HELLO!

OpenInfra SUMMIT › VANCOUVER '23

PERFORMANCE TUNING AND RESOURCE MANAGEMENT FOR VARIANT DEDICATED AND SHARED-CORE FLAVORS

Khomkrit Viangvises

PRINCIPLE OPENSTACK ENGINEER

Charnsilp Chinprasert

CHIEF INNOVATION OFFICER

NIPA Cloud Team

WHO ARE WE?
WE ARE PIONEERS OF OPENSTACK IN THAILAND.

>> WHO ARE WE?

Charnsilp Chinprasert

- NIPA Cloud Creator & Product manager
- Initiate innovative product of Nipa Cloud
- Strong OpenStack & Ceph Specialist
- Over 5 years experiences in OpenStack
- Understand OpenStack in code level

Certification
Certified OpenStack Administrator (COA)
Certified Kubernetes Administrator (CKA)

>> WHO ARE WE?

Khomkrit Viangvises

- Develop new features for NIPA OpenStack Public Cloud
- The most experience OpenStack Software in Thailand
- Over 7 years experiences in OpenStack

Certification

Mirantis Certified Administrator for OpenStack (MCA200)
Red Hat Certified System Administrator
Red Hat Certified System Administrator in Red Hat OpenStack

>> TOPICS TO TALK ABOUT

Our Public Cloud Design

public cloud with demand from variant requirements such as database, dev server

Demand of variant requirements

public cloud with demand from variant requirements such as database, dev server

Shared Core tuning guide

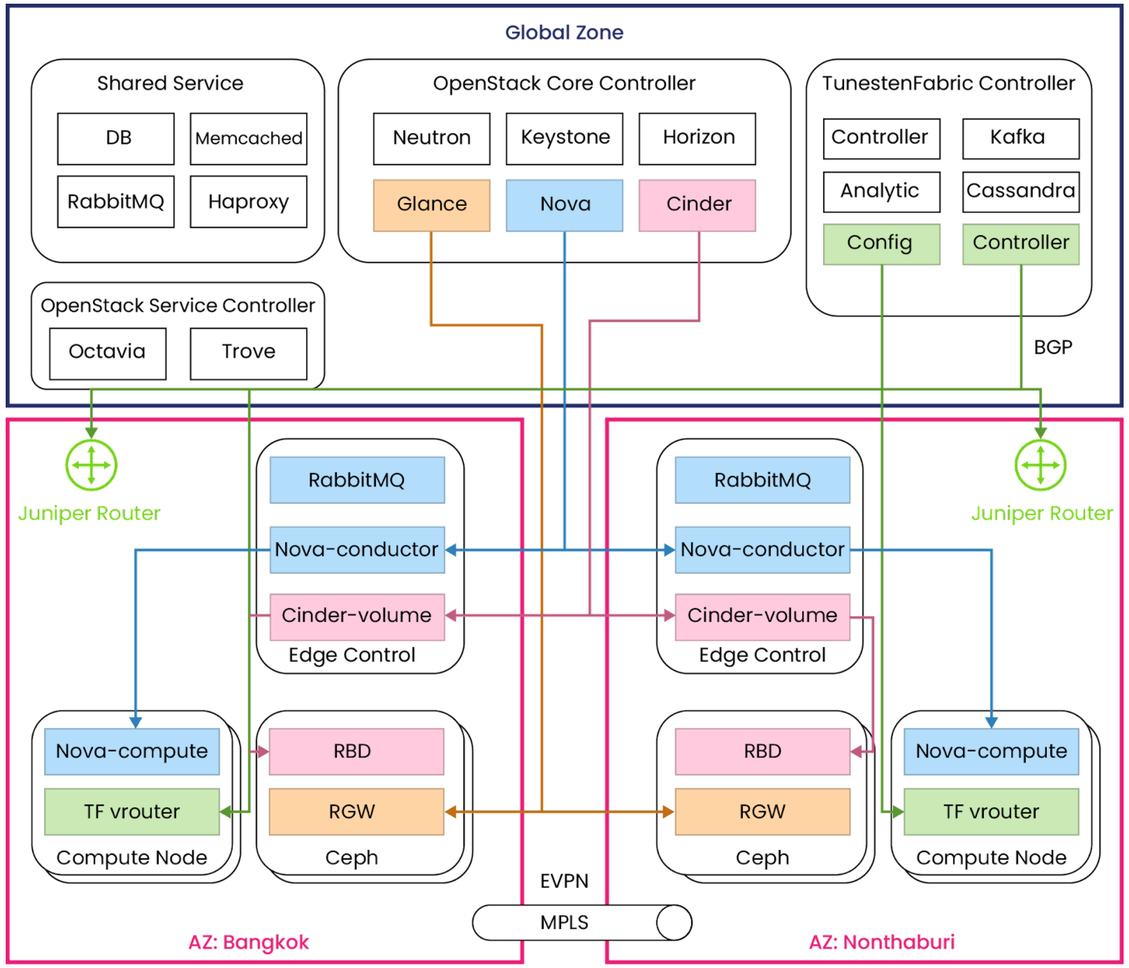
CPU Quota limit

Dedicated Core tuning guide

CPU pinning, Huge page, Resource Classes, Inventories, Allocations and Traits

Conclusion & further work

Our Public Cloud Design



CLOUD SOFTWARE

- OpenStack Victoria
- Ceph Octopus
- Tungsten Fabric

AVAILABILITY ZONE

- Multi-AZ
- DWDM between DCs

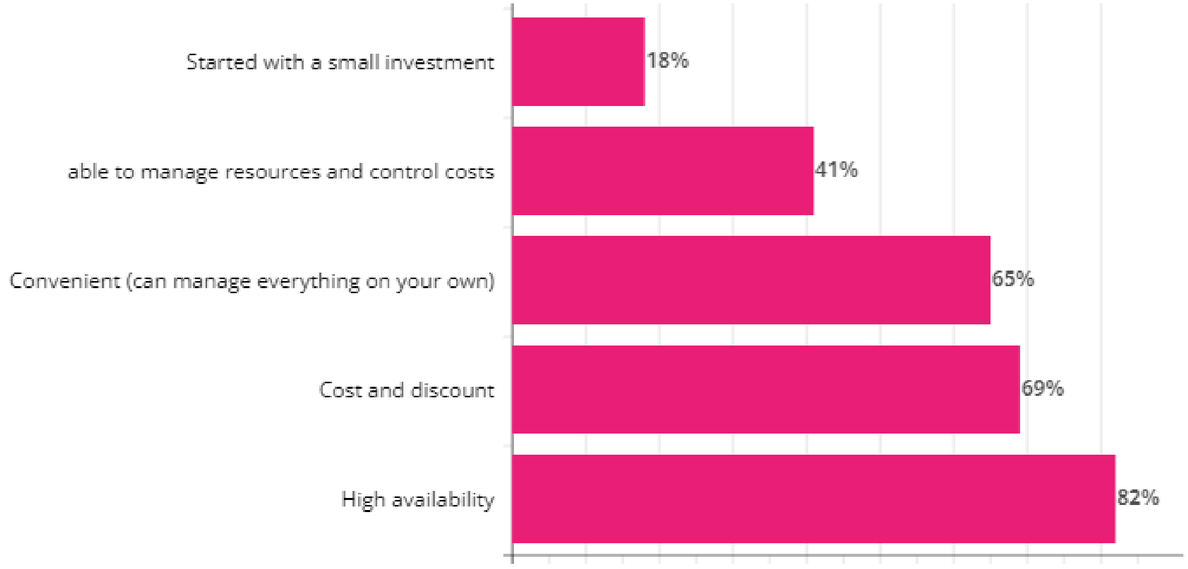
4 OpenStack Services

1. Instances (Nova)
2. Volume (Cinder)
3. Image (Glance)
4. Network (Neutron, Tungsten Fabric)

Demand of Variant Customers

1. HIGH RELIABILITY (HA)

No need to manage the infrastructure yourself, whereas previously a server was required but there is no HA. Cloud technology gives you better performance also making it easier to scale.



