More Multiprocessing on seL4: Are efficient SMP Virtual Machines Possible on Verifiable seL4 Kernels today?

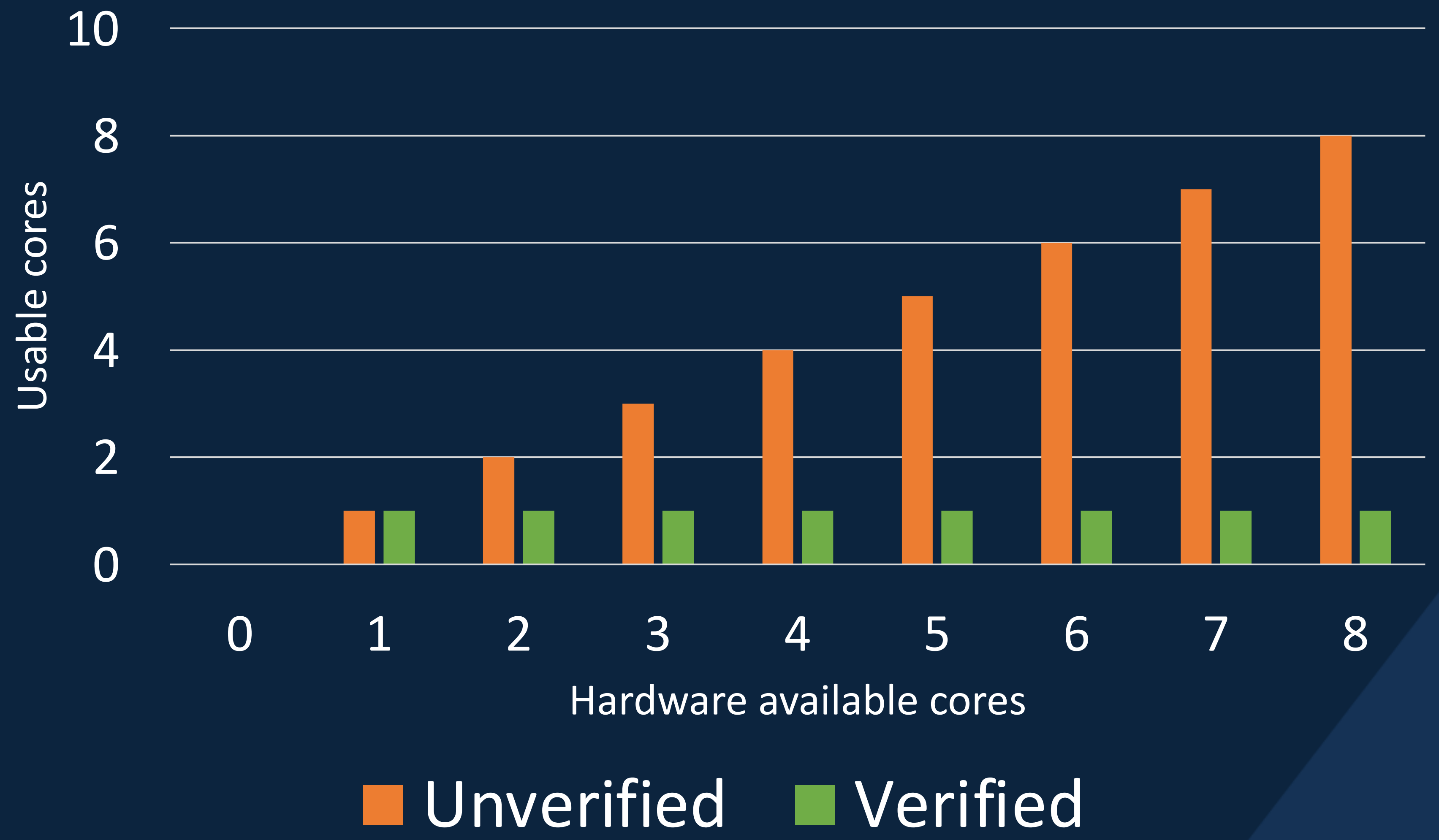
Kent McLeod | seL4 Summit 2023 | Minneapolis, USA

