Configuration Management Evolution at CERN

Gavin McCance
gavin.mccance@cern.ch
@gmccance
Agile Infrastructure

- Why we changed the stack
- Current status
- Technology challenges
- People challenges
- Community
Why?

- Homebrew stack of tools
  - Twice the number machines, no new staff
  - New remote data-centre
  - Adopting more dynamic Cloud model

- “We’re not special”
  - Existence of open source tool chain: OpenStack, Puppet, Foreman, Kibana

- Staff turnover
  - Use standard tools – we can hire for it and people can be hired for it when they leave
Agile Infrastructure “stack”

- Our current stack has been stable for one year now
  - See plenary talk at last CHEP (Tim Bell et al)
- Virtual server provisioning
  - Cloud “operating system”: OpenStack -> (Belmiro, next)
- Configuration management
  - Puppet + ecosystem as configuration management system
  - Foreman as machine inventory tool and dashboard
- Monitoring improvements
  - Flume + Elasticsearch + Kibana -> (Pedro, next++)
Puppet

• Puppet manages nodes’ configuration via “manifests” written in Puppet DSL

• All nodes check in frequently (~1-2 hours) and ask for configuration
  • Configuration applied frequently to minimise drift

• Using the central puppet master model
  • ..rather than masterless model
  • No shipping of code, central caching and ACLs
Separation of data and code

- Puppet “Hiera” splits configuration “data” from “code”
  - Treat Puppet manifests really as code
  - More reusable manifests
    - Heira is quite new: old manifests are catching up

- Hiera can use multiple sources for lookup
  - Currently we store the data in git
  - Investigating DB for “canned” operations
Modules and Git

• Manifests (code) and hiera (data) are version controlled

• Puppet can use git’s easy branching to support parallel environments
  • Later…
Foreman

• Lifecycle management tool for VMs and physical servers

• External Node Classifier – tells the puppet master what a node should look like

• Receives reports from Puppet runs and provides dashboard
## Hosts

<table>
<thead>
<tr>
<th>Name</th>
<th>Operating system</th>
<th>Environment</th>
<th>Model</th>
<th>Host group</th>
<th>Last report</th>
<th>Owner</th>
</tr>
</thead>
<tbody>
<tr>
<td>lxplus0353.cern.ch</td>
<td>SLC 6.4</td>
<td>production</td>
<td>OpenStack Nova</td>
<td>bi/inter/plus/live/login</td>
<td>about 2 hours ago</td>
<td>it-dep-pes-ps-support</td>
</tr>
<tr>
<td>lxplus0375.cern.ch</td>
<td>SLC 6.4</td>
<td>production</td>
<td>OpenStack Nova</td>
<td>bi/inter/plus/live/login</td>
<td>42 minutes ago</td>
<td>it-dep-pes-ps-support</td>
</tr>
<tr>
<td>lxplus0407.cern.ch</td>
<td>SLC 6.4</td>
<td>production</td>
<td>OpenStack Nova</td>
<td>bi/inter/plus/live/acron</td>
<td>about 1 hour ago</td>
<td>it-dep-pes-ps-support</td>
</tr>
<tr>
<td>lxplus0409.cern.ch</td>
<td>SLC 6.4</td>
<td>production</td>
<td>OpenStack Nova</td>
<td>bi/inter/plus/live/login</td>
<td>about 1 hour ago</td>
<td>it-dep-pes-ps-support</td>
</tr>
<tr>
<td>lxplus0410.cern.ch</td>
<td>SLC 6.4</td>
<td>production</td>
<td>OpenStack Nova</td>
<td>bi/inter/plus/live/login</td>
<td>about 2 hours ago</td>
<td>it-dep-pes-ps-support</td>
</tr>
<tr>
<td>lxplus0411.cern.ch</td>
<td>SLC 6.4</td>
<td>production</td>
<td>OpenStack Nova</td>
<td>bi/inter/plus/live/login</td>
<td>about 1 hour ago</td>
<td>it-dep-pes-ps-support</td>
</tr>
<tr>
<td>lxplus0412.cern.ch</td>
<td>SLC 6.4</td>
<td>production</td>
<td>OpenStack Nova</td>
<td>bi/inter/plus/live/login</td>
<td>about 2 hours ago</td>
<td>it-dep-pes-ps-support</td>
</tr>
<tr>
<td>lxplus0416.cern.ch</td>
<td>SLC 6.4</td>
<td>production</td>
<td>OpenStack Nova</td>
<td>bi/inter/plus/live/login</td>
<td>about 2 hours ago</td>
<td>it-dep-pes-ps-support</td>
</tr>
<tr>
<td>lxplus0417.cern.ch</td>
<td>SLC 6.4</td>
<td>production</td>
<td>OpenStack Nova</td>
<td>bi/inter/plus/live/login</td>
<td>about 1 hour ago</td>
<td>it-dep-pes-ps-support</td>
</tr>
<tr>
<td>lxplus0418.cern.ch</td>
<td>SLC 6.4</td>
<td>production</td>
<td>OpenStack Nova</td>
<td>bi/inter/plus/live/login</td>
<td>about 2 hours ago</td>
<td>it-dep-pes-ps-support</td>
</tr>
<tr>
<td>lxplus0419.cern.ch</td>
<td>SLC 6.4</td>
<td>production</td>
<td>OpenStack Nova</td>
<td>bi/inter/plus/live/login</td>
<td>about 2 hours ago</td>
<td>it-dep-pes-ps-support</td>
</tr>
<tr>
<td>lxplus0420.cern.ch</td>
<td>SLC 6.4</td>
<td>production</td>
<td>OpenStack Nova</td>
<td>bi/inter/plus/live/login</td>
<td>about 1 hour ago</td>
<td>it-dep-pes-ps-support</td>
</tr>
</tbody>
</table>
p01001532021656.cern.ch

