

Tools for Distributed, Open Source Systems Administration

LISA15

Friday, November 13, 2015

Elizabeth K. Joseph
lyz@princessleia.com
@pleia2

Elizabeth K. Joseph

- OpenStack Infrastructure Team systems administrator, paid by Hewlett Packard Enterprise
- Long time contributor to various open source projects
- Co-author of The Official Ubuntu Book, 8th ed

1. Our team
2. The OpenStack CI system
3. Code Review
4. Automated testing
5. Other collaboration tools
6. Communication

How most open source projects do infrastructure

- Team (or company) manages it ...or they just use code hosting
- Requests are submitted via mailing list, bug report or ticketing system
- Request priority is determined by the core team

This may be similar to your organization.

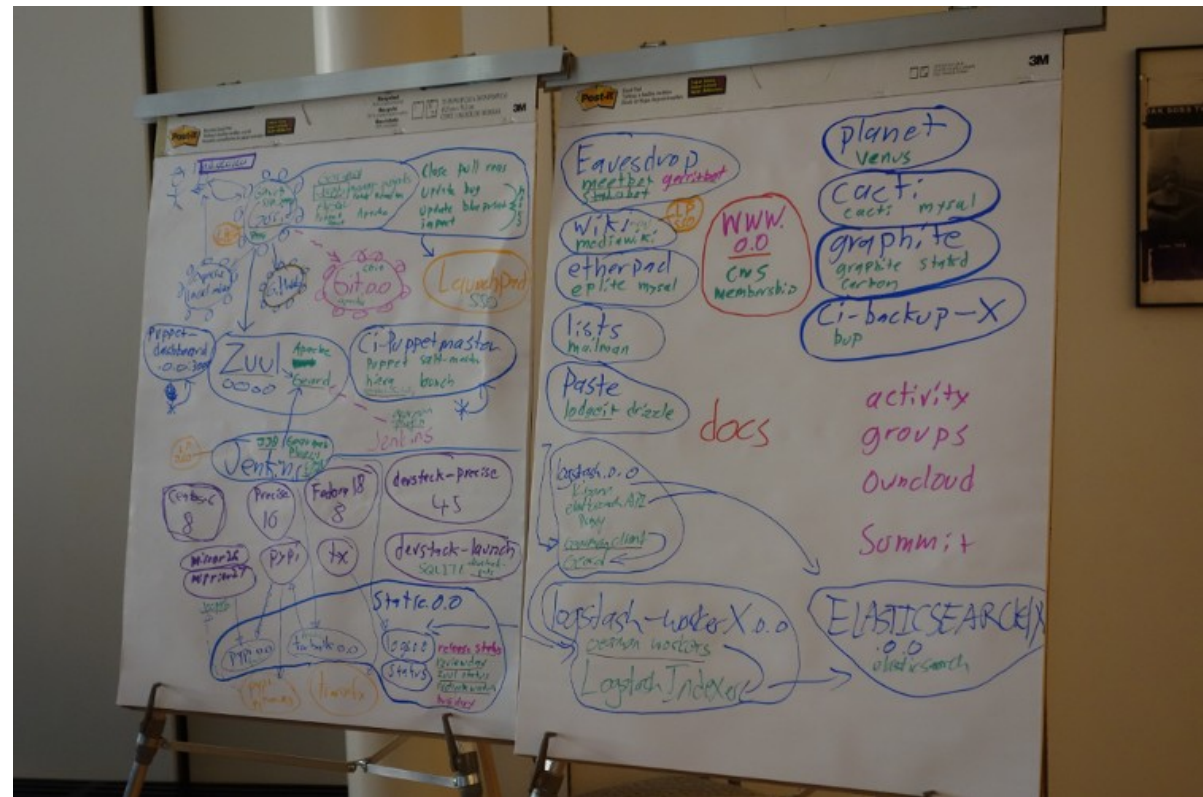
Is there a better way?

