ORACLE

What does vNUMA actually mean?

November 2019

Wim ten Have <wim.ten.have@oracle.com>
Software Developer / Consulting Member of Technical Staff
Oracle Linux and VM Development

Safe harbor statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions.

The development, release, timing, and pricing of any features or functionality described for Oracle’s products may change and remains at the sole discretion of Oracle Corporation.

Program agenda

1. **Virtualization – libvirt QEMU/KVM**
2. System Architecture – UMA / NUMA
3. “Host” topology – Processor Topology / NUMA Topology
4. Partitioning the NUMA host
5. vNUMA automatic host partitioning

Virtualization

Why is virtualization useful?

- Virtualization reduces the number of physical servers
- Running multiple operating systems simultaneously
- Ability to live migrate Virtual Machines without perceived downtime
- Fast Server Provisioning and Deployment
- Streamline and maximize resources

Virtualization can significantly reduce hardware and electricity costs. Most of the time, computers today only use a fraction of their potential power and run with low average system loads. A lot of hardware resources as well as electricity is thereby wasted. So, instead of running many such physical computers that are only partially used, one can pack many virtual machines onto a few powerful hosts and balance the loads between them.

Virtualization – QEMU/KVM - libvirt

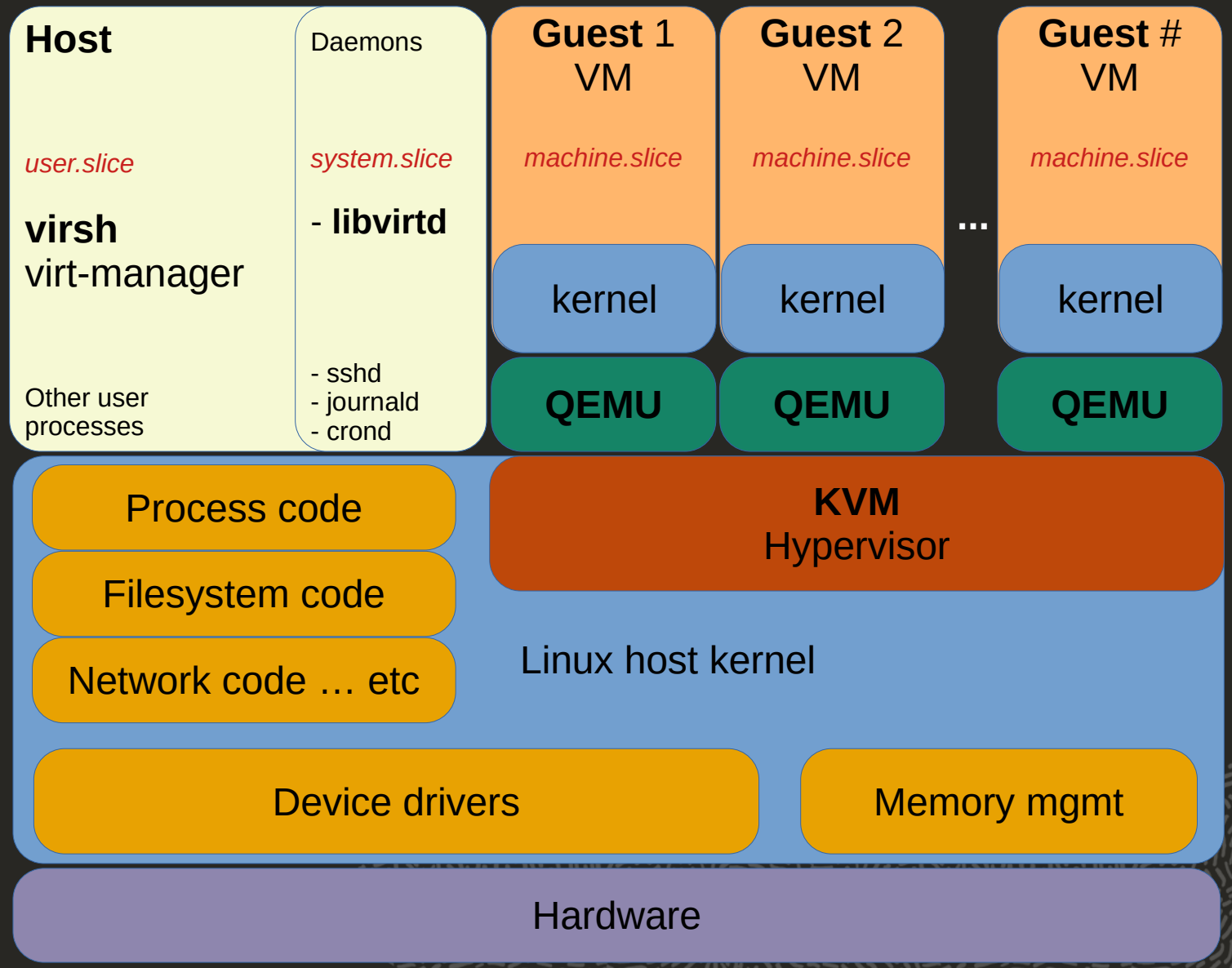
Terminology

- Kernel-based Virtual Machine is an open source virtualization technology built into Linux
  - <https://linux-kvm.org/>
- QEMU - KVM is an operational mode of QEMU for virtualization via a kernel module
  - <https://qemu.org/>
- Libvirt is a daemon ‘libvirtd’ and a command line tool ‘virsh’ and an API library
  - <https://libvirt.org/>

- “host” - the s/w part of the installation that hosts the Linux KVM system
- “guest” - a virtual machine instance running a guest operating system under the “host”

Libvirt QEMU/KVM

Components

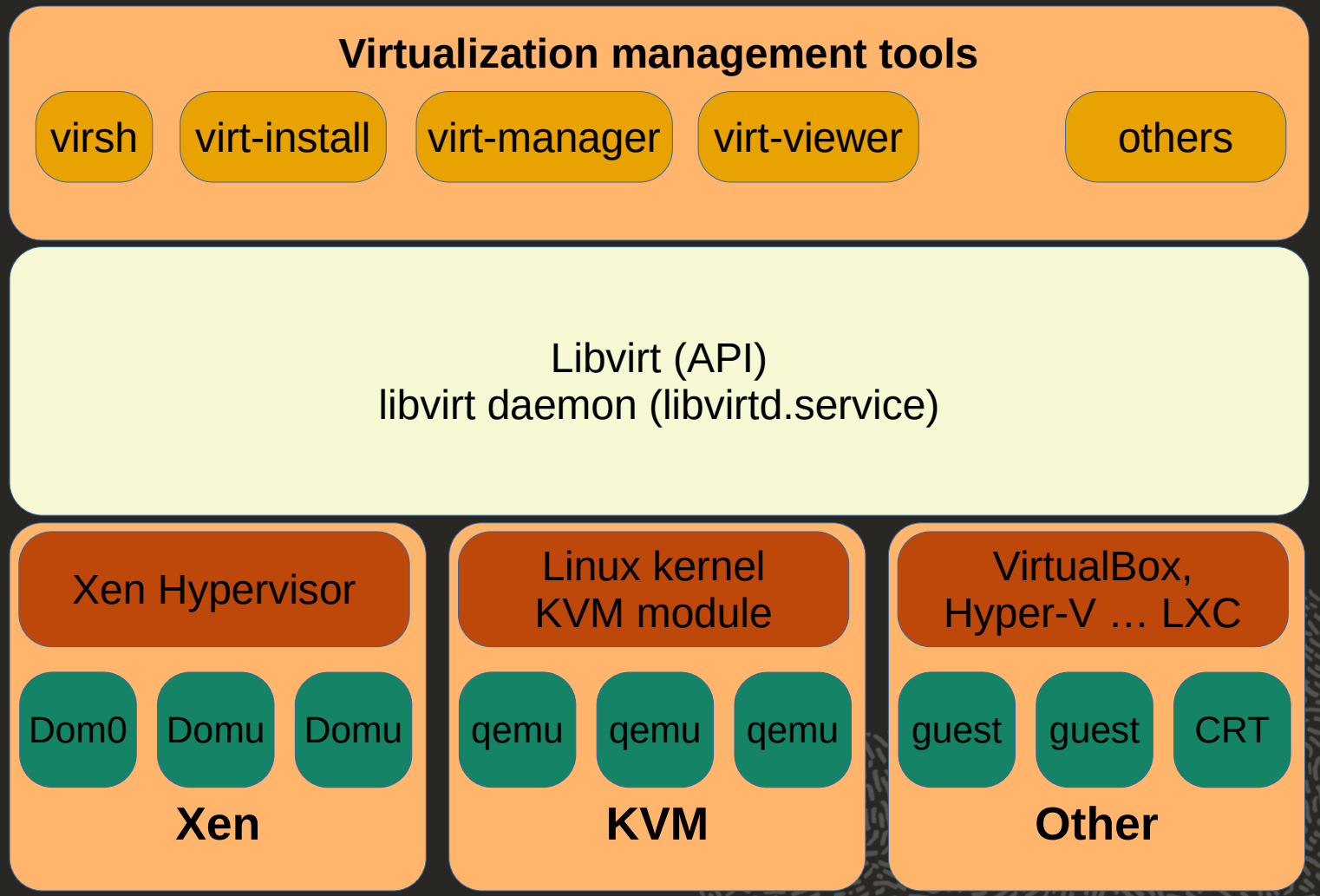


Libvirt

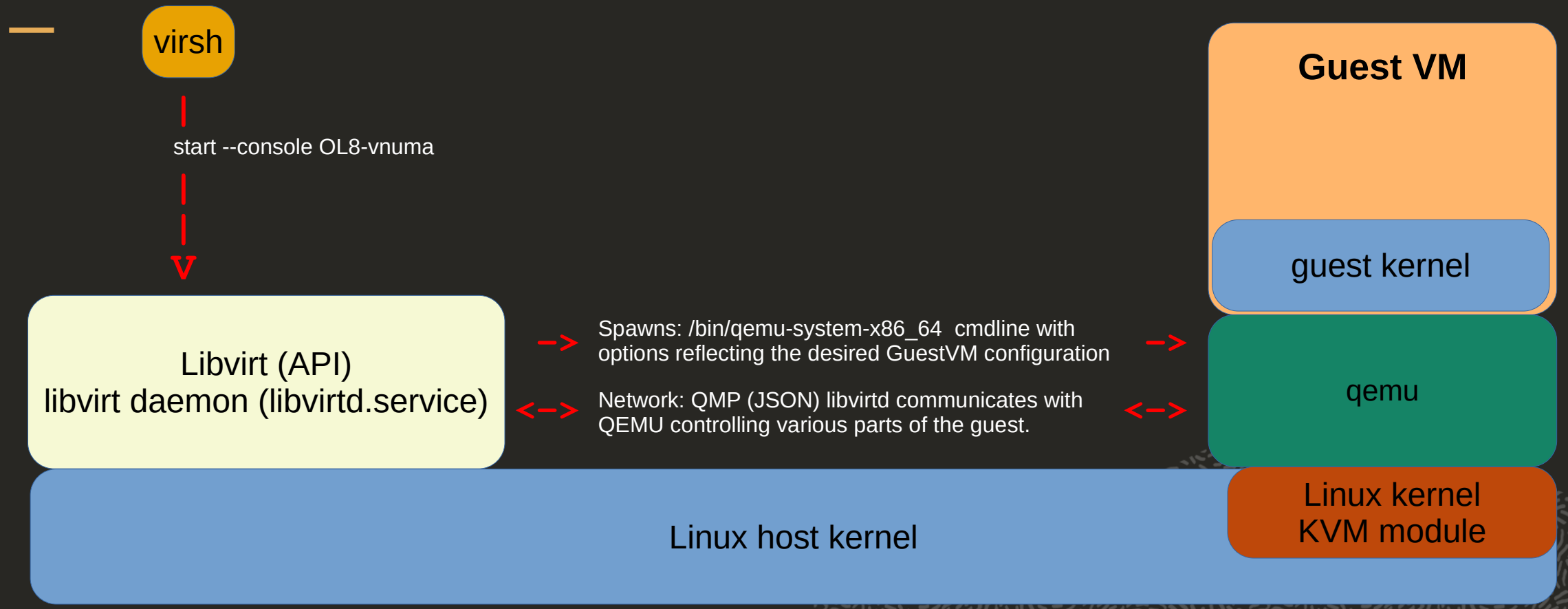
A high-level toolstack to manage virtualization platforms