Reports from the last 7 days - 163 reports found

Details
Audits Facts Reports YAML

Properties
Remote control | Console
Monitoring | Lemon
Console Username | noci
Console Password | noci
Domain | cern.ch
IP Address | 128.142.35.220
MAC Address | 00:00:00:00:00:00
Puppet Environment | ceph_beesty
Host Architecture | x86_64

Runtime
last 7 days

Resources
last 7 days
Deployment at CERN

- Puppet 3.2
- Foreman 1.2

- Been in real production for 6 months
- Over 4000 hosts currently managed by Puppet
  - SLC5, SLC6, Windows
  - ~100 distinct hostgroups in CERN IT + Expts
  - New EMI Grid service instances puppetised
  - Batch/Lxplus service moving as fast as we can drain it
  - Data services migrating with new capacity
  - AI services (Openstack, Puppet, etc)

14/10/2013
CHEP 2013
Key technical challenges

• Service stability and scaling

• Service monitoring

• Foreman improvements

• Site integration
Scalability experiences

- Most stability issues we had were down to scaling issues
- Puppet masters are easy to load-balance
  - We use standard apache mod_proxy_balancer
  - We currently have 16 masters
  - Fairly high IO and CPU requirements
- Split up services
  - Puppet – critical vs. non critical
SERVING LARGE FILES OVER PUPPET

THAT'S A PADDLIN'
Scalability experiences

• Foreman is easy to load-balance
• Also split into different services
  • That way Puppet and Foreman UI don’t get affected by e.g. massive installation bursts
PuppetDB

- All puppet data sent to PuppetDB
  - Querying at compile time for Puppet manifests
    - e.g. configure load-balancer for all workers

- Scaling is still a challenge
  - Single instance – manual failover for now
  - Postgres scaling
    - Heavily IO bound (we moved to SSDs)
    - Get the book
Monitor response times

- Monitor response, errors and identify bottlenecks

- Currently using Splunk – will likely migrate to Elasticsearch and Kibana
Upstream improvements

- CERN strategy is to run the main-line upstream code
- Any developments we do gets pushed upstream
- e.g. Foreman power operations, CVE reported
Site integration

- Using Opensource doesn’t get *completely* get you away from coding your own stuff

- We’ve found every time Puppet touches our existing site infrastructure a new “service” or “plugin” is born
  - Implementing our CA audit policy
  - Integrating with our existing PXE setup and burn-in/hardware allocation process - possible convergence on tools in the future – Razor?
  - Implementing Lemon monitoring “masking” use-cases – nothing upstream, yet..
People challenges