Based on the customer database of NIPA Cloud (N > 10,000 customers)

2. COST AND DISCOUNTS

when planning for annual payments huge discount will be offers.

3. CONVENIENT

Capable of handling everything according to your needs.

4. SELF SERVICE

Able to manage resources and control cost

5. PAY AS YOU GO

no need for upfront investment or pay-as-you-go model.

Customer requirements

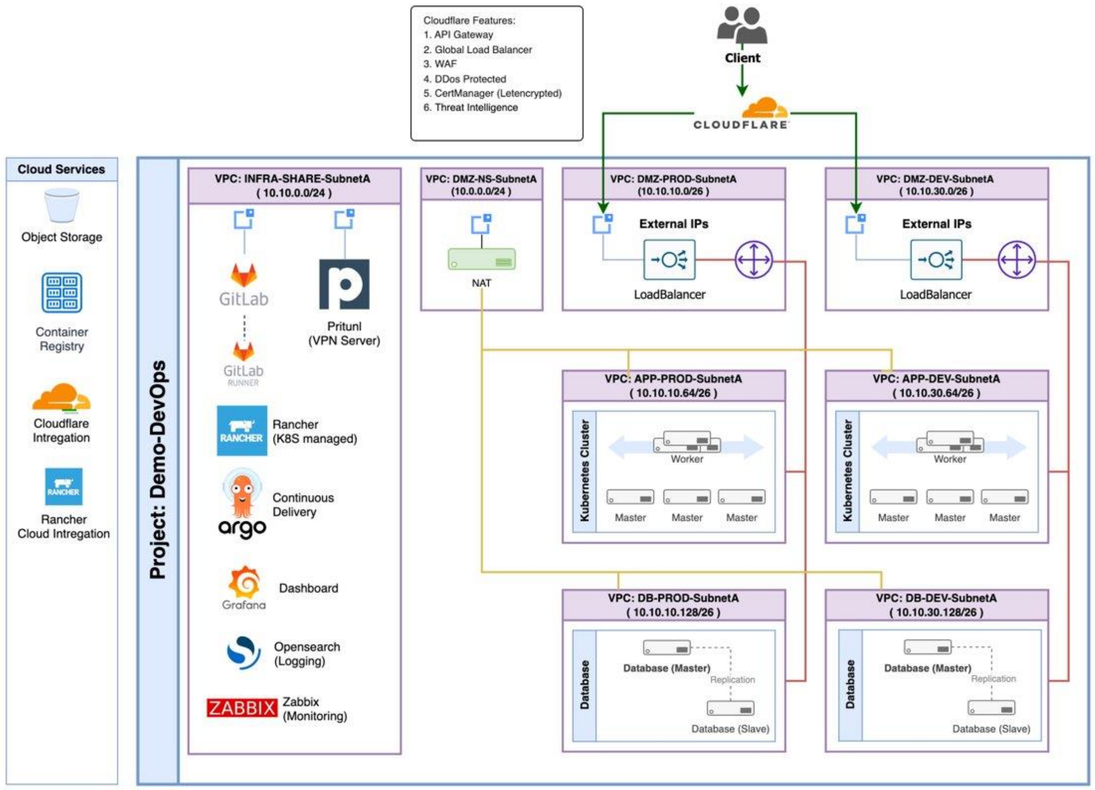
COST-EFFECTIVE

- Low traffic web servers
- CMS / Blogs
- Dev/Test servers
- Small Databases
- Microservices
- Repository

PERFORMANCE

- High performance SQL / NoSQL
- In-memory caches and indexes
- Real-time big data processing
- Mission-critical applications, like JVM
- CI/CD
- Video encoding
- Batch processing
- Monitoring and analytics software
- E-commerce site

Customer case:



CUSTOMER REQUIREMENT

- Optimize cost of non-mission critical service.
- Performance of NFV and K8S production

SHARED CORE FLAVORS

- K8S development cluster
- Infra server: logging, Git server, Opensearch and monitoring

