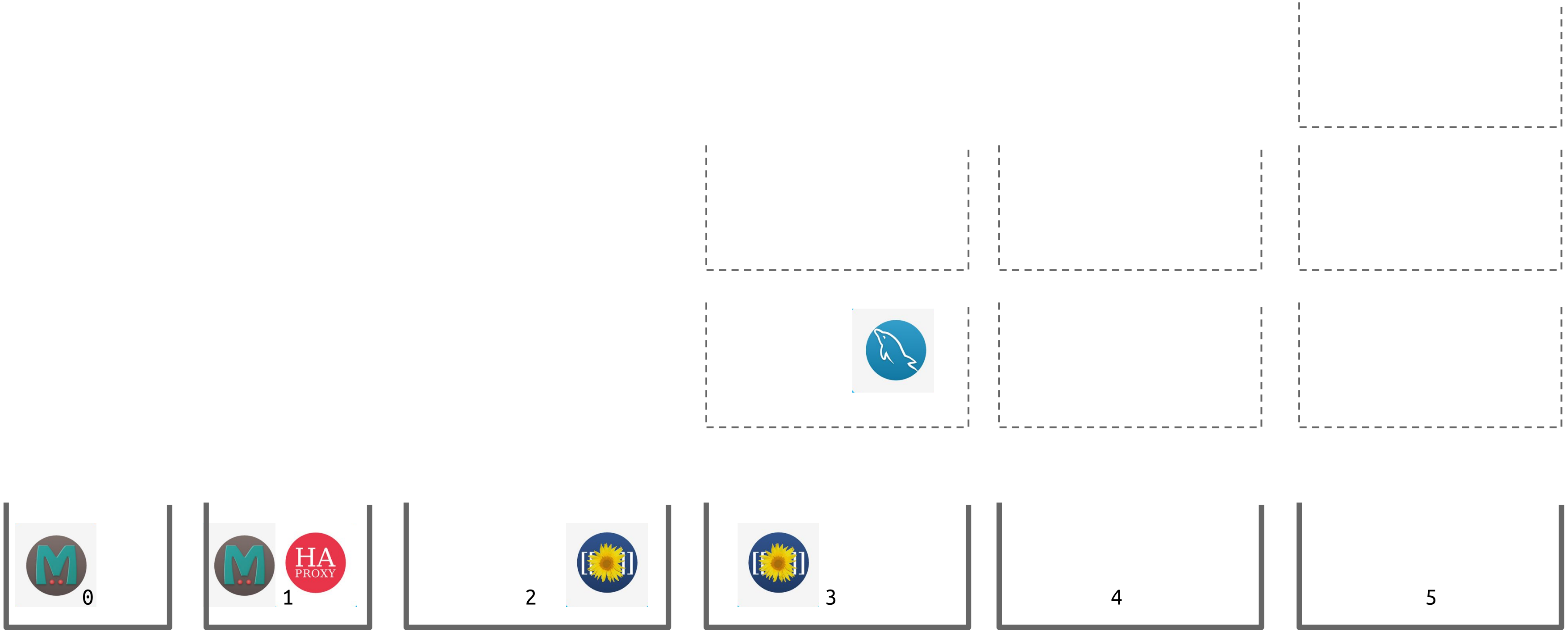
That right you can deploy **Microsoft** services!

