Lightweight on-demand computing with Elasticluster and Nordugrid ARC

On behalf of the ATLAS Collaboration

Maiken Pedersen, University of Oslo (NO)
David Cameron, University of Oslo (NO)
Andrej Filipcic, Jozef Stefan Institute (SI)
Overview

• Types of ATLAS sites in WLCG including the Nordugrid ARC and aCT INTERNAL mode grid site
• Overview of the different ARC-CE submission interfaces
• Setup and configuration of OpenStack grid site with Elasticluster
• INTERNAL submission interface in use
• Conclusion
A handful different types of ATLAS sites in the WLCG

TRADITIONAL GRID MODE
- Middleware on WN
- Inbound connectivity on WN and frontend
- Information publishing service for discovery

NORDUGRID STANDARD MODE
- NO middleware on WN
- NO inbound connectivity on WN
- Inbound connectivity on frontend
- Information publishing service for discovery

ARC: Advanced Resource Connector
aCT: ARC/ATLAS Control Tower

- A site might offer several grid flavours
  - Grid
  - HPC
  - Cloud
**TRADITIONAL GRID MODE**
- Middleware on WN
- Inbound connectivity on WN and frontend
- Information publishing service for discovery

**NORDUGRID INTERNAL MODE**
- NO middleware on WN
- NO inbound connectivity neither on WN nor frontend
- NO information publishing

**NORDUGRID STANDARD MODE**
- NO middleware on WN
- NO inbound connectivity on WN
- Inbound connectivity on frontend
- Information publishing service for discovery
Overview of the ARC-CE submission interfaces
INTERNAL submission interface

With aCT and ARC-CE installed at site running in “internal” mode: system administrator can run aCT and ARC-CE as non-root
  • All files and jobs owned by this user

Since aCT and ARC are run on the same machine no host certificate is required

→ Minimal set of services, no gridftp server, no emi-es, no ldap, no host certificate

Lightweight ARC-CE beneficial for installation, configuration and maintenance
Setup and configuration of OpenStack grid site with Elasticluster
Elasticluster

Tool that uses ansible scripts to set up a cluster on a cloud service from inside or outside the cloud

- Elasticluster supported cloud providers
  - ec2_boto
  - Google
  - Openstack
  - Libcloud

- Batch system – slurm/gridengine/htcondor

- NFS setup

- HPC common software (... lmod, ...), ganglia

Available roles in Elasticluster:

anaconda  easybuild  glusterfs-server  hadoop.yml  htccondor.yml  jupyter
ansible  ganglia-gmetad  gridengine-common  iptables  jupyterhub  lua
ansible.yml  ganglia-gmond  gridengine-exec  ipython  jupyterhub.yml  mcr
bigtop  ganglia-web  gridengine-master  hive  kubernetes-common  mcr.yml
ceph  ganglia.yml  gridengine.yml  hive-server  kubernetes-master  mcr
ceph.yml  glusterfs-client  hpc-common  jenkins  kubernetes-worker
common  glusterfs-common  hadoop-common  jenkins.yml  nis
        htcconductor.yml  hdfs-datanode  kubernetes.yml  ntgp
        hdfs-namenode  hive  lambda  pbs
        ipython  hive-server  lmod  pbs+maui
        jupyter  hive  lmod  r
        jupyter.yml  hive  lmod  r.yml
        jupyterhub.yml  htcconductor.yml  lua  r
        jupyterhub.yml  mcr
        jupyterhub.yml  mcr.yml
        jupyterhub.yml  mcr
        jupyterhub.yml  mcr
        jupyterhub.yml  mcr
        jupyterhub.yml  mcr

Maiken Pedersen - UiO - CHEP 2018
Elasticluster in work for SLURM grid site

• Elasticluster contacts the cloudprovider through the API
• Fires up specified number of frontends and compute nodes with specified OS, size, memory, and what ports to open (through predefined security group)
• Installs slurm server for frontend and client on compute nodes, NFS, ganglia (+ whatever else specified)
• Elasticluster “after” play used to customize the frontend and compute elements
Steps to create an ARC-CE INTERNAL site
Ansible script tasks

**On frontend**
- Install, configure ARC, aCT
- Mounting of extra block storage for shared session directory, cache and runtime directory
- Install CA’s for verification of incoming jobs
- Modify $PATH and $PYTHONPATH for non-default installation and as non-root
- Create griduser and add user to SLURM

**On compute node**
- Cvmfs setup plus extra block storage to contain it
- Create griduser and add user to SLURM
Elasticluster and ansible sequence

step1)
```
elasticcluster -v start slurm -n $clusternam
```

step2)
```
elasticcluster -v setup $clusternamen -- elasticcluster/src/elasticcluster/share/playbooks/after_custom.yml
  --tags "after"
  --extra-vars="localuser=centos lrms_type=slurm cluster_name=$clusternam"
  --extra-vars="@$play_vars/blockstorage.yml"
  --extra-vars="@$play_vars/griduser_local.yml"
  --extra-vars="@$play_vars/os_env.yml"
  --extra-vars="@$play_vars/nfs_export_mounts_local.yml"
```

step3)
```
ansible-playbook grid-uh-cloud/ansible/site_arc-ce_act.yml
-i ~/.elasticluster/storage/$clusternam.inventory
  --skip-tags="installarc,private-act,cvmfs,apache"
  --extra-vars="localuser=centos installationtype=local arc_major=6 lrms_type=slurm"
  --extra-vars="@$play_vars/griduser_local.yml"
  --extra-vars="@$play_vars/os_env.yml"
  --extra-vars="@$play_vars/host_env.yml"
  --extra-vars="@$play_vars/slurm_pwd.yml"
```

