

EU grant n.822070

Cloud Technologies for DIRAC

Dr. Silvio Pardi Jennifer2 Computing WorkShop - Task 5.1 CERN – 12/12/2019

Summary

- Cloud Resources in DIRAC
- Technologies considered in Belle II
- VMDIRAC
- CE-like Cloud I/Fs
- VCYCLE



Integration with grid and cluster

DIRAC works like a big cluster over WAN

- Central task queue
- Pilot director
 - Submit proper pilots job to different resources dealing with heterogeneous
- Match service
 - Match proper user jobs to resources

The main difference between Cloud and Cluster is that the WNs of a cluster are mostly static while VM use to be dynamically created and destroyed.



Technologies evaluated in Belle II

- VMDIRAC
 - Academic clouds and AWS
- CE-like Cloud I/Fs
 - Cloud Scheduler (UVic)
 - Dynamic Torque (Melbourne/CoEPP)
- VCYCLE
 - Small Academic Cloud and HNSciCloud



BELLE II Distributed Component



VMDIRAC

- VMDIRAC is an extension of the DIRAC system. It is used to manage a pool of different kind of clouds by API drivers like EC2, OCCI, rOCCI, Nova
- <u>https://github.com/DIRACGrid/VMDIRAC</u>



VMDIRAC Integration with cloud

- "VM director" instead of "Pilot director"
 - start VMs, instead of submitting pilot jobs
- VMs at boot time start "pilot job"
 - This makes the instantiated VMs behave just as other WNs with respect to the DIRAC WMS
- VM scheduler need to manage dynamic virtual machines according to job situation



VMDIRAC Tested in Torino

VMDIRAC: DIRAC system exstention

Use of private and public cloud resources

Contextualization done with Cloud-init (on VM)

rOCCI remote cloud management

Authentication with secure HTTPS

Tests performed with VMDIRAC v2r1p1

DIRAC job submission succesful

Life cycle:

- job submission via VMDIRAC
- VM created
- job execution
- VM destroyed

Performances:

- 1. start the VM: few sec.
- 2. OS availability: ~ 1 min
- 3. DIRAC on VM: ~ 5 min
- 4. Pilot 2.0 components: ~ 3 min
- 5. Job matched by JobAgent: ~ 10 min from start



Cloud Toy - Easy Cloud Infrastructure Setup

ks.cfa

- create kickstart file
- prepare customized ISO
- . make bootable usb key
- . install on server
- check everything works
- B2CT (Belle 2 Cloud Toy with setup optimized and ready to work with BELLE2DIRAC)



Cloud Scheduler (UVic)

- Cloud Scheduler manages virtual machines on clouds like Nimbus, OpenNebula, Eucalyptus or EC2 to create an environment for batch job execution.
- Users submit their jobs to a batch job queue like Condor, Sun Grid Engine, or Platform LSF, and Cloud Scheduler boots VMs to suit those jobs.

https://hep-gc.github.io/cloud-scheduler/



Cloud Scheduler (UVic)



Randall Sobie University of Victoria

3

~ ~

Cloud Scheduler (UVic)

Cloudscheduler Version 2

- In-memory data of CSV1 are replaced with a MariaDB database
 - Keep track of system state in MariaDB
 - More responsive with MariaDB
 - Resilient to outages and easier to maintain
- Independent scheduling, polling, and user interface processes that track the state of the clouds and the HTCondor pool
 - Improved VM and job scheduling
- RESTful web user interface for
 - Improved administration and management
 - Monitoring of the clouds and jobs
- Expanded functionality
 - Multiple projects (experiments)
 - Multiple HTCondor instances
 - Opportunistic sharing in a cloud



VCYCLE

VCYLCE is VM lifecycle manager developed by GRIDPP, it is designed to create VMs on Cloud endpoints offering EC2, Openstack or Azure interface.

VCYCLE can be easily integrated in DIRAC and the accounting system is compliant with APEL. VCYCLE is currently uses in production by LHCb.

https://www.gridpp.ac.uk/vcycle/



VCYCLE



VCYCLE Configuration: /etc/vcycled/vcycle.conf

[space prisma-controller01.na.infn.it]

api = openstack url = https://prismaidentity01.na.infn.it:5000/v2.0/ tenancy_name = vcycle username = vcycle password_base64 = XYZ network_uuid = b47c3bae-4f42-4e88-bd62-31ce3b0b97ca max_processors = 16

[machinetype prisma-controller01.na.infn.it belle] processors limit = 10 processors per machine = 1 flavor names = vcycle.nano backoff seconds=100 fizzle_seconds=100 max_wallclock_seconds=86400 https x509dn = /DC=org/DC=terena/DC=tcs/C=IT/L=Frascati/O=Istituto Nazionale di Fisica Nucleare/CN=recas-vmvcycle.na.infn.it root_public_key = /root/.ssh/id_rsa.pub target_share = 1.0 root_image = https://repo.gridpp.ac.uk/vacproject/example/cernvm3.iso user data = https://recas-vcycle01.na.infn.it:8443/HNSciCloud/pilotkek2/user data user data option cvmfs proxy = http://squid-cvmfs01.na.infn.it:3128 user data machine hostname = recas-vmvcycle.na.infn.it user data manager hostname = recas-vcycle01.na.infn.it user_data_file_hostkey = /opt/vcyclefile/vmhostkey.pem user_data_file_hostcert = /opt/vcyclefile/vmhostcert.pert

VCYCLE Contextualization

The golden image is the standard CERNVM that is completed by a contextualization process via cloud-init.

Before to lunch the VM, vcycle server checks if there is an updated version of the golden image.

Contextualization allows to configure the VMs to start the dirac-pilot.py script at boot time, providing the configuration parameters.

A customized VM could be used in order accelerate boot.

•••••

Now run the pilot script

python /scratch/plt/dirac-pilot.py \

--debug \

-o '/LocalSite/SubmitPool=Test' \

--Name '##user_data_space##' \

--Queue default \

--MaxCycles 1 \

--CEType Sudo \

--cert \

--certLocation=/scratch/plt/etc/grid-security \

-r 'v4r6p5' \

-e 'Belle' \

-g '2016-11-03' \

>/var/spool/joboutputs/dirac-pilot.log 2>&1



DIRAC Configuration





Jennifer2 Computing Workshop

12 December 2019

Conclusion

- There are multiple solutions to integrate Cloud Computing resources in DIRAC.
- In the context of Belle II we have tested some of them.
- Inside the Jennifer2 there are expertise to use some of those technologies tested in Belle II

