



Docker & openstack™

CLOUD SOFTWARE

January 2014



What?

- Docker driver for Nova accepted in Havana
- Docker plugin for Heat accepted for Icehouse
- Docker support in devstack
- **Now spec'ing a project to improve containers support in OpenStack**

Why Docker?

(an incomplete list)

- Cross-Cloud compatibility
- Dockerfiles provide flexible, repeatable image authoring
- Global image portability and delivery through the Registry
- Incremental images & builds
- Completeness of vision
(Runtime, Configuration, Build, Deployment, Orchestration...)



Nova Integration

Docker driver for OpenStack Compute



What?

Enables control of
Docker via OpenStack:

- Nova API
- Horizon UI

Supports:

- launch
- terminate
- reboot
- fetch logs
- snapshot
- glance

<https://wiki.openstack.org/wiki/HypervisorSupportMatrix>



UI Integration (Horizon)

Instances - OpenStack Dashboard

dev/project/instances/

Logged in as: admin

Launch Instance

Details * Access & Security Post-Creation

Availability Zone: nova

Instance Name *: myinstance

Flavor *: m1.nano

Instance Count *: 1

Instance Boot Source *: Boot from image.

Image Name: docker-busybox:latest (2.2 MB)

Specify the details for launching an instance. The chart below shows the resources used by this project in relation to the project's quotas.

Flavor Details	
Name	m1.nano
VCPUs	1
Root Disk	0 GB
Ephemeral Disk	0 GB
Total Disk	0 GB
RAM	64 MB

Project Limits

- Number of Instances: 0 of 10 Used
- Number of VCPUs: 0 of 20 Used
- Total RAM: 0 of 51,200 MB Used

Cancel Launch

UI Integration (Horizon)

Instances - OpenStack Dashboard

dev/project/instances/

Logged in as: admin Settings Help Sign Out

Instances

Filter Filter [Launch Instance](#) [Soft Reboot Instances](#) [Terminate Instances](#)

<input type="checkbox"/>	Instance Name	Image Name	IP Address	Size	Keypair	Status	Task	Power State	Uptime	Actions
<input type="checkbox"/>	myinstance	docker-busybox:latest	10.0.0.2	m1.nano 64MB RAM 1 VCPU 0 Disk	-	Active	None	Running	0 minutes	Create Snapshot More