DEDICATED CORE FLAVORS

- K8S Production cluster
- Database nodes
- Nat gateway
- VPN site-to-site

Design Compute node

COST-EFFECTIVE

- Low-medium workload
- Burstable CPU
- Lower cost
- Overcommitting CPU and RAM

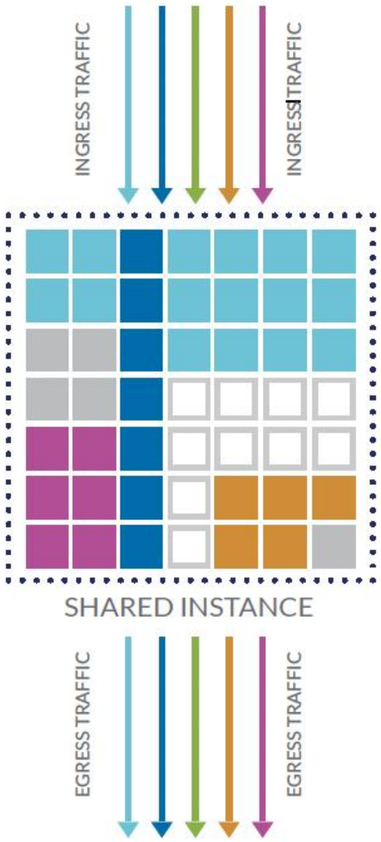
SHARED-CORE FLAVORS

PERFORMANCE

- Medium-High workload
- Sustained CPU Performance
- Minimize CPU and RAM latency
- Dedicated CPU and RAM

DEDICATED-CORE FLAVORS

Resource Type



SHARED-CORE FLAVORS

- Cost-saving
- Dev environment
- CI/CD
- Compare to on-premise or VMware

DEDICATED-CORE FLAVORS

- Performance stability
- Reducing memory latency
- Production environment
- Compare to Hyper scaler Cloud