- Supports multi-hypervisor KVM, QEMU, Xen, VMWare ESX, LXC ...
- Providing access through a single API
- Is Accessible from C, Python, Perl, Java ...
- Open Source
- Runs as a Linux platform daemon providing the libvirtd.service
- Facilitates serial console “guest” domain services
- Facilitates a variety of virtual graphical console “guest” domain services
- Configuration, Life cycle management
- Guests are described in XML

Libvirt overview



Libvirt overview



Libvirt

Component overview

User components

- virsh - VIRtual SHell, management CLI for virtual machines
- virt-install – CLI to provision new virtual machines
- virt-manager – GUI to provision and manage virtual machines
- virt-viewer – Graphical console for virtual machines
  - <https://virt-manager.org/>

System daemons

- libvirtd – management daemon, orchestrating the “guest” domains
- virtlogd – libvirt log management daemon

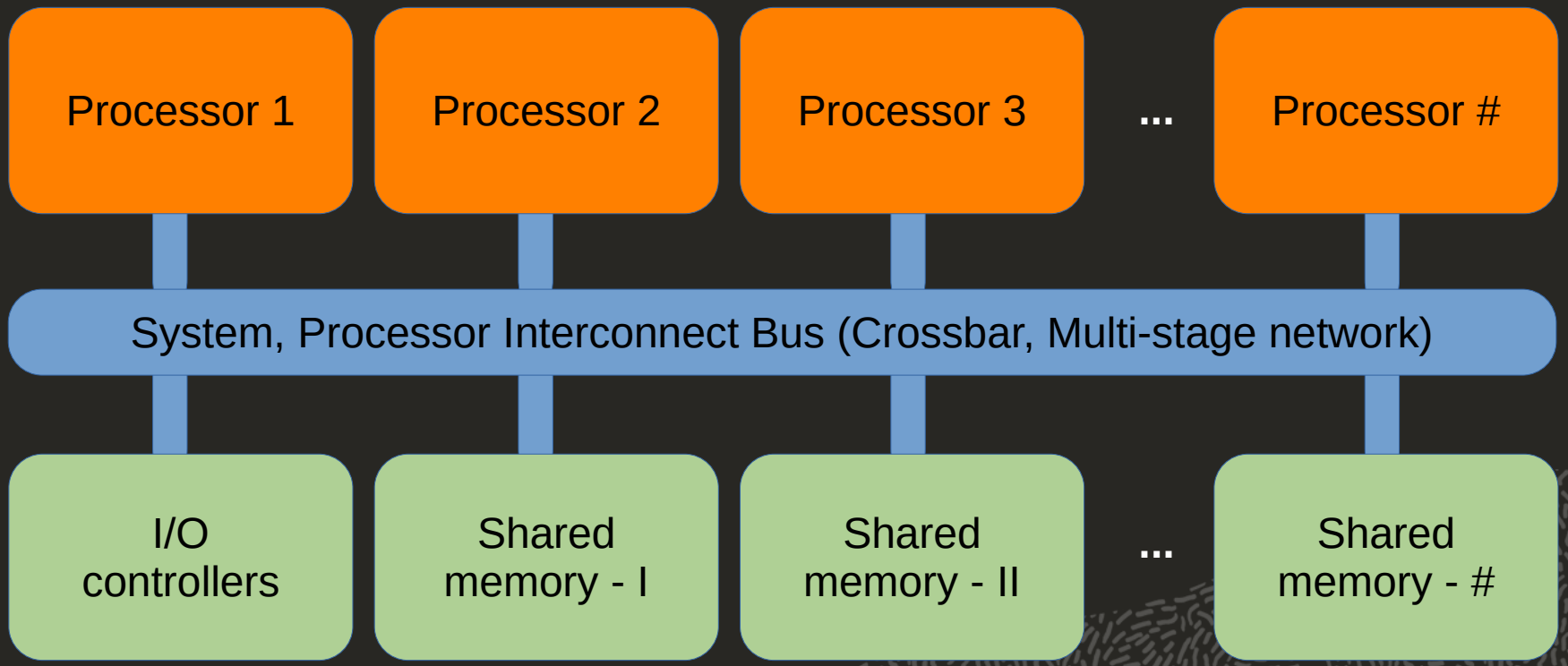
Guest processes

- qemu – Quick EMUlator, under KVM the execution of the “guest” code is done by the KVM hypervisor under the control of QEMU

Program agenda

1. Virtualization – libvirt QEMU/KVM
2. **System Architecture – UMA / NUMA**
3. “Host” topology – Processor Topology / NUMA Topology
4. Partitioning the NUMA host
5. vNUMA automatic host partitioning

UMA – Uniform Memory Access



UMA – Uniform Memory Access

Shared memory multiprocessor architecture

Advantages:

- All the processors share same physical memory uniformly
- Access time to memory is same for all processors
- Peripherals are also shared in some fashion
- Relatively easy programming model

Disadvantages:

- Adding more processors and I/O causes bus contention

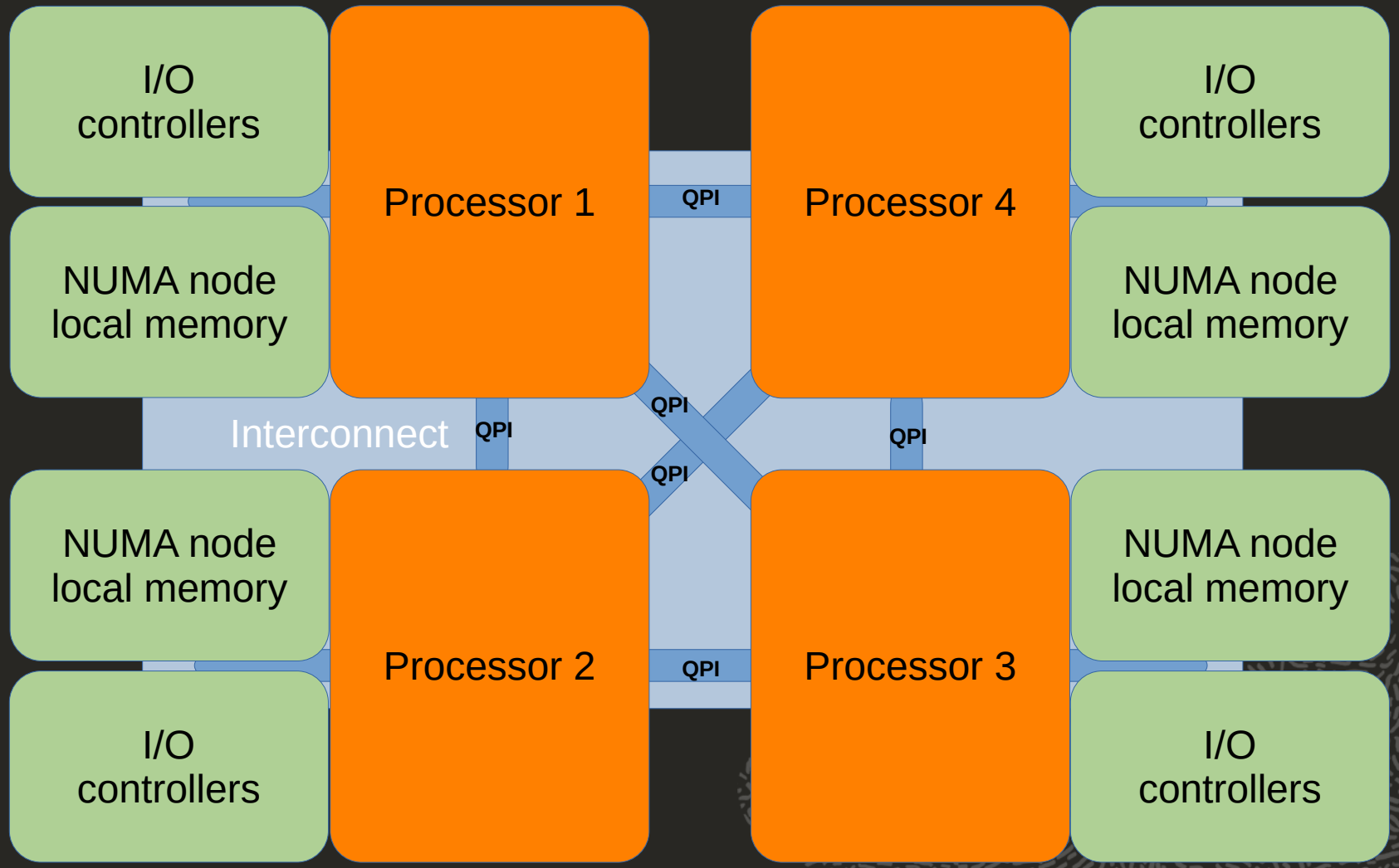
UMA v.s. NUMA machine architectures

Historical Software considerations

- Single processor architectures
- Multiprocessor architectures (HWP - HeavyWeight Processes)
  - Multi-process programs
  - Inter-process communication using System V Shared Memory
- SMP architectures (LWP - Light-Weight Processes)
  - Multi-threaded multi-process applications / database engines, JavaVM, LLVM, programs build on top of multi-threading programming libraries

Ultimately complex multi-process and multi-threading “enterprise” s/w started to outweigh this UMA architecture rooting for NUMA (Non-Uniform Memory Access)

NUMA – Non-Uniform Memory Access



NUMA – Non-Uniform Memory Access

The fundamental building block of a NUMA machine is a Uniform Memory Access region that we call a “node”. These nodes have point-to-point Interconnects to other nodes.

Point-to-point Interconnect
- Intel - QuickPath Interconnect (QPI)
- AMD – HyperTransport-technology (HT)

Disadvantages:
- Maintaining cache coherence across nodes
- Remote node memory access has a significant overhead
- Memory placement in processes that cross NUMA nodes
- Multi-threaded programs should take care of thread binding

“vNUMA” - What is it good for?

- Guests should have NUMA characteristics too!
- The deployment of NUMA architecture optimized execution environments under virtualized setups requires vNUMA
- Application performance!