Displaying 1 item

Project Admin

CURRENT PROJECT admin

Manage Compute

- Overview
- Instances
- Volumes
- Images & Snapshots
- Access & Security



Nova+Docker Architecture Overview

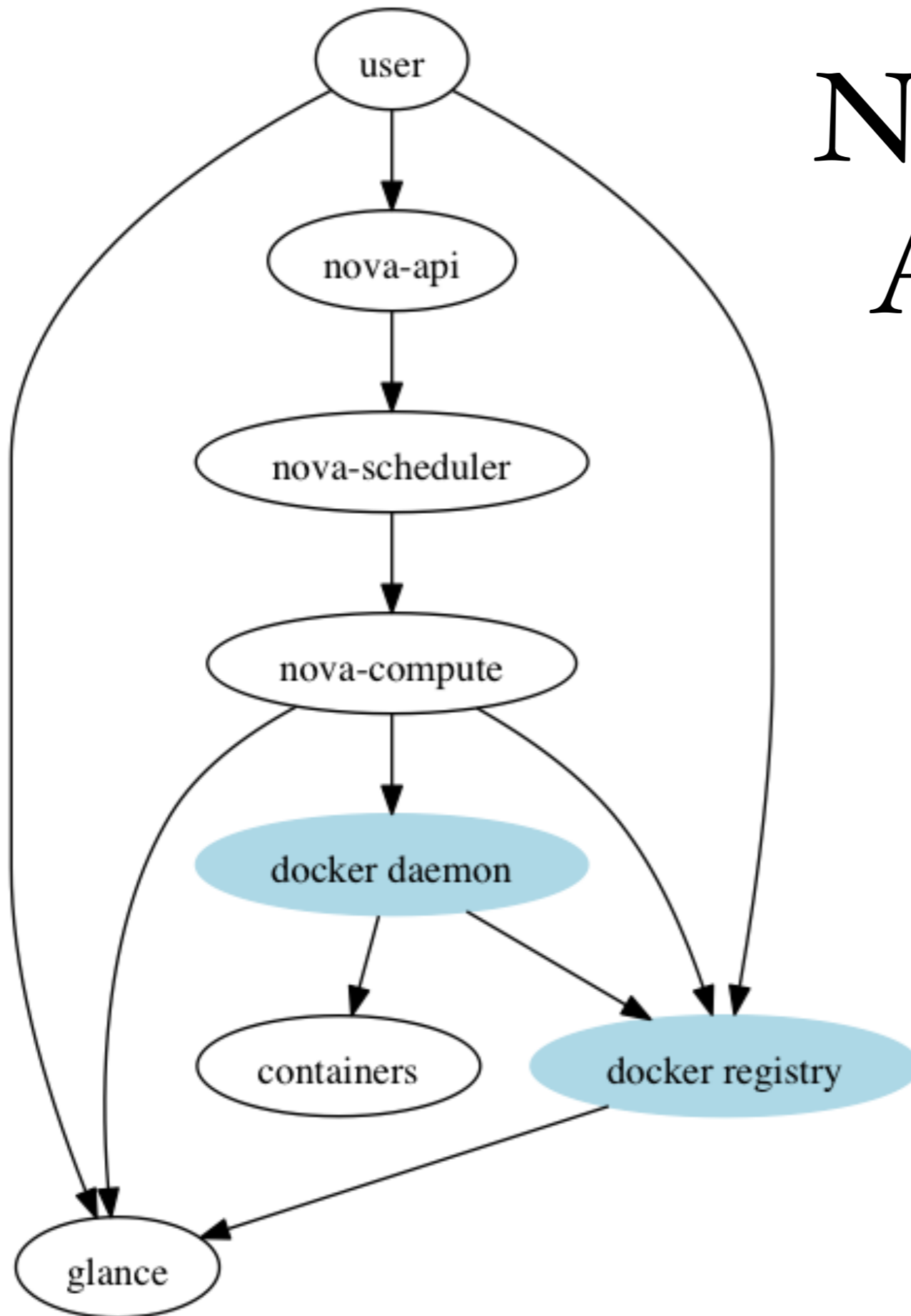
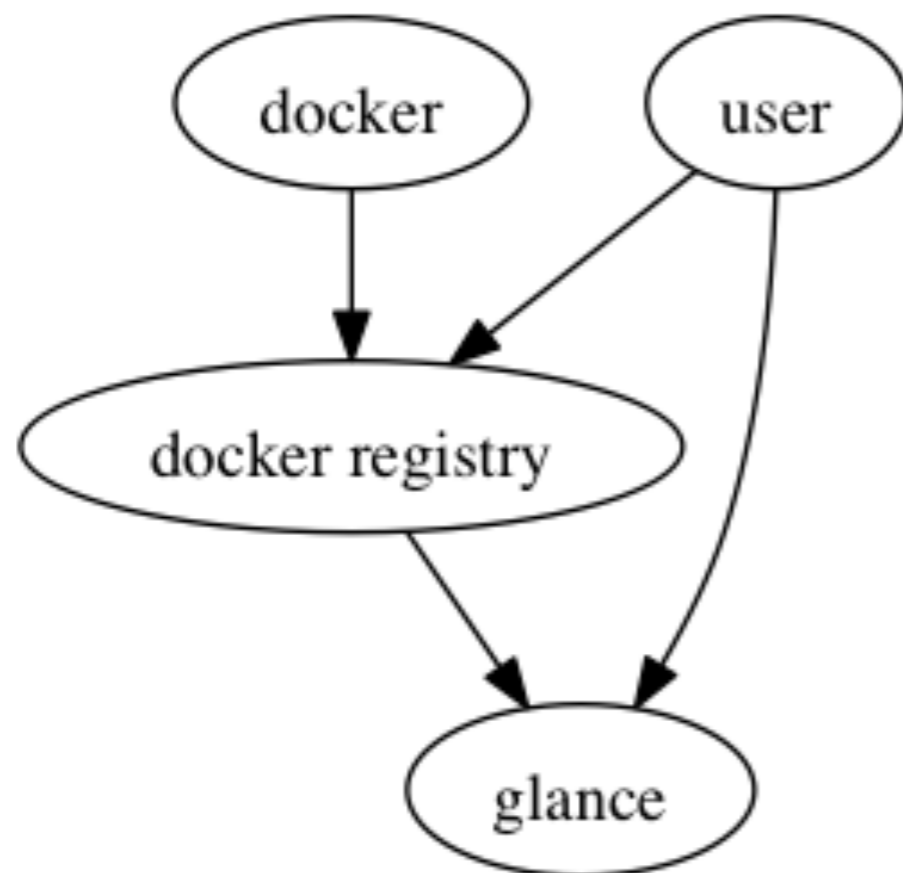