http://www.cloudbase.it/juju/

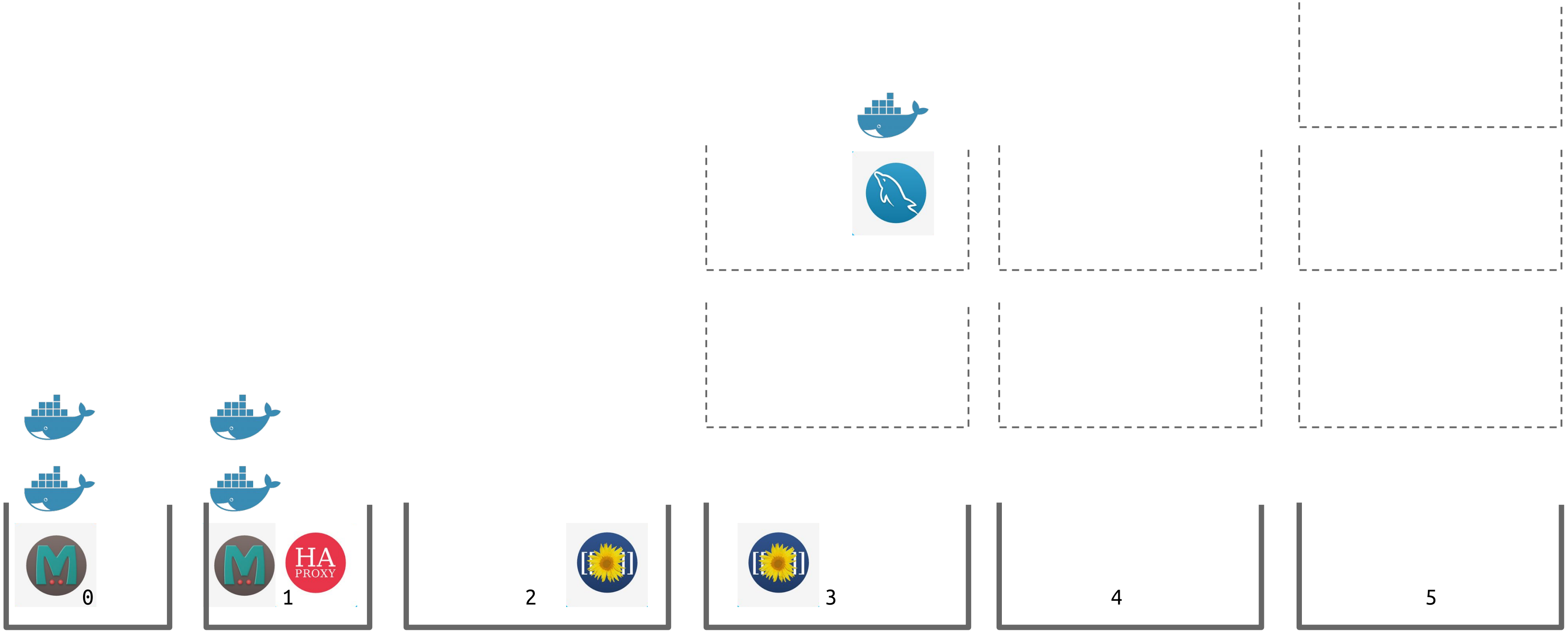
Machines and LXD Containers



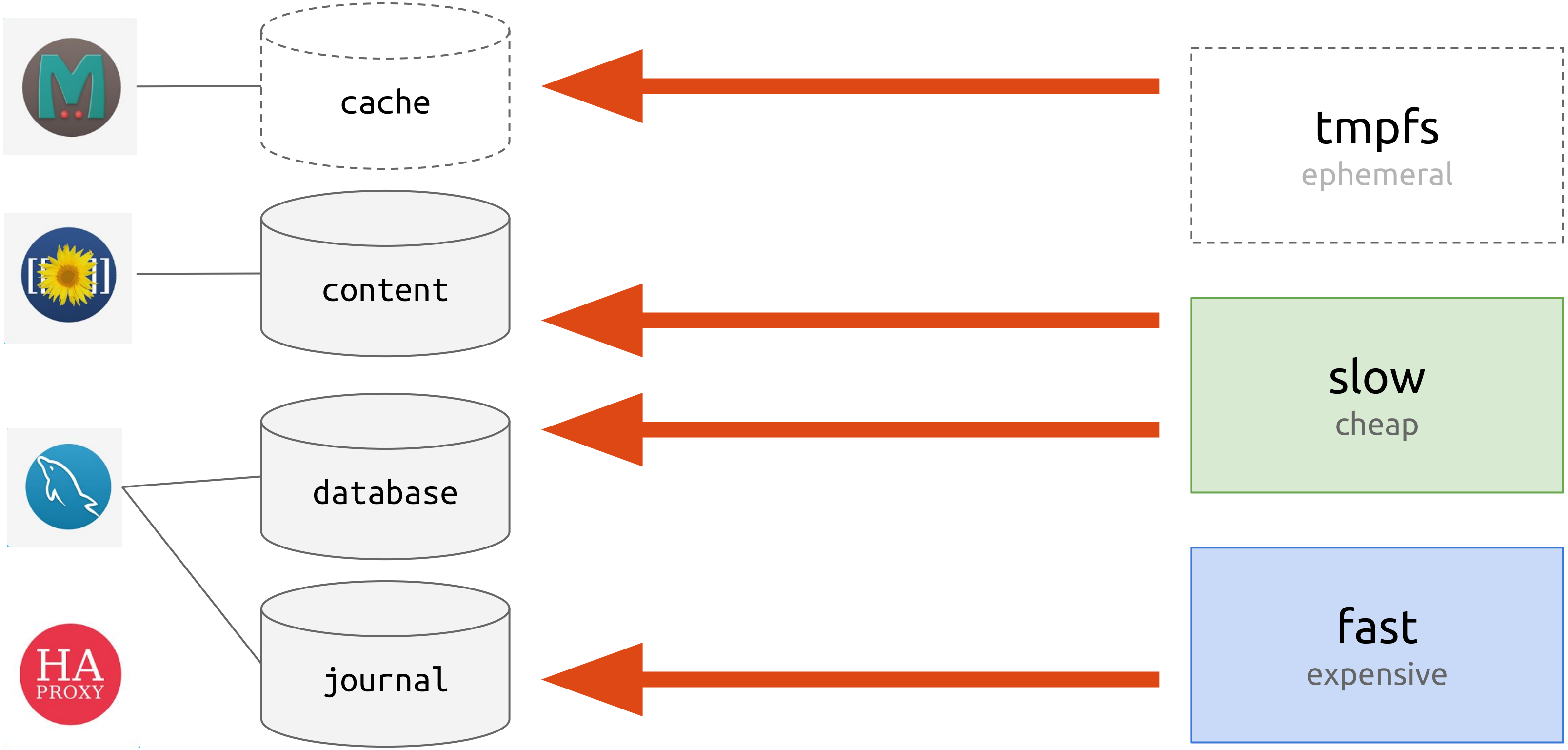
Unit colocation or isolation via LXD containers



