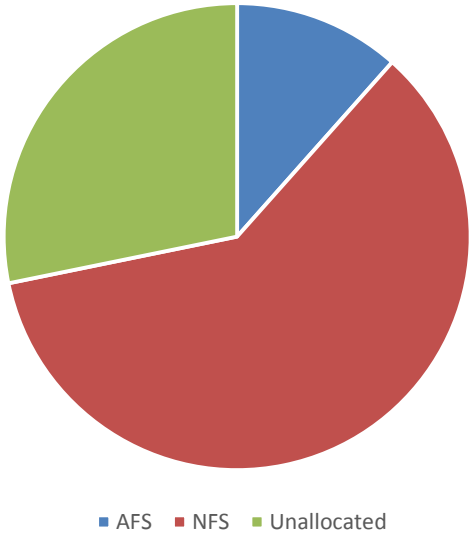
PPE

- Local group:
  - c. 50 servers (50% virtualised)
  - 70 – 80 users (laptops and desktops)
  - Small batch system (10 nodes, 208 cores, ancient TORQUE)
  - 576 TB storage



PPE

- CFEngine 3:
  - 5,500 lines of configuration
  - 3,000 managed files
  - Makes my head hurt

Ansible

- Ansible: A Brief Overview

  Robin Long, University of Lancaster
  HEPSYSMAN (June 2016)

  http://indico.cern.ch/event/518392/contributions/2189473/attachments/1296714/1933786/presentation.pdf

Inventory: Hosts File

```
[bespoke]
node034

[compute]
node[001:004]

[desktop]
ppepc3
ppepc141

[inventory]
ook

[ldap-master]
lipwig

[monitoring]
mort

[provisioning]
vetinari

[storage]
ponder
rincewind

[vm_hosts]
carrot
#vimes

[afs_clients:children]
compute
desktop
```

Inventory: Hosts File

```
[bespoke]
node034

[compute]
node[001:004]

[desktop]
ppepc3
ppepc141

[inventory]
ook

[ldap-master]
lipwig

[monitoring]
mort
```

```
[provisioning]
vetinari

[storage]
ponder
rincewind

[vm_hosts]
carrot
#vimes

[afs_clients:children]
compute
desktop
```

Ad Hoc Commands

```
# ansible compute -m shell -a 'hostname'
node001 | SUCCESS | rc=0 >>
node001.ppe.gla.ac.uk

node002 | SUCCESS | rc=0 >>
node002.ppe.gla.ac.uk

node003 | SUCCESS | rc=0 >>
node003.ppe.gla.ac.uk

node004 | SUCCESS | rc=0 >>
node004.ppe.gla.ac.uk
```

Variables

vars/ldap.yml (extract)

```
ldap_master_hostname:         lipwig.ppe.gla.ac.uk
ldap_hostnames:
  - "{{ ldap_master_hostname }}"
  - adora.ppe.gla.ac.uk
  - dibbler.ppe.gla.ac.uk
  - groat.ppe.gla.ac.uk
```

group_vars/compute

```
virtual_machines:
  - vm_hostname:              ook
    vm_kvm_host:              carrot
    vm_domain:                ppe.gla.ac.uk
    vm_no_cpus:               2
    vm_memory_mb:             2048
    vm_mac_address:           52:54:00:c2:20:00
    vm_disk_size:             50G
    vm_os_family:             rhel7
    vm_vlan:                  210
```

Variables

host_vars/node034

```
packages_compute_devel:      true
packages_desktop:            true

host_ssh_users:
  - abuckley
  - tfletcher
```

group_vars/compute

```
automount_data:              true
automount_home:              true
automount_nfs_admin:         true
automount_nfs_research:      true
```

Secure Variables

- Encrypt variables, tasks, handlers, etc. using Ansible vault:

```
# ansible-vault create vars/secure.yml
# ansible-vault edit vars/secure.yml

# ansible-playbook --vault-password-file /etc/ansible/pw
/etc/ansible/monitoring.yml
```

Roles

```
# ls -1 roles/
afs
ansible-prereq
auks
authn-client
authn-keytab
authn-util
autofs
common
cvmfs
firewall
glasgow-dir
letsencrypt
munge
nagios-config
nagios-nrpe
packages
packages-compute
repo-cern
repo-egi-trustanchors
repo-epel
repo-glasgow
repo-wlcg
rsyslog
snipe-it
snipe-it-upgrade
ssh-authorized-keys
ssh-config
sudo-config
torque-compute
vm-creation
yum-cron
```