OpenStack Infrastructure Team

- Our job is to make sure the OpenStack developers can do *their* job
- All of our system configurations are open source and tracked in git: <https://git.openstack.org/cgit/openstack-infra>
- Anyone in the world can propose patches for direct inclusion in our infrastructure, instructions at: <http://docs.openstack.org/infra/manual/developers.html>
- We all work remotely. Worldwide: US, Russia, Australia, Spain.

What we run

- Askbot
- Continuous Integration systems
- Cacti
- Elasticsearch, Logstash and Kibana
- IRC Bots
- Etherpad
- Git
- Paste
- Planet
- Puppetboard
- Mailing Lists
- Translations platform
- Various web services
- Wiki



OpenStack Continuous Integration (CI) System

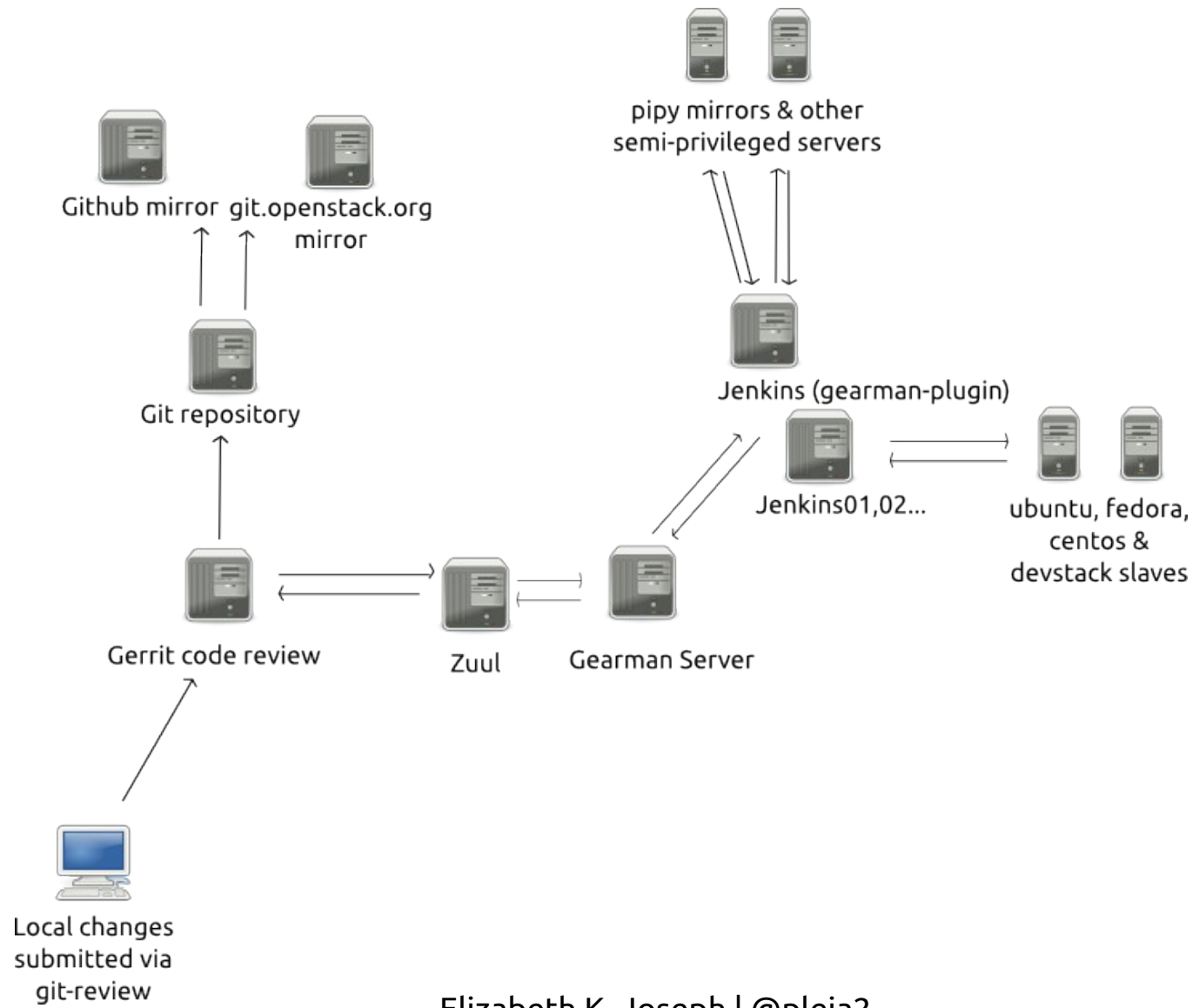
- Lots of individual projects (800+)
- All projects must work together
- Changes can't break master branch
- Code should be syntactically clean
- Testing must be completed automated

