



# Sharing OpenStack Reference Architectures

Using OpenStack Orchestration  
(aka Heat) as a RA baseline

Discussion by OpenStack Board Members:

Monty Taylor

Rob Hirschfeld

# OpenStack + Common Ops...

## Multiple Interoperable OpenStack Clouds

- Prevents vendor lock-in
- Establishes best practices
- Provides basis for sharing
- Scales OpenStack Ecosystem

Panel Today @ 5:20

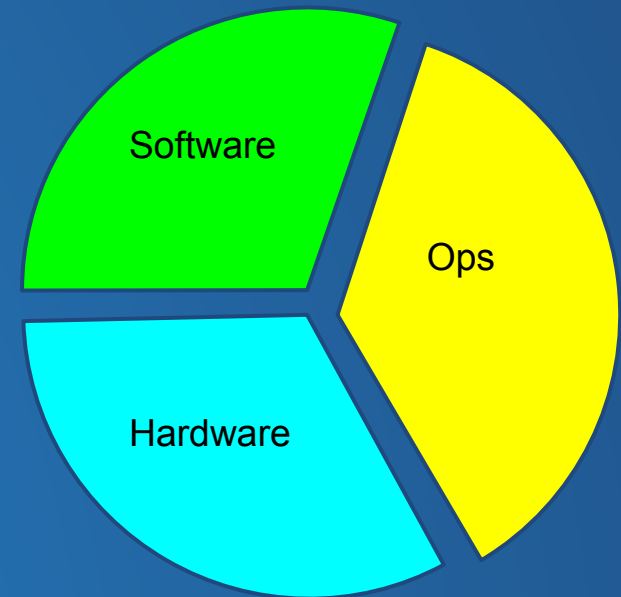
Room A107-A109

# What is the problem?

Everyone does OpenStack differently!  
We share code, but it takes a lot more to run a cloud.

Small differences in hardware, versioning, and implementation

- breaks portability
- limits collaboration
- hampers support



Why solve it?

OpenStack sites as fault zones

Interoperability is valuable

If we can't deploy OpenStack consistently,  
we can't test OpenStack

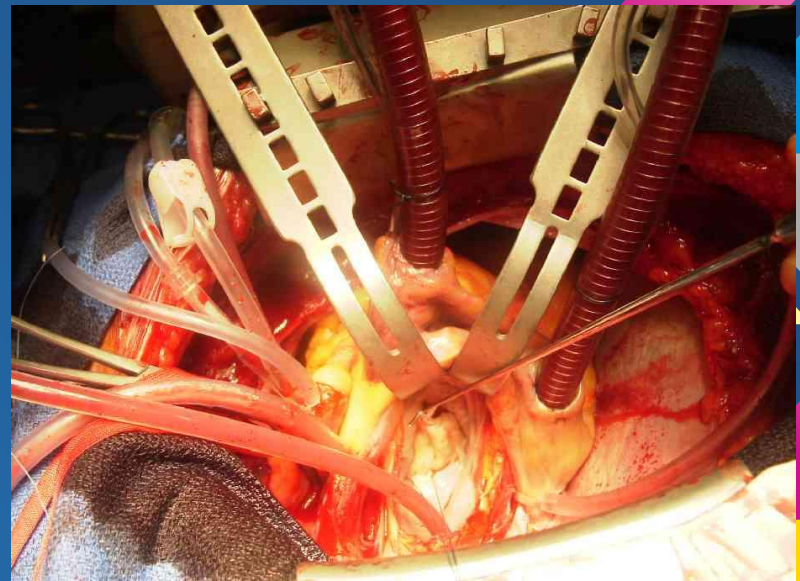
If we can't compare deployments,  
we cannot help or validate OpenStack



<http://www.vibrant.com/>

# Open Operations

Collaborating directly  
on operations code



<http://www.fitchicago.com/heart%20surgery%203.jpg>

## Extends Reference Architectures

- Sharing the "what"
- Enabling environment-specific config
- Your deploy isn't that different
  - Cows and Cats - cows are still snowflakes!
- Helps new users know what works
- Streamlines addressing vulnerabilities

# Upstreaming Ops



If we can agree on a physical configuration  
AND on the code to install then...