Program agenda

1. Virtualization – libvirt QEMU/KVM
2. System Architecture – UMA / NUMA
3. **“Host” topology – Processor Topology / NUMA Topology**
4. Partitioning the NUMA host
5. vNUMA automatic host partitioning

Describing the “*host*”

Four NUMA node machine:

- Disk, USB, serial/console I/O under P0
- Network I/O, SR-IOV under P1

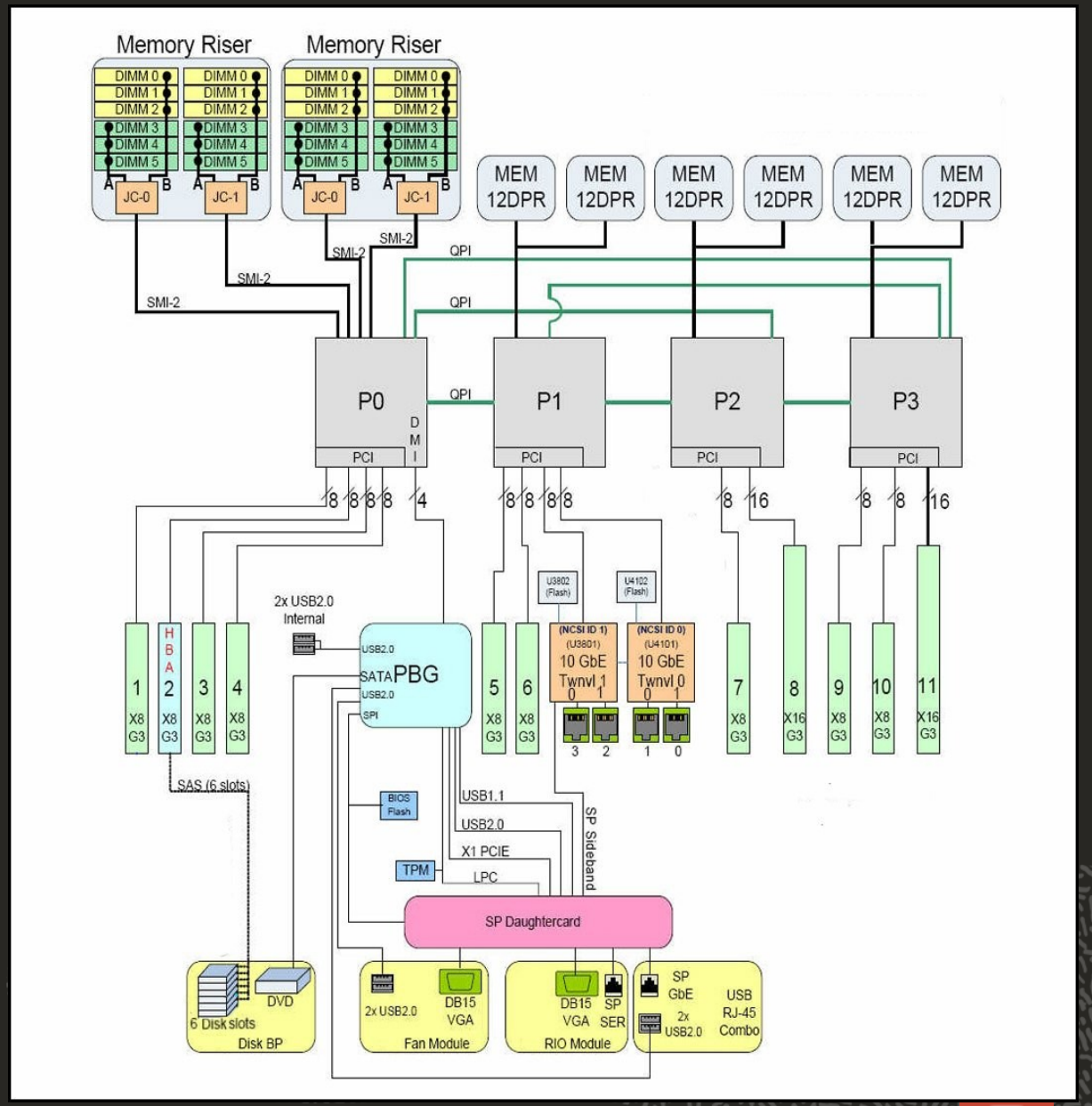
With help of host local commands:

- `lscpu`
- `lspci -tv`
- `lstopo (lstopo-no-graphics)`
- `numactl -H`

From libvirt local or remote:

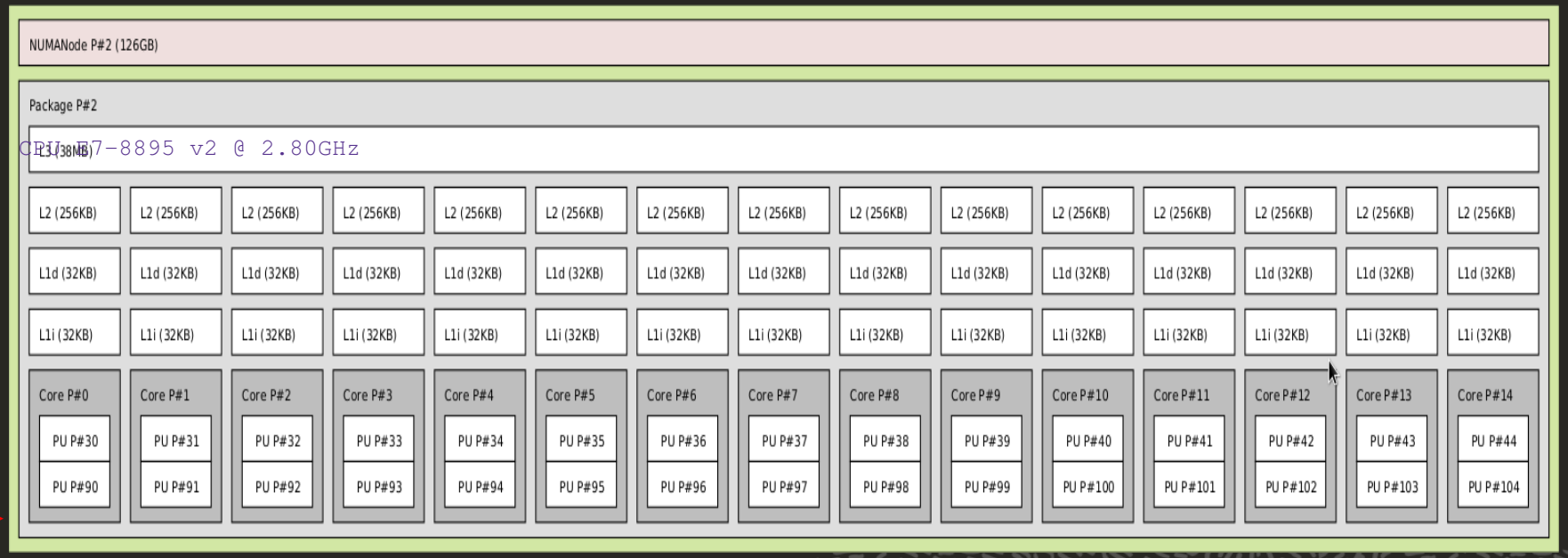
`virsh -c qemu+ssh://user@hostname/system`

- `help`
- `capabilities`
- `domcapabilities`
- `nodedev-list --tree`
- `nodedev-list --cap net (--cap scsi)`
- `nodedev-dumpxml pci_domain_bus_slot_function`
- `sysinfo`



The “host” Processor Topology

```
Architecture:          x86_64
CPU op-mode(s):        32-bit, 64-bit
Byte Order:            Little Endian
CPU(s):                120
On-line CPU(s) list:   0-119
Thread(s) per core:    2
Core(s) per socket:    15
Socket(s):             4
NUMA node(s):          4
Vendor ID:             GenuineIntel
CPU family:            6
Model:                 62
Model name:            Intel(R) Xeon(R) CPU E7-8895 v2 @ 2.80GHz
Stepping:              7
CPU MHz:               2794.385
CPU max MHz:           3600.0000
CPU min MHz:           1200.0000
BogoMIPS:              5586.32
Virtualization:        VT-x
L1d cache:             32K
L1i cache:             32K
L2 cache:              256K
L3 cache:              38400K
NUMA node0 CPU(s):     0-14,60-74
NUMA node1 CPU(s):     15-29,75-89
NUMA node2 CPU(s):     30-44,90-104 >>>
NUMA node3 CPU(s):     45-59,105-119
Flags:                 fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush dts acpi mmx fxsr sse sse2 ss ht tm pbe
syscall nx pdpe1gb rdtscp lm constant_tsc arch_perfmon pebs bts rep_good nopl xtopology nonstop_tsc cpuid aperfmperf pni pclmulqdq dtes64
monitor ds_cpl vmx smx est tm2 ssse3 cx16 xtpr pdcm pcid dca sse4_1 sse4_2 x2apic popcnt tsc_deadline_timer aes xsave avx f16c rdrand lahf_lm
cpuid_fault epb pti intel_ppin ssbd ibrs ibpb stibp tpr_shadow vnmi flexpriority ept vpid fsgsbase smep erms xsaveopt dtherm ida arat pln pts
md_clear flush_l1d
```



The “host” NUMA node Topology

Physical CPU(s) and memory per NUMA node

```
NUMA node0 CPU(s):     0-14,60-74
NUMA node1 CPU(s):     15-29,75-89
NUMA node2 CPU(s):     30-44,90-104
NUMA node3 CPU(s):     45-59,105-119

<wtenhave@peppi:2> numactl -H
available: 4 nodes (0-3)
node 0 cpus: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74
node 0 size: 128602 MB
node 0 free: 125954 MB
node 1 cpus: 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89
node 1 size: 129018 MB
node 1 free: 118358 MB
node 2 cpus: 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104
node 2 size: 129018 MB
node 2 free: 123758 MB
node 3 cpus: 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119
node 3 size: 128985 MB
node 3 free: 127365 MB
node distances:
node   0   1   2   3
  0:  10  21  21  21
  1:  21  10  21  21
  2:  21  21  10  21
  3:  21  21  21  10
```

Program agenda

1. Virtualization – libvirt QEMU/KVM
2. System Architecture – UMA / NUMA
3. “Host” topology – Processor Topology / NUMA Topology
4. **Partitioning the NUMA host**
5. vNUMA automatic host partitioning

Partitioning the NUMA host

Building the initial “guest” XML

```
<domain type='kvm'>
  <name>OL8-vnuma</name>
  <memory unit='GiB'>32</memory>
  <vcpu placement='static'>16</vcpu>
  <os>
    <type arch='x86_64' machine='q35'>hvm</type>
    <boot dev='hd'/>
  </os>
  <features>
    <acpi/>
    <apic/>
  </features>
  <cpu mode='host-passthrough'/>
  <clock offset='utc'/>
  <devices>
    <emulator>/bin/qemu-system-x86_64</emulator>
    <disk type='file' device='disk'>
      <driver name='qemu' type='qcow2'/>
      <source file='/local/ocfs2/images/repos/OL8-vnuma.qcow2'/>
      <target dev='sda' bus='sata'/>
    </disk>
    <controller type='pci' index='0' model='pcie-root'/>
    <controller type='pci' index='1' model='pcie-root-port'/>
    <interface type='bridge'>
      <source bridge='uteng0'/>
      <model type='virtio'/>
    </interface>
    <serial type='pty'/>
    <console type='pty'/>
    <input type='mouse' bus='ps2'/>
    <input type='keyboard' bus='ps2'/>
    <memballoon model='none'/>
  </devices>
</domain>
```

```
[root@OL8-vnuma ~]# dmesg | grep -i numa
[    0.000000] No NUMA configuration found

[root@OL8-vnuma ~]# lscpu  ###reduced output
 ...
CPU(s):                16
On-line CPU(s) list:   0-15
Thread(s) per core:    1
Core(s) per socket:    1
Socket(s):             16
NUMA node(s):          1
 ...
Model name:            Intel(R) Xeon(R) CPU E7-8895 v2 @ 2.80GHz
 ...
Virtualization:        VT-x
Hypervisor vendor:     KVM
Virtualization type:   full
 ...
NUMA node0 CPU(s):     0-15
 ...

[root@OL8-vnuma ~]# numactl -H
available: 1 nodes (0)
node 0 cpus: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
node 0 size: 31897 MB
node 0 free: 31587 MB
node distances:
node   0
  0:  10
```