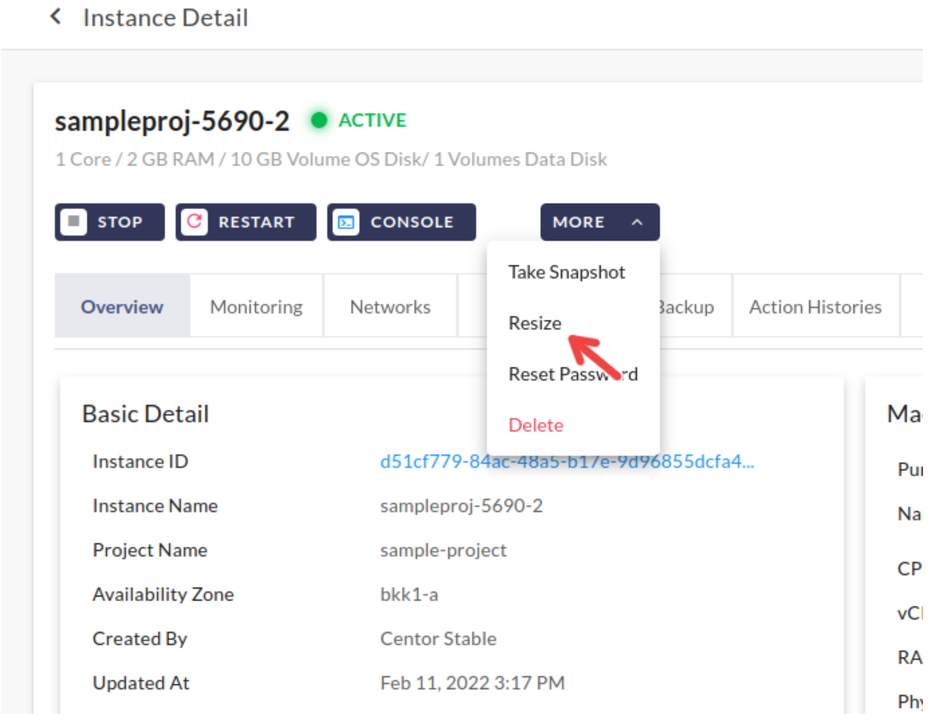
Performance and Price Ratio

Region	Asia Pacific (Singapore)							
Machine type	t4g	t3	t3a	M6g	M6i	M5	M5d	M5a
CPU model	64-bit Arm Neoverse cores	Intel Xeon Scalable processor (Skylake 8175M or Cascade Lake 8259CL)	AMD EPYC 7000 series processor (7571)	64-bit Arm Neoverse cores	3rd gener-ation Intel Xeon Scalable processors (Ice Lake 8375C)	Intel Xeon Scalable processor (Skylake 8175M or Cascade Lake 8259CL)	Intel Xeon Scalable processor (Skylake 8175M or Cascade Lake 825	AMD EPYC 7000 series processor (7571)
CPU count	4	4	4	4	4	4	4	4
RAM	16	16	16	16	16	16	16	16
Storage type	EBS only	EBS only	EBS only	EBS only	EBS only	EBS only	NVMe SSD	EBS only
Storage (GB)	80	80	80	80	80	80	80	80
Pay-as-you-go price in USD (monthly)	133.41	163.78	147.72	149.76	184.8	184.8	215.46	167.28
Pay-as-you-go price in THB (monthly)	4,912.16	6,036.38	5,439.05	5,514.16	6,804.34	6,804.34	7,933.24	6,159.25
Average multi-core score (Geekbench 5)	2,539	1,770	1,464	2,848	2,722	2,021	1,987	1,692
Performance / price (USD)	19.03	10.81	9.91	19.02	14.73	10.94	9.22	10.11
Performance / price (THB)	0.52	0.30	0.27	0.52	0.40	0.30	0.25	0.27

PERFORMANCE / PRICE

- 4 vCPU 16 GB of RAM
- Variant CPU type
- Based on AWS Singapore region

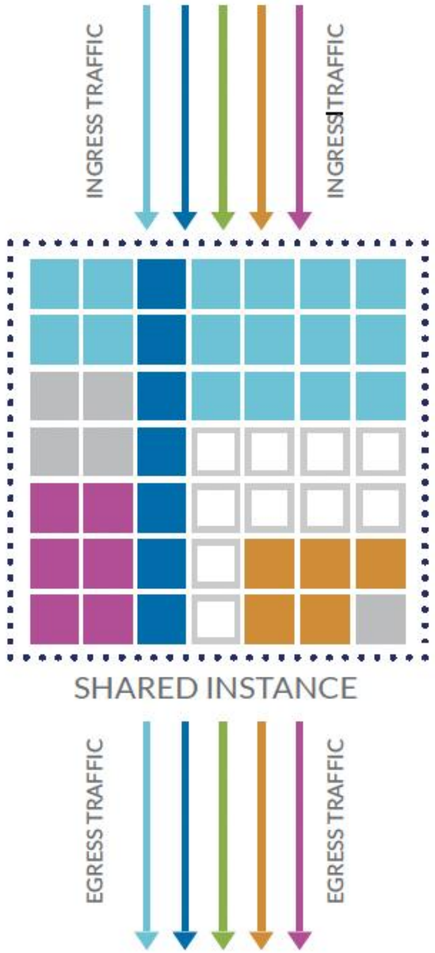
On-demand resize



SELF SERVICE RESIZE

- Customer can resize between both resource type
- Self service portal
- Optimize costs

Shared cores



OVERCOMMITTING

By default, instance vCPU processes are not assigned to any host CPU

- Overcommitting of CPUs (default 8)
- Cost optimize: Single Host core can share in many instance's vCPUs.
- Total CPU utilization may be high, then CPU power consumption per core is also high.

```
$ openstack flavor set m1.large --property hw:cpu_policy=shared
```

Limit CPU quota for Shared core

```
# openstack flavor create <name>
    --vcpu 4
    --ram 8192
    --property quota:cpu_period=100000
    --property quota:cpu_quota=50000
    --property quota:cpu_shares=50000
    --property hw:cpu_policy=shared
```

PROBLEM

- If all VMs in same shared core are high utilization.
- High congestion
- Performance not stable. Depend on how other VM CPU utilization.