Shouldn't we also be able to collaborate on  
the installation scripts?

That would make it easier to interoperate!

# What is an RA?

Resident Advisor? :)

Rich Aunt? B)

Reference Architecture :P

Reference Implementation?



<http://jeffjonas.typepad.com/RubeGoldberg.bmp>

- Generally hardware focused
- Physical layout and wiring
- Defines logical operations and interconnections
- Recommends components
- May include configuration guidance
- May also discuss performance

Not likely to include HA config, upgrades or operation

# RefStack

<https://github.com/openstack-ops/refstack>  
(currently empty)

- One or More RA's
- Defined in OpenStack terms
- Mechanism for testing against an RA
- Vendor neutral & vendor accessible
- Intended as best practices (not only way)



# Flavors of RA

We don't know all of them,  
but need to limit number!



[http://blogs.laweekly.com/squidink/2011/11/5\\_best\\_ice\\_cream\\_shops\\_losan\\_angeles.php](http://blogs.laweekly.com/squidink/2011/11/5_best_ice_cream_shops_losan_angeles.php)

What matters in a favor?

- networking?
- hypervisor?
- scale?
- workload?

# Flavors Requirement

Two OpenStack clouds of the same flavor should be able to *interoperate!*

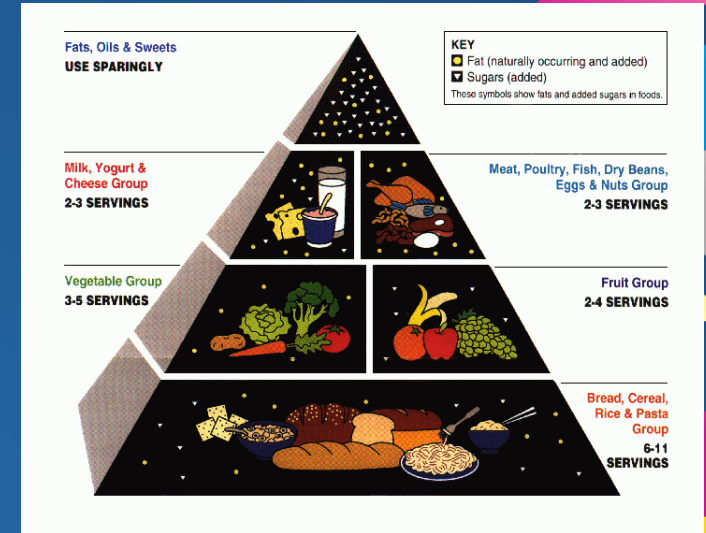
Operators using similar favors should be able to share scripts and processes

There should be as few flavors as possible.

Flavor should be cited in bugs

# Flavor Table ???

Mix and discuss....



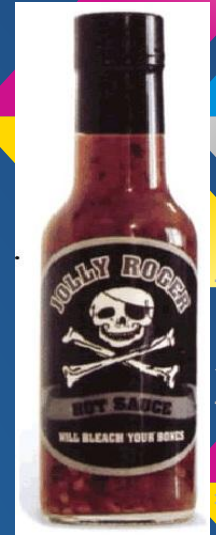
Function	Hypervisor	Networking	Scale
Mixed Use	?	?	?
Compute	?	?	?
Storage	?	?	?
?			

# What is Heat?

- Makefile for a Cloud Application
- OpenStack Orchestration
- Coordinates OpenStack API calls
- Provides Metadata to Resources

Express a multi-node application

- Nodes + Relationships + Personas + Order



# Heat Orchestration

- In service dependency order, heat:
  - Provisions node
  - Provides JSON config metadata
  - Runs config-change trigger
  - Reports success/collected node information

chef/ohai/puppet can read JSON config ...

Write recipes/modules to expect standard input parameters

# Interoperability = Nirvana

## Interop is a *Major Focus* for Foundation

Panel on Tuesday @ 5:20

Room A107-A109

Interoperability is a sustaining process  
it is not an release feature

- shared reference architectures -> refstack
- open operations -> upstreamed DevOps
- comprehensive test -> CI & API tests
- community participation -> you!