Last year:

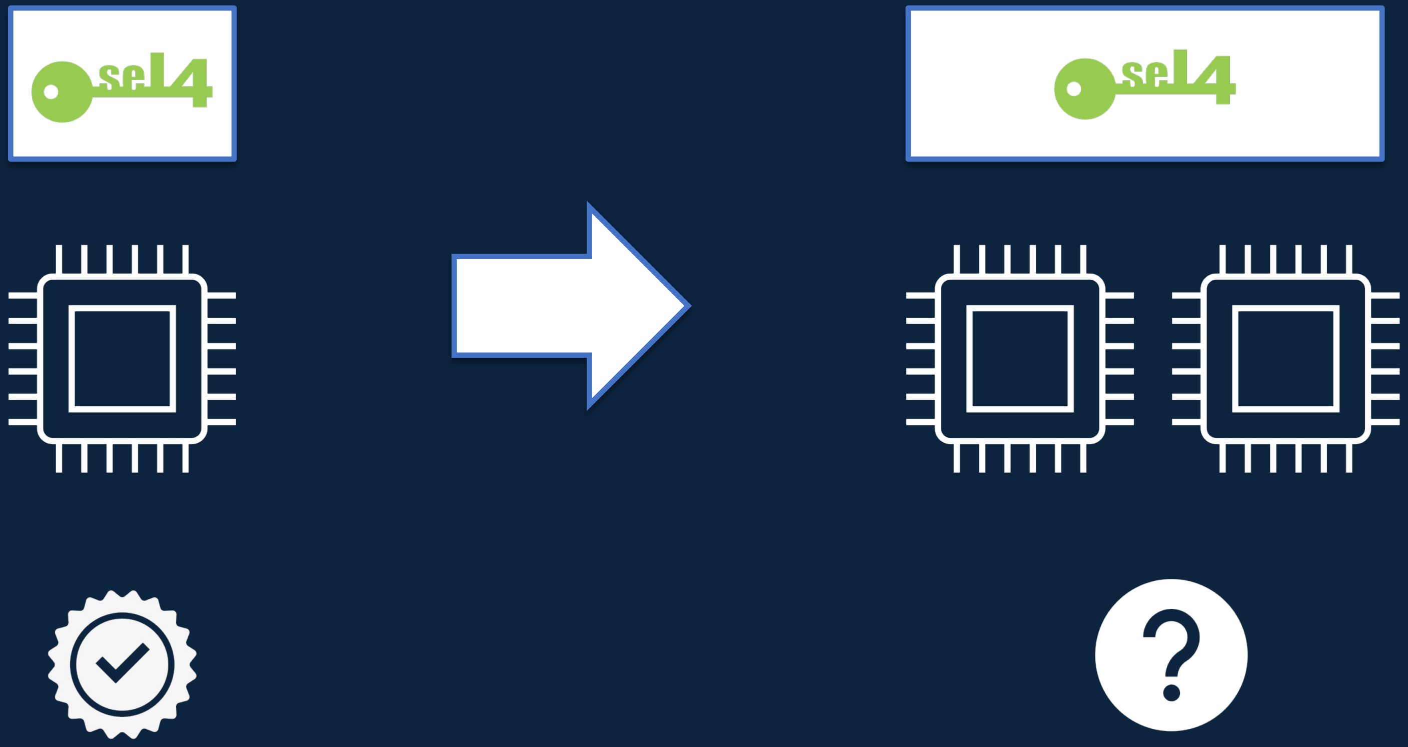
Multiprocessing on seL4 with verified kernels