CPU QUOTA

- Limit Quota of each shared vCPU
- hw:cpu_policy: shared (CPU pinning of cpu_shared_set)
- quota:cpu_period
- quota:cpu_quota
- quota:cpu_shares

Dedicated cores

CPU PINNING

CPU pinning control over how instances run on hypervisor CPUs and the topology of virtual CPUs available to instances.

- Minimize latency and maximize performance.
- Pinning virtual CPUs to Hypervisor CPUs

```
$ openstack flavor set m1.large \
  --property hw:cpu_policy=dedicated \
  --property hw:cpu_thread_policy=require
```

Dedicated cores

```
<memory unit='KiB'>33554432</memory>
<currentMemory unit='KiB'>33554432</currentMemory>
<memoryBacking>
  <hugepages>
    <page size='1048576' unit='KiB' nodeset='0'/>
  </hugepages>
</memoryBacking>
<vcpu placement='static'>8</vcpu>
<cputune>
  <shares>8192</shares>
  <vcpupin vcpu='0' cpuset='88'/>
  <vcpupin vcpu='1' cpuset='184'/>
  <vcpupin vcpu='2' cpuset='162'/>
  <vcpupin vcpu='3' cpuset='66'/>
  <vcpupin vcpu='4' cpuset='180'/>
  <vcpupin vcpu='5' cpuset='84'/>
  <vcpupin vcpu='6' cpuset='56'/>
  <vcpupin vcpu='7' cpuset='152'/>
  <emulatorpin cpuset='56,66,84,88,152,162,180,184'/>
</cputune>
```

Libvirt domain XML

CPU PINNING

- Instance vCPUs pin to Hypervisor CPUs
- Nova-compute automatic choose available Hypervisor CPUs

vCPU	Hypervisor CPU
0	88
1	184
2	162
3	66
4	180
5	84
6	56
7	152

CPU mapping configuration on nova.conf

```
[DEFAULT]
cpu_allocation_ratio=8.0

[compute]
cpu_dedicated_set=2-17
cpu_shared_set=18-47
```

```
COMPUTE NODE provider
    PCPU:
        total: 16
        reserved: 0
        min_unit: 1
        max_unit: 16
        step_size: 1
        allocation_ratio: 1.0
    VCPU:
        total: 30
        reserved: 0
        min_unit: 1
        max_unit: 30
        step_size: 1
        allocation_ratio: 8.0
```

SHARED CORE

- cpu_shared_set : specific host CPUs should used for vCPU
- cpu_allocation_ratio : overcommitting of CPU

