Moving from CellsV1 to CellsV2 at CERN

OpenStack Summit - Vancouver 2018
Belmiro Moreira
belmiro.moreira@cern.ch  @belmiromoreira
## Cloud resources

<table>
<thead>
<tr>
<th>Category</th>
<th>Used</th>
<th>Available</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cores</td>
<td>299.9 K</td>
<td>316.6 K</td>
</tr>
<tr>
<td>RAM</td>
<td>829.9 TiB</td>
<td>924.6 TiB</td>
</tr>
<tr>
<td>Disk</td>
<td>10.0 PiB</td>
<td>15.3 PiB</td>
</tr>
</tbody>
</table>

## Openstack services stats

<table>
<thead>
<tr>
<th>Service</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Users</td>
<td>3264</td>
</tr>
<tr>
<td>Projects</td>
<td>4296</td>
</tr>
<tr>
<td>VMs</td>
<td>37067</td>
</tr>
<tr>
<td>Magnum clusters</td>
<td>260</td>
</tr>
<tr>
<td>Hypervisors</td>
<td>9113</td>
</tr>
<tr>
<td>Images</td>
<td>2843</td>
</tr>
<tr>
<td>Baremetal nodes</td>
<td>1147</td>
</tr>
<tr>
<td>Volumes</td>
<td>5729</td>
</tr>
<tr>
<td>Volume size</td>
<td>1.77 PiB</td>
</tr>
<tr>
<td>Fileshares</td>
<td>103</td>
</tr>
<tr>
<td>Fileshares size</td>
<td>191 TiB</td>
</tr>
</tbody>
</table>

## Resource overview by time

- **VM changes**
- **Total VMs**
- **Average VM boot time**
- **Projects and users**
- **Hypervisors**
- **Magnum clusters**
Cells at CERN

- CERN uses cells since 2013
- Why cells?
  - Single endpoint. Scale transparently between different Data Centres
  - Availability and Resilience
  - Isolate failure domains
  - Dedicate cells to projects
  - Hardware type per cell
  - Easy to introduce new configurations
Cells at CERN

● Disadvantages
  ○ Unmaintained upstream
  ○ Only few deployments using Cells
  ○ Several functionality missing
    ■ Flavor propagation
    ■ Aggregates
    ■ Server groups
    ■ Security groups
    ■ ...

CellsV1 architecture at CERN

- Nova API Servers
- TOP Cell controllers
- CellA controller
- CellA compute nodes
- CellB controller
- CellB compute nodes
- nova DB
- RabbitMQ
CellsV1 architecture at CERN (Newton)
Journey to CellsV2
Before Ocata Upgrade

- Enable Placement
  - Introduced in Newton release
  - Required in Ocata
  - nova-scheduler runs per cell in cellsV1

- How to deploy Placement with cellsV1 in a large production environment?
  - Placement retrieves the allocation candidates to the scheduler
  - Placement is not cell aware
  - Global vs Local (in the Cell)
    - Global: scheduler gets all allocation candidates available in the cloud
    - Local: scheduler gets only the allocation candidates available in the cloud
Setup Placement per cell

- Create a region per cell
- Create a placement endpoint per region
- Configure a “nova_api” DB per cell
- Run a placement service per cell in each cell controller
- Configure the compute nodes of the cell to use the cell placement
Enable placement per cell

- Issues
  - “build_requests” not deleted in the top “nova_api”
  - [https://review.openstack.org/#/c/523187/](https://review.openstack.org/#/c/523187/)
- Keystone needs to scale accordingly
CellsV1 architecture with local placement

- **API nodes**
  - nova-api

- **top cell controller**
  - nova-novncproxy
  - nova-consoleauth
  - nova-cells

- **child cell controller**
  - nova-api
  - nova-conductor
  - nova-scheduler
  - nova-network
  - nova-cells
  - RabbitMQ
  - placement-api