Payloads - Docker or KVM

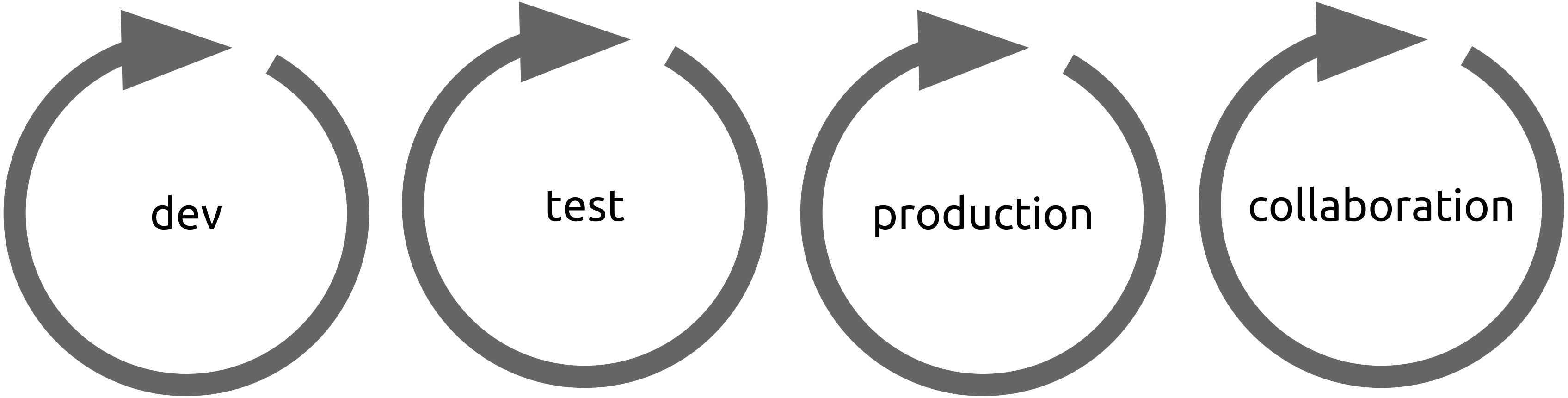


Stores and Storage Pools



Reusable operational components

Faster. Smarter. Better. Everywhere.