• Debugging tools and docu needed!
  • PuppetDB helpful here

• Can we have X’, Y’ and Z’?
  • Just because the old thing did it like that, doesn’t mean it was the only way to do it
  • Real requirements are interesting to others too
  • Re-understanding requirements and documentation and training

• Great tools – how do 150 people use them without stepping on each other?
  • Workflow and process
Use git branches to define isolated puppet environments
Easy git cherry pick
eth2 and eth3 used for bond
added latest tomcat packages mwmgr project
added latest tomcat packages mwmgr project
added latest tomcat packages mwmgr project
AI-2468 set flume user soft/hard limits
added latest tomcat packages mwmgr project
Merge branch 'devel' of ssh://gitgw.cern.ch:12345/...
AI-2431 fix hg_dashboard
BI-1163 - Enable lxplus-large, xlarge and acron
BI-1163 - Add lxplus-large, xlarge and acron
AI-2431 fix on the hg_belle hostgroup
Merge branch 'devel' of ssh://gitgw.cern.ch:12345/...
AI-2431 changes from the vocs_devel branch
AM-108 lemon metrics new flume format
AI-1680 - xrootd now handled by c26s.
New flume gw metrics
AM-121 new flume-extra jar file
Stopping Honeypot System
Merge branch 'baporici' into devel
added perl packages needed for mwmgr project
Merge branch 'devel' of ssh://gitgw.cern.ch:12345/...
add gw log file monitoring
AM-108 json serialization of flume lemon metrics
AI-1680 - Use the c26s module on plus and beta
Merge branch 'devel' into baporici
Add DNS alias foremanlb in front of the balancer
Add foremanlb to the LB server
Remove cron job
Merge branch 'devel' of ssh://gitgw.cern.ch:12345/...
Grizzly : Updated rabbitmq module
New FQDN yaml files for RAC50
AM-121 lemon flume agent get a custom log4j conf
AM-121 allow to specify custom log4j and env vars
File renamed
AM-106 changed itmon rsyslog module to pick up
AM-106 created module hiera file
AM-106 changed in /etc/_hosts to add a new entry
Git model and flexible environments

• For simplicity we made it more complex
  • Each Puppet module / hostgroup now has its own git repo (~200 in all)
    • Simple git-merge process within module
    • Delegated ACLs to enhance security
  • Standard “QA” and “production” branches that machines can subscribe to
    • Flexible tool (Jens, to be open-sourced by CERN) for defining “feature” developments
    • Everything from “production” except for the change I’m testing on my module
Strong QA process

- Mandatory QA process for “shared” modules
  - Recommended for non-shared modules
  - Everyone is expected to have some nodes from their service in the QA environment
  - Normally changes are QA’d for at least 1 week. Hit the button if it breaks your box!

- Still iterating on the process
  - Not bound by technology
  - Is one week enough? Can people “freeze”?
Community collaboration

- Traditionally one of HEPs strong points

- There’s a large existing Puppet community with a good model - we can join it and open-source our modules

- New HEPiX working group being formed now
  - Engage with existing Puppet community
  - Advice on best practices
  - Common modules for HEP/Grid-specific software
  - [https://twiki.cern.ch/twiki/bin/view/HEPIX/ConfigManagement](https://twiki.cern.ch/twiki/bin/view/HEPIX/ConfigManagement)
  - [https://lists.desy.de/sympa/info/hepix-config-wg](https://lists.desy.de/sympa/info/hepix-config-wg)

14/10/2013  CHEP 2013  27
http://github.com/cernops
for the modules we share
Pull requests welcome!
Summary

• The Puppet / Foreman / Git / Openstack model is working well for us
  • 4000 hosts in production, migration ongoing
• Key technical challenges are scaling and integration which are under control
• Main challenge now is people and process
  • How to maximise the utility of the tools
• The HEP and Puppet communities are both strong and we can benefit if we join them together

https://twiki.cern.ch/twiki/bin/view/HEPIX/ConfigManagement
http://github.com/cernops
Backup slides