Partitioning the NUMA host

Defining CPU Topology

```
<domain type='kvm'>
  <name>OL8-vnuma</name>
  <memory unit='GiB'>32</memory>
  <vcpu placement='static'>16</vcpu>
   ...
  <cpu mode='host-passthrough'/>
    <topology sockets='4' cores='2' threads='2'/>
  </cpu>
   ...

</domain>
```

```
[root@OL8-vnuma ~]# dmesg | grep -i numa
[    0.000000] No NUMA configuration found

[root@OL8-vnuma ~]# lscpu  ###reduced output
 ...
CPU(s):                16
On-line CPU(s) list:   0-15
Thread(s) per core:    2
Core(s) per socket:    2
Socket(s):             4
NUMA node(s):          1
 ...
Model name:            Intel(R) Xeon(R) CPU E7-8895 v2 @ 2.80GHz
 ...
Virtualization:        VT-x
Hypervisor vendor:     KVM
Virtualization type:   full
 ...
NUMA node0 CPU(s):     0-15
 ...

[root@OL8-vnuma ~]# numactl -H
available: 1 nodes (0)
node 0 cpus: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
node 0 size: 31897 MB
node 0 free: 31587 MB
node distances:
node   0
  0:  10
```

Partitioning the NUMA host

Defining NUMA Topology

```
<domain type='kvm'>
  <name>OL8-vnuma</name>
  <memory unit='GiB'>32</memory>
  <vcpu placement='static'>16</vcpu>
   ...
  <cpu mode='host-passthrough'>
    <topology sockets='4' cores='2' threads='2'/>
    <numa>
      <cell id='0' cpus='0-3' memory='8388608' unit='KiB'/>
      <cell id='1' cpus='4-7' memory='8388608' unit='KiB'/>
      <cell id='2' cpus='8-11' memory='8388608' unit='KiB'/>
      <cell id='3' cpus='12-15' memory='8388608' unit='KiB'/>
    </numa>
  </cpu>
   ...
</domain>
```

```
[root@OL8-vnuma ~]# lscpu
 ...
CPU(s):                16
On-line CPU(s) list:   0-15
Thread(s) per core:    2
Core(s) per socket:    2
Socket(s):             4
NUMA node(s):          4
 ...
Model name:            Intel(R) Xeon(R) CPU E7-8895 v2 @ 2.80GHz
 ...
Virtualization:        VT-x
Hypervisor vendor:     KVM
Virtualization type:   full
 ...
NUMA node0 CPU(s):     0-3
NUMA node1 CPU(s):     4-7
NUMA node2 CPU(s):     8-11
NUMA node3 CPU(s):     12-15
```

NUMA node distances beteen cells can be described as below

```
<cell id='#' cpus='range' memory='#size' unit='#unit'>
  <distances>
    <sibling id='0' value='10'/>
    <sibling id='1' value='21'/>
    <sibling id='2' value='21'/>
    <sibling id='3' value='21'/>
  </distances>
</cell>
```

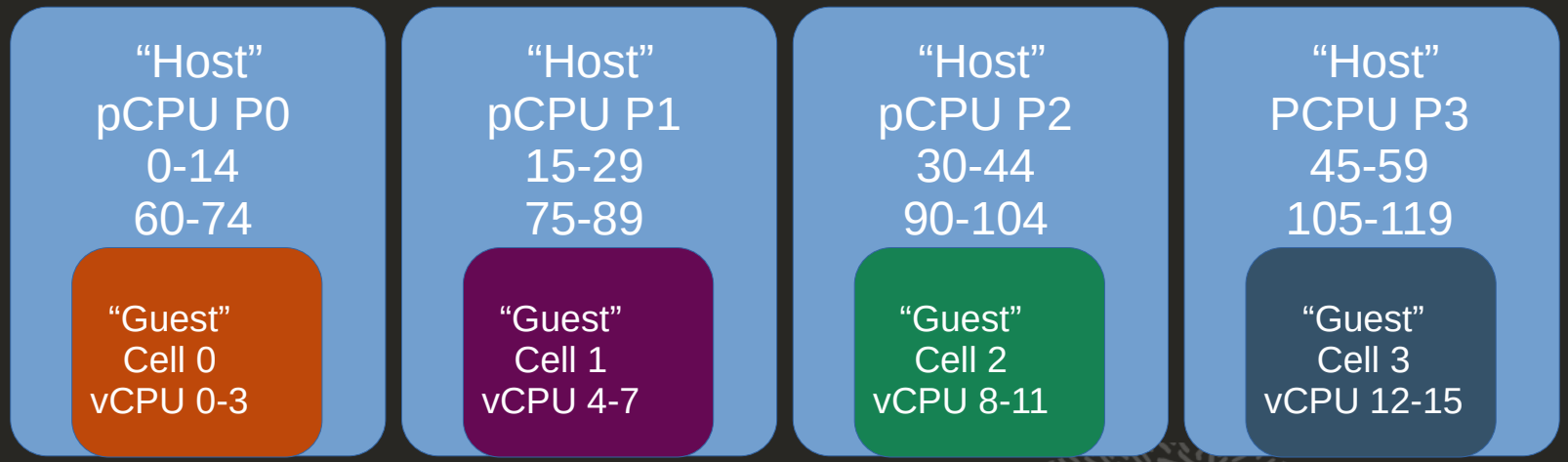
```
[root@OL8-vnuma ~]# numactl -H
available: 4 nodes (0-3)
node 0 cpus: 0 1 2 3
node 0 size: 7708 MB
node 0 free: 7587 MB
node 1 cpus: 4 5 6 7
node 1 size: 8063 MB
node 1 free: 7994 MB
node 2 cpus: 8 9 10 11
node 2 size: 8063 MB
node 2 free: 7999 MB
node 3 cpus: 12 13 14 15
node 3 size: 8062 MB
node 3 free: 8002 MB
node distances:
node   0   1   2   3
  0:  10  21  21  21
  1:  21  10  21  21
  2:  21  21  10  21
  3:  21  21  21  10
```

Partitioning the NUMA host

Pinning the “guest” vcpus to the “host” NUMA node cpuset

```
<domain type='kvm'>
  <name>OL8-vnuma</name>
  <memory unit='GiB'>32</memory>
  <vcpu placement='static'>16</vcpu>
  <cputune>
    <vcpupin vcpu='0' cpuset='0-14,60-74'/>
    <vcpupin vcpu='1' cpuset='0-14,60-74'/>
    <vcpupin vcpu='2' cpuset='0-14,60-74'/>
    <vcpupin vcpu='3' cpuset='0-14,60-74'/>
    <vcpupin vcpu='4' cpuset='15-29,75-89'/>
    <vcpupin vcpu='5' cpuset='15-29,75-89'/>
    <vcpupin vcpu='6' cpuset='15-29,75-89'/>
    <vcpupin vcpu='7' cpuset='15-29,75-89'/>
    <vcpupin vcpu='8' cpuset='30-44,90-104'/>
    <vcpupin vcpu='9' cpuset='30-44,90-104'/>
    <vcpupin vcpu='10' cpuset='30-44,90-104'/>
    <vcpupin vcpu='11' cpuset='30-44,90-104'/>
    <vcpupin vcpu='12' cpuset='45-59,105-119'/>
    <vcpupin vcpu='13' cpuset='45-59,105-119'/>
    <vcpupin vcpu='14' cpuset='45-59,105-119'/>
    <vcpupin vcpu='15' cpuset='45-59,105-119'/>
  </cputune>
   ...
  <cpu mode='host-passthrough'>
    <topology sockets='4' cores='2' threads='2'/>
    <numa>
      <cell id='0' cpus='0-3' memory='8388608' unit='KiB'/>
      <cell id='1' cpus='4-7' memory='8388608' unit='KiB'/>
      <cell id='2' cpus='8-11' memory='8388608' unit='KiB'/>
      <cell id='3' cpus='12-15' memory='8388608' unit='KiB'/>
    </numa>
  </cpu>
   ...
</domain>
```

>>>

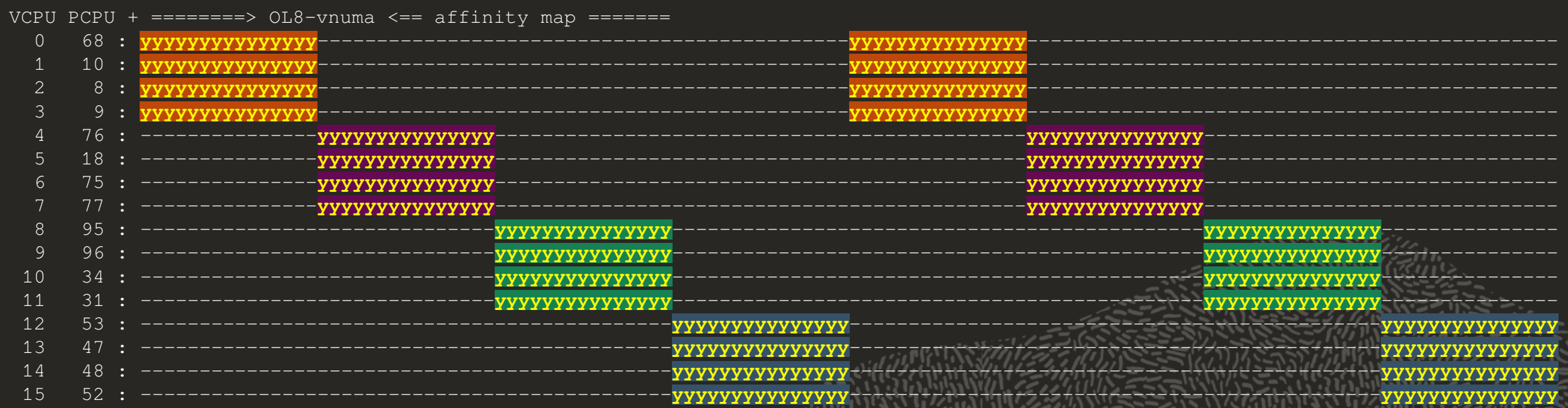


Partitioning the NUMA host

Pinning the “guest” vcpus to the “host” NUMA node cpuset

Inspect the “guest” vCPU : pCPU pinning from virsh:

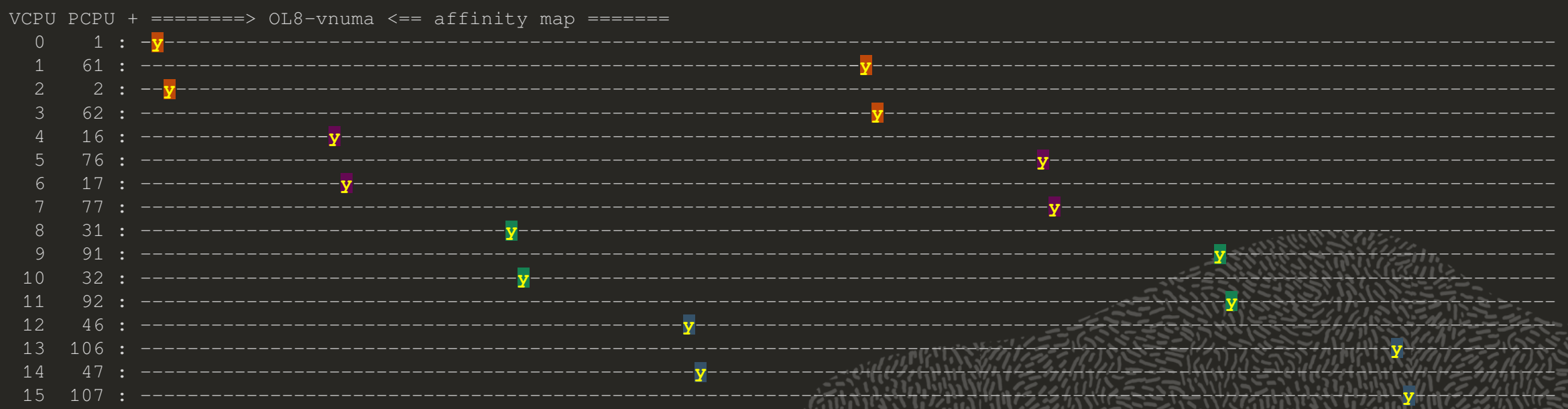
Virsh-affinity based on ‘`virsh vcpuinfo OL8-vnuma`’ info