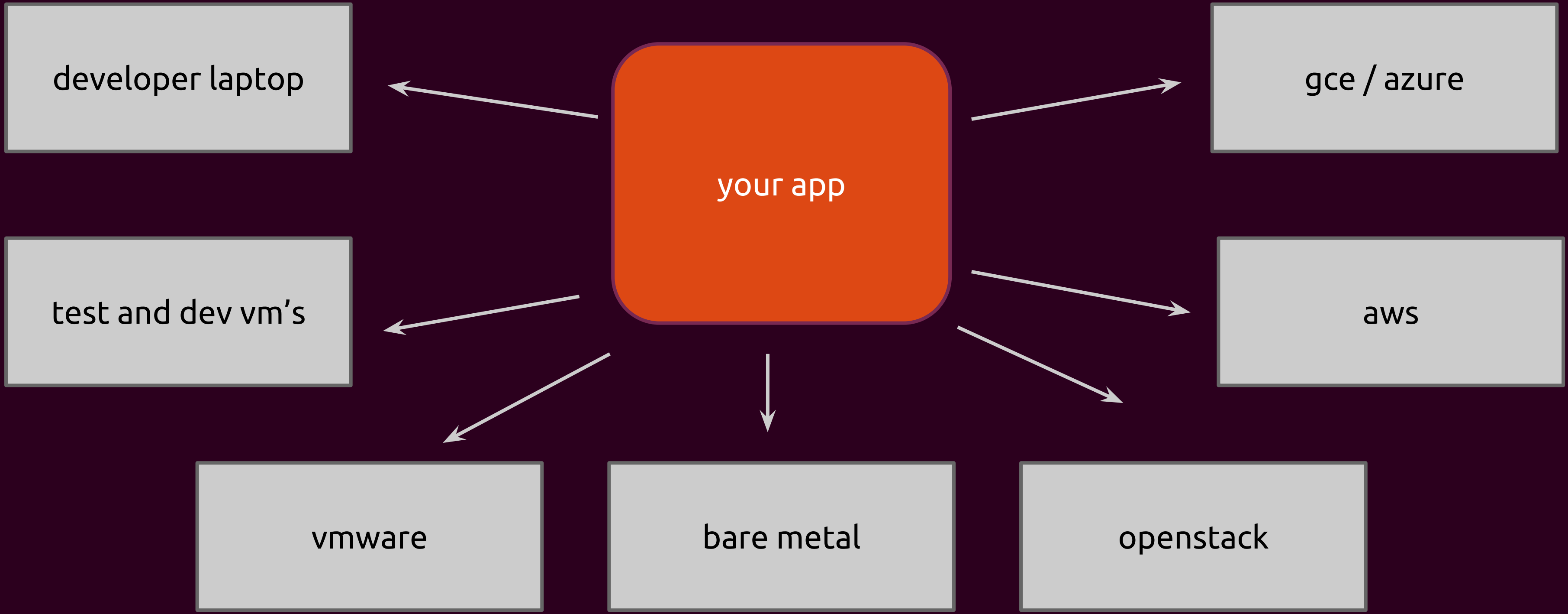
dev

test

production

collaboration

Reuse across clouds



your app

developer laptop

test and dev vm’s

vmware

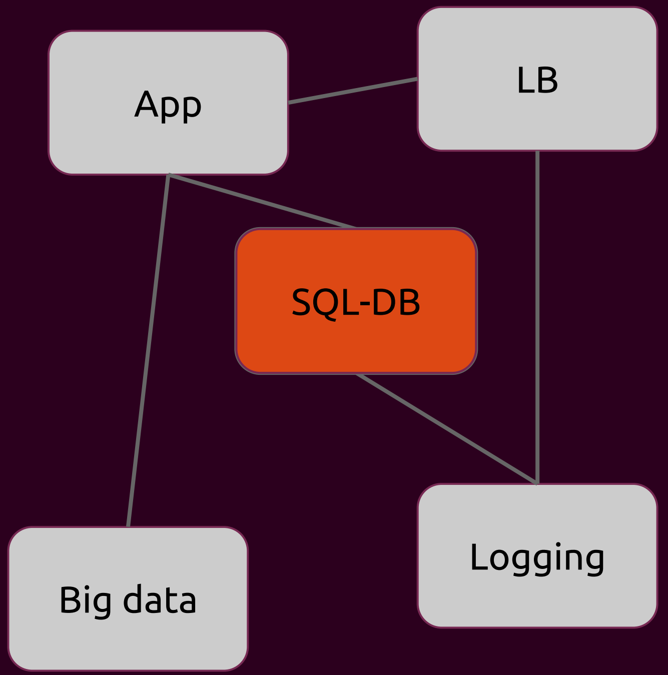
bare metal

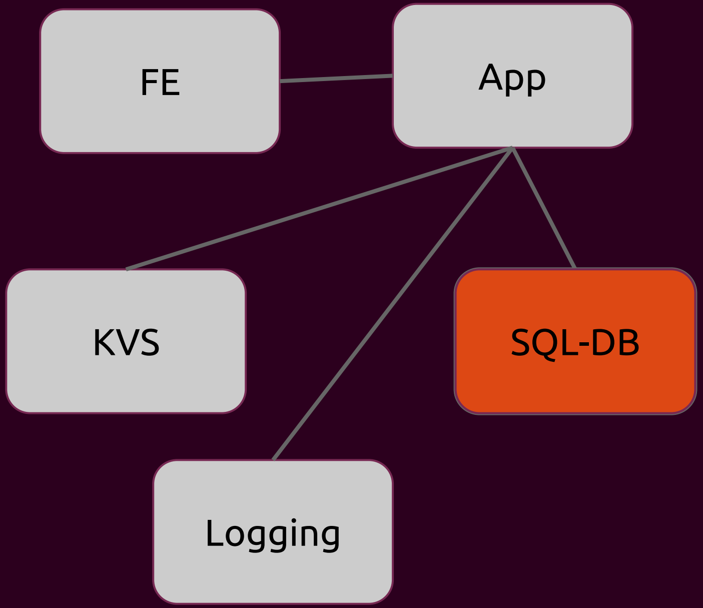
openstack

aws

gce / azure

Reuse across scenarios





beyond automation - reuse & sharing

this is apt-get for the cloud

and it is absolutely amazing!

juju

Demonstration

Questions?

Where to go for more information:

