NUMA and CPU Pinning in OpenStack

Saputro Aryulianto
ary@btech.id

OICNDI
November 7th 2020

vmware boer technology ONF

Agenda

- NUMA
- CPU Pinning
- OpenStack Use Cases

What is NUMA ?

What is NUMA ?

OICNDI

UMA

In the past, processors had been designed as Symmetric Multi-processing or Uniform Memory Architecture (UMA) machines, which mean that all processors shared the access to all memory available in the system over the single bus

NUMA

Non-Uniform Memory Access is a derivative of the SMP design that is found in many multi-socket systems. In a NUMA system, system memory is divided into cells or nodes that are associated with particular CPUs. Requests for memory on other nodes are possible through an interconnect bus.

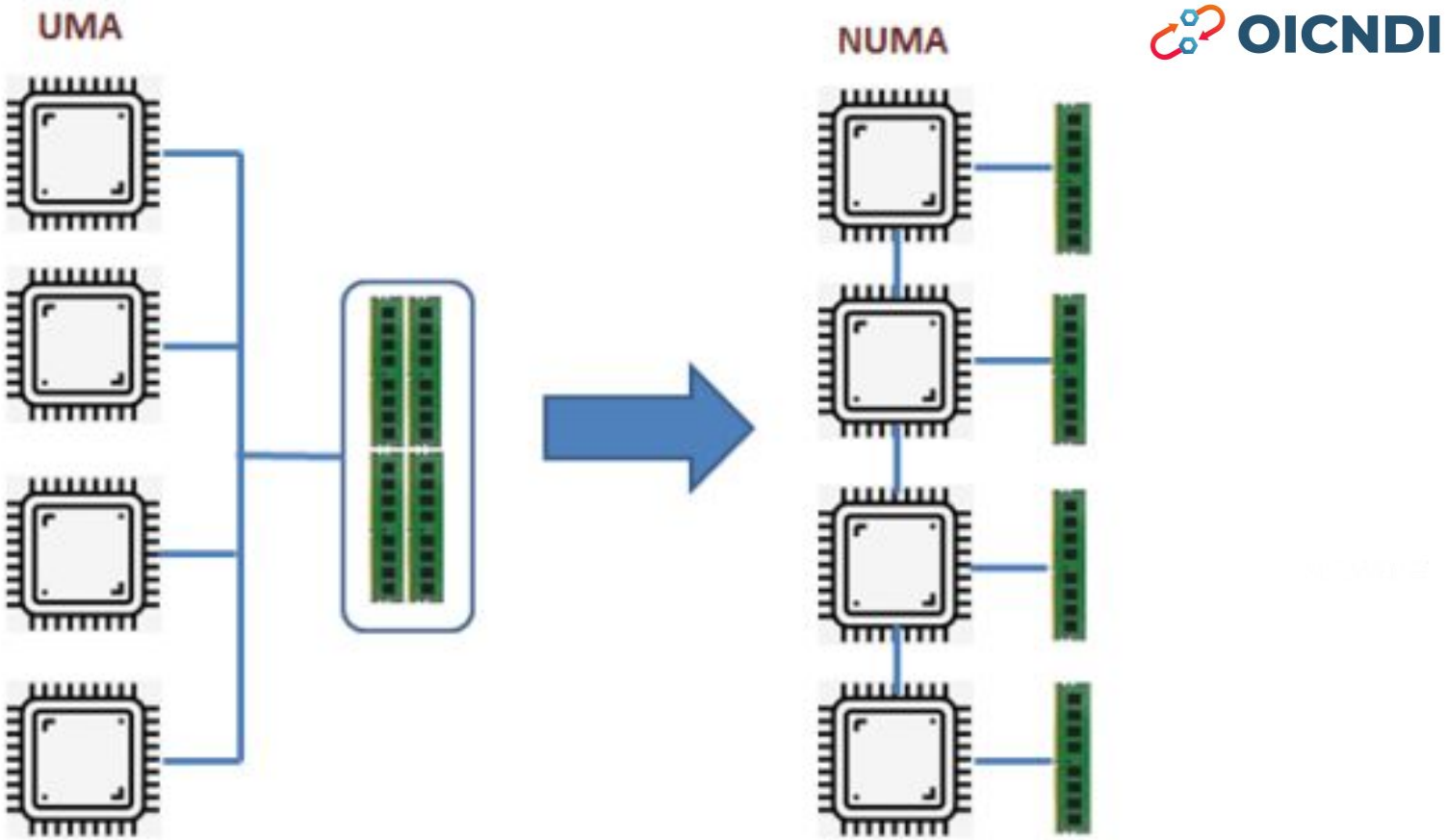


Image Source

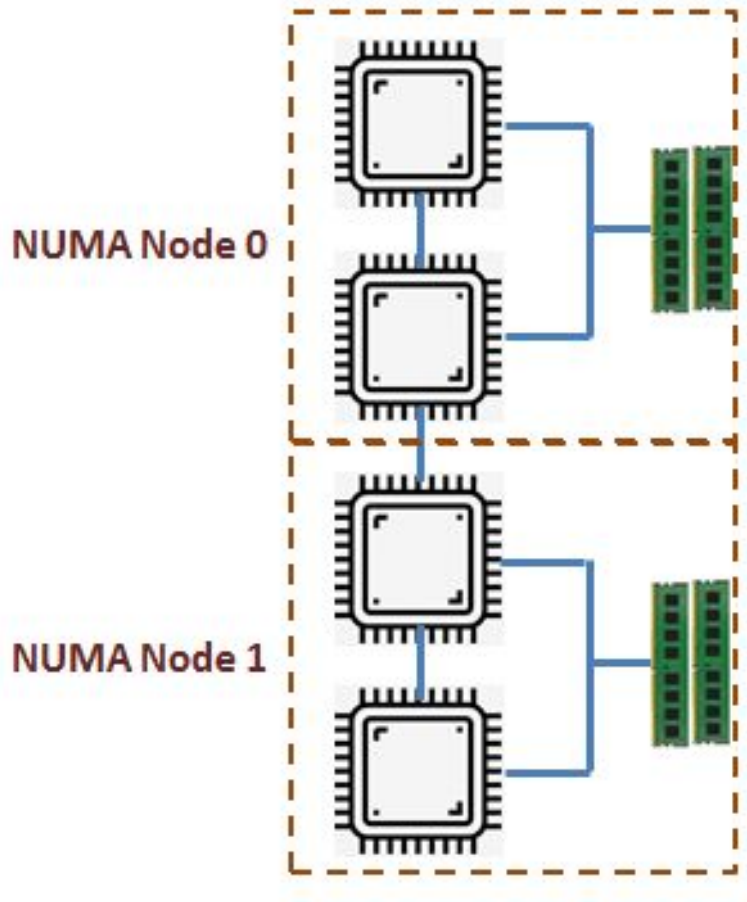
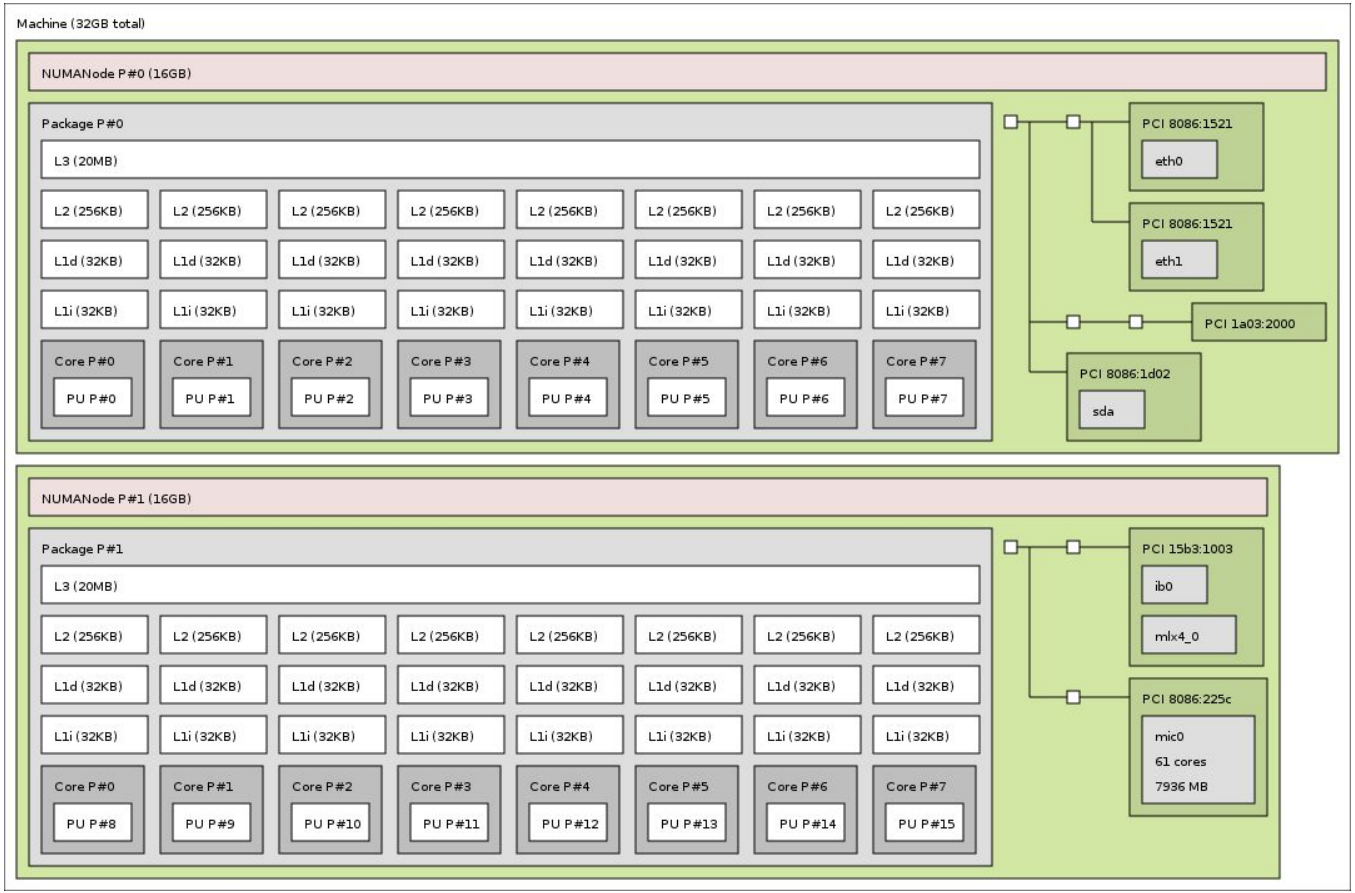
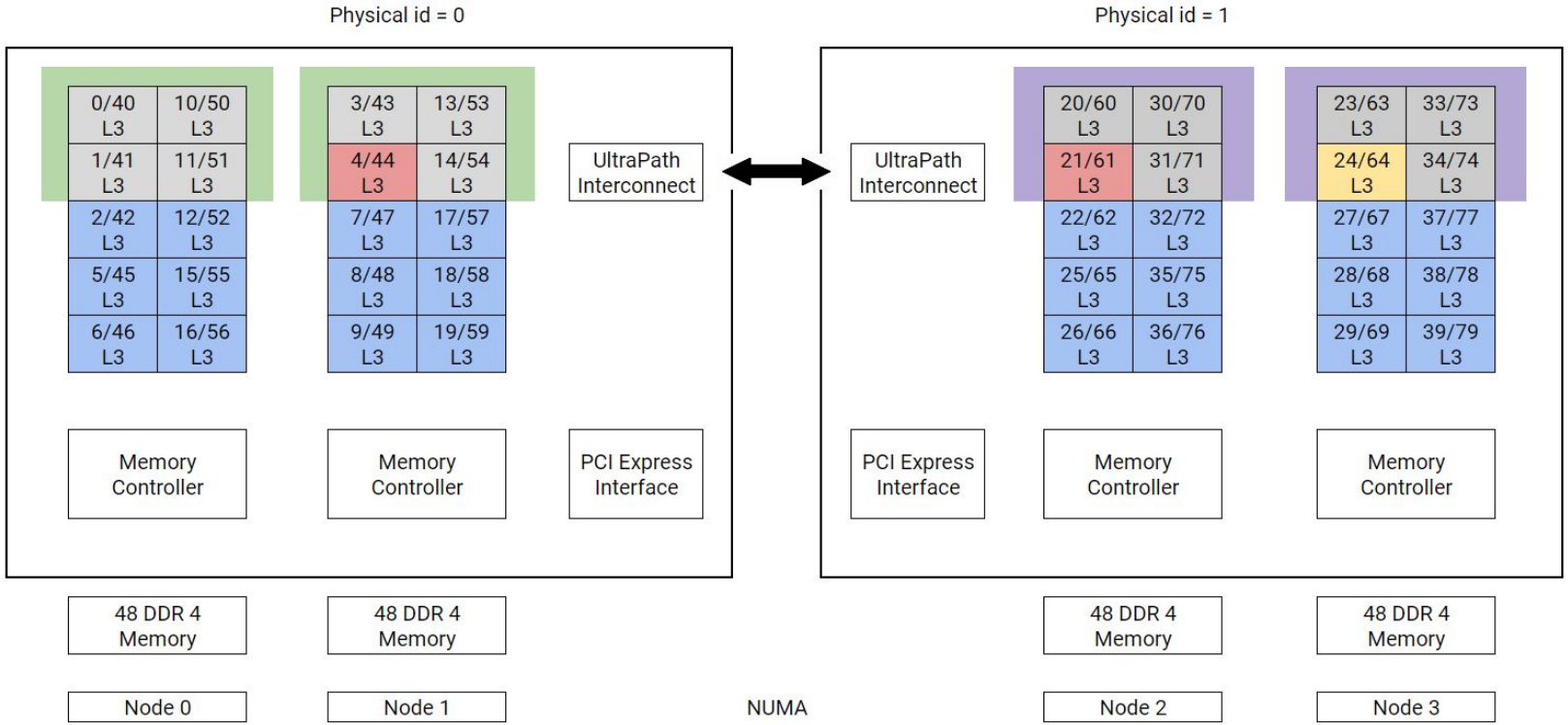