- **compute node**
  - nova-compute

- **RabbitMQ cluster**
  - nova_api DB
  - nova DB

- **nova_api DB**

- **nova DB**

Upgrade to Ocata

- **Data migrations**
  - flavors, keypairs, aggregates moved to nova_api (Top cell DB)
  - migrate_instance_keypairs required to run in cells DBs
    - However keypairs only exist in Top cell DB
    - [https://bugs.launchpad.net/nova/+bug/1761197](https://bugs.launchpad.net/nova/+bug/1761197)
    - Migration tool that populates cells “instance_extra” table from “nova_api” DB
  - No data migrations required in cells DBs
  - “db sync” in child cells fails because there are flavors not moved to nova_api (local)

- **DB schema**
  - migration 346 can take a lot of time (remove 'schedule_at' column from instances table)
    - consider archive and then truncate shadow tables
  - “api_db sync” fails if cells not defined even if running cellsV1
Upgrade to Ocata

● Add cells mapping in all “nova_api” DBs
  ○ cell0 (will not be used) and Top cell
  ○ Other cells mapping are not required

● “use_local” removed in Ocata
  ○ Changed nova-network to continue to support it!

● Inventory min_unit, max_unit and step_size constraints are enforced in Ocata
  ○ https://bugs.launchpad.net/nova/+bug/1638681
  ○ Problematic if not all compute nodes are upgraded to Ocata
Consolidate Placement

API nodes
- nova-api

Top cell controller
- nova-novncproxy
- nova-consoleauth
- nova-cells

Child cell controller
- nova-api
- nova-conductor
- nova-scheduler
- nova-network
- nova-cells
- RabbitMQ

Compute node
- nova-compute

RabbitMQ cluster
- nova_api DB
- nova DB

Top cell controller
- nova_api DB
- nova DB
- placement-api
Consolidate Placement

- Change endpoints to “central” placement
  - “ placement_region” and “nova_api”
  - Applied per cell (few cells per day)
    - Need to learning how to scale placement-api
  - Scheduling time expected to go up
Consolidate Placement

- Local placement disabled in all cells
  - Moved last 15 cells to “central” placement
  - Scheduler time increased
  - Placement request time also increased
Consolidate Placement

- Fell apart during the night...
- Memcached reached the “max_connections”
  - Increased “max_connections”
  - Increased the number of “placement-api” servers
Consolidate Placement

- Moved 70 local Placements to the central Placement
  - Didn’t copied the data from the local nova_api DBs
  - resource_providers, inventory and allocations are recreated
- Running on apache WSGI
- 10 servers (VMs 4 vcpus/8 GiB)
  - 4 processes/20 threads
  - Increased number of connections on “nova_api” DB
- ~1000 compute nodes per placement-api server
- memcached cluster for keystone auth_token
Scheduling time after Consolidate Placement

- Scheduling time in few cells was better than expected
- Ocata scheduler only uses Placement after all compute nodes are upgraded

```python
if service_version < 16:
    LOG.debug("Skipping call to placement, as upgrade in progress.")
```
CellsV2 in Queens

- **Advantages**
  - Finally using the “loved” code
  - Can remove all internal cellsV1 patches

- **Concerns**
  - Is someone else running cellsV2 with more than one cell?
  - Scheduling limitations
  - Availability/Resilience issues
Scheduling

- How to dedicate cells to projects?
  - No cell_filters equivalent in cellsV2

- Scheduler is global
  - Scheduler doesn’t know about cells
  - Placement doesn’t know about cells
  - Scheduler needs to receive all available allocation candidates from placement
    - https://review.openstack.org/#/c/531517/ (scheduler/max_placement_results)
  - Availability zone selection is a scheduler filter

- Can’t enable/disable scheduler filters per cell
- Can’t enable/disable a cell
  - https://review.openstack.org/#/c/546684/
Scheduling

- Placement request-filter
  - [https://review.openstack.org/#/c/544585/](https://review.openstack.org/#/c/544585/)
- Initial work already done for Rocky
- CERN backported it for Queens
- Created our own filters
  - AVZ support
  - project-cell mapping
  - flavor-cell mapping
- Few commits you may want to consider to backport to Queens
Scheduling

- Placement request-filter uses aggregates
  - Create an aggregate per cell
  - Add hosts to the aggregates
  - Add the aggregate metadata for the request-filter
  - Placement aggregates are created and resource providers mapped
    - Mirror host aggregates to placement: https://review.openstack.org/#/c/545057/