Kent McLeod | seL4 Summit 2022 | Munich, Germany

Usable CPU count by kernel configuration



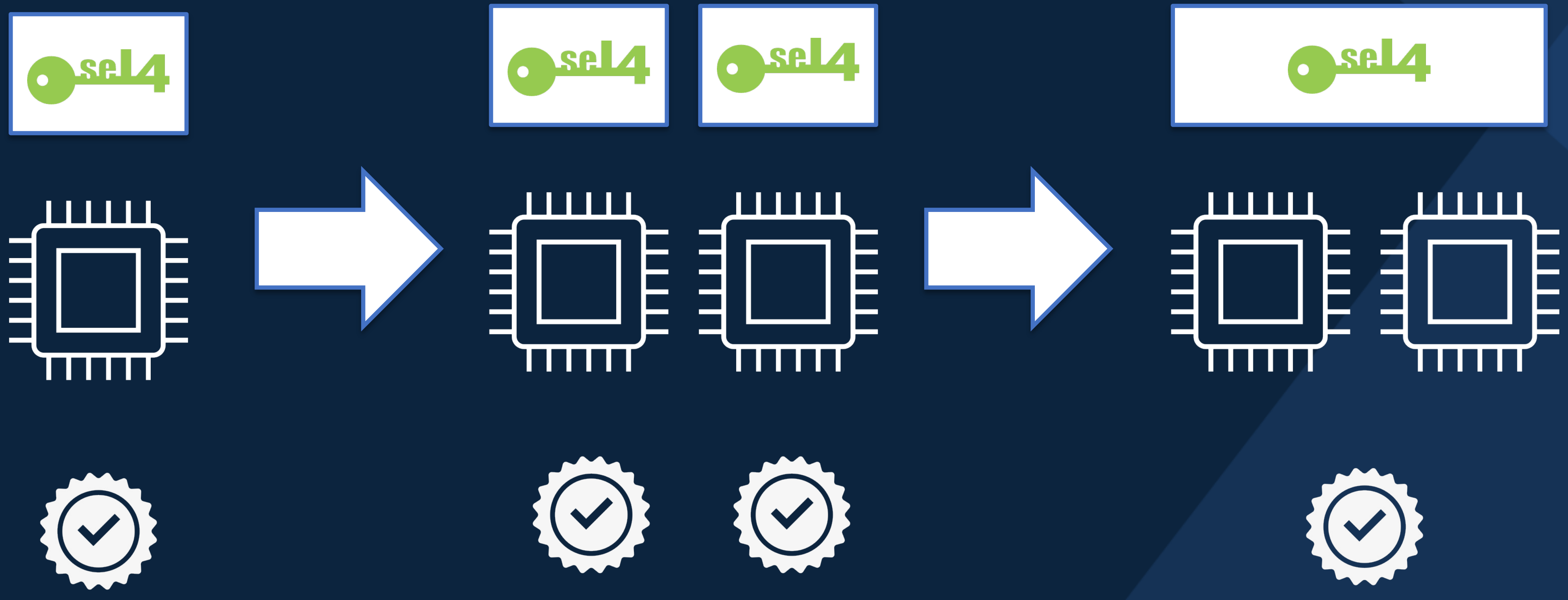
SMP seL4 configurations



Unicore

SMP

(Re)Introducing: Partitioned multikernel



Unicore

Multikernel (AMP)

