

External Plugin Interfaces for OpenStack QA Projects

Matthew Treinish
mtreinish@kortar.org
mtreinish on Freenode

Andrea Frittoli
andrea.frittoli@hp.com
andreaf on Freenode

Oct 27, 2015

https://github.com/mtreinish/external_plugins

What is OpenStack QA?

- ▶ Official Mission Statement:
Develop, maintain, and initiate tools and plans to ensure the upstream stability and quality of OpenStack, and its release readiness at any point during the release cycle.

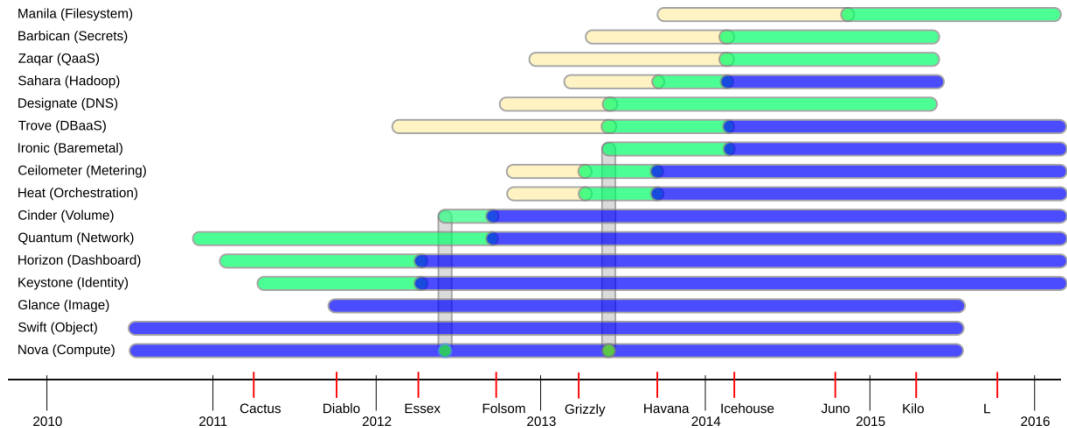
Current QA Projects

- ▶ bashate
- ▶ devstack
- ▶ devstack-plugin-cookiecutter
- ▶ devstack-vagrant
- ▶ grenade
- ▶ tempest
- ▶ tempest-lib
- ▶ tempest-plugin-cookiecutter
- ▶ stackviz
- ▶ hacking
- ▶ eslint-config-openstack
- ▶ os-testr
- ▶ openstack-health dashboard

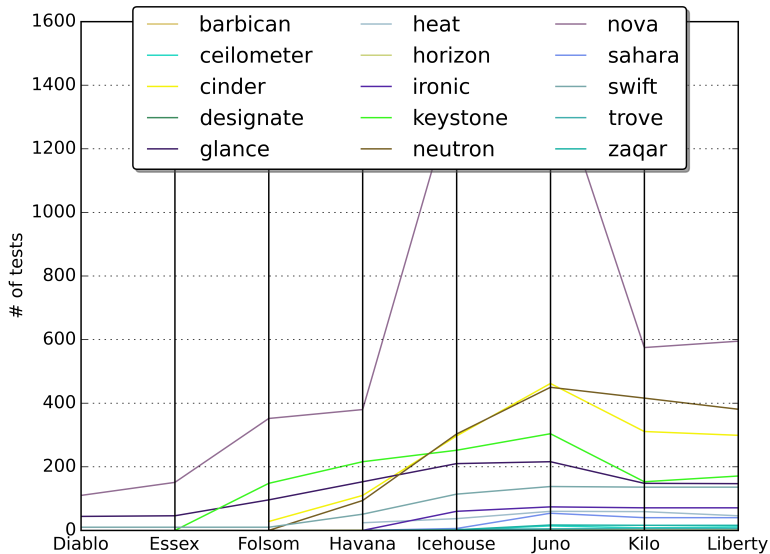
Scaling Limitations

- ▶ Previously QA projects supported all incubated and integrated OpenStack projects
- ▶ This works with 5 projects
- ▶ At 10 things are stretched very thinly

OpenStack Project Growth



Tempest Tests per Project



The Big Tent. . .



QA in the Big Tent

- ▶ QA projects now directly support base
- ▶ Provide stable plugin interfaces to expand functionality
- ▶ Better fits with the growth in projects

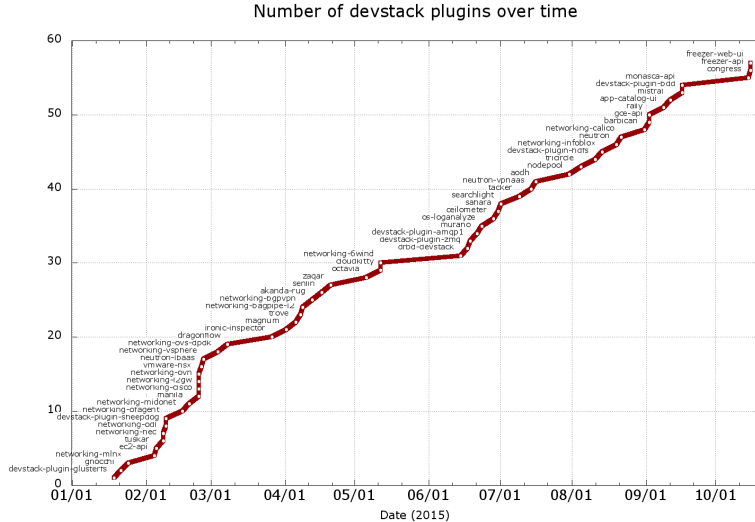
Current Projects QA directly supports in-tree