Partitioning the NUMA host

Pinning the “guest” vcpus to the “host” NUMA node cpuset

Oracle enhanced libvirt qemu driver does vCPU:pCPU pinning respecting the CPU SMT topology

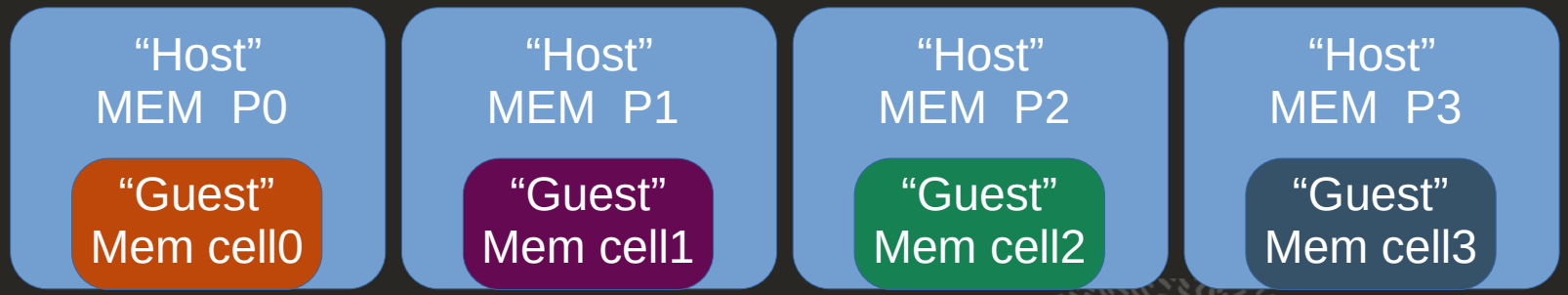


Partitioning the NUMA host

Binding the “guest” cell memory to the “host” NUMA nodeset

```
<domain type='kvm'>
  <name>OL8-vnuma</name>
  <memory unit='GiB'>32</memory>
  <memoryBacking>
    <locked/>
  </memoryBacking>
  <vcpu placement='static'>16</vcpu>
  <cputune>
    <vcpupin vcpu='0' cpuset='0-14,60-74'/>
    <vcpupin vcpu='1' cpuset='0-14,60-74'/>
    <vcpupin vcpu=… … cpuset=… …,… …/>
    <vcpupin vcpu='15' cpuset='45-59,105-119'/>
  </cputune>
  <numatune>
    <memnode cellid='0' mode='strict' nodeset='0'/>
    <memnode cellid='1' mode='strict' nodeset='1'/>
    <memnode cellid='2' mode='strict' nodeset='2'/>
    <memnode cellid='3' mode='strict' nodeset='3'/>
  </numatune>
   ...
  <cpu mode='host-passthrough'>
    <topology sockets=’4’ cores=’2’ threads=’2’/>
    <numa>
      <cell id='0' cpus='0-3' memory='8388608' unit='KiB'/>
      <cell id='1' cpus='4-7' memory='8388608' unit='KiB'/>
      <cell id='2' cpus='8-11' memory='8388608' unit='KiB'/>
      <cell id='3' cpus='12-15' memory='8388608' unit='KiB'/>
    </numa>
  </cpu>
   ...
</domain>
```

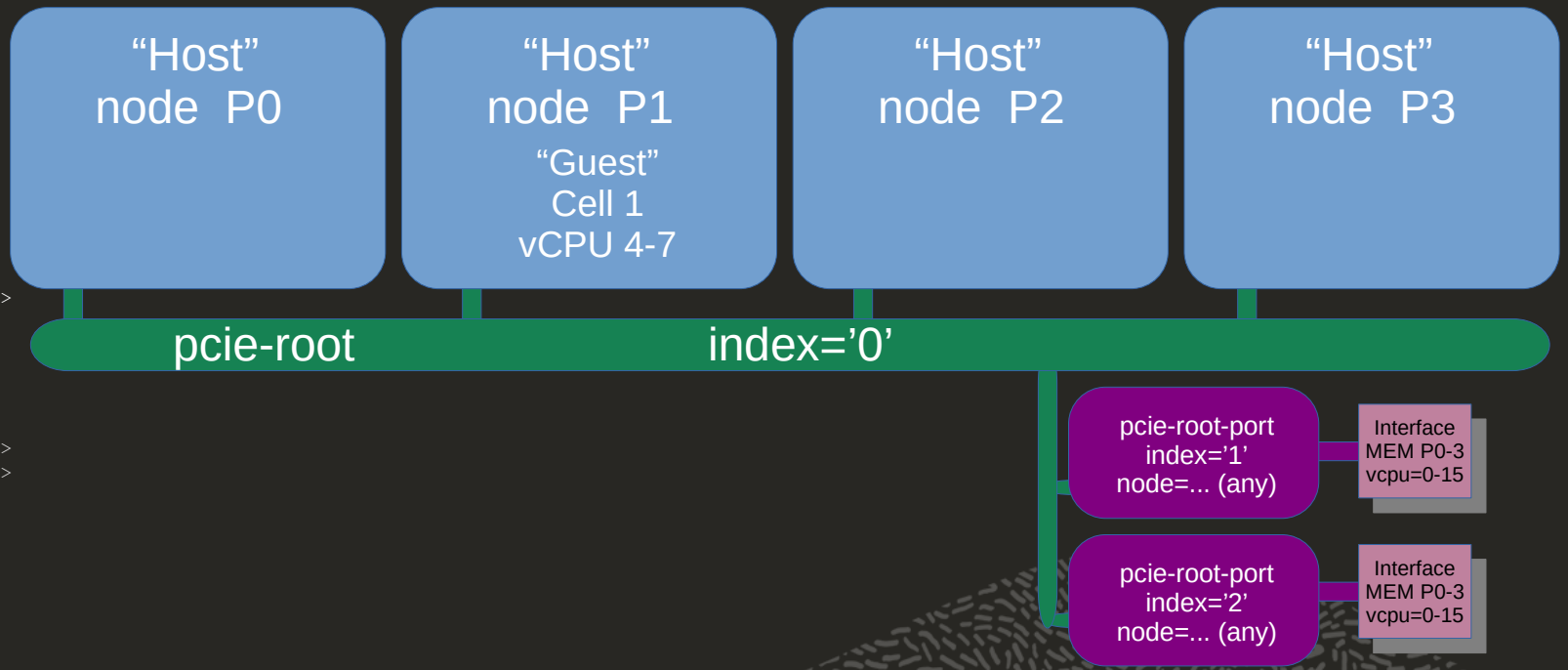
>>>



Partitioning the NUMA host

Adding a NUMA node targeted pcie-expander-bus

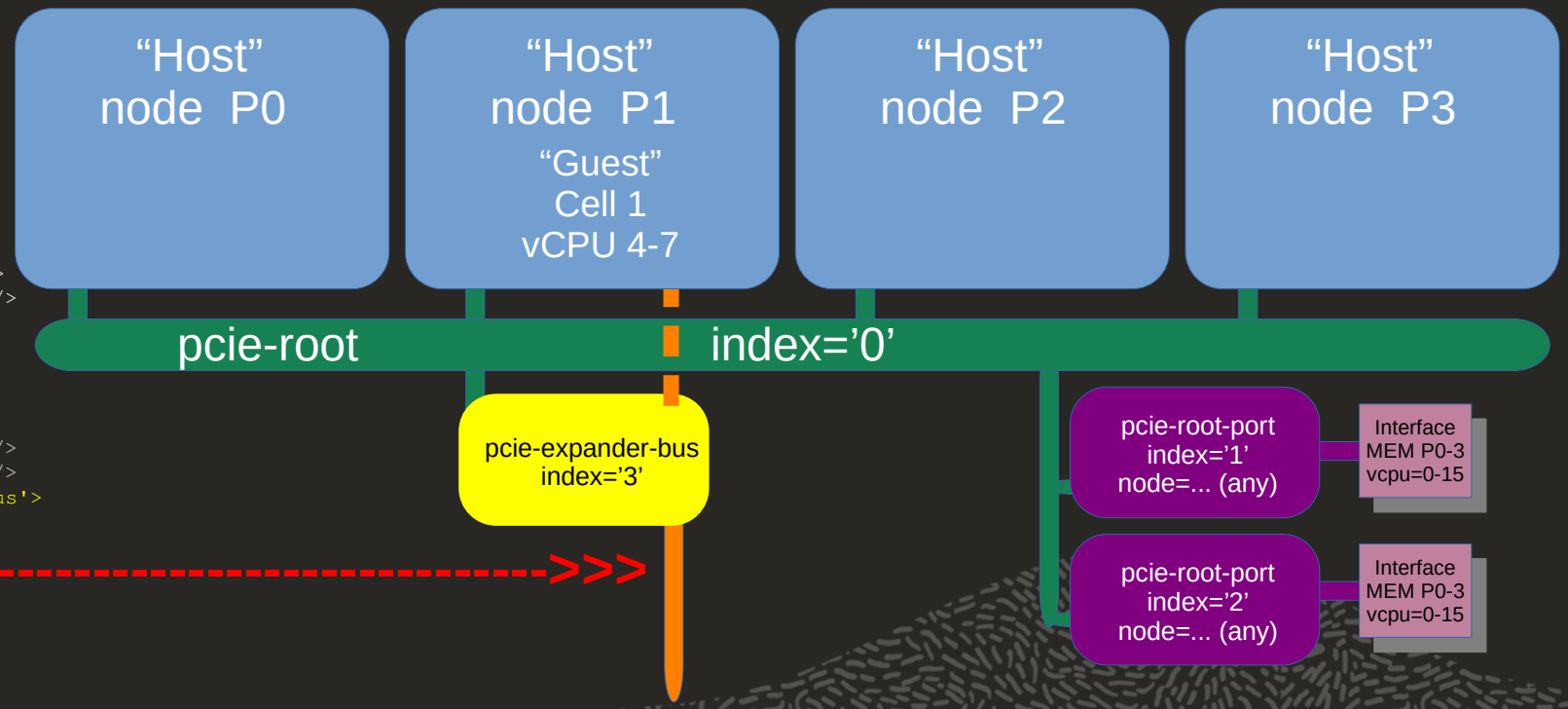
```
<domain type='kvm'>
  <name>OL8-vnuma</name>
  <memory unit='GiB'>32</memory>
  <vcpu placement='static'>16</vcpu>
   ...
  <cpu mode='host-passthrough'>
    <topology sockets='4' cores='2' threads='2'/>
    <numa>
      <cell id='0' cpus='0-3' memory='8388608' unit='KiB'/>
      <cell id='1' cpus='4-7' memory='8388608' unit='KiB'/>
      <cell id='2' cpus='8-11' memory='8388608' unit='KiB'/>
      <cell id='3' cpus='12-15' memory='8388608' unit='KiB'/>
    </numa>
  </cpu>
   ...
  <devices>
    <controller type='pci' index='0' model='pcie-root'/>
    <controller type='pci' index='1' model='pcie-root-port'/>
    <controller type='pci' index='2' model='pcie-root-port'/>
     ...
  </devices>
   ...
</domain>
```