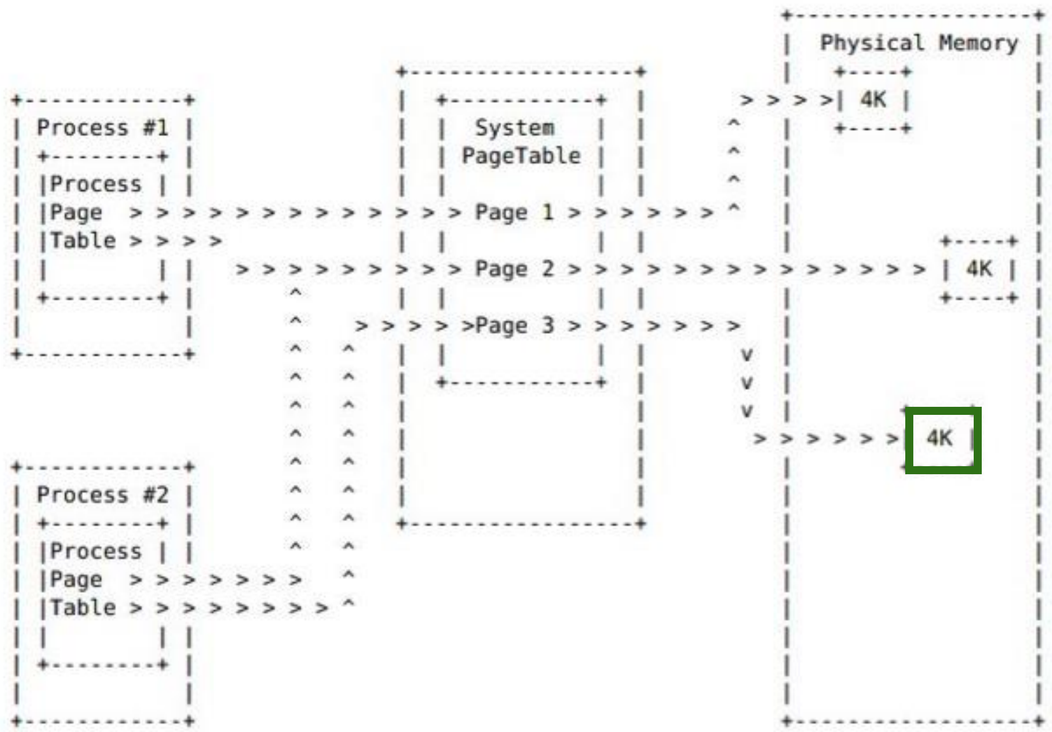
DEDICATED CORE

- cpu_dedicated_set : specific host CPUs should used for vCPU

VCPU & PCPU

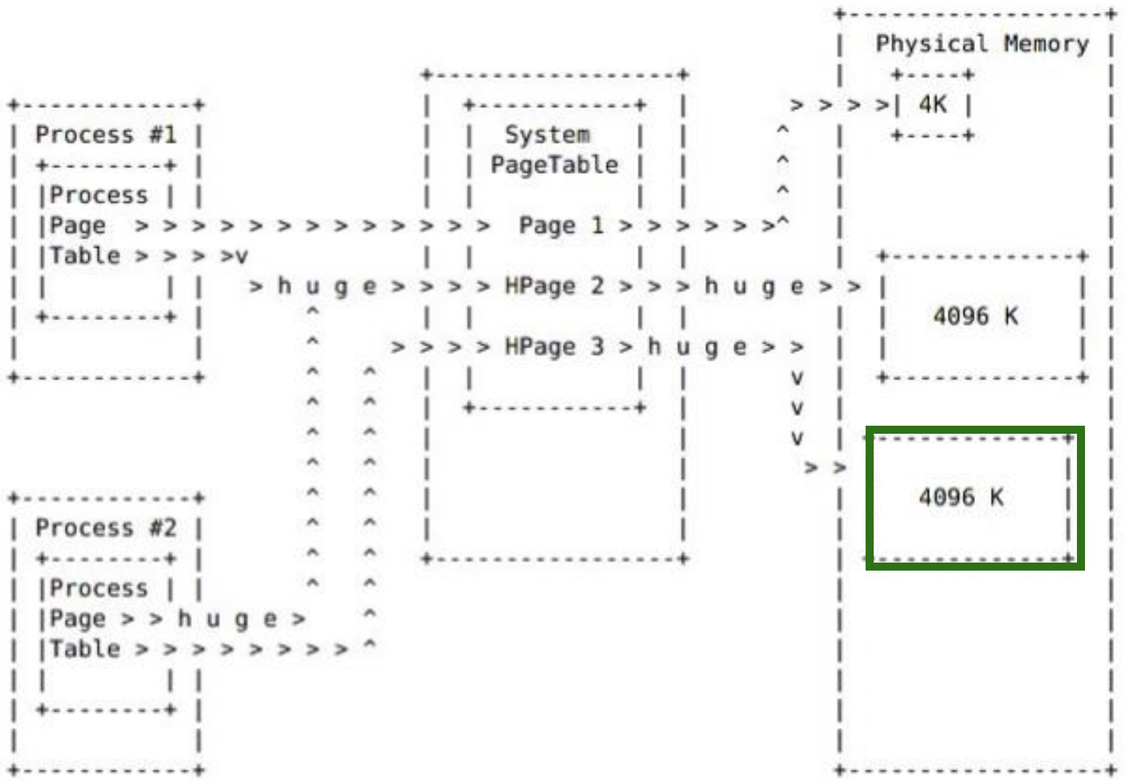
- VCPU is inventory for shared core
- PCPU is inventory for dedicated core