Tools we're using for CI

- Launchpad (someday: openstackid)
- Git
- Gerrit
- Zuul*
- Gearman
- Jenkins (with jenkins-job-builder*, devstack-gate*)
- Nodepool*

* Started by the OpenStack Infrastructure team

Workflow



Why do I care?

The Infrastructure team uses it too!

Automated tests for infrastructure

- flake8 (pep 8 and pyflakes)
- puppet parser validate
- puppet lint
- Puppet application tests
- XML checkers
- Alphabetized files
- IRC channel permissions

Peer review means...

- Multiple eyes on changes prior to merging
- Good infrastructure for developing new solutions
- No special process to go through for commit access
- Trains us to be collaborative by default
- Since anyone can contribute, anyone can devote resources to it

Gerrit in-line comments

19	}	}	20
20			21
... skipped 25 common lines ...			
46	content => template('lodgeit/manage.py.erb'),	content => template('lodgeit/manage.py.erb'),	47
47	notify => Service["\${name}-paste"],	notify => Service["\${name}-paste"],	48
48	}	}	49
49			50
50	file { "/srv/lodgeit/\${name}/lodgeit/views/layout.html":	file { "/srv/lodgeit/\${name}/lodgeit/views/layout.html":	51
51	ensure => present,	ensure => present,	52
52	replace => true,	replace => true,	53
53	content => template('lodgeit/layout.html.erb'),	content => template('lodgeit/layout.html.erb'),	54
54	}	}	55
55			56
		if \$robotstxt != undef {	57
		Spencer Krum Sep 21 4:32 PM	
		Probably just want if \$robotstxt here.	
		<input type="button" value="Reply ..."/> <input type="button" value="Reply 'Done'"/>	
		file { "/srv/lodgeit/\${name}/robots.txt":	58
		ensure => present,	59
		owner => 'root',	60
		group => 'root',	61
		mode => '0444',	62
		source => 'puppet:///modules/lodgeit/robots.txt',	63
		require => File["/srv/lodgeit/\${name}/"],	64
		}	65
		}	66
56	cron { "update_backup_\${name}":	cron { "update_backup_\${name}":	67
57	ensure => absent,	ensure => absent,	68
58	user => root,	user => root,	69
59	}	}	70
60			71

