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
• Nordugrid ARC and aCT in INTERNAL mode on cloud resource
• Overview of the 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

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

- A site might offer several grid flavours depending on availability
  - 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
- 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

→ 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:
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
- Custom Elasticluster “after” play can be configured to attach extra storage volumes needed for the ARC-CE, and distribute shared folders
Creating an ARC-CE with aCT and preparing compute nodes

**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
Elasticcluster and ansible sequence

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

step2)
```
elasticcluster -v setup $clusternname -- elasticcluster/src/elasticcluster/share/playbooks/after_custom.yml \
--tags "after" \
--extra-vars="localuser=centos lrms_type=slurm cluster_name=$clusternname" \
--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 ~/.elasticcluster/storage/$clusternname.inventory \
--skip-tags="installarc,private-act,cvmfs,apache" \
--extra-vars="localuser=centos installationtype=local arc_major=6 lrms_type=slurm" \
--extra-vars="@$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"
```
Testing submission with the INTERNAL submission interface

On compute-element use INTERNAL job submission

```shell
arcsub --direct -c localhost -S org.nordugrid.internal hello.xrls

[[centos@frontend001 arctesting]$ arcstat --c localhost --long

Job: local://localhost/q5ZNDmJ4CdrnzfEJwm4kCpGoABFKDmABFKDm6SIKdmABFKDmmXOthn
Name: hello_LOCAL-CLOUD-ARC
State: Finishing
Specific state: FINISHING
ID on service: q5ZNDmJ4CdrnzfEJwm4kCpGoABFKDmABFKDm6SIKdmABFKDmmXOthn
Service information URL: local://localhost (org.nordugrid.local)
Job status URL: local://localhost (org.nordugrid.local)
Job management URL: local://localhost (org.nordugrid.local)

Status of 1 jobs was queried, 1 jobs returned information
```
Hammercloud jobs with local submission in PanDA monitor

- An ARC-CE and aCT INTERNAL test cluster has successfully been installed
- 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.
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 at 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=ld

[infosys]
logfile=/grid/infoprovider.log

[queue:main]

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

```yaml
[cloud/iaas]
provider=openstack
auth_url=https://api.uh-iaas.no:5000/v3
username=maiken.pedersen@usit.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=df3dedc6-f98c-4eb0-b77e-7f8f24f857e4
frontend_nodes=1
compute_nodes=1
ssh_to=frontend
network_ids=97fa886-592e-4ad1-a995-6d5651bed78

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

[cluster/slurm/compute]
flavor=m2.4xlarge
```

Maiken Pedersen - UiO - CHEP 2018
Configuration of aCT for INTERNAL mode

```xml
<config>
  <db>
    <type>mysql</type>
    <name>act</name>
    <user>centos</user>
    <password>secret</password>
    <host>localhost</host>
    <port>3306</port>
  </db>

  <periodicrestart>
    <actsubmitter>120</actsubmitter>
    <actstatus>600</actstatus>
    <actfetcher>600</actfetcher>
    <acctcleaner>600</acctcleaner>
  </periodicrestart>

  <tmp>
    <dir>/tmp</dir>
  </tmp>

  <actlocation>
    <dir>/grid/software/aCT/src</dir>
    <pidfile>/grid/act.pid</pidfile>
  </actlocation>

  <logger>
    <level>debug</level>
  </logger>

  <atlasglis>
    <timeout>20</timeout>
  </atlasglis>

  <queuesreject>
    <item>bgmem</item>
    <item>tier3</item>
    <item>infiniband</item>
    <item>gridsim</item>
  </queuesreject>

  <jobs>
    <checkinterval>30</checkinterval>
    <checkmtime>20</checkmtime>
    <maxtimerunning>259200</maxtimerunning>
    <maxtimehold>172800</maxtimehold>
    <maxtimeundefined>3600</maxtimeundefined>
  </jobs>

  <voms>
    <vo>atlas</vo>
    <roles>
      <item>production</item>
    </roles>
  </voms>
</config>
```
Nordugrid ARC CE modes

Pilot factory

True pilot

NDGF mode
Nordugrid ARC CE modes for restrictive (HPC) sites and lightweight sites, including clouds

Maiken Pedersen - UiO - CHEP 2018