

CERN/CNAF meeting - 14/04/2021 (Wednesday) @ 11:00

Topic 1.2 - "Large scale deployments"

<https://indico.cern.ch/event/1026281/>

Participants:

- Belmiro Moreira
- Luis Fernandez Alvarez
- Stefano Dal Pra
- Arne Wiebalck
- Jaroslava Schovancova
- Doina Cristina Duma
- Andrea Chierici
- Gavin McCance
- Stefano Dal Pra
- Carmelo Pellegrino
- Daniele Cesini
- Diego Michelotto
- Federico Versari

1. Introduction / Catch up! Progress since the last meeting

- CNAF planning to use OpenStack Ironic
- Interested in the CERN Cloud architecture
- Ballance between Batch worker nodes / normal compute nodes in the infrastructure
- Looking into to run docker containers on baremetal
- Understand how CERN configures the cloud infrastructure and services on top

2. Questions sent by CNAF

(Generic CLOUD Infrastructure questions to set some context)

--> What's the CERN Cloud architecture / use cases?

- CERN Cloud Infrastructure started 8 years ago
- The vision was to run all the services on top of the Cloud Infrastructure
- Several advantages: IaaS model, server consolidation, improve resource utilization, common infrastructure for services, ...
- Based on OpenStack
- Currently with 3 regions (>8000 compute nodes)
- OpenStack Nova scales using Cells (>80 cells in total. Each cell with a max of 200 compute nodes)
- Cells also used to isolate different use cases and as failure domains
- Control plane run on VMs on top of the same cloud that it manages
- Service cells: CPU overcommit (to x4), custom CPU definition
- Batch cells: No overcommit!, Huge pages (2MB), scheduling considers NUMA topology, CPU passthrough
- Some cells dedicated to special use cases (Network, GPUs, Hyper-converge, ...). Requires special configuration for this special cases.
- Some pointers:
 - <https://techblog.web.cern.ch/techblog/post/10-years-of-openstack-at-cern-part-1/>
 - <https://techblog.web.cern.ch/techblog/post/10-years-of-openstack-at-cern-part-2/>

--> How dynamic is the life of a RH can be?

Examples:

The same RH can work as HV hosting (only) VWNs (Once installed, the RH is expected to work only as HV for VWNs)

The same RH can work as HV hosting (only) VMs (VWN and a mix of other VMs too)

Once installed as HV, it is expected to host both VWNs and generic VMs

The same RH can work as HV or BM (mutually exclusive)

The same RH can host both VMs and Container on bare metal at the same time

- For services/batch the hardware is the same.
- Generally, compute nodes don't change use case.
- Sometimes they are moved from batch (when a physical node is needed) because it's easier to drain.

--> Do you reserve a CPU for OpenStack (and/or other) core services?

- No. We don't reserve a CPU for the services in the compute nodes.
- We reserve memory. Ex: 8GB for 128GB RAM compute nodes.
- There are some special considerations for Batch workers on baremetal (outside the cloud):
 - For batch running in file servers there are spare SSDs used for special purposes.

--> Access to shared posix filesystems (gpfs, nfs, cvmfs, ...) happens via: A mountpoint on the RH, then the VM accesses it through its physical host; A mountpoint on the VM

- Services (VMs) don't have access to the compute nodes (Also managed by different teams).
- In this particular case, these mounts need to be done inside the VM.

(CLOUD Infrastructure - Baremetal)

--> Some context regarding the baremetal infrastructure. Why? What changed?

- Some services still require baremetal.
- Using OpenStack Ironic gave us the opportunity
 - to "rethink" how we manage the hardware lifecycle and get rid of some in-house machinery
 - to leverage the tools and workflows already built for VMs (as we have the same API)
 - to streamline the accounting
 - to enable new use cases (like k8s on baremetal)
- Most of the servers in the data centre integrated (adopted) into OpenStack Ironic.

--> How do you use ironic for bare metal provisioning?

- Baremetal service (OpenStack Ironic) is completely integrated with the Cloud Infrastructure.
- User continues to use the same OpenStack APIs to create a VM/Baremetal.

--> Can you explain the interaction with network? How many physical interfaces are connected? How do you manage network in openstack if bare metal needs more than one network?

- CERN network tight with CERN LanDB (network database).
- OpenStack Ironic at CERN skips network configuration (it uses the info from the CERN

LanDB).

- CERN uses its own PXE infrastructure (AIMS) for Ironic.
- No plan to use SDN on baremetal.

--> How do you contextualize a bare metal node after the Ironic provisioning? Cloud-init, Puppet-Foreman, other...

- Cloud-init is used (config drive). However, the metadata service is not available.
- Can be used to bootstrap Puppet for some services.
- The same image is used to create VMs or bare metal instances.

--> How do you compute HS06 computing power on nodes?

- HS06 benchmark is triggered by Ironic in every node.
- Ironic "cleaning" step downloads a container image that benchmarks the node and uploads the results to Elastic Search.
- HS06 is presented as project metadata in OpenStack per flavour. An intermediate API pulls and exposes that information to our worker nodes via Puppet. This HS06 value is then injected as HTCondor job attributes.
- Q: is this cleaning and benchmarking done individually on each machine?
-- A: Yes, then the average is used as benchmark for accounting. If large differences on equal models are observed, reason is investigated.

(BATCH infrastructure)

--> Can you describe the software setup of a generic worker node at CERN?

- Batch nodes look like lxplus at CERN.

--> Do you run jobs on bare metal?

- Yes, also Batch will run only on baremetal for new deliveries.
- It also runs in storage nodes.

--> Is the BATCH service running only on top of OpenStack?

- Not only on top of OpenStack.
- Baremetal, Storage servers, external Clouds, ...

--> What's the size of BATCH VMs? (CPU, RAM, Disk)

- Size depends in the NUMA nodes of the compute node CPU.
- Usually, one VM per NUMA node.

--> Do you host multi-node jobs in the HTC (Batch) farm?

- No.
- If it needs multi-node it should run on HPC (Slurm).

--> Do you run HPC batch systems nodes on OpenStack (e.g., Slurm)? If so, are there special tunings to interface with infiniband or similar high-performance networking?

- No.
- HPC has dedicated hardware that is not managed by OpenStack. (No VMs).
- Maybe in the future the physical nodes are deployed with Ironic.

--> Container WN: what size do you allow for WN containers on Bare Metal? one Container as a "whole node"? one container per CPU?

(Note: This choice determines the overall number of STARTDs in the HTCondor pool: are there observable effects on the reactivity / readiness of HTCondor the pool?)

- Tried, but will not follow this path.
- Difficult to sync packages... (new images required all the time).
- Critical points: swap, AFS, cvmfs,...

--> Do you use a specific docker container orchestrator? How do you handle common batch system admin task such as "security upgrade of kernel and other packages"? More precisely the process should go through these steps: CWNs are drained (at HTCondor level); A new docker image is set up, with the upgraded sw package (at "puppet" level); When a CWN is drained, the container can be shut off (the container orchestrator level does that, however, the information "drain done" which is needed to trigger the shutdown comes from the HTCondor level); When a BM does not run any active container, it can be rebooted with the new kernel; this information should be known to the Openstack/Ironic layer; When rebooted, a Bare Metal should re-instantiate its CWNs; Somehow one must coordinate activities at different "layers" (HTCondor to drain WNs, docker to poweroff/start containers, Ironic to reboot physical nodes). Do you have specific tools to handle this?

- Tried K8s.
- Independently of K8s, one approach we are considering at the moment is to wrap automatically the jobs in containers at the HTCondor level. This would decouple the worker node requirements from OS (wrap job with CentOS7 on a CentOS8 worker, ...)
- Exploring orchestration alternatives: stackstorm, zeebe and custom software APIs.

--> Do you provision batch (HTCondor) access to GPU resources?

- A lot of requests for GPU resources.
- Batch uses GPUs in few nodes.

--> is there an optimal VWN size (how many VWN per HV)

- No overcommit. Mostly depends in the size of the compute node CPU NUMA node.

--> Have you evaluated the impact of virtualization on production WN? According to our investigations 2 to 5% performance is lost. Is this acceptable?

- Presentation from HEPIX (Arne)
- On baremetal to have the best performance still need to map NUMA nodes to the jobs.
- Running on baremetal there is only a single CERNVM cache (Better performance)
- CPU affinity on Condor? Not clear
- 1 job, one CPU. Hyper-threading enabled.
- Batch team expects to increase the usage of baremetal. Batch will run on Baremetal for the new deliveries.

It was agreed that we may need to follow up in some of these topics.

New meetings should be scheduled after CNAF feedback.