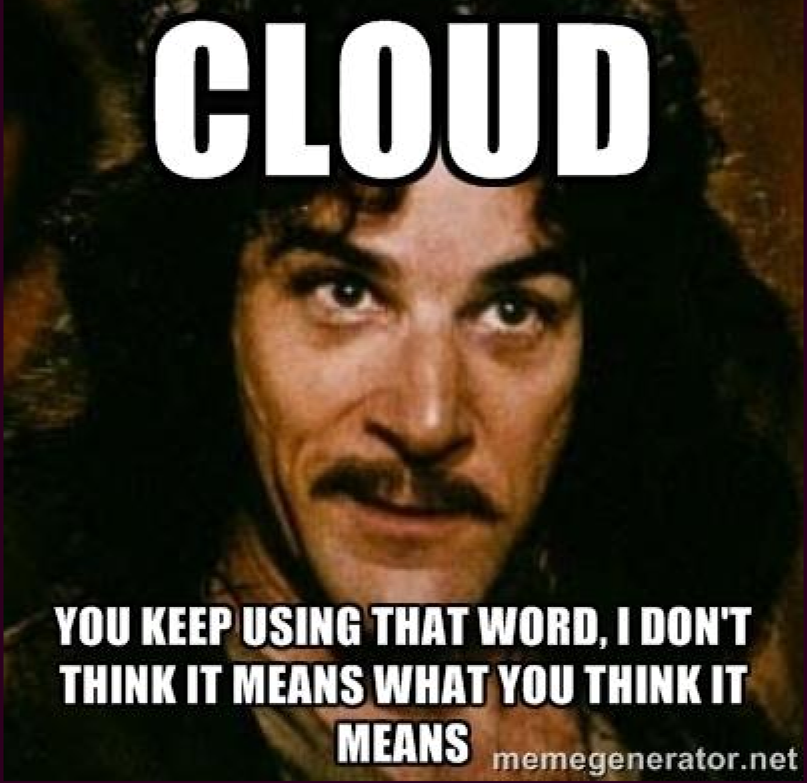
Web site: https://jujucharms.com

The code: https://github.com/juju/juju

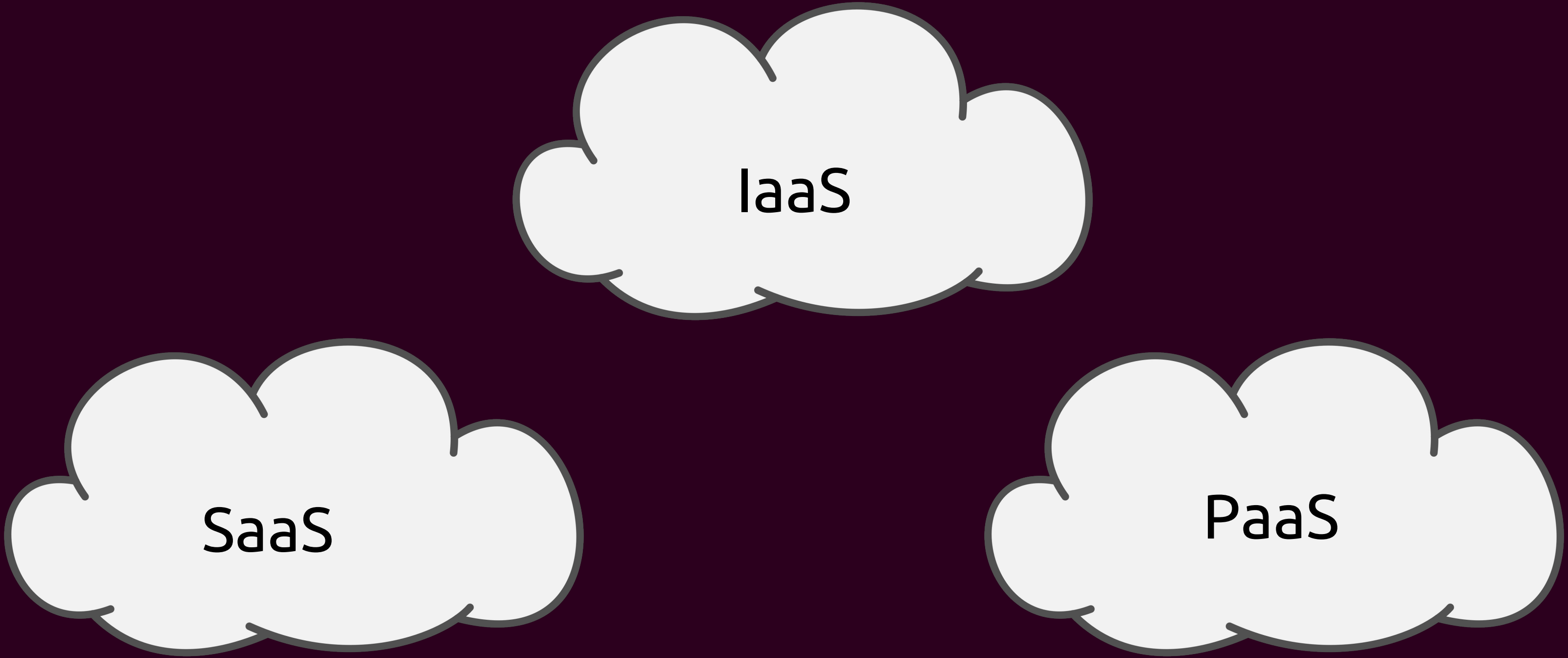
Mailing list: https://lists.ubuntu.com/mailman/listinfo/juju

IRC: **#juju** on freenode.net

Backup slides


CLOUD
YOU KEEP USING THAT WORD, I DON'T THINK IT MEANS WHAT YOU THINK IT MEANS
memegenerator.net