Partitioning the NUMA host

Adding a NUMA node targeted pcie-expander-bus

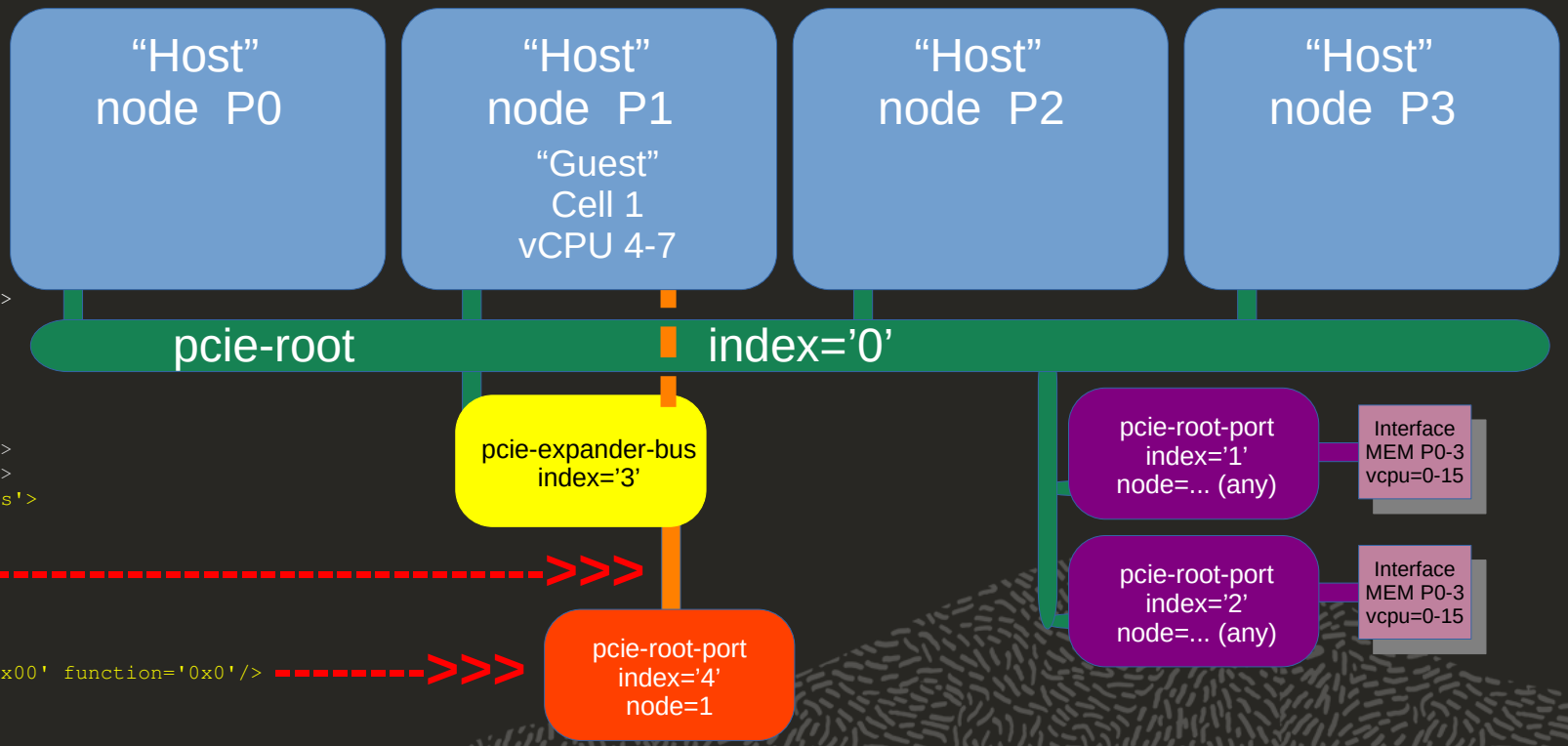
```
<domain type='kvm'>
  <name>OL8-vnuma</name>
  <memory unit='GiB'>32</memory>
  <vcpu placement='static'>16</vcpu>
   ...
  <cpu mode='host-passthrough'>
    <topology sockets='4' cores='2' threads='2'/>
    <numa>
      <cell id='0' cpus='0-3' memory='8388608' unit='KiB'/>
      <cell id='1' cpus='4-7' memory='8388608' unit='KiB'/>
      <cell id='2' cpus='8-11' memory='8388608' unit='KiB'/>
      <cell id='3' cpus='12-15' memory='8388608' unit='KiB'/>
    </numa>
  </cpu>
   ...
  <devices>
    <controller type='pci' index='0' model='pcie-root'/>
    <controller type='pci' index='1' model='pcie-root-port'/>
    <controller type='pci' index='2' model='pcie-root-port'/>
    <controller type='pci' index='3' model='pcie-expander-bus'>
      <model name='pxb-pcie'/>
      <target busNr='254'>
        <node>1</node>
      </target>
    </controller>
     ...
  </devices>
   ...
</domain>
```



Partitioning the NUMA host

Adding a NUMA node targeted pcie-expander-bus

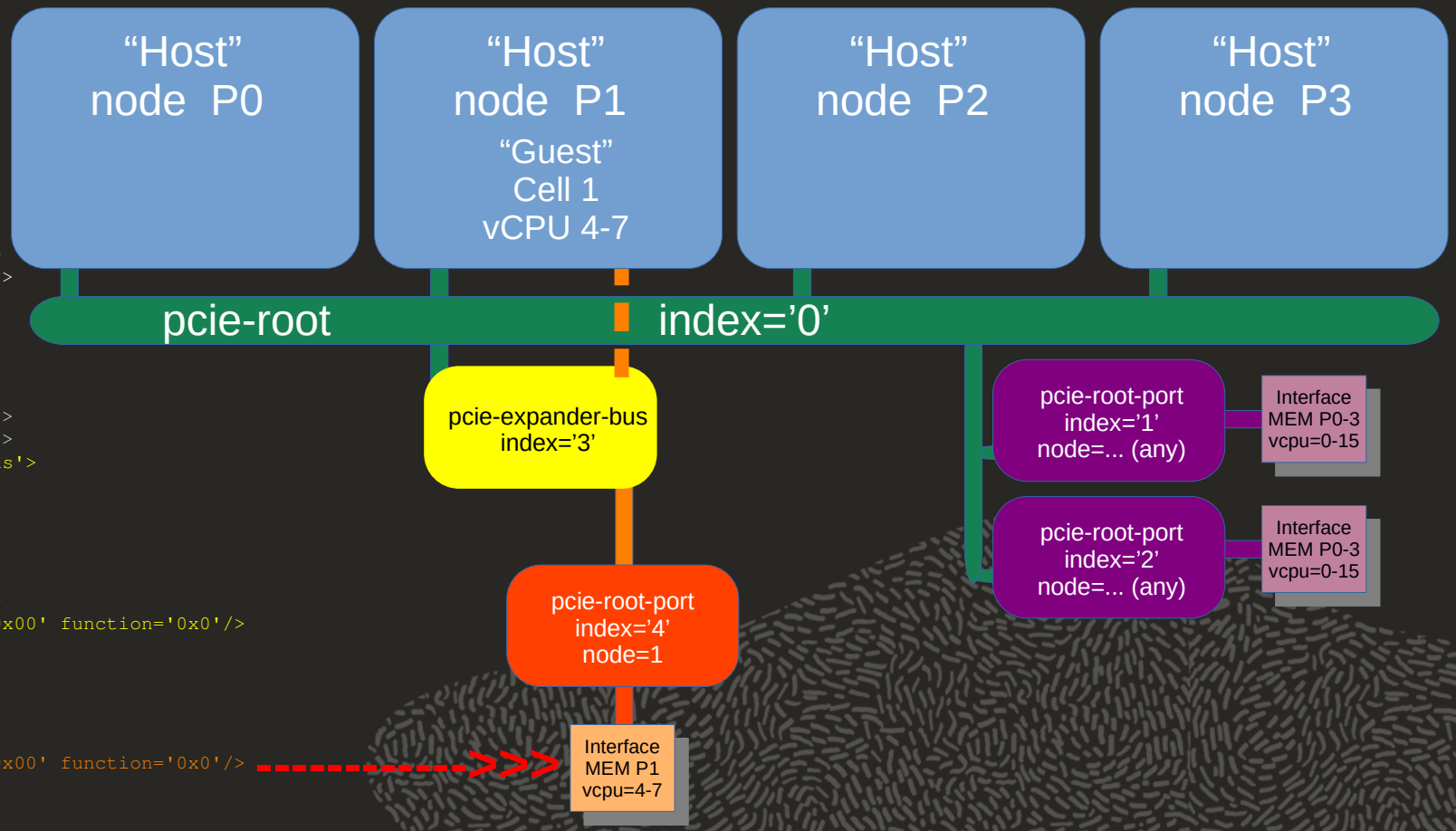
```
<domain type='kvm'>
  <name>OL8-vnuma</name>
  <memory unit='GiB'>32</memory>
  <vcpu placement='static'>16</vcpu>
   ...
  <cpu mode='host-passthrough'>
    <topology sockets='4' cores='2' threads='2'/>
    <numa>
      <cell id='0' cpus='0-3' memory='8388608' unit='KiB'/>
      <cell id='1' cpus='4-7' memory='8388608' unit='KiB'/>
      <cell id='2' cpus='8-11' memory='8388608' unit='KiB'/>
      <cell id='3' cpus='12-15' memory='8388608' unit='KiB'/>
    </numa>
  </cpu>
   ...
  <devices>
    <controller type='pci' index='0' model='pcie-root'/>
    <controller type='pci' index='1' model='pcie-root-port'/>
    <controller type='pci' index='2' model='pcie-root-port'/>
    <controller type='pci' index='3' model='pcie-expander-bus'>
      <model name='pxb-pcie'/>
      <target busNr='254'>
        <node>1</node>
      </target>
    </controller>
    <controller type='pci' index='4' model='pcie-root-port'>
      <address type='pci' domain='0x0000' bus='0x03' slot='0x00' function='0x0'/>
    </controller>
     ...
  </devices>
   ...
</domain>
```



Partitioning the NUMA host

Adding a NUMA node targeted pcie-expander-bus

```
<domain type='kvm'>
  <name>OL8-vnuma</name>
  <memory unit='GiB'>32</memory>
  <vcpu placement='static'>16</vcpu>
   ...
  <cpu mode='host-passthrough'>
    <topology sockets='4' cores='2' threads='2'/>
    <numa>
      <cell id='0' cpus='0-3' memory='8388608' unit='KiB'/>
      <cell id='1' cpus='4-7' memory='8388608' unit='KiB'/>
      <cell id='2' cpus='8-11' memory='8388608' unit='KiB'/>
      <cell id='3' cpus='12-15' memory='8388608' unit='KiB'/>
    </numa>
  </cpu>
   ...
  <devices>
    <controller type='pci' index='0' model='pcie-root'/>
    <controller type='pci' index='1' model='pcie-root-port'/>
    <controller type='pci' index='2' model='pcie-root-port'/>
    <controller type='pci' index='3' model='pcie-expander-bus'>
      <model name='pxb-pcie'/>
      <target busNr='254'>
        <node>1</node>
      </target>
    </controller>
    <controller type='pci' index='4' model='pcie-root-port'>
      <address type='pci' domain='0x0000' bus='0x03' slot='0x00' function='0x0'/>
    </controller>
    <interface type='bridge'>
      <mac address='52:54:00:0f:fb:22'/>
      <source bridge='uteng0'/>
      <model type='virtio'/>
      <address type='pci' domain='0x0000' bus='0x04' slot='0x00' function='0x0'/>
    </interface>
  </devices>
   ...
</domain>
```