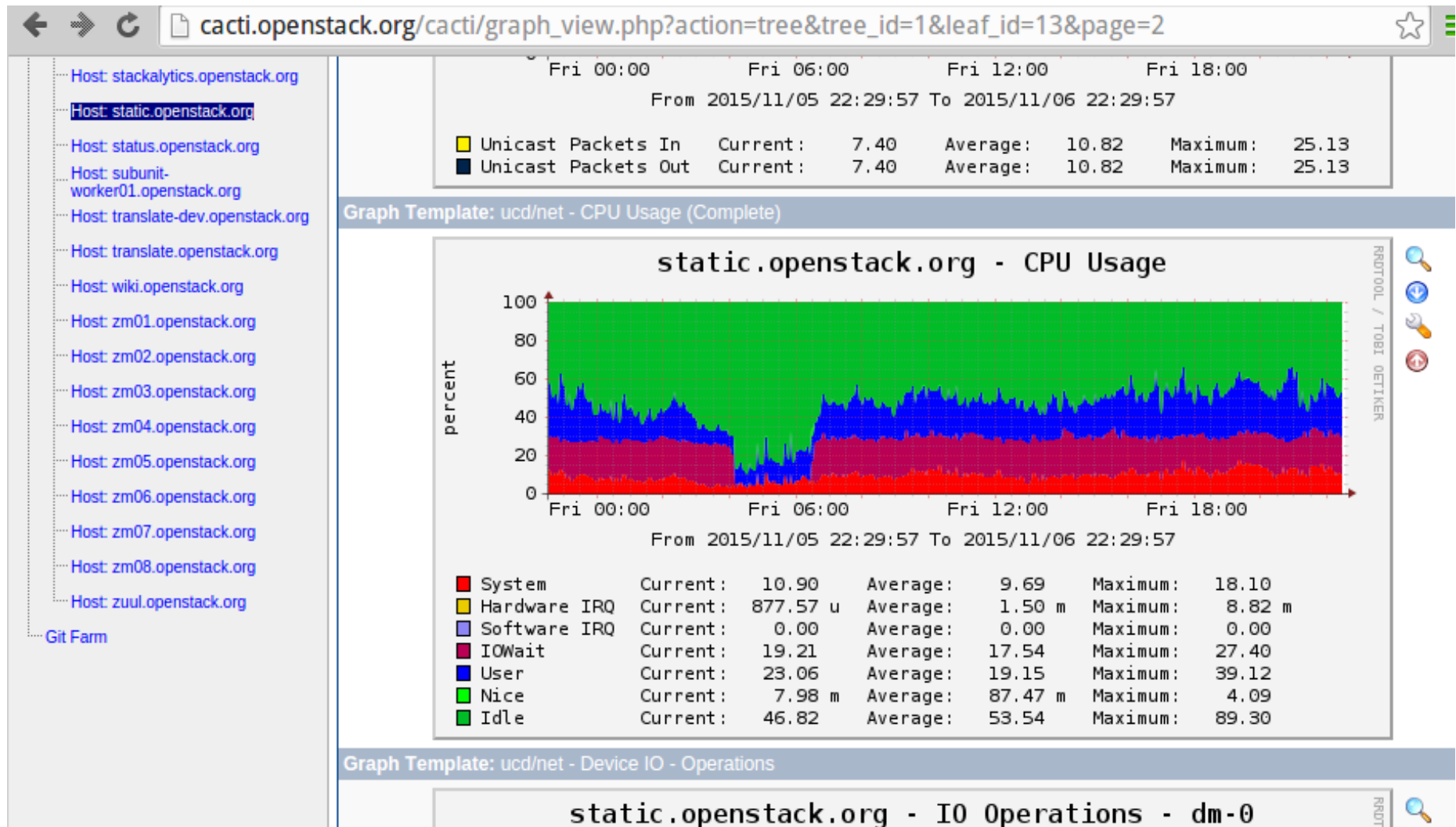
Automated deployment

- Change gets approved, tested and merged
- ...Either puppet master gets updated and applies change
- ...Or vcsrepo module in puppet pulls in latest version of project

Can you really manage an infrastructure via git commits?