Playbooks

```
- hosts: compute
  roles:
    - common
    - ansible-prereq
    - yum-cron
    - authn-client
    - authn-keytab
    - autofs
    - ssh-authorized-keys
    - ssh-config
    - sudo-config
    - cvmfs
    - afs
    - munge
    - packages
    - torque-compute
    - firewall
    - nagios-nrpe
  vars_files:
    - vars/afs.yml
    - vars/infrastructure.yml
    - vars/kerberos.yml
    - vars/ldap.yml
    - vars/nagios.yml
    - vars/path.yml
    - vars/torque.yml
    - vars/users.yml
```

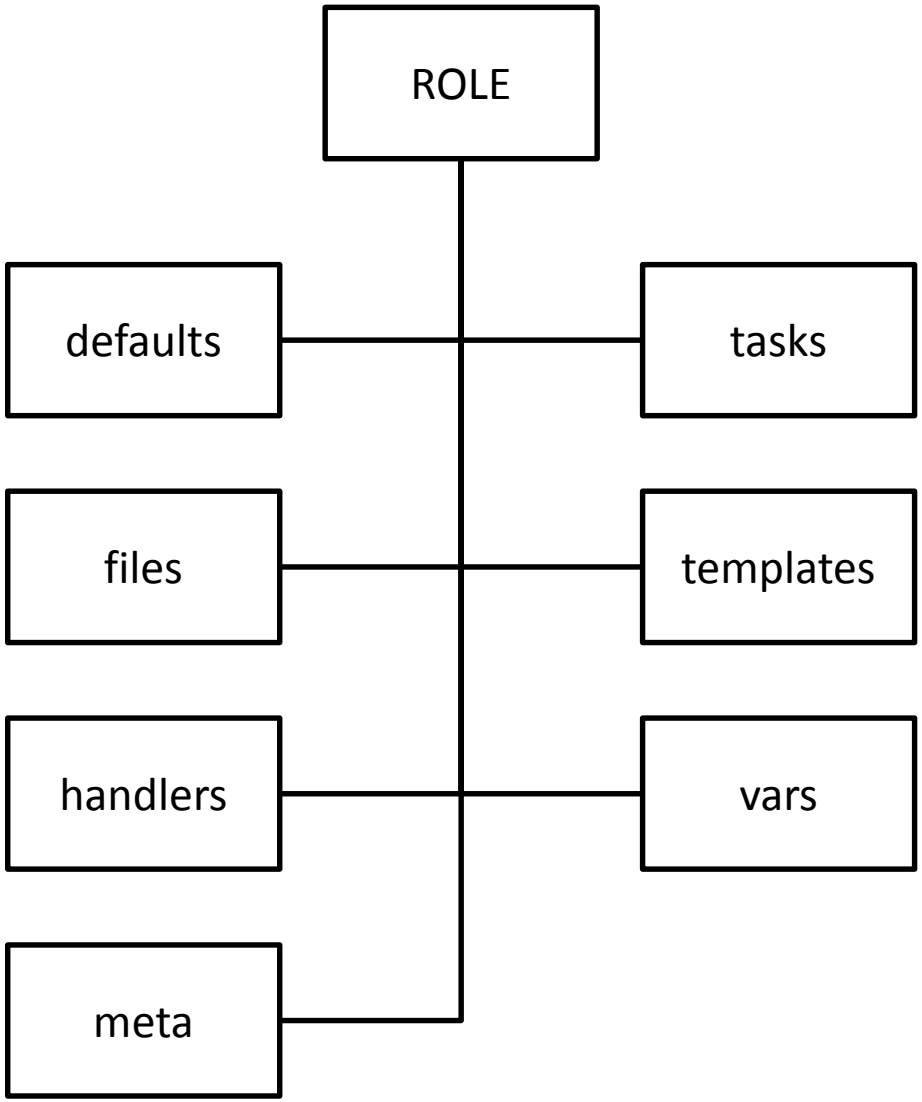
Playbooks

```
# ansible-playbook --vault-password-file /etc/ansible/pw
/etc/ansible/compute.yml

# ansible-playbook --vault-password-file /etc/ansible/pw
/etc/ansible/site.yml --tags mount

# ansible-playbook --vault-password-file /etc/ansible/pw
/etc/ansible/desktop.yml -l ppepc3
```

