



HUAWEI CLOUD PROJECT

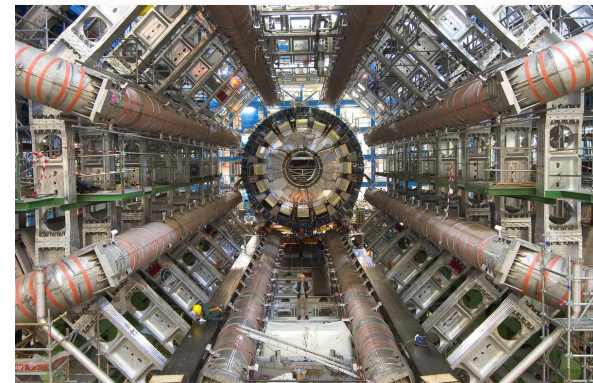
THEODOROS TSIOUTSIAS
theodoros.tsioutsias@cern.ch
SURYA SEETHARAMAN
surya.seetharaman@cern.ch

12/01/2018

PROJECTS

Currently there are two projects

- *Preemptible Instances*
 - Theodoros Tsioutsias
- *Migration from Cells_v1 to Cells_v2*
 - Surya Seetharaman



Openstack at CERN

Some Interesting Numbers

Statistics from Scaling Point of View

Compute Nodes	~ 8,500
Cores	~ 250,000
VMs	~ 33,000
Users	~ 3,100
Projects	~ 3,800
Images	~ 3,800
Volumes	~ 4,800

Preemptible Instances

Introduction

Cloud Computing gives the illusion of infinite capacity

- Quota System:

1. Sets limits on resources
2. Ensures everyone makes use of their fair share of the resources

- Operators use **quotas** per project to:

1. Prevent system capabilities from being exhausted without notification
2. Manage the resource allocations
3. Avoid “Over-committing” resources
4. Reserving Resources for operations with higher priority

Challenges

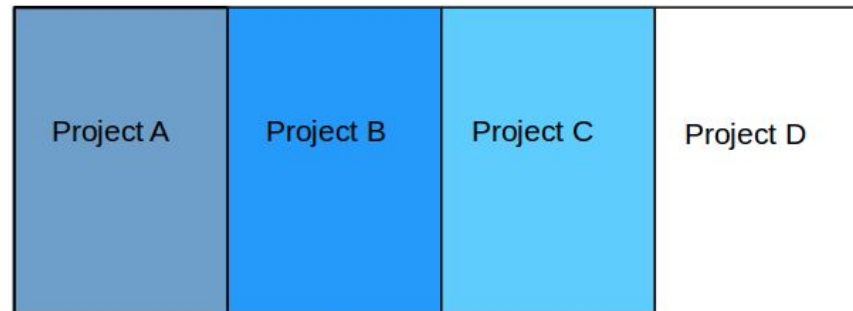
Keeping track of the available resources

- Quotas are hard limits:

Even if there are free resources, they cannot be allocated to a project whose quota is exceeded

- This leads to a reduction in cloud utilization:

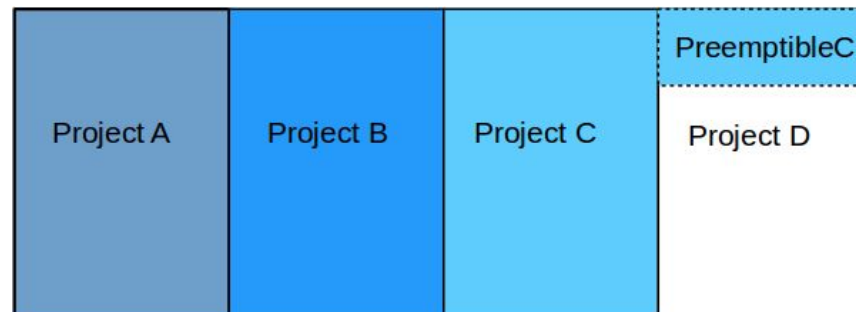
There are available resources in idle state!



*Project D uses no resources
25% of the resources are idle*

Solution

- Introducing the concept of **Preemptible Instances**:
 - created even after the quota for a project is exceeded
 - use idle resources
 - terminated as soon as the resources are requested for higher priority tasks
- The result of this:
 - handling the demand for extra resources
 - increasing the cloud utilization
 - maintaining the fair sharing of the infrastructure