Partitioning the NUMA host

Adding a NUMA node targeted pcie-expander-bus

```
<domain type='kvm'>
  <name>OL8-vnuma</name>
  <memory unit='GiB'>32</memory>
  <vcpu placement='static'>16</vcpu>
   ...
  <cpu mode='host-passthrough'>
    <topology sockets='4' cores='2' threads='2'/>
    <numa>
      <cell id='0' cpus='0-3' memory='8388608' unit='KiB'/>
      <cell id='1' cpus='4-7' memory='8388608' unit='KiB'/>
      <cell id='2' cpus='8-11' memory='8388608' unit='KiB'/>
      <cell id='3' cpus='12-15' memory='8388608' unit='KiB'/>
    </numa>
  </cpu>
   ...
  <devices>
    <controller type='pci' index='0' model='pcie-root'/>
    <controller type='pci' index='1' model='pcie-root-port'/>
    <controller type='pci' index='2' model='pcie-root-port'/>
    <controller type='pci' index='3' model='pcie-expander-bus'>
      <model name='pxb-pcie'/>
      <target busNr='254'>
        <node>1</node>
      </target>
    </controller>
    <controller type='pci' index='4' model='pcie-root-port'>
      <address type='pci' domain='0x0000' bus='0x03' slot='0x00' function='0x0'/>
    </controller>
    <interface type='bridge'>
      <mac address='52:54:00:0f:fb:22'/>
      <source bridge='uteng0'/>
      <model type='virtio'/>
      <address type='pci' domain='0x0000' bus='0x04' slot='0x00' function='0x0'/>
    </interface>
  </devices>
   ...
</domain>
```

Inspecting the NUMA node bound interface

```
[root@OL8-vnuma ~]# dmesg | grep bus | grep -i node
[    0.305006] pci_bus 0000:fe: on NUMA node 1
```

```
[root@OL8-vnuma ~]# lspci -tv
-+-[0000:fe]---00.0-[ff]----00.0  Red Hat, Inc. Virtio network device
 \-[0000:00]-+-00.0  Intel Corporation 82G33/G31/P35/P31 Express DRAM Controller
             +-01.0-[01]----00.0  Red Hat, Inc. QEMU XHCI Host Controller
             +-01.1-[02-03]----00.0-[03]--
             +-01.2-[04]--
             +-02.0  Red Hat, Inc. QEMU PCIe Expander bridge
             +-1f.0  Intel Corporation 82801IB (ICH9) LPC Interface Controller
             +-1f.2  Intel Corporation 82801IR/IO/IH (ICH9R/DO/DH) 6 port SATA Controller [AHCImode]
             \-1f.3  Intel Corporation 82801I (ICH9 Family) SMBus Controller
```

Libvirt invoking QEMU

How did the XML transform into the QEMU command line?

```
/bin/qemu-system-x86_64 -name guest=OL8-vnuma,debug-threads=on -S \
    -object secret,id=masterKey0,format=raw,file=/var/lib/libvirt/qemu/domain-64-OL8-vnuma/master-key.aes \
    -machine pc-q35-3.1,accel=kvm,usb=off,dump-guest-core=off \
    -cpu host -m 32768 -overcommit mem-lock=on \
    -smp 1,maxcpus=16,sockets=4,cores=2,threads=2 \
    -object memory-backend-ram,id=ram-node0,size=8589934592,host-nodes=0,policy=bind \
    -numa node,nodeid=0,cpus=0-3,memdev=ram-node0 \
    -object memory-backend-ram,id=ram-node1,size=8589934592,host-nodes=1,policy=bind \
    -numa node,nodeid=1,cpus=4-7,memdev=ram-node1 \
    -object memory-backend-ram,id=ram-node2,size=8589934592,host-nodes=2,policy=bind \
    -numa node,nodeid=2,cpus=8-11,memdev=ram-node2 \
    -object memory-backend-ram,id=ram-node3,size=8589934592,host-nodes=3,policy=bind \
    -numa node,nodeid=3,cpus=12-15,memdev=ram-node3 \
    -numa dist,src=0,dst=0,val=10 -numa dist,src=0,dst=1,val=21 -numa dist,src=0,dst=2,val=21 -numa dist,src=0,dst=3,val=21 \
    -numa dist,src=1,dst=0,val=21 -numa dist,src=1,dst=1,val=10 -numa dist,src=1,dst=2,val=21 -numa dist,src=1,dst=3,val=21 \
    -numa dist,src=2,dst=0,val=21 -numa dist,src=2,dst=1,val=21 -numa dist,src=2,dst=2,val=10 -numa dist,src=2,dst=3,val=21 \
    -numa dist,src=3,dst=0,val=21 -numa dist,src=3,dst=1,val=21 -numa dist,src=3,dst=2,val=21 -numa dist,src=3,dst=3,val=10 \
    -uuid 2dae4266-6da5-4893-8616-bc2871c4b80f \
    -display none -no-user-config -nodefaults \
    -chardev socket,id=charmonitor,fd=35,server,nowait \
    -mon chardev=charmonitor,id=monitor,mode=control -rtc base=utc -no-shutdown -boot strict=on \
    -device pcie-root-port,port=0x8,chassis=1,id=pci.1,bus=pcie.0,multifunction=on,addr=0x1 \
    -device pcie-root-port,port=0x9,chassis=2,id=pci.2,bus=pcie.0,addr=0x1.0x1 \
    -device pxb-pcie,bus_nr=254,id=pci.3,numa_node=1,bus=pcie.0,addr=0x2 \
    -device pcie-root-port,port=0x0,chassis=4,id=pci.4,bus=pci.3,addr=0x0 \
    -device pcie-root-port,port=0xa,chassis=5,id=pci.5,bus=pcie.0,addr=0x1.0x2 \
    -device pcie-pci-bridge,id=pci.6,bus=pci.2,addr=0x0 \
    -device qemu-xhci,id=usb,bus=pci.1,addr=0x0 \
    -drive file=/local/ocfs2/images/repos/OL8-vnuma.qcow2,format=qcow2,if=none,id=drive-sata0-0-0 \
    -device ide-hd,bus=ide.0,drive=drive-sata0-0-0,id=sata0-0-0,bootindex=1 \
    -netdev tap,fd=37,id=hostnet0,vhost=on,vhostfd=38 \
    -device virtio-net-pci,netdev=hostnet0,id=net0,mac=52:54:00:0f:fb:22,bus=pci.4,addr=0x0 \
    -chardev pty,id=charserial0 \
    -device isa-serial,chardev=charserial0,id=serial0 \
    -sandbox on,obsolete=deny,elevateprivileges=deny,spawn=deny,resourcecontrol=deny -msg timestamp=on
```

Program agenda

1. Virtualization – libvirt QEMU/KVM
2. System Architecture – UMA / NUMA
3. “Host” topology – Processor Topology / NUMA Topology
4. Partitioning the NUMA host
5. **vNUMA automatic host partitioning**

vNUMA automatic host partitioning by libvirt

Proposed libvirt version adds <vnuma> element for automatic vNUMA partitioning

```
<domain type='kvm'>
  <name>OL8-vnuma</name>
  <memory unit='GiB'>32</memory>
  <vnuma mode='host'/>
  <vcpu placement='static'>16</vcpu>
   ...
</domain>
```

>>>

```
<domain type='kvm'>
  <name>OL8-vnuma</name>
  <memory unit='GiB'>32</memory>
  <memoryBacking>
    <locked/>
  </memoryBacking>
  <vcpu placement='static'>16</vcpu>
  <vcpus>
    <vcpu id='0' enabled='yes' hotpluggable='no'/>
    <vcpu id='1' enabled='yes' hotpluggable='yes'/>
    <vcpu id=… … enabled='...' hotpluggable='...'/>
    <vcpu id='15' enabled='yes' hotpluggable='yes'/>
  </vcpus>
  <cputune>
    <vcpupin vcpu='0' cpuset='0-14,60-74'/>
    <vcpupin vcpu='1' cpuset='0-14,60-74'/>
    <vcpupin vcpu=… … cpuset=… …,… …/>
    <vcpupin vcpu='15' cpuset='45-59,105-119'/>
  </cputune>
  <numatune>
    <memnode cellid='0' mode='strict' nodeset='0'/>
    <memnode cellid='1' mode='strict' nodeset='1'/>
    <memnode cellid='2' mode='strict' nodeset='2'/>
    <memnode cellid='3' mode='strict' nodeset='3'/>
  </numatune>
   ...
  <cpu mode='host-passthrough'>
    <topology sockets='4' cores='2' threads='2'/>
    <numa>
      <cell id='0' cpus='0-3' memory='8388608' unit='KiB'/>
      <cell id='1' cpus='4-7' memory='8388608' unit='KiB'/>
      <cell id='2' cpus='8-11' memory='8388608' unit='KiB'/>
      <cell id='3' cpus='12-15' memory='8388608' unit='KiB'/>
    </numa>
  </cpu>
   ...
</domain>
```

vNUMA automatic host partitioning by libvirt

In addition to the “host” partitioning mode the vNUMA enhancement also adds the “node” partitioning mode

```
<domain>
  <name>OL8-vnuma</name>
   ...
  <vnuma mode='host|node'
         distribution='contiguous|siblings|round-robin|interleave'>
    <memory unit='GiB'>32</memory>
    <partition nodeset="0-3,^2" cells="6"/>
  </vnuma>
  <vcpu placement='static'>16</vcpu>
   ...
</domain>
```

Example of <vnuma mode=’node’>

```
<domain>
  <name>OL8-vnuma</name>
   ...
  <vnuma mode='node'>
    <memory unit='GiB'>32</memory>
    <partition nodeset="1"/>
  </vnuma>
  <vcpu placement='static'>16</vcpu>
   ...
</domain>
```

---------->>>

```
<domain type='kvm'>
  <name>OL8-vnuma</name>
  <memory unit='GiB'>32</memory>
  <memoryBacking>
    <locked/>
  </memoryBacking>
  <vcpu placement='static'>16</vcpu>
  <vcpus>
    <vcpu id='0' enabled='yes' hotpluggable='no'/>
    <vcpu id='1' enabled='yes' hotpluggable='yes'/>
    <vcpu id=… … enabled=’...’ hotpluggable=’...’/>
    <vcpu id='15' enabled='yes' hotpluggable='yes'/>
  </vcpus>
  <cputune>
    <vcpupin vcpu='0' cpuset='0-14,60-74'/>
    <vcpupin vcpu='1' cpuset='0-14,60-74'/>
    <vcpupin vcpu=… … cpuset=… …,… …/>
    <vcpupin vcpu='15' cpuset='0-14,60-74'/>
  </cputune>
  <numatune>
    <memnode cellid='0' mode='strict' nodeset='0'/>
  </numatune>
   ...
  <cpu mode='host-passthrough' check='none'>
    <topology sockets='1' cores='8' threads='2'/>
    <numa>
      <cell id='0' cpus='0-15' memory='33554432' unit='KiB'/>
    </numa>
  </cpu>
   ...
</domain>
```