Image Source

Discover NUMA topology

- lstopo
- numactl
- lscpu


Machine (32GB total)
NUMANode P#0 (16GB)
Package P#0
L3 (20MB)
L2 (256KB)
L1d (32KB)
L1i (32KB)
Core P#0
PU P#0
Core P#1
PU P#1
Core P#2
PU P#2
Core P#3
PU P#3
Core P#4
PU P#4
Core P#5
PU P#5
Core P#6
PU P#6
Core P#7
PU P#7
PCI 8086:1521
eth0
PCI 8086:1521
eth1
PCI 1a03:2000
PCI 8086:1d02
sda
NUMANode P#1 (16GB)
Package P#1
L3 (20MB)
L2 (256KB)
L1d (32KB)
L1i (32KB)
Core P#0
PU P#8
Core P#1
PU P#9
Core P#2
PU P#10
Core P#3
PU P#11
Core P#4
PU P#12
Core P#5
PU P#13
Core P#6
PU P#14
Core P#7
PU P#15
PCI 15b3:1003
ib0
mlx4_0
PCI 8086:225c
mic0
61 cores
7936 MB


Physical id = 0
Physical id = 1
0/40 L3
10/50 L3
1/41 L3
11/51 L3
2/42 L3
12/52 L3
5/45 L3
15/55 L3
6/46 L3
16/56 L3
3/43 L3
13/53 L3
4/44 L3
14/54 L3
7/47 L3
17/57 L3
8/48 L3
18/58 L3
9/49 L3
19/59 L3
UltraPath Interconnect
UltraPath Interconnect
20/60 L3
30/70 L3
21/61 L3
31/71 L3
22/62 L3
32/72 L3
25/65 L3
35/75 L3
26/66 L3
36/76 L3
23/63 L3
33/73 L3
24/64 L3
34/74 L3
27/67 L3
37/77 L3
28/68 L3
38/78 L3
29/69 L3
39/79 L3
Memory Controller
Memory Controller
PCI Express Interface
PCI Express Interface
Memory Controller
Memory Controller
48 DDR 4 Memory
48 DDR 4 Memory
48 DDR 4 Memory
48 DDR 4 Memory
Node 0
Node 1
NUMA
Node 2
Node 3