Link to playbook to install ARC and aCT (step 3)
https://source.coderefinery.org/nordugrid/contrib/tree/master/ansible/arc-ce
Testing submission with the INTERNAL submission mode

Specify local interface –S org.nordugrid.internal or leave blank as it is the default

[centos@frontend001 testing]$ arcsub --c localhost -S org.nordugrid.internal hello.xrls
[[centos@frontend001 testing]$ arcsub --c localhost hello.xrls
Job submitted with jobid: file:///wlcg/session/5a1NDm1r9vsnrp02tmaBI5UnABFKDmABFKDmB2KKDmABFKDmXr0rKm

[[centos@frontend001 testing]$ arcstat --long --all
Job: file:///wlcg/session/5a1NDm1r9vsnrp02tmaBI5UnABFKDmABFKDmB2KKDmABFKDmXr0rKm
Name: hello_ARCTEST1
State: Queuing
Specific state: INLRMS
ID on service: 5a1NDm1r9vsnrp02tmaBI5UnABFKDmABFKDmB2KKDmABFKDmXr0rKm
Service information URL: file:///localhost (org.nordugrid.internal)
Job status URL: file:///localhost (org.nordugrid.internal)
UIO_CLOUD queue
Hammercloud jobs with local submission in PanDA monitor

- An ARC-CE and aCT INTERNAL test cluster has successfully been installed in the University of Oslo’s Openstack cloud service
- Collects jobs from PanDA as the UIO_CLOUD queue
- The jobs are so-called Hammercloud jobs
  - Testing framework using realistic ATLAS jobs
  - Jobs require cvmfs, download of input files etc.

<table>
<thead>
<tr>
<th>Job</th>
<th>Datasets:</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>3977944112</td>
<td>In: mc15_13TeV.361106.PowhegPythia8EvtGen_AZNLOCTEQ6L1_Zee.evgen.EVTN.a3601.tid04972714_00</td>
<td></td>
</tr>
<tr>
<td>3977936140</td>
<td>In: mc15_13TeV.361106.PowhegPythia8EvtGen_AZNLOCTEQ6L1_Zee.evgen.EVTN.a3601.tid04972714_00</td>
<td></td>
</tr>
<tr>
<td>3977928003</td>
<td>In: mc15_13TeV.361106.PowhegPythia8EvtGen_AZNLOCTEQ6L1_Zee.evgen.EVTN.a3601.tid04972714_00</td>
<td></td>
</tr>
</tbody>
</table>
Conclusion

• ARC and aCT gives a new site configuration option for ATLAS sites
  • Lightweight
  • Good option for restrictive sites
  • Suitable for cloud and HPC

• Will be available in upcoming release of ARC 6
  • Pre-release version already available
  • https://source.coderefinery.org/nordugrid/arc
Extra material
Minimalistic configuration of ARC for INTERNAL submission only running ARC as normal user

```
[lrms]
lrms=slurm

[arex]
logfile=/grid/arex.log
joblog=/grid/gm-jobs.log
controldir=/grid/control
sessiondir=/wlcg/session
runtimedir=/wlcg/runtime
shared_scratch=/wlcg

[arex/cache]
logfile=/grid/cache-clean.log
cachedir=/wlcg/cache
cachesize=80 70
cachelifetime=1d

[infosys]
logfile=/grid/infprovider.log

[queue:main]
```

For production site you would add VO configuration
Example configuration of elasticcluster

[cloud/iaas]
provider=openstack
auth_url=https://api.uh-iaas.no:5000/v3
username=maiken.pedersen@uisit.uio.no
password=xxxxxxx
project_name=uio-test-hpc-grid
user_domain_name=dataporten
project_domain_name=dataporten
region_name=osl
identity_api_version=3

[login/centos]
image_user=centos
image_user_sudo=root
image_sudo=True
user_key_name=cloud
user_key_private=~/.ssh/cloud.key
user_key_public=~/.ssh/cloud.key.pub

[setup/ansible-slurm]
provider=ansible
frontend_groups=slurm_master,ganglia_master,ganglia_monitor,frontend,cluster
compute_groups=slurm_worker,ganglia_monitor,compute,cluster
global_var_multiuser_cluster=no

[cluster/slurm]
cloud=iaas
login=centos
setup=ansible-slurm
security_group=default
image_id=df3deddc-f98c-4eb8-b77e-7f8f24f857e4
frontend_nodes=1
compute_nodes=1
ssh_to=frontend
network_ids=97faa886-592e-4ad1-a99b-6d56851bed78

[cluster/slurm/frontend]
flavor=m1.medium

[cluster/slurm/compute]
flavor=m2.4xlarge
Configuration of aCT for INTERNAL mode

Maiken Pedersen, UiO - CHEP 2018
Nordugrid ARC CE modes

Pilot factory

True pilot

NDGF mode

Maiken Pedersen - UiO - CHEP 2018
Nordugrid ARC CE modes for restrictive (HPC) sites and lightweight sites, including clouds

**INTERNAL mode HPC**

**INTERNAL mode cloud**