INDIGO-Datacloud

Improving IaaS resources to accommodate scientific applications

Alvaro Lopez Garcia ¹, Lisa Zangrando ², Sara Vallero ³, Massimo Sgaravatto ², Stefano dal Pra ⁴, Stefano Bagnasco ³, Sonia Taneja ⁴, Vincent Llorens ⁵, Davide Salomoni ⁴, Giacinto Donvito ⁶

¹IFCA (CSIC-UC), ²INFN-Padova ³INFN-Torino, ⁴INFN-CNAF ⁵CNRS, ⁶INFN-Bari
Outline

Introduction

FairShare Scheduling
  Synergy: an extensible manager for OpenStack
  FairShare Scheduler for OpenNebula

Preemptible Instances in OpenStack

Partition Director

Conclusions
Introduction

FairShare Scheduling
  Synergy: an extensible manager for OpenStack
  FairShare Scheduler for OpenNebula

Preemptible Instances in OpenStack

Partition Director

Conclusions
• Collecting and consolidating evolving user requests.
• Creation of a new **sustainable cloud competence in Europe for PaaS**, for both the scientific and industrial sectors.
• Many **technology gaps**, for example:
  – Existing lock-ins: vendor, hypervisor or technology.
  – Inflexible way of distributing applications.
    ▶ Difficult to port applications between providers.
    ▶ Different interfaces, different APIs.
  – **Naive resource allocation strategies (i.e. scheduling)**.
    ▶ Prevents users from exploiting effectively the resources.
    ▶ Makes difficult for providers to partition the resources.
    ▶ Under-utilization of resources

For more details check the presentation "**INDIGO-DataCloud: a Cloud-based Platform as a Service oriented to scientific computing for the exploitation of heterogeneous resources:**" (13 Oct 2016, 11:15).
https://indico.cern.ch/event/505613/contributions/2227435/
Why is current scheduling a problem?

• Current cloud scheduling strategies are too naive, focused on satisfying industry needs.

• Schedulers are too simple: no priorities, static partitioning, static quotas, simplistic queuing.

• From a resource provider point of view:
  – Difficult to partition resources.
  – Compromise between using the infrastructure as much as possible and give enough share to each group.
  – Difficult to get a good utilization of the infrastructure (e.g. backfilling not possible).
  – Difficult to prioritize workloads from different users/groups.

• From a user’s point of view:
  – Difficult to exploit resources: limited quota even if free resources.
  – Opportunistic usage not possible.
Three complementary approaches:

**Fair Sharing scheduling** to implement resource allocation based on fair-share approach rather than a static quota.

**Preemptible instances** to allow a better utilization of idle resources.

**Partition Director** to easily move resources between different infrastructures and adjust the user shares.
Outline

Introduction

FairShare Scheduling
   Synergy: an extensible manager for OpenStack
   FairShare Scheduler for OpenNebula

Preemptible Instances in OpenStack

Partition Director

Conclusions
Outline

Introduction

FairShare Scheduling
  Synergy: an extensible manager for OpenStack
  FairShare Scheduler for OpenNebula

Preemptible Instances in OpenStack

Partition Director

Conclusions
Synergy is an extensible general purpose management service designed for executing tasks in OpenStack.

It is composed by a collection of pluggable managers with specific functionality.

6 different managers provide together an advanced resource allocation model

- Cloud resources shared among different OpenStack projects.
- Requests that cannot be immediately fulfilled are enqueued.
- Resource allocation based on fair-share approach → avoiding the static partitioning.
- Maximize the resource utilization.

Projects can access to two distinct quota kinds:

- **Private quota** the standard OpenStack quota, fixed and statically allocated.
- **Shared quota** access to extra resources shared among different projects.
Shared Quota

- Synergy allows the IaaS administrators to allocate a subset of resources to be shared among different projects, besides the ones statically partitioned.
- Resources are fairly distributed among users by complying specific fair-share policies:
  - list of projects allowed to access to the shared quota,
  - definition of shares on resource usages for the relevant projects,
  - max lifetime for resources (VM or container).
- User requests **are not rejected** if not immediately satisfied:
  - Persistent priority queue with backfill strategy.
  - SLURM fair-share algorithm.
- Synergy **does not replace** any existing OpenStack service (e.g. Nova) → complements OpenStack functionality as an independent service.
Synergy Architecture

- RESTFul
- Queue Manager
- Quota Manager
- Scheduler Manager
- FairShare Manager
- Nova Manager
- Keystone Manager