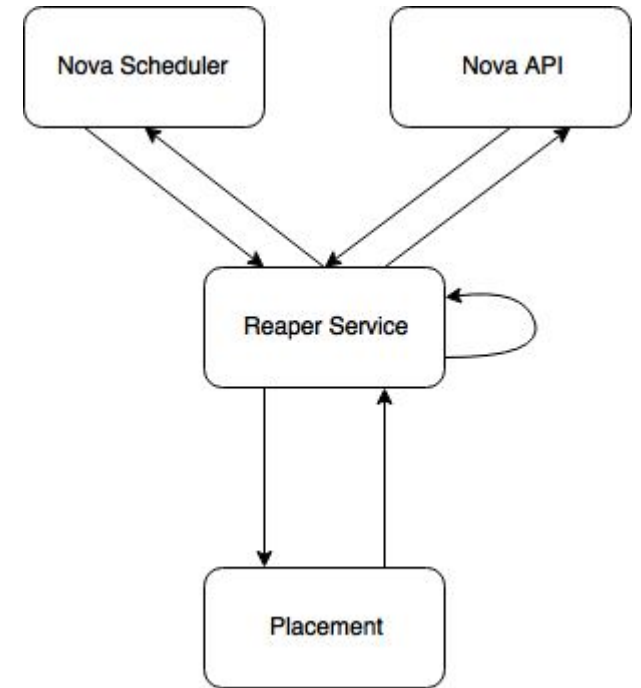
*Idle resources can be used and
they are freed as soon as they are needed*

Project Status

- Implementing a prototype service as decided by Nova Team
- Exploring different approaches
- Preparation for the Dublin PTG (February 2018)
- Upstream contributions

Next Steps:

- Upstream implementation
- Deploy Compute Intensive Services using **preemptible** servers (e.g. Batch Processing)



NOVA

an OpenStack Community Project



openstack™

Migration to Cells_v2

Compute Service is scaled using Nova Cells

- What are Nova Cells ?

1. A functionality to scale the OpenStack compute cloud
2. Hosts in the compute cloud are divided into groups i.e cells
3. Each cell has its own DB and MQ.
4. Cells are in a tree-like structure having a top 'api' cell and child cells

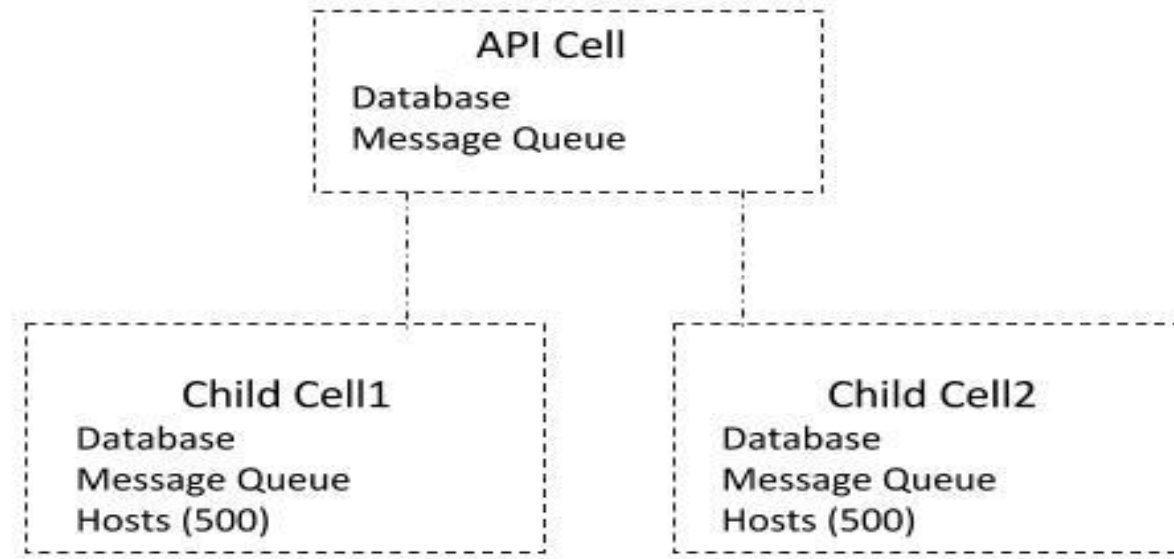
- Why do we use Cells ?

1. A strategy for scaling
2. Failure Isolation
3. Provides Elasticity/Flexibility

CERN Cells

Existing Infrastructure - Currently on Nova Cells_v1

- Two Data Centers; One top level cell and ~62 child cells
- A simplified version of Cells_v1 concept is shown below



★ Note: What we have at CERN is of course more complicated

Limitations of Cells_v1

Upstream development has stopped for v1

- Lacks certain features like:
 - Host aggregates
 - Security Groups
 - Availability Zones
- Top level cell scheduler has limitations
- Race Conditions since it's not built in a core manner
- Not the default architecture - difficulty in development
- No path for upgrade from no cells to cells_v1

Nova Cells_v2

Completely integrated with the whole nova code - it's in the main code path

- All data in one place, no duplication
- This is the default/only way to deploy nova
- Single scheduler - knows about all nodes
- Basically overcomes all the issues with Cells_v1

Project Status :

- Contributing to upstream cells_v2 development
- Preparation for the Dublin PTG (February 2018)



QUESTIONS?

theodoros.tsioutsias@cern.ch
ttsiouts @ freenode

surya.seetharaman@cern.ch
tssurya on IRC