Cloud types



IaaS

SaaS

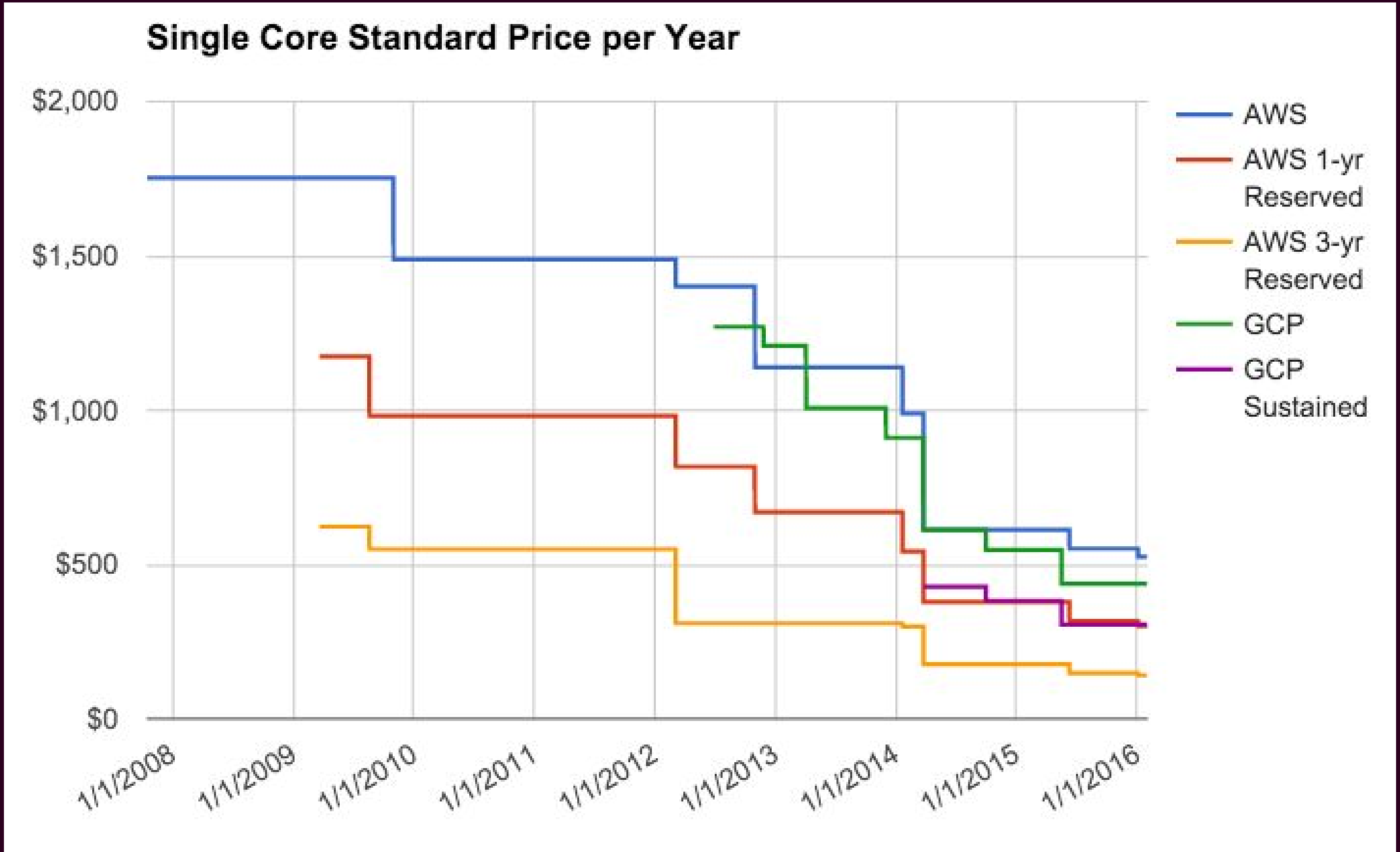
PaaS

HOW DO THEY STORE MY IPHONE PICTURES IN THE CLOUD
WHEN IT IS SUNNY
imgflip.com

We learn by breaking things

It must be cheap to try things, that may break

How much does the cloud cost?



Source: https://quizlet.com/blog/whats-the-best-cloud-probably-gcp

Containers?

Containers

One kernel, multiple isolated processes.

Technology used:

- kernel namespaces
- cgroups
- chroot
- Similar to BSD jails

What they are good for:

- Isolation
- Resource management (CPU, Memory, etc).
- Software delivery mechanism
- **Standardization**