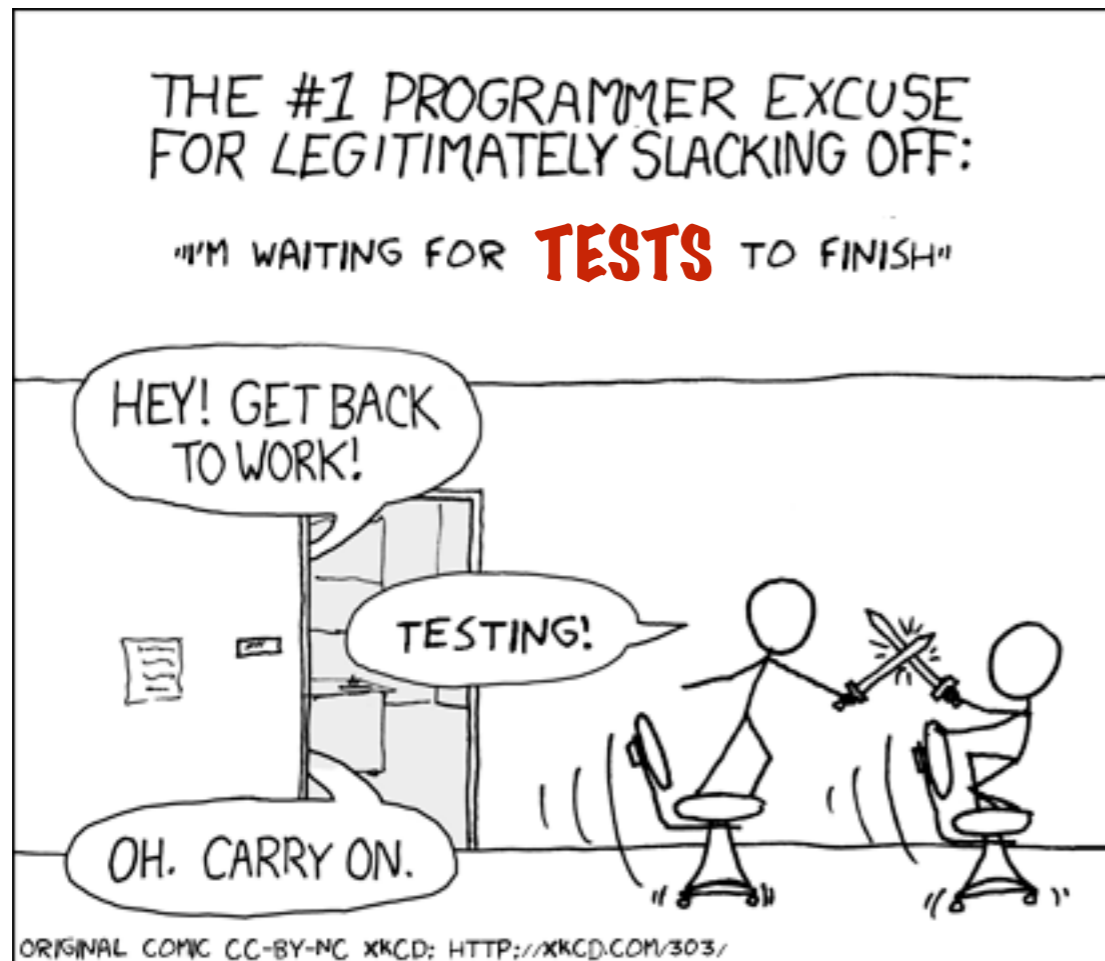
Image Management



docker-registry is a proxy

users can upload through docker-registry or to glance directly

docker pulls images through the docker-registry proxy



Gate Tests

passing >90% of tests in Tempest

- not perfect, but we have visibility
- most failures are !docker

non-voting gate coming in Icehouse

Unit Tests

38 unit tests, gating commits

(vs 541 for libvirt and 10 specific to lxc)



Not supported. (yet)

Neutron (+OVS)

Cinder Volumes

**(Work in progress
but patches welcome!)**



Docker Basics

Example: Run Tempest
(the OpenStack integration test suite)

Based on the project
“Dockenstack” written by Paul Czarkowski
forked and improved by Eric Windisch

Run OpenStack tempest against:

- ... a freshly provisioned OpenStack install
- ... installed by Devstack
- ... in a Docker container



Dockerfile

```
FROM ubuntu:raring
MAINTAINER Eric Windisch "ewindisch@docker.com"
[...]
RUN apt-get -qqy install mysql-server git socat curl ...
RUN useradd devstack && usermod -a -G docker devstack
ADD devstack.sudo /etc/sudoers.d/devstack
RUN chown root /etc/sudoers.d/devstack
ADD tempest /usr/local/bin/start-devstack
RUN chmod 755 /usr/local/bin/start-devstack
VOLUME /var/lib/docker
[...]
RUN git clone https://github.com/openstack-dev/devstack
RUN /devstack/tools/install_prereqs.sh
ADD localrc /devstack/localrc
CMD ["/usr/local/bin/start-devstack"]
```



Build the container image:

```
$ docker build -t dockenstack .
```

Run it:

```
$ docker run -privileged dockenstack
```

Publish it on the public Registry:

```
$ docker tag dockenstack ewindish/dockenstack
```

```
$ docker push ewindisch/dockenstack
```



It's on the public registry!

```
another-host$ docker search dockenstack
```

NAME	DESCRIPTION
ewindisch/dockenstack	OpenStack development environment (using D...

```
another-host$ docker pull ewindisch/dockenstack
```

... and on github!

<https://github.com/ewindisch/dockenstack>



ewindisch/dockenstack Re x

https://index.docker.io/u/ewindisch/dockenstack/

DOCKER index

Pricing [samalba](#) [admin](#) [automated builds](#) [logout](#) [help](#)

repository **ewindisch/dockenstack** Pull this repository: `docker pull ewindisch/dockenst`

Information Tags Settings 0 0

Maintainer
ewindisch

Trusted Build
[Source Repo](#)
[Project Page](#)
[Build Bundle](#)

Downloaded
11

Last updated
2014-01-25 07:51:58

Created
2014-01-22 17:54:30

ewindisch/dockenstack

Short Description
OpenStack development environment (using Docker virt driver); Forked from paulczar's dockenstack

Full Description

Description

Dockenstack provides an ephemeral devstack container with cached, pre-installed requirements (apt and pip packages/modules). By default, it uses and supports the Docker virt driver.

This image is usable for development and local testing, but also happens to be vital to our gate. This means that your testing environment precisely matches what is used in the gate! (Note: in the gate for the Docker driver)

A great *Thank You* to Paul Czar for creating the original dockenstack.