Role Structure



Tasks: Modules

- Almost 600 modules
- Cloud: Amazon (EC2, S3), Azure, Docker, OpenStack, VMWare
- Commands: command, expect, raw, script, shell
- Database: MySQL, PostgreSQL, MongoDB
- Files: copy, find, lineinfile, patch, unarchive, xattr
- Monitoring: BigPanda, Nagios, Monit, Zabbix
- Notification: Campfire, IRC, Jabber, Mail, Slack, osx_say
- Systems: firewalld, GlusterFS, IPTables, Puppet, ZFS
- Windows: IIS, MSI, regedit, Robocopy

Roles: Tasks

```
- name: Copy PAM configuration files
  template: src=centos-7/pam.d/{{ item }}.j2 dest=/etc/pam.d/{{ item }} owner=root
group=root mode=0644
  with_items:
    - password-auth-ac
    - postlogin-ac
    - system-auth-ac


- name: Update authconfig
  lineinfile: dest=/etc/sysconfig/authconfig line="{{ item.key }}={{ item.value }}"
regexp="^{{ item.key }}"
  with_items:
    - { key: "PASSWDALGORITHM", value: "md5" }
    - { key: "USELDAPAUTH", value: "no" }
    - { key: "USEKERBEROS", value: "yes" }


- name: Enable and start services
  service: name={{ item }} enabled=yes state=started
  with_items:
    - nslcd
    - sssd
```

Templating

```
#! /bin/bash

# {{ ansible_managed }}

VM_HOSTNAME="{{ item.vm_hostname }}"
VM_DOMAIN="{{ item.vm_domain }}"
VM_NO_CPUS="{{ item.vm_no_cpus }}"
VM_MEMORY_MB="{{ item.vm_memory_mb }}"
VM_MAC_ADDRESS="{{ item.vm_mac_address }}"
VM_DISK_SIZE="{{ item.vm_disk_size }}"
VM_OS_FAMILY="{{ item.vm_os_family }}"
VM_VLAN="{{ item.vm_vlan }}"

# Check whether a volume group exists for the VM, and create one if not
VG_INFO=`/sbin/lvdisplay | grep $VM_HOSTNAME`
if [ "$VG_INFO" == "" ]; then
    echo "Creating volume group"
    /sbin/lvcreate -L $VM_DISK_SIZE -n $VM_HOSTNAME virtvg
else
    echo "Volume group exists"
fi
```

```
- name: Prepare VM creation scripts
  template: src=vm-creation-script.j2
dest=/etc/libvirt/machine_scripts/{{ item.vm_hostname }}
owner=root group=root mode=0744
  with_items: "{{ virtual_machines }}"
  when: item.vm_kvm_host == ansible_hostname
  tags: vm-creation
```

Templating

```
#! /bin/bash

# Ansible managed: /etc/ansible/roles/vm-creation/templates/vm-creation-script.j2
modified on 2016-05-21 23:49:31 by root on vetinari.ppe.gla.ac.uk

VM_HOSTNAME="ook"
VM_DOMAIN="ppe.gla.ac.uk"
VM_NO_CPUS="2"
VM_MEMORY_MB="2048"
VM_MAC_ADDRESS="52:54:00:c2:20:00"
VM_DISK_SIZE="50G"
VM_OS_FAMILY="rhel7"
VM_VLAN="210"

# Check whether a volume group exists for the VM, and create one if not
VG_INFO=`/sbin/lvdisplay | grep $VM_HOSTNAME`
if [ "$VG_INFO" == "" ]; then
    echo "Creating volume group"
    /sbin/lvcreate -L $VM_DISK_SIZE -n $VM_HOSTNAME virtvg
else
    echo "Volume group exists"
fi
```