Container technologies

Application containers

- One process in per container

docker

rkt

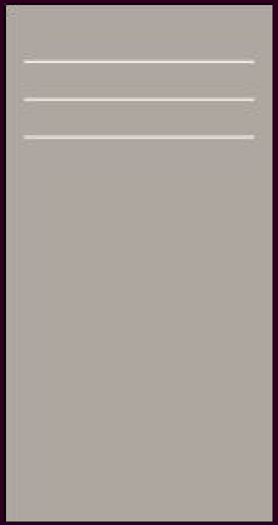
System containers

- Complete OS in a container, including boot sequence

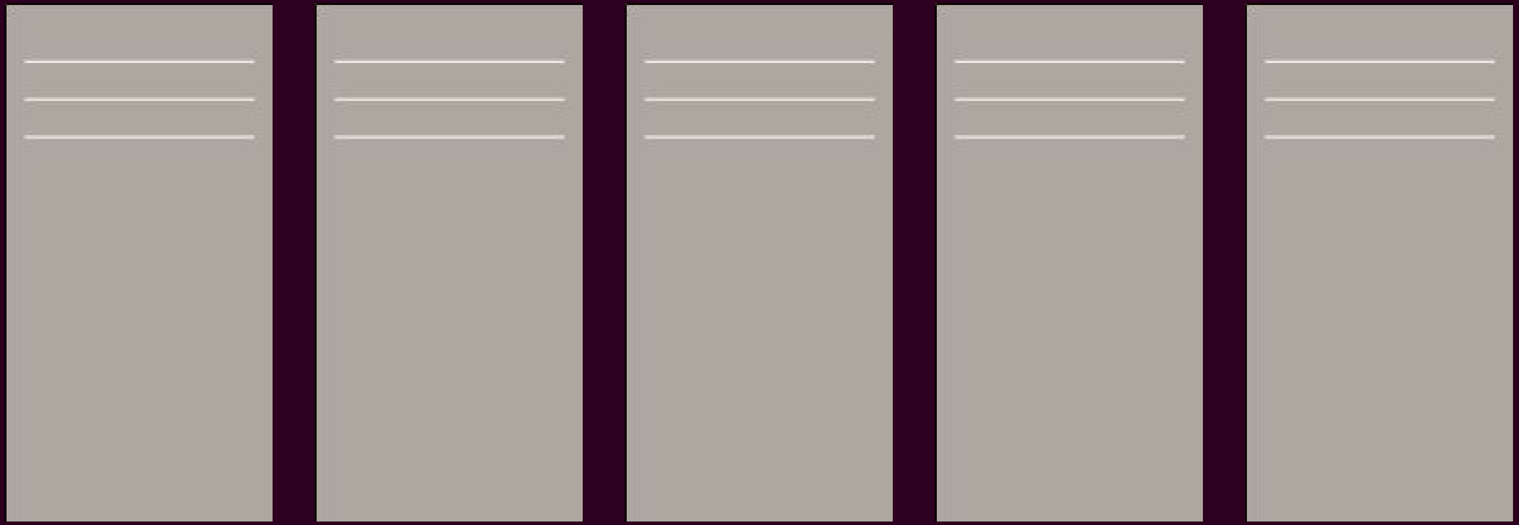
LXC

Computing infrastructure is only getting more complex

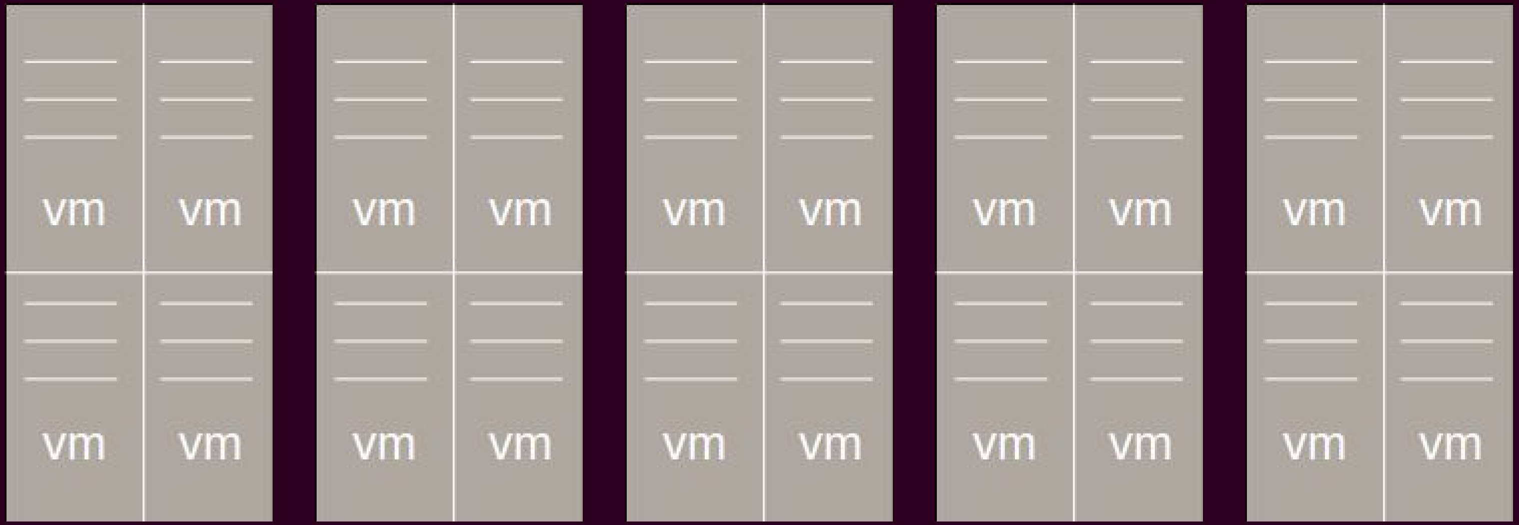
Server



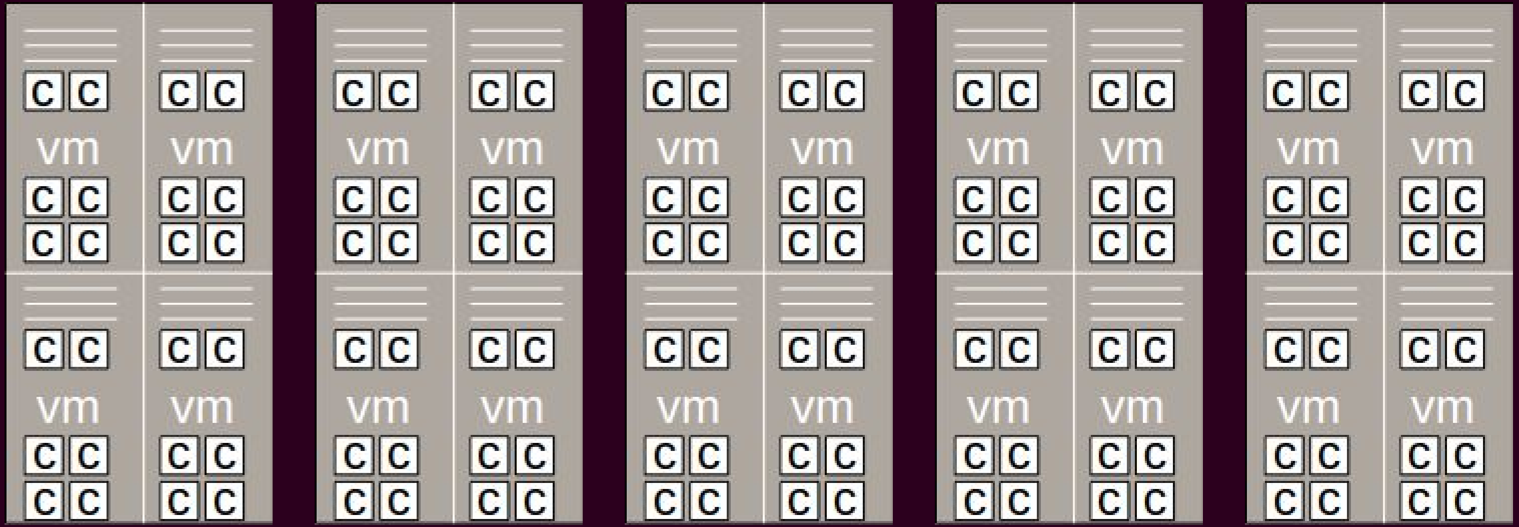
Many servers