Running the container

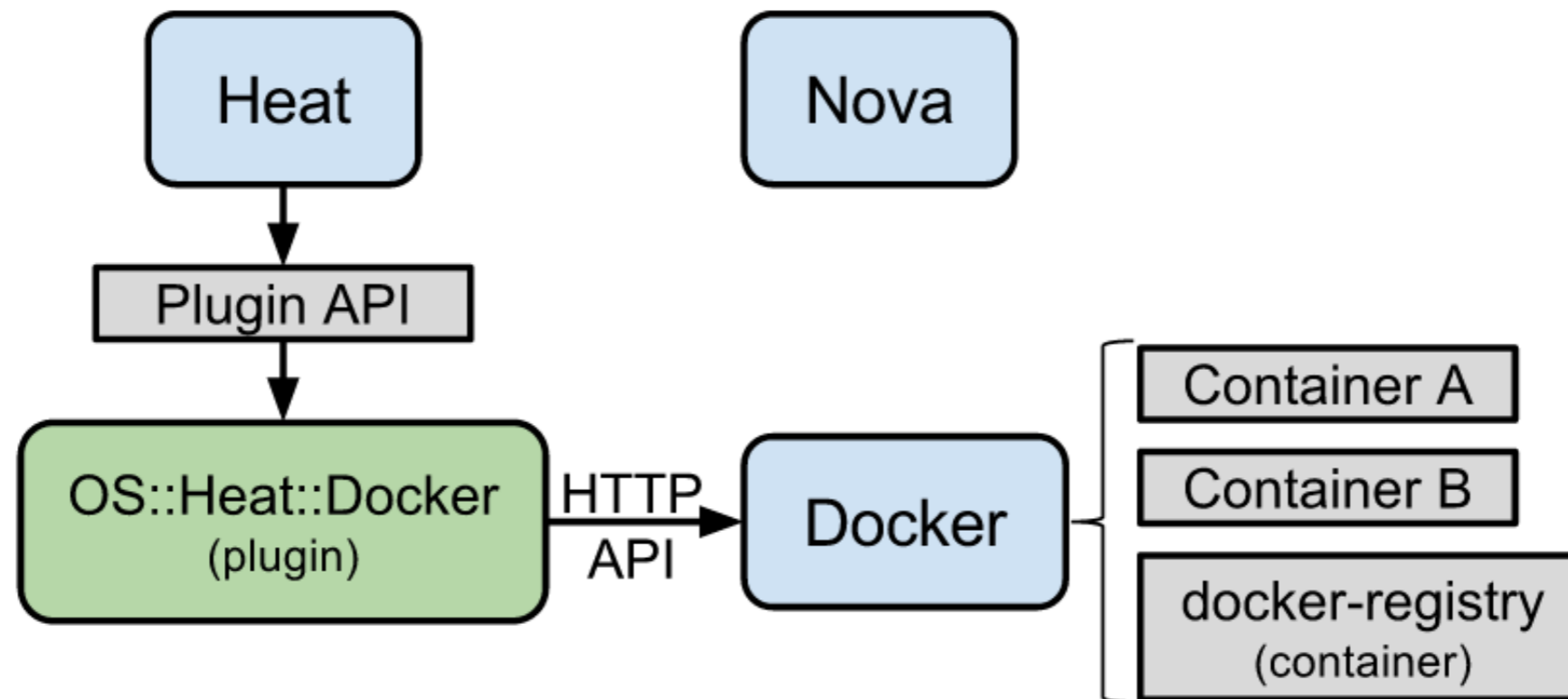
```
docker run -privileged -lxc-conf=aa_profile=unconfined -t -i ewindisch/dockenstack
```


Applying Heat

Orchestration for Docker API using OpenStack Heat



Heat Plugin Flow



Heat Template

```
heat_template_version: 2013-05-23
```

```
description: Single compute instance with one docker container-  
runs devstack /w docker and tests it with Tempest!
```

```
resources:
```

```
  my_instance:
```

```
    type: OS::Nova::Server
```

```
    properties:
```

```
      key_name: ewindisch_key1
```

```
      image: ubuntu-precise-docker
```

```
      flavor: m1.large
```

```
  my_docker_container:
```

```
    type: OS::Heat::Docker
```

```
    docker_endpoint: { get_attr: [my_instance, first_address] }
```

```
    image: ewindisch/dockenstack-tempest
```



Heat vs Nova

Comparison

Nova driver	Heat plugin
<p>Integration with other services</p> <p>Nova features (quota, auth, etc...)</p> <p>Abstraction layer for other hypervisors</p> <p>Integrated scheduling</p>	<p>Closer to the Docker workflow</p> <p>Hybrid-cloud compatible</p> <p>Scheduled by backing cloud</p>

- Both approaches are simply different
- One does not replace the other

Q & A