VCondor - an implementation of dynamic virtual computing cluster

Yaodong Cheng, CC-IHEP, CAS
chyd@ihep.ac.cn
5-10 March 2017, Taipei (ISGC2017)
Problems

- **Static resource management**
  - Fixed CPU cores in each queue

- **Resource access rights**
  - Resources are owned by the group who invested
  - Only the group can use their own resources
  - Each person is allowed to submit some number of jobs. If one user submits too many jobs, these jobs will be queued

- **Run environment of different experiments are not compatible with each other**
  - Difficult to provide one physical machine to different experiments

**Virtualization and Cloud computing**
Motivation

**GOALS we want to achieve**

- Simplify the maintenance of computing resources
  - Virtualization (virtual computing cluster)

- Share resources between different experiments
  - Dynamic allocation of resources (dynamic virtual computing cluster)

- Meet the peak requirements of resources
  - Integration of external resource transparent to user (cloud federation)
Virtual Computing Cluster

- Computing nodes are installed in virtual machines
- **Seamless integration** with the existing middleware stacks.
- **Completely transparent** to the computing service and end users.
Dynamic virtual computing cluster

RMS

PBS/HTCondor

VPManager

Virtual resource Pool Manager

Virtual machines

Openstack

hypervisor

Physical machines
VPManager components

- **VM Pool**
  - Manages one or more openstack instances, which hides the detailed information of openstack from upper applications.
  - Makes it possible to deploy multiple and different versions of openstack.

- **VM Quota**
  - Checks the information of VM Pool and requirements of different applications to allocate or reserve resources.

- **Virtual job manager, VPBS and VCondor**
  - Checks the status of different queue and get the available VM number and create new VMs or destroy existing VMs.

- **VM node manager**
  - Checks and controls all the VM run environment such network status, affiliated job queue by an agent running in the virtual machine.

- **Accounting system**
  - Keeps all the usage information of each virtual machine and generate bills to user.
VCondor

I. JobMonitor: query and record job information and HTCondor queue changes

II. NodeManager: use plugins (openstack api occi, or other interface) to create or destroy virtual machines

III. DAEMON: Main module, periodically executed
VMQuota

- Resource Quota management for different experiments
- Different experiments have different resource queues
- Allocate and reserve resources for different queues

<table>
<thead>
<tr>
<th>Queue Name</th>
<th>Low threshold</th>
<th>High threshold</th>
<th>Available</th>
<th>Reserve time(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>BES</td>
<td>100</td>
<td>400</td>
<td>200</td>
<td>600</td>
</tr>
<tr>
<td>JUNO</td>
<td>100</td>
<td>300</td>
<td>150</td>
<td>600</td>
</tr>
</tbody>
</table>

VMQuota Client

Socket

{{"ResID":"juno"}}

Response

VMQuota Server

Socket

JSON

Request

{{"ResID":"juno","MIN":100,"AVAILABLE":50}}
Resource pool expansion

CONDOR JOBS POOL (schedd)

CONDOR NODES POOL (collector)

VCondor

VMQuota

Openstack
OpenNebula
...

BES

JUNO

Condor Job
Submit Node

Job

Compute Job

Condor Worker Node

Virtual Machine

Compute Job Transfer

VCondor Workflow

Virtual Machines

startd

VM Creation

Job Scheduler

HTCondor

High Throughput Computing

Job

WN

Job

Job

Job

Job

Job

Job

Job

Job

Job

VM

Creation

Virtual Machines

1

2

3

4

5

6

7

8

9

10

11
Resource pool shrink

CONDOR JOBS POOL (schedd)

CONDOR NODES POOL (collector)

Openstack
OpenNebula

VCondor

VMQuota

Virtual Machines

BES

Condor Job
Submit Node

Condor Worker
Node

Virtual Machine

Compute Job

Compute Job Transfer

VCondor Workflow

Low threshold

VM destroy
Deployment

@IHEP, China
- Virtual computing cluster, ~ 1000 CPU cores
- HTCondor
- IHEPCloud: Openstack Kilo
- Support LHAASO, JUNO, BES, CEPC, ...

@JINR, Russia
- testbed
- HTCondor
- OpenNebula
VCondor monitoring

~50% resource utility with legacy management

Resource high threshold
Resource low threshold
Job queued, automatically create virtual machines
VCondor: How to use

- Download VCondor from https://github.com/hepgnu/VCondor.git

- Make sure HTCondor and Openstack or OpenNebula are well configured

- Setup a VM Image with HTCondor startd setup pre-configured

- Setup a VM Template with Image in the above

- The VCondor configuration file allows us to configure most of its functionalities

- Start VCondor and submit jobs, then resource pool scale up and down dynamically on-demand
Ongoing Work: cloud federation

Condor Pool

payload

Check load

Start VM

VCondor

VMQuota

Info

IHEP

Firewall

Condor Connection Broker

CCB

router

Glidein startd

VM

Glidein startd

VM

Openstack, EC2, OpenNebula, ...

Other Sites
Conclusion

- Traditional cluster use static resource management, which leads to some problems
- VCondor is a middleware for dynamic virtual computing cluster
  - add or remove computing nodes depending on queue load
  - Improve resource utilization by sharing between different experiments
- VQuota coordinates the allocation of resources to make sure the fair and efficient use
- VCondor and HTCondor glidein is an optional solution to integrate remote cloud resources
Thank you!

Any Questions?