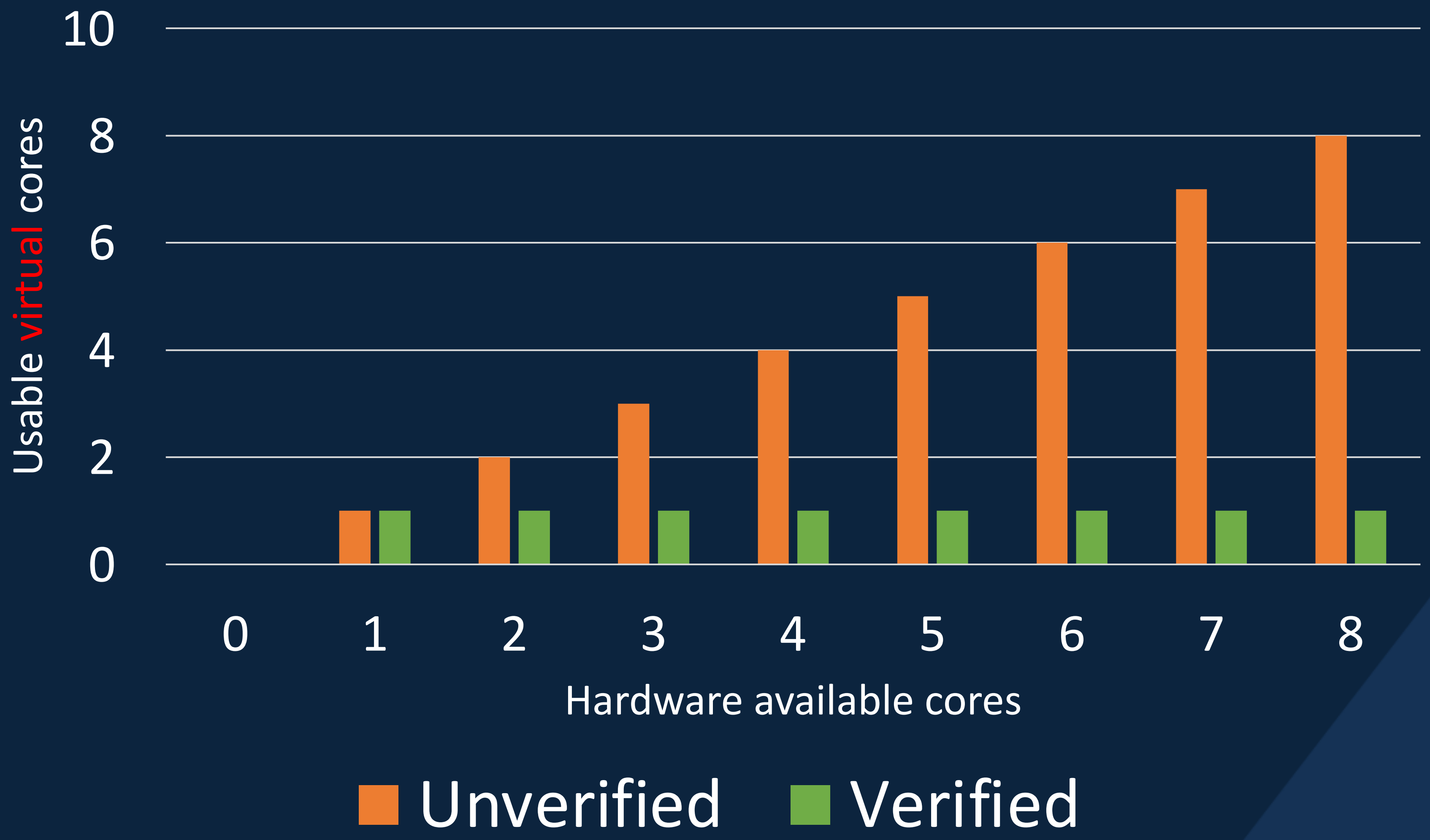
SMP

Last year:

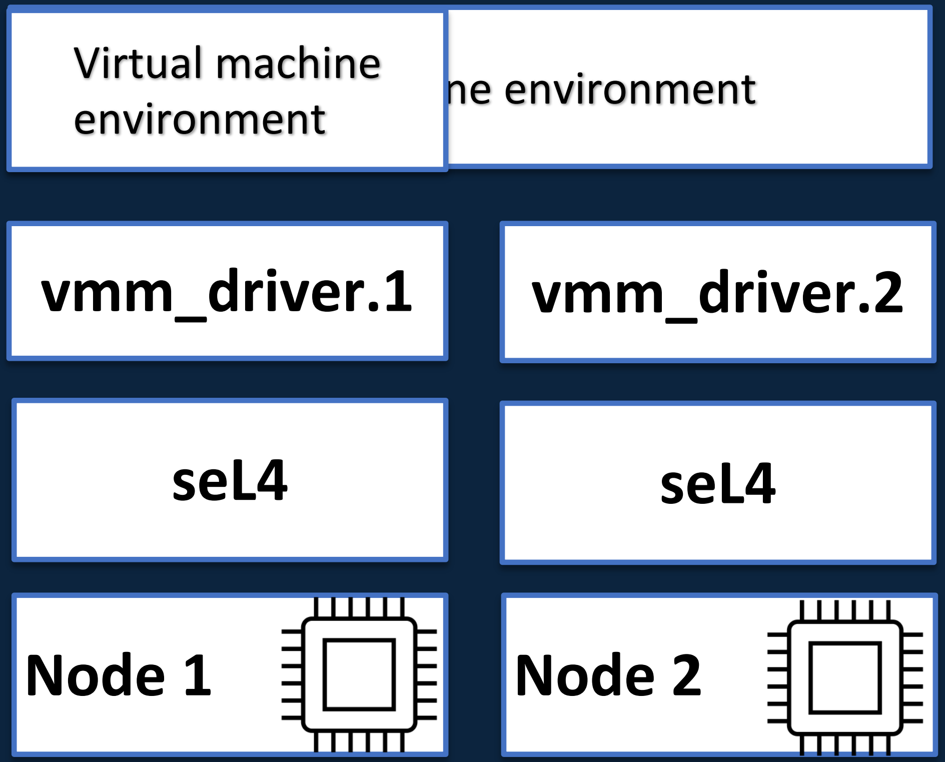
Follow-up steps

- SMP-like user apps
- Scalable cross-core notifications
- Investigating impact of replicated data on shared caches
- Transparent cross-core seL4RPCCall CAmkES connectors
- Finish off multi-vm multicore example

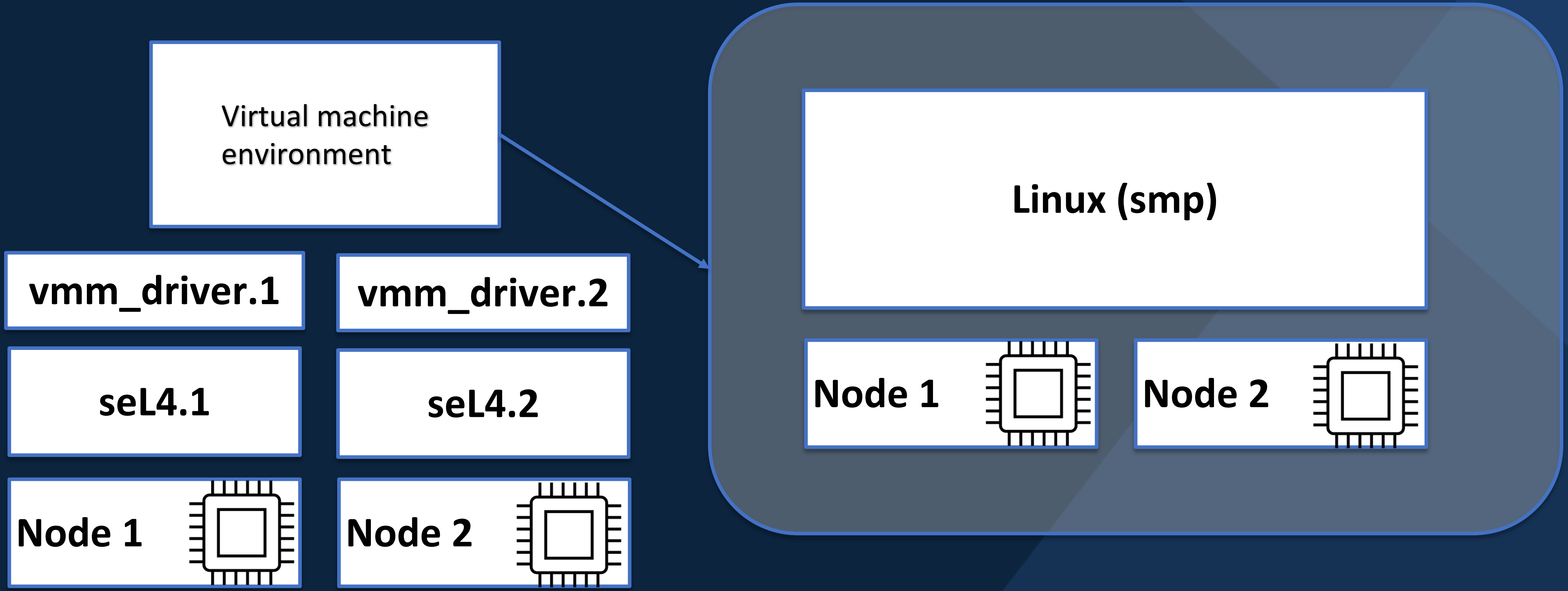
Usable vCPU count by kernel configuration