Cacti (<http://cacti.openstack.org/>) to keep an eye on server usage

Cacti



Can you really manage an infrastructure via git commits? (2/3)

PuppetBoard (<http://puppetboard.openstack.org/>) so you can watch your changes get applied, or not

PuppetBoard

The screenshot shows the PuppetBoard interface with a navigation bar and a list of nodes. The browser address bar shows `puppetdb.openstack.org/nodes`. The navigation bar includes links for PUPPETBOARD, OVERVIEW, NODES, FACTS, REPORTS, METRICS, and QUERY. The main content area displays a table of nodes with columns for status, node name, last report time, and a 'Reports' button. The nodes listed are:

Status	Node Name	Last Report Time	Reports
changed	afbdb02.openstack.org	Friday, November 6 2015 2:32 PM	Latest Report Reports
unchanged	afstest.openstack.org	Friday, November 6 2015 2:32 PM	Latest Report Reports
unchanged	apps.openstack.org	Friday, November 6 2015 2:32 PM	Latest Report Reports
changed	ask-staging.openstack.org	Friday, November 6 2015 2:32 PM	Latest Report Reports
unchanged	ask.openstack.org	Friday, November 6 2015 2:32 PM	Latest Report Reports
changed	cacti.openstack.org	Saturday, July 5 2014 12:20 PM	Latest Report Reports

Copyright © 2013 Daniele Sluijters. Live from PuppetDB.

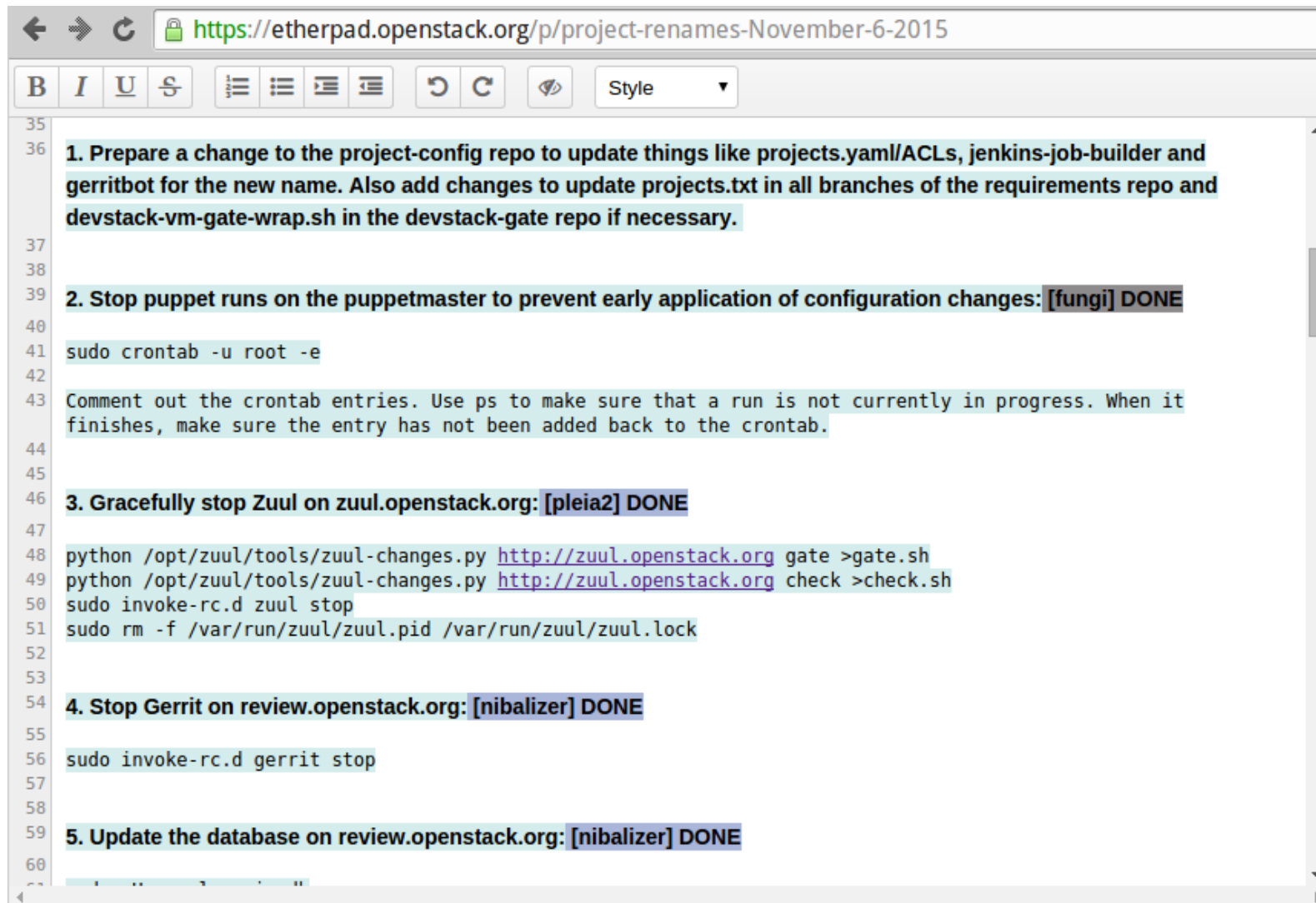
Can you really manage an infrastructure via git commits? (3/3)