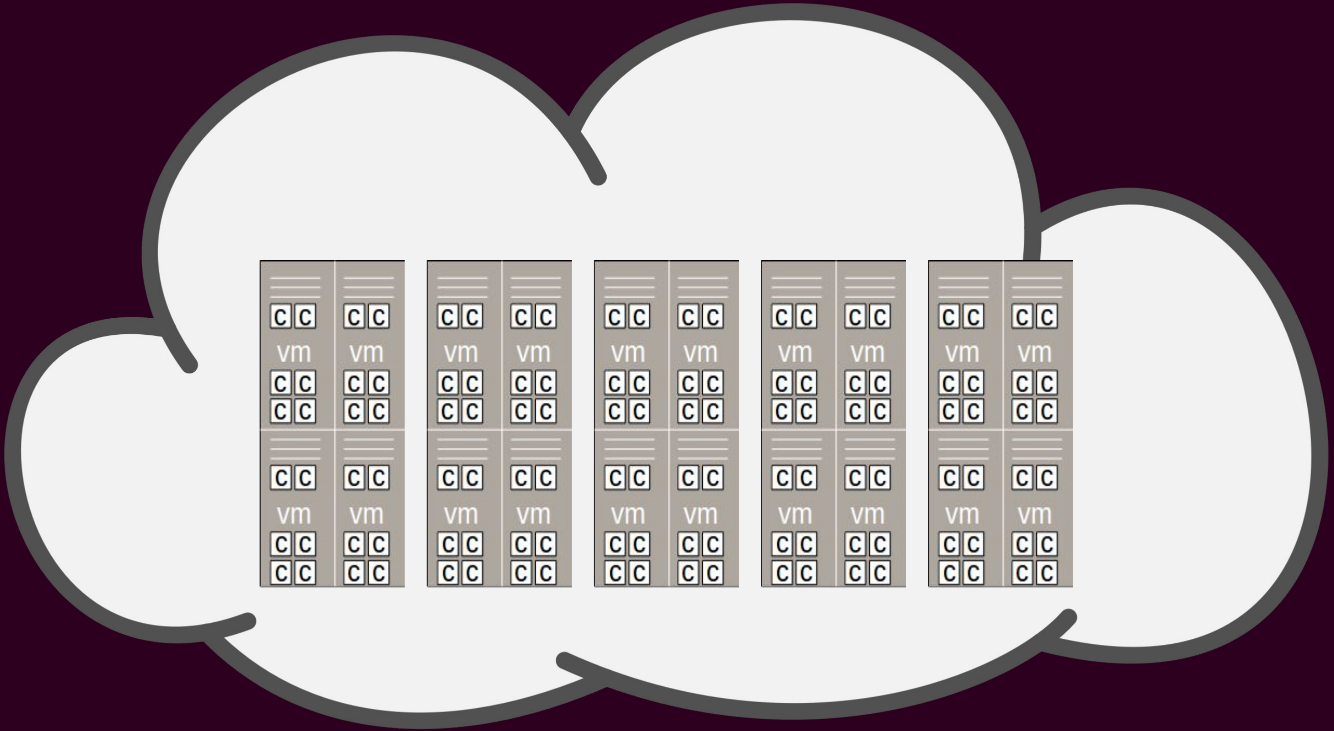
Virtualization



Containers



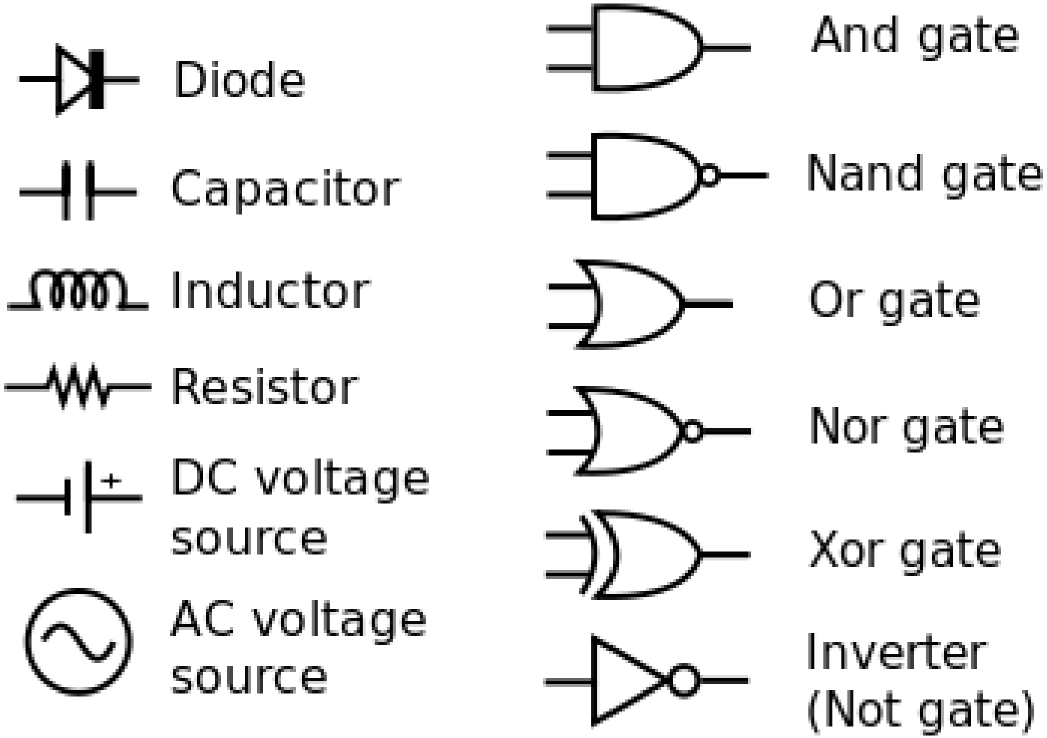
Cloud



> “All problems in computer science can be solved by another level of indirection”

David Wheeler

building blocks you might be familiar with...


Diode
Capacitor
Inductor
Resistor
DC voltage source
AC voltage source
And gate
Nand gate
Or gate
Nor gate
Xor gate
Inverter (Not gate)