Partitioning the NUMA host “Hotpluggable” vCPUs

Dynamically controlling guest resources “Hotpluggable” vcpus

```
<domain type='kvm'>
  <name>OL8-vnuma</name>
  <memory unit='GiB'>32</memory>
  <vcpu placement='static'>16</vcpu>
   ...
  <vcpus>
    <vcpu id='0' enabled='yes' hotpluggable='no'/>
    <vcpu id='1' enabled='yes' hotpluggable='yes'/>
    <vcpu id='2' enabled='yes' hotpluggable='yes'/>
    <vcpu id='3' enabled='yes' hotpluggable='yes'/>
    <vcpu id='4' enabled='yes' hotpluggable='yes'/>
    <vcpu id='5' enabled='yes' hotpluggable='yes'/>
    <vcpu id='6' enabled='yes' hotpluggable='yes'/>
    <vcpu id='7' enabled='yes' hotpluggable='yes'/>
    <vcpu id='8' enabled='yes' hotpluggable='yes'/>
    <vcpu id='9' enabled='yes' hotpluggable='yes'/>
    <vcpu id='10' enabled='yes' hotpluggable='yes'/>
    <vcpu id='11' enabled='yes' hotpluggable='yes'/>
    <vcpu id='12' enabled='yes' hotpluggable='yes'/>
    <vcpu id='13' enabled='yes' hotpluggable='yes'/>
    <vcpu id='14' enabled='yes' hotpluggable='yes'/>
    <vcpu id='15' enabled='yes' hotpluggable='yes'/>
  </vcpus>
   ...
  <cpu mode='host-passthrough'>
    <topology sockets='4' cores='2' threads='2'/>
    <numa>
      <cell id='0' cpus='0-3' memory='16777216' unit='KiB'/>
      <cell id='1' cpus='4-7' memory='16777216' unit='KiB'/>
      <cell id='2' cpus='8-11' memory='16777216' unit='KiB'/>
      <cell id='3' cpus='12-15' memory='16777216' unit='KiB'/>
    </numa>
  </cpu>
   ...
</domain>
```

Partitioning the NUMA host “Hotpluggable” vCPUs

GUEST CPU inspection and kernel reporting

```
[root@OL8-vnuma ~]# lscpu | grep "NUMA node"
NUMA node(s):          4
NUMA node0 CPU(s):     0-3
NUMA node1 CPU(s):     4-7
NUMA node2 CPU(s):     8-11
NUMA node3 CPU(s):     12-15

[  813.279408] Unregister pv shared memory for cpu 15
[  813.281994] smpboot: CPU 15 is now offline
[  813.309308] Unregister pv shared memory for cpu 14
[  813.311788] smpboot: CPU 14 is now offline
[  813.331365] Unregister pv shared memory for cpu 11
[  813.335913] IRQ 44: no longer affine to CPU11
[  813.338149] smpboot: CPU 11 is now offline
[  813.354323] Unregister pv shared memory for cpu 10
[  813.356435] smpboot: CPU 10 is now offline
[  813.377425] Unregister pv shared memory for cpu 7
[  813.379121] IRQ 43: no longer affine to CPU7
[  813.381184] smpboot: CPU 7 is now offline
[  813.396407] Unregister pv shared memory for cpu 6
[  813.397787] IRQ 51: no longer affine to CPU6
[  813.399752] smpboot: CPU 6 is now offline
[  813.418346] Unregister pv shared memory for cpu 3
[  813.421430] smpboot: CPU 3 is now offline
[  813.437299] Unregister pv shared memory for cpu 2
[  813.440075] smpboot: CPU 2 is now offline

root@OL8-vnuma ~]# lscpu | grep "NUMA node"
NUMA node(s):          4
NUMA node0 CPU(s):     0,1
NUMA node1 CPU(s):     4,5
NUMA node2 CPU(s):     8,9
NUMA node3 CPU(s):     12,13
```

HOST QEMU/KVM vCPU Hypervisor threads

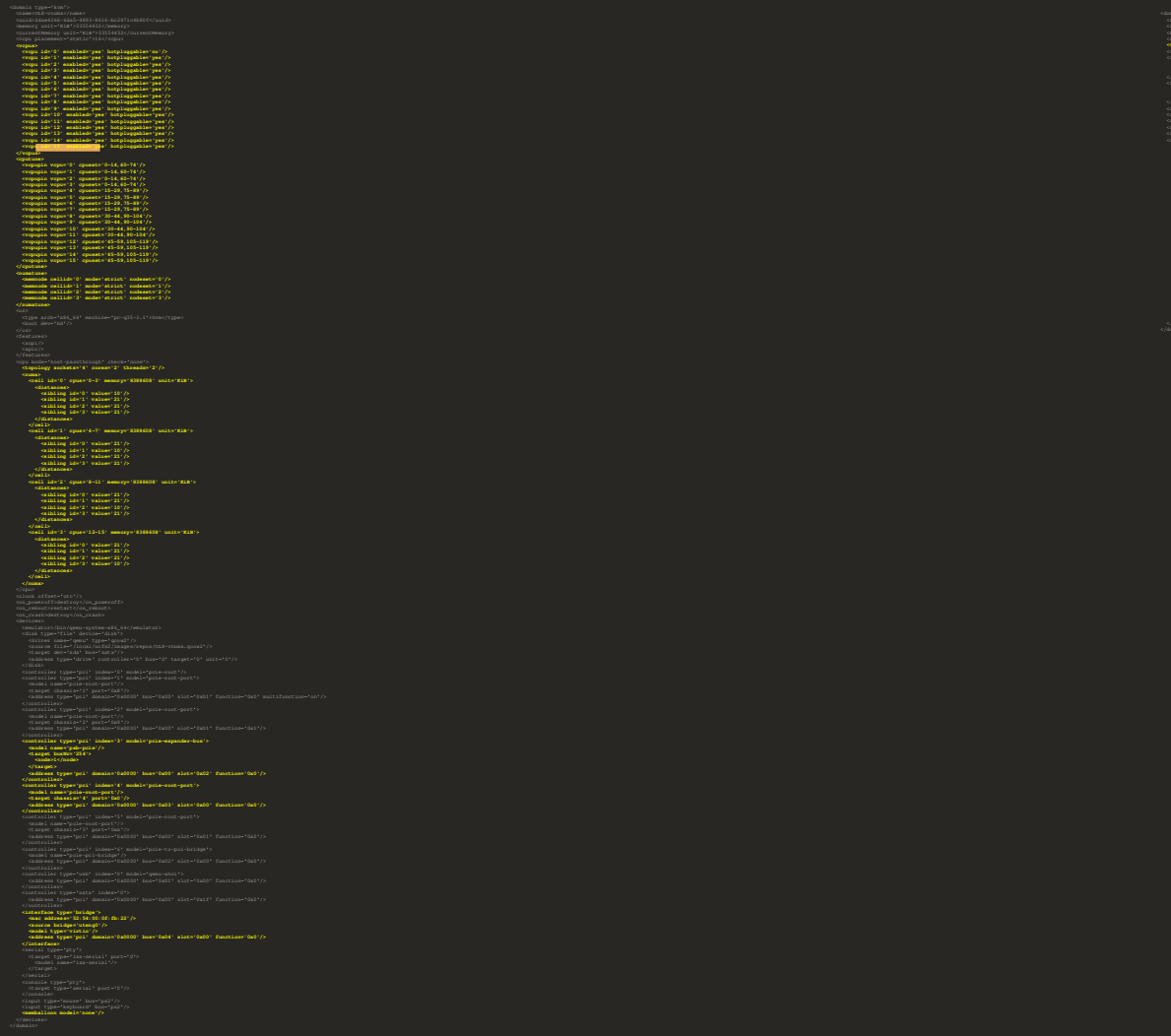
```
<wtenhave@peppi:51> ps -C qemu-system-x86 -L
   PID    LWP TTY          TIME CMD
111612 111612 ?        00:00:07 qemu-system-x86
111612 111614 ?        00:00:00 qemu-system-x86
111612 111618 ?        00:00:06 CPU 0/KVM
111612 111635 ?        00:00:01 CPU 1/KVM
111612 111636 ?        00:00:01 CPU 2/KVM
111612 111637 ?        00:00:01 CPU 3/KVM
111612 111638 ?        00:00:01 CPU 4/KVM
111612 111639 ?        00:00:01 CPU 5/KVM
111612 111640 ?        00:00:01 CPU 6/KVM
111612 111641 ?        00:00:02 CPU 7/KVM
111612 111642 ?        00:00:02 CPU 8/KVM
111612 111643 ?        00:00:01 CPU 9/KVM
111612 111644 ?        00:00:02 CPU 10/KVM
111612 111645 ?        00:00:01 CPU 11/KVM
111612 111646 ?        00:00:01 CPU 12/KVM
111612 111647 ?        00:00:01 CPU 13/KVM
111612 111648 ?        00:00:01 CPU 14/KVM
111612 111649 ?        00:00:01 CPU 15/KVM

<wtenhave@peppi:52> sudo virsh setvcpus OL8-vnuma 8 --live

<wtenhave@peppi:53> ps -C qemu-system-x86 -L
   PID    LWP TTY          TIME CMD
110936 110936 ?        00:00:06 qemu-system-x86
110936 110938 ?        00:00:00 qemu-system-x86
110936 110942 ?        00:00:09 CPU 0/KVM
110936 110987 ?        00:00:02 CPU 1/KVM
110936 110990 ?        00:00:02 CPU 4/KVM
110936 110991 ?        00:00:01 CPU 5/KVM
110936 110994 ?        00:00:03 CPU 8/KVM
110936 110995 ?        00:00:02 CPU 9/KVM
110936 110998 ?        00:00:02 CPU 12/KVM
110936 110999 ?        00:00:01 CPU 13/KVM
```

Partitioning the NUMA host “Expanding” Memory

```
<domain type='kvm'>
  <name>OL8-vnuma</name>
   ...
  <maxMemory slots='16' unit='GiB'>512</maxMemory>
   ...
  <cpu>
    <topology sockets='4' cores='2' threads='2'/>
    <numa>
      <cell id='0' cpus='0-3' memory='16' unit='GiB'/>
      <cell id='1' cpus='4-7' memory='16' unit='GiB'/>
      <cell id='2' cpus='8-11' memory='16' unit='GiB'/>
      <cell id='3' cpus='12-15' memory='16' unit='GiB'/>
    </numa>
  </cpu>
   ...
  <device>
   ...
    <memory model="dimm">
      <target>
        <size unit="GiB">32</size>
        <node>0</node>
      </target>
    </memory>
     ...
    <memballoon model='none'/>
  </device>
</domain>
```

```
<domain type='kvm'>
  <name>OL8-vnuma</name>
   ...
  <maxMemory slots='16' unit='GiB'>512</maxMemory>
   ...
  <cpu>
    <topology sockets='4' cores='2' threads='2'/>
    <numa>
      <cell id='0' cpus='0-3' memory='16' unit='GiB'/>
      <cell id='1' cpus='4-7' memory='16' unit='GiB'/>
      <cell id='2' cpus='8-11' memory='16' unit='GiB'/>
      <cell id='3' cpus='12-15' memory='16' unit='GiB'/>
    </numa>
  </cpu>
   ...
  <device>
   ...
    <memory model='nvdimm' access='shared'>
      <source>
        <path>/dev/dax0.0</path>
        <alignsize unit='KiB'>2048</alignsize>
        <pmem/>
      </source>
      <target>
        <size unit='KiB'>524288</size>
        <node>2</node>
        <label>
          <size unit='KiB'>128</size>
        </label>
      </target>
    </memory>
     ...
    <memballoon model='none'/>
  </device>
</domain>
```

“vNUMA” - What is it good for?



Questions ?

https://www.oracle.com/linux/