Thorough, specific documentation at <http://docs.openstack.org/infra/system-config>

Well, not everything

- Automation is not perfect, sometimes you just need to log into a server
- Complicated migrations and upgrades need manual components (but we automate more every time!)
- Initial persistent server deployment still has manual components
- Passwords need to be privately managed (but we use git!)

Maintenance collaboration on Etherpad



The screenshot shows a web browser window with the URL <https://etherpad.openstack.org/p/project-renames-November-6-2015>. The document content is as follows:

```
35
36 1. Prepare a change to the project-config repo to update things like projects.yaml/ACLs, jenkins-job-builder and gerritbot for the new name. Also add changes to update projects.txt in all branches of the requirements repo and devstack-vm-gate-wrap.sh in the devstack-gate repo if necessary.
37
38
39 2. Stop puppet runs on the puppetmaster to prevent early application of configuration changes: [fungi] DONE
40
41 sudo crontab -u root -e
42
43 Comment out the crontab entries. Use ps to make sure that a run is not currently in progress. When it finishes, make sure the entry has not been added back to the crontab.
44
45
46 3. Gracefully stop Zuul on zuul.openstack.org: [pleia2] DONE
47
48 python /opt/zuul/tools/zuul-changes.py http://zuul.openstack.org gate >gate.sh
49 python /opt/zuul/tools/zuul-changes.py http://zuul.openstack.org check >check.sh
50 sudo invoke-rc.d zuul stop
51 sudo rm -f /var/run/zuul/zuul.pid /var/run/zuul/zuul.lock
52
53
54 4. Stop Gerrit on review.openstack.org: [nibalizer] DONE
55
56 sudo invoke-rc.d gerrit stop
57
58
59 5. Update the database on review.openstack.org: [nibalizer] DONE
60
```

Human collaboration

- Main IRC channel (#openstack-infra)
- Incident IRC channel (#openstack-infra-incident)
- Sprint IRC channel (#openstack-sprint)
- Weekly IRC-based meetings (#openstack-meeting)
- Channel/Meeting logs on <http://eavesdrop.openstack.org/>
- Pastebin <http://paste.openstack.org/>
- In person collaboration at the OpenStack Design Summit every 6 months

No voice or video calls.

And then there are time zones :(

- Increased coverage is great, but...
 - The first core/root member in a particular region struggles to feel cohesion with the team
 - Increased reluctance to land changes into production
 - Makes for slower on-boarding for key tasks
- Only solved by increasing coverage in that time zone so they're not alone

But mostly it's pretty great!

(and HPE is hiring)

Questions

My *real* email: lyz@princessleia.com

My HPE email: lyz@hpe.com

Freenode: pleia2

OpenStack CI Resources:
<http://docs.openstack.org/infra/system-config/>