Hugepages



SMALL PAGE 4KB (DEFAULT)

- Example 2GB
- 2GB / 4 KB = 524,288 pages



HUGE PAGE 2MB

- Example 2GB
- 2GB / 2 MB = 1,000 pages

https://blogs.sap.com/2013/06/17/oracle-myths-and-common-misconceptions-about-transparent-huge-pages-for-oracle-databases-on-linux-uncovered/

Benefit of Hugepages

```
# cat /proc/meminfo | grep Huge
AnonHugePages:         0 kB
ShmemHugePages:        0 kB
FileHugePages:         0 kB
HugePages_Total:     470
HugePages_Free:      206
HugePages_Rsvd:        0
HugePages_Surp:        0
Hugepagesize:    1048576 kB
Hugetlb:       492830720 kB
```

HUGEPAGES

- Larger Page Size and Less # of Pages
- Reduced Page Table walking
- Less overhead of Mem operations
- No Swapping
- No 'kswapd' operations

```
$ openstack flavor set m1.large --property hw:mem_page_size=large
```

/etc/default/grub

GRUB_CMDLINE_LINUX_DEFAULT="**default_hugepagesz=1G hugepagesz=1G hugepages=470**"

Create Dedicate-core Flavors

```
# openstack flavor create <name>
    --vcpu 4
    --ram 8192
    --property hw:mem_page_size=1GB
    --property hw:cpu_policy=dedicated
```

FLAVOR PROPERTIES

- hw:cpu_policy: dedicated (CPU pinning of cpu_dedicated_set)
- hw:mem_page_size: 1GB (using Hugepage 1G)

Resource Provider Usage

PCPU

- no cpu_allocation_ratio
- Dedicate core inventory

VCPU

- cpu_allocation_ratio
- Shared core inventory

- openstack resource provider inventory list xxx

```
(osclient) root@gos-bkk-kolla-deploy1:~# openstack resource provider inventory list  39cb4c8c-a4c6-455f-a358-ab09dc966053
+------------------------+------------------+----------+----------+----------+-----------+------------+--------+
| resource_class         | allocation_ratio | min_unit | max_unit | reserved | step_size |      total |   used |
+------------------------+------------------+----------+----------+----------+-----------+------------+--------+
| VCPU                   |              8.0 |        1 |        6 |        0 |         1 |          6 |      2 |
| MEMORY_MB              |              1.0 |        1 |   515810 |    32768 |         1 |     515810 | 331776 |
| DISK_GB                |              1.0 |        1 |      437 |       40 |         1 |        437 |      0 |
| PCPU                   |              1.0 |        1 |      146 |        0 |         1 |        146 |    132 |
| MEM_ENCRYPTION_CONTEXT |              1.0 |        1 |        1 |        0 |         1 | 2147483647 |      0 |
+------------------------+------------------+----------+----------+----------+-----------+------------+--------+
```

Conclusion

- Initial requirement from customer is just how many are cores that you provide. Doesn't care the performance per cost.
- Help customer design their architect for optimize current costs.
- CPU utilization of Shared compute node is higher of Dedicated compute node.
- Power consumption per cores of shared core will be higher than others

Further works

- Plan to upgrade nova to 2023.1 release.
- strategy to reduce CPU power consumption when unused
    - Libvirt.cpu_power_management_strategy: cpu_state / governor
    - Cpu_state can be offline / online
    - Governor can be powersave / performance

```
[compute]
cpu_dedicated_set=2-17

[libvirt]
cpu_power_management=True
cpu_power_management_strategy=cpu_state
```

THANKS!

OpenInfra
SUMMIT › VANCOUVER '23