- Difficult to manage in large deployments
  - “Forgotten” nodes will not receive instances
  - Mistakes can lead to wrong scheduling
  - Deleting a cell doesn’t delete resource_providers, resource_provider_aggregates, aggregate_hosts
    - https://bugs.launchpad.net/nova/+bug/1749734
Availability

- If a cell/DB is down all cloud is affected
  - Can’t list instances
  - Can’t create instances
  - ...
- Looking back we only had few issues with DBs
  - Felt confident to move to CellsV2
- Upstream discussion on how to fix/improve the availability problem
  - https://review.openstack.org/#/c/557369/
Upgrade to Queens

- “Shutdown the cloud”
- Steps we followed for the upgrade
  - Upgrade packages
  - Data migrations / DB schema
    - Pike/Queens data migrations
      - Quotas, service UUIDs, block_device UUIDs, migrations UUIDs
    - Top cell DB will be removed
  - Create cells in nova_api DB
  - Delete current instance_mappings
  - Recreate instance_mappings per cell
  - Discover hosts
Upgrade to Queens

- Create aggregates per cell and populate aggregate_hosts, aggregate_metadata
- Create placement aggregates and populate resource_provider_aggregates
- Setup AVZs
- Enable nova-scheduler and nova-conductor services in the top control plane
- Remove nova-cells service from parent and child cells
- Remove nova-scheduler from child cells controllers
- Upgrade compute nodes

- Start the cloud
After Queens upgrade

CPU load in nova-api servers

CPU load in placement-api servers
Placement - number of requests
What changed in Placement?

- Refresh aggregates, traits and aggregate-associated sharing providers
  - ASSOCIATION_REFRESH = 5m
  - Made the option configurable:
    - Master: https://review.openstack.org/#/c/565526/
    - Backported to Queens: https://review.openstack.org/#/c/566288/
      - Set it to a very large value

- However it still runs when nova-compute restarts
  - Problematic with Ironic
  - At the end we removed this code path
Placement - number of requests
Placement

- Doubled the number of placement-api nodes
  - ~500 compute nodes per placement-api server
- In average request time < 100ms
Nova API request time
Database load pattern

- Number of queries in Cell DBs more than double after the upgrade
  - APIs only available to few users
- Connection rate increased
  - Clients could not connect. API calls failed
  - Reviewed DB configuration. Related with ulimits of mysql processes
Nova list / Nova boot

- To list instances the request goes to all cells DBs
  - Problematic if a group of DBs is slow or has connection issues
  - Fails if a DB is down
- DBs for Wigner data centre cells are located in Wigner
  - API servers are located in Geneva
- To minimize the impact deployed few patches
  - Nova list only queries the cells DBs where the project has instances
    - [https://review.openstack.org/#/c/509003](https://review.openstack.org/#/c/509003)
  - Quota calculation only queries the cells DBs where the project has instances
    - [https://bugs.launchpad.net/nova/+bug/1771810](https://bugs.launchpad.net/nova/+bug/1771810)
Minor issues

- Availability zones in api-metadata
  - [https://bugs.launchpad.net/nova/+bug/1768876](https://bugs.launchpad.net/nova/+bug/1768876)
- nova-compute (ironic) creates new resource provider when failover
  - resource_provider_aggregate lost
  - [https://bugs.launchpad.net/nova/+bug/1771806](https://bugs.launchpad.net/nova/+bug/1771806)
- Scheduler host_manager gathering info
  - Makes it parallel. Ignore cells down: [https://review.openstack.org/#/c/539617/](https://review.openstack.org/#/c/539617/)
- Service list
  - Not parallel. Fails if a cell is down: [https://bugs.launchpad.net/nova/+bug/1726310](https://bugs.launchpad.net/nova/+bug/1726310)
- nova-network doesn’t start when using cellsV2
Today - metrics
CellsV2 architecture at CERN (Queens)
Summary

CERN cloud is running Nova Queens with CellsV2

- Moving from CellsV1 is not a trivial upgrade
- CellsV2 works at scale
- Availability/Resilience issues

Thanks to all Nova Team!