- ▶ Keystone
- ▶ Nova
- ▶ Glance
- ▶ Cinder
- ▶ Neutron
- ▶ Swift

Devstack Plugins

- ▶ Bits of bash code that live outside of devstack tree
- ▶ Called via a strong contract
- ▶ Maintained by the project owners
- ▶ Tested via OpenStack CI
- ▶ Project plugins
 - ▶ Installs new OpenStack service as part of devstack
- ▶ Driver / configuration plugins
 - ▶ Configures existing OpenStack service to use a different driver / backend
- ▶ Registry of plugins maintained by the community:
<http://docs.openstack.org/developer/devstack/plugin-registry.html>

Projects using Devstack Plugins

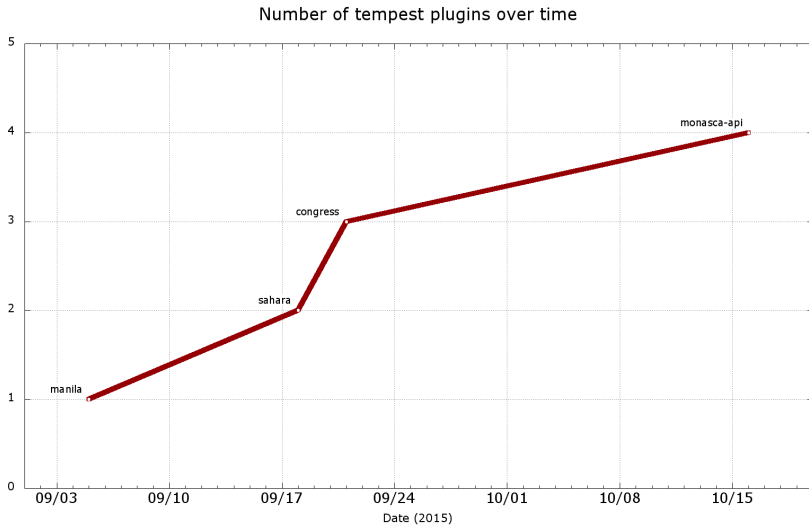


Howto write a Devstack Plugin

Tempest Plugins

- ▶ Integrates any external tests into a tempest run
- ▶ Unifies configuration between plugin(s) and in-tree
- ▶
- ▶ Based on stevedore plugin model
- ▶ Automatically discovered when installed
- ▶ Test code built on top of tempest-lib

Projects using tempest plugins



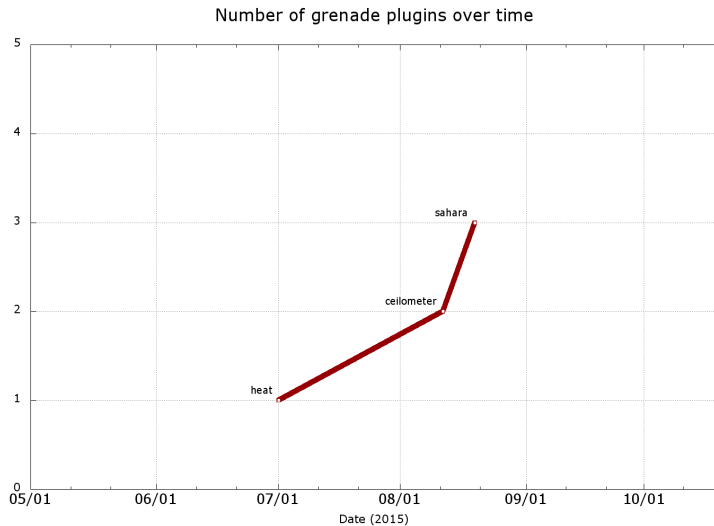
Source: `git blame -L '/^tempest/,+1' setup.cfg | awk '{print $1}' | xargs git log -1 --format=%cd --date=short`

Howto Write a Tempest Plugin

- ▶ Code can be hosted in existing or dedicated git project
- ▶ Extend `plugins.TempestPlugin` from `tempest` (or `tempest-lib`)
- ▶ Implement all abstract methods
 - ▶ Test discovery:
 - ▶ Configuration options:
- ▶ Check if everything you need is tempest lib
- ▶ Setup CI to test with your plugin, and keep it tested

Grenade Plugins

Projects using Grenade Plugins



Source: `git log -diff-filter=A -format=%cd -date=short -1 - devstack/upgrade/upgrade.sh`

Howto Write a Grenade Plugin

Where to get more information

- ▶ Tempest Plugin Docs: <http://docs.openstack.org/developer/tempest/plugin.html>
- ▶ Devstack Plugin Docs:
<http://docs.openstack.org/developer/devstack/plugins.html>
- ▶ Grenade Plugin Docs: <http://docs.openstack.org/developer/grenade/plugins.html>
- ▶ openstack-dev ML: openstack-dev@lists.openstack.org
- ▶ #openstack-qa on Freenode

Questions?