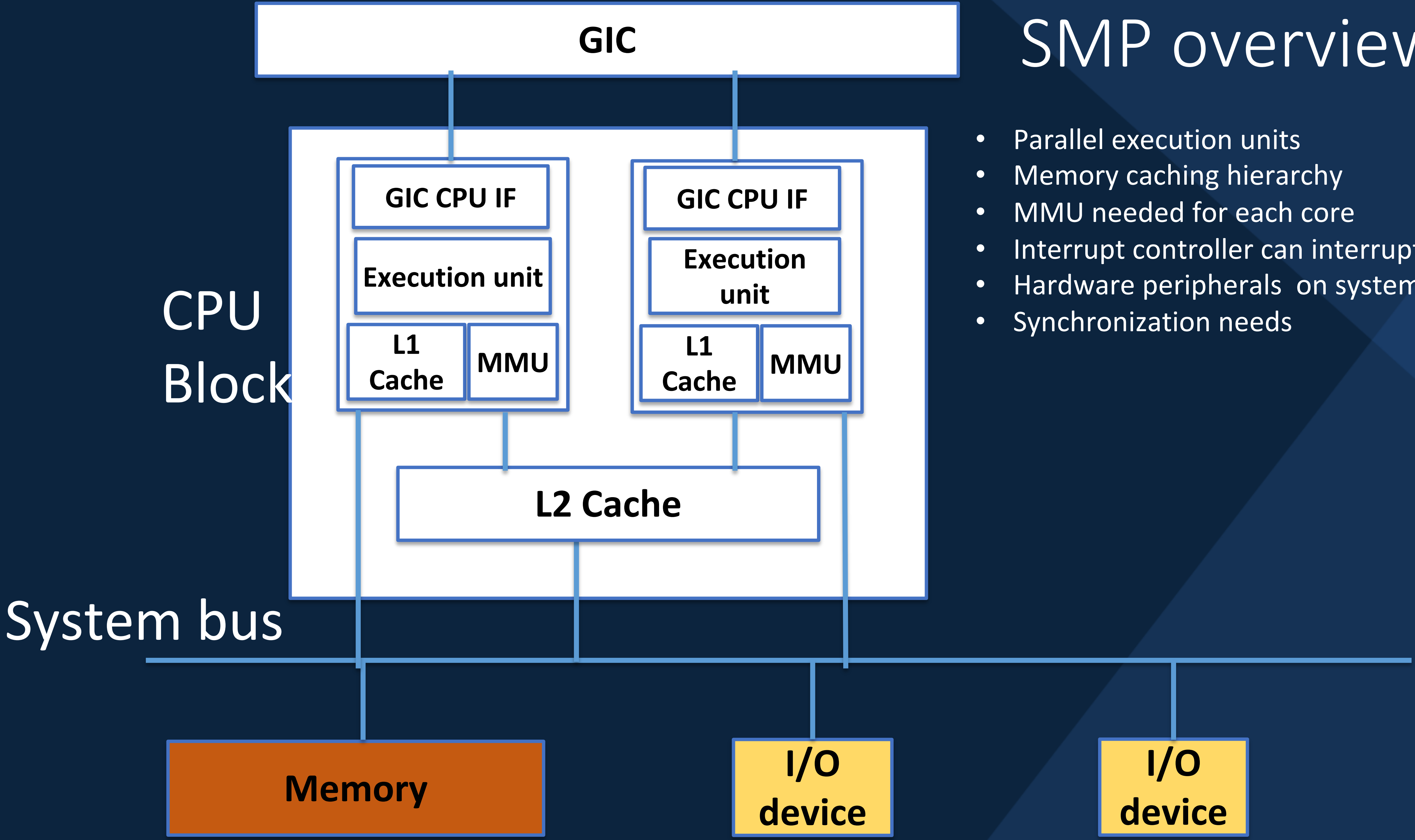
SMP guests on partitioned hosts



SMP guests on partitioned hosts