CPU Pinning for OpenStack Perspective

- CPU Pinning is introduced with JUNO release
- Default Scheduler node does not take in consideration whether assigned to VM vCPU are from same NUMA node or different
- Scheduler doesn’t know whether process used local to CPU memory
- End user doesn’t have any visibility for underlying Hardware

Configuring CPU Pinning in OpenStack

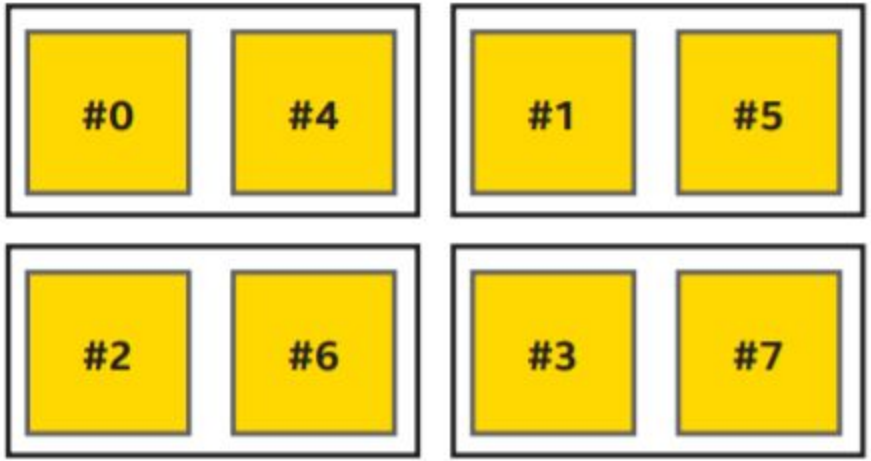
- Tell systemd about CPUAffinity. Where we will indicate those CPU's which we want to assign for Host Process; (/etc/systemd/system.conf)
- Exclude CPU's which we want to reserve for VM's from Kernel; (/etc/default/grub isolcpus)
- Configure Nova Compute configuration file for CPU pinning; (vcpu_pin_set)

Configuring CPU Pinning in OpenStack (2)

- Enabling the required scheduler filters;
- Creating a host aggregate to add all hosts configured for CPU pinning to;
- Creating a performance focused flavor to target this host aggregate; and
- Launching an instance with CPU pinning!

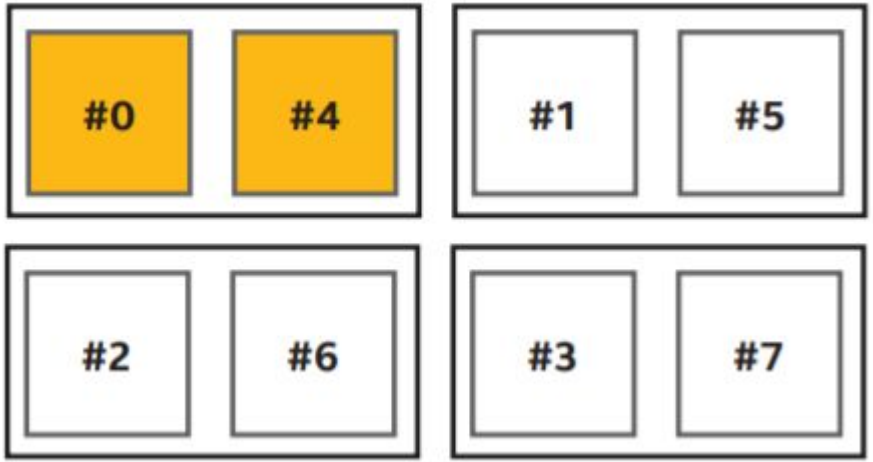
https://docs.openstack.org/nova/ussuri/admin/cpu-topologies.html

w/o CPU Pinning



Without CPU Pinning, vCPUs are “floating” across host cores.

CPU Pinning



With CPU Pinning, vCPUs are tied to pCPUs.

Benefit CPU Pinning

- It Improve the VM Performance as it is using a dedicated CPU
- It Removes the Latency because complete CPU cycle is used by a single VM
- It Maximize the CPU cache efficiency
- Separation from Noisy neighbors
- Improved memory access speed

Thank You

Sponsored by:

vmware boer technology ONF

OICNDI

November 7th 2020