Templating

```
# {{ ansible_managed }}
{% if automount_data %}
/data               {{ autofs_config_dir }}/auto.data         {{ autofs_config_opts }}
{% endif %}
{% if automount_home %}
/home               {{ autofs_config_dir }}/auto.home         {{ autofs_config_opts }}
/home/grid          {{ autofs_config_dir }}/auto.home.grid    {{ autofs_config_opts }}
/home/phas          {{ autofs_config_dir }}/auto.home.phas    {{ autofs_config_opts }}
/home/ppe           {{ autofs_config_dir }}/auto.home.ppe     {{ autofs_config_opts }}
{% endif %}
{% if automount_nfs_research %}
/nfs/archive        {{ autofs_config_dir }}/auto.nfs.archive  {{ autofs_config_opts }}
/nfs/atlas          {{ autofs_config_dir }}/auto.nfs.atlas    {{ autofs_config_opts }}
/nfs/cdf            {{ autofs_config_dir }}/auto.nfs.cdf      {{ autofs_config_opts }}
/nfs/detdev         {{ autofs_config_dir }}/auto.nfs.detdev   {{ autofs_config_opts }}
/nfs/lhcb           {{ autofs_config_dir }}/auto.nfs.lhcb     {{ autofs_config_opts }}
{% endif %}
{% if automount_nfs_admin %}
/nfs/ppemgr         {{ autofs_config_dir }}/auto.nfs.ppemgr   {{ autofs_config_opts }}
{% endif %}
{% if automount_nfs_web %}
/nfs/www            {{ autofs_config_dir }}/auto.nfs.www      {{ autofs_config_opts }}
{% endif %}
```

Templating

```
# Ansible managed: /etc/ansible/roles/autofs/templates/auto.master.j2 modified on 2016-
06-14 09:32:58 by root on vetinari.ppe.gla.ac.uk
/home               /etc/autofs/auto.home         --ghost rw,tcp,nfsvers=3
/home/grid          /etc/autofs/auto.home.grid    --ghost rw,tcp,nfsvers=3
/home/phas          /etc/autofs/auto.home.phas    --ghost rw,tcp,nfsvers=3
/home/ppe           /etc/autofs/auto.home.ppe     --ghost rw,tcp,nfsvers=3
/nfs/ppemgr         /etc/autofs/auto.nfs.ppemgr   --ghost rw,tcp,nfsvers=3
```

roles/autofs/defaults/main.yml

```
autofs_config_dir:              /etc/autofs
autofs_config_opts:             --ghost rw,tcp,nfsvers=3
autofs_mount_opts:              -rw,tcp

automount_data:                 false
automount_home:                 false
automount_nfs_admin:            false
automount_nfs_research:         false
automount_nfs_web:              false
```

group_vars/inventory

```
automount_home:             true
automount_nfs_admin:        true
```

Templating

```
# {{ ansible_managed }}

[domain/default]
ldap_id_use_start_tls = False
cache_credentials = True
ldap_search_base = {{ ldap_base_dn }}
krb5_realm = {{ krb_realm }}
krb5_server = {{ krb_hostnames | join(':88,') }}:88
id_provider = ldap
auth_provider = krb5
chpass_provider = krb5
ldap_uri = ldaps://{{ ldap_hostnames | join(',ldaps://') }}
krb5_kpasswd = {{ krb_master_hostname }}
ldap_tls_cacertdir = /etc/openldap/cacerts
```

Templating

```
# Ansible managed: /etc/ansible/roles/authn-client/templates/sssd.conf.j2 modified on
2016-05-11 22:09:35 by root on vetinari.ppe.gla.ac.uk

[domain/default]
ldap_id_use_start_tls = False
cache_credentials = True
ldap_search_base = dc=phas,dc=glasgow
krb5_realm = PHAS.GLA.AC.UK
krb5_server = kdc1.phas.gla.ac.uk:88,kdc2.phas.gla.ac.uk:88,kdc3.phas.gla.ac.uk:88
id_provider = ldap
auth_provider = krb5
chpass_provider = krb5
ldap_uri =
ldaps://lipwig.ppe.gla.ac.uk,ldaps://adora.ppe.gla.ac.uk,ldaps://dibbler.ppe.gla.ac.uk
krb5_kpasswd = kdc1.phas.gla.ac.uk
ldap_tls_cacertdir = /etc/openldap/cacerts
```

Templating

```
# {{ ansible_managed }}

# Batch compute nodes

{% for host in groups['compute'] %}
# {{ host }}
define host {
    address             {{ hostvars[host].ansible_default_ipv4.address }}
    host_name           {{ hostvars[host].ansible_fqdn }}
    use                 linux-server
}


{% endfor %}
```

Templating

```
# Ansible managed: /etc/ansible/roles/nagios-
config/templates/objects/hosts/compute.cfg.j2 modified on 2016-06-03
12:14:29 by root on vetinari.ppe.gla.ac.uk

# Batch compute nodes

# node001
define host {
    address             10.194.1.1
    host_name           node001.ppe.gla.ac.uk
    use                 linux-server
}


# node002
define host {
    address             10.194.1.2
    host_name           node002.ppe.gla.ac.uk
    use                 linux-server
}
```

Tasks: Dependencies

meta/main.yml

```
dependencies:
  - { role: glasgow-dir }
  - { role: repo-glasgow }
```

Tasks: Multi-OS

```
- include: install.yml
  tags: afs


- include: configure-c6.yml
  tags: afs
  when: ((ansible_distribution == "CentOS" or ansible_distribution ==
"Scientific") and ansible_distribution_major_version|int <= 6)


- include: configure-c7.yml
  tags: afs
  when: ((ansible_distribution == "CentOS" or ansible_distribution ==
"Scientific") and ansible_distribution_major_version|int == 7)
```

Tasks: Conditional

```
- name: Install packages (general)
  yum: name={{ item }} state=present
  with_items:
    - nano
    - net-tools
    - vim
    - vim-enhanced
  when: packages_usability


- name: Install packages (admin)
  yum: name={{ item }} state=present
  with_items:
    - nmap
    - nmap-cat
  when: packages_admin


- name: Install packages (compute)
  yum: name={{ item }} state=present
  with_items:
    - atlas
    - blas
    - ca-certificates
    - HEP_OSlibs
    - java-1.8.0-openjdk
    - lapack
    - lcg-CA
    - openblas
    - xrootd-client
    - xrootd-client-libs
    - xrootd-fuse
    - xrootd-libs
  when: inventory_hostname in
groups('compute')
```

Handlers

```
- name: Restart AFS
  service: name={{ item }} state=restarted
  with_items:
    - afs
    - openafs-client
  ignore_errors: yes


- name: Restart autofs
  service: name=autofs state=restarted


- name: Restart HTTPD
  service: name=httpd state=restarted


- name: Restart iptables
  service: name=iptables state=restarted


- name: Restart NRPE
  service: name=nrpe state=restarted


- name: Restart rsyslog
  service: name=rsyslog state=restarted


- name: Restart SSH
  service: name=sshd state=restarted
```

Handlers

```
- name: Restrict users permitted to connect via SSH
  lineinfile: dest=/etc/ssh/sshd_config state=present regexp="AllowUsers (.*)"
line="AllowUsers root ppeadmin {{ ppe_admins | join(' ') }} {{ other_admins | join(' ')
}} {{ host_ssh_users | default() | join(' ') }}"
  tags: ssh
  notify: Restart SSH


- name: Configure SSH daemon
  lineinfile: dest=/etc/ssh/sshd_config state=present regexp="^(#)?{{ item.key }}(
)+.*" line="{{ item.key }} {{ item.value }}"
  with_items:
    - { key: "GSSAPIKeyExchange", value: "yes" }
  tags: ssh
  notify: Restart SSH
```

Playbooks

```
- hosts: compute
  roles:
    - common
    - ansible-prereq
    - yum-cron
    - authn-client
    - authn-keytab
    - autofs
    - ssh-authorized-keys
    - ssh-config
    - sudo-config
    - cvmfs
    - afs
    - munge
    - packages
    - torque-compute
    - firewall
    - nagios-nrpe
  vars_files:
    - vars/afs.yml
    - vars/infrastructure.yml
    - vars/kerberos.yml
    - vars/ldap.yml
    - vars/nagios.yml
    - vars/path.yml
    - vars/torque.yml
    - vars/users.yml
```

Restart autofs

Restart autofs

Fact Caching

monitoring.yml

```
- hosts: all
  tasks: []

- hosts: monitoring
  roles:
    - common
    - ansible-prereq
    - nagios-config
  vars_files:
    - vars/infrastructure.yml
    - vars/kerberos.yml
    - vars/ldap.yml
    - vars/nagios.yml
    - vars/path.yml
    - vars/users.yml
    - vars/vm.yml
```