AMQP

keystone

nova
Outline

Introduction

FairShare Scheduling
  Synergy: an extensible manager for OpenStack
  FairShare Scheduler for OpenNebula

Preemptible Instances in OpenStack

Partition Director

Conclusions
FairShare Scheduler

Design Considerations

- Not intrusive in the OpenNebula core (independent from the ONE release).
- Self-contained module interacting only with the ON XML-RPC interface.
- Keep the original scheduler implementation for matching resources to requests.
- Implement an algorithm which prioritizes tasks according to an initial weight and to the historical resource usage per project.
- Order the requests queue according to priorities.
- Do not add new states to the VM life-cycle.
- Similar to Synergy with the ONE authentication, quotas, monitoring and accounting systems.
• **Priority Manager**: calculate priorities for queued jobs, use pluggable algorithms.

• **XML-RPC interface**: catch scheduler requests or redirect to ONE XML-RPC.

• **Priority DB**: internal data back-end.

• **Clients**: interface to the FSS core.

• **Sunstone extension**: monitoring.
Outline

Introduction

FairShare Scheduling
Synergy: an extensible manager for OpenStack
FairShare Scheduler for OpenNebula

Preemptible Instances in OpenStack

Partition Director

Conclusions
A preemptible instance is a special kind of instance, that can be terminated by a higher priority task.

Priority can be determined by a bid price (spot market), a fair-sharing algorithm or simply considering spot VS non spot.

Useful for opportunistic usage: get as much as you want, but be aware that you can be killed.

OPIE implemented as a pluggable OpenStack Compute scheduler + API extensions.

Aiming for introducing basic support for spot instances into OpenStack core (blueprint submitted, under discussion).

Policies implementation using external services (e.g. bid price) and scheduler weighters.
Preemptible instances scheduling flowchart

- new request
  - is spot?
    - Yes: get hosts full state
    - No: get hosts state without spot instances
    - filter
      - ERROR
  - hosts pass?
    - Yes: weigh and select hosts
      - overcommit?
        - Yes: filter and weigh spots on host
          - trigger spot termination
        - No: schedule
    - No: schedule
Outline

Introduction

FairShare Scheduling
  Synergy: an extensible manager for OpenStack
  FairShare Scheduler for OpenNebula

Preemptible Instances in OpenStack

Partition Director

Conclusions
Partition Director

- Providing a computing center or a user community with the ability to share physical resources over different interface types, such as cloud and batch, while keeping the overall user quota unchanged.
Partition Director

Functionality

- Switch role of selected physical resource from batch interface (WN) to the cloud one (CN) and vice versa
- Manage intermediate transition states, ensuring consistency

Working principles

- On each node, both batch and cloud services are active. However, they are enabled in a mutually exclusive manner.
- A Draining phase is needed before switching role.
- When a WN is assigned to the Cloud partition, batch system stops dispatching jobs to it (Draining) and when there are no more running batch jobs it becomes available (nova-compute enable) to a Cloud controller.
- When a CN is assigned to the batch partition, cloud controller disables (nova-compute disable) the node. After a known TTL limit, VMs still active are destroyed.
Introduction

FairShare Scheduling
   Synergy: an extensible manager for OpenStack
   FairShare Scheduler for OpenNebula

Preemptible Instances in OpenStack

Partition Director

Conclusions
Development Status

- Synergy, Partition Director and OPIE part of the first INDIGO-DataCloud release (August 2016).
- Code available in INDIGO-DataCloud github:
  - https://github.com/indigo-dc/synergy-service
  - https://github.com/indigo-dc/synergy-scheduler-manager
  - https://github.com/indigo-dc/dynpart
  - https://github.com/indigo-dc/one-fass
  - https://github.com/indigo-dc/opie
  - https://github.com/indigo-dc/python-openstackclient
  - https://github.com/indigo-dc/python-novaclient
- Launchpad pages:
  - https://launchpad.net/synergy-service
  - https://launchpad.net/synergy-scheduler-manager
Outlook and Conclusions

• INDIGO-DataCloud is overcoming cloud management frameworks scheduling gaps with three different complementary approaches.
• The three solutions can be used separately to get each individual functionality or they can be further combined to get more advanced features.
• Aim to be non-invasive with existing deployments, or integrated into upstream projects whenever possible.
• INDIGO-Datacloud CHEP posters:
  – A Fair Share Scheduler for OpenNebula
  – Optimizing the resource usage in Cloud based environments: the Synergy approach
Questions?

INDIGO-DataCloud Web: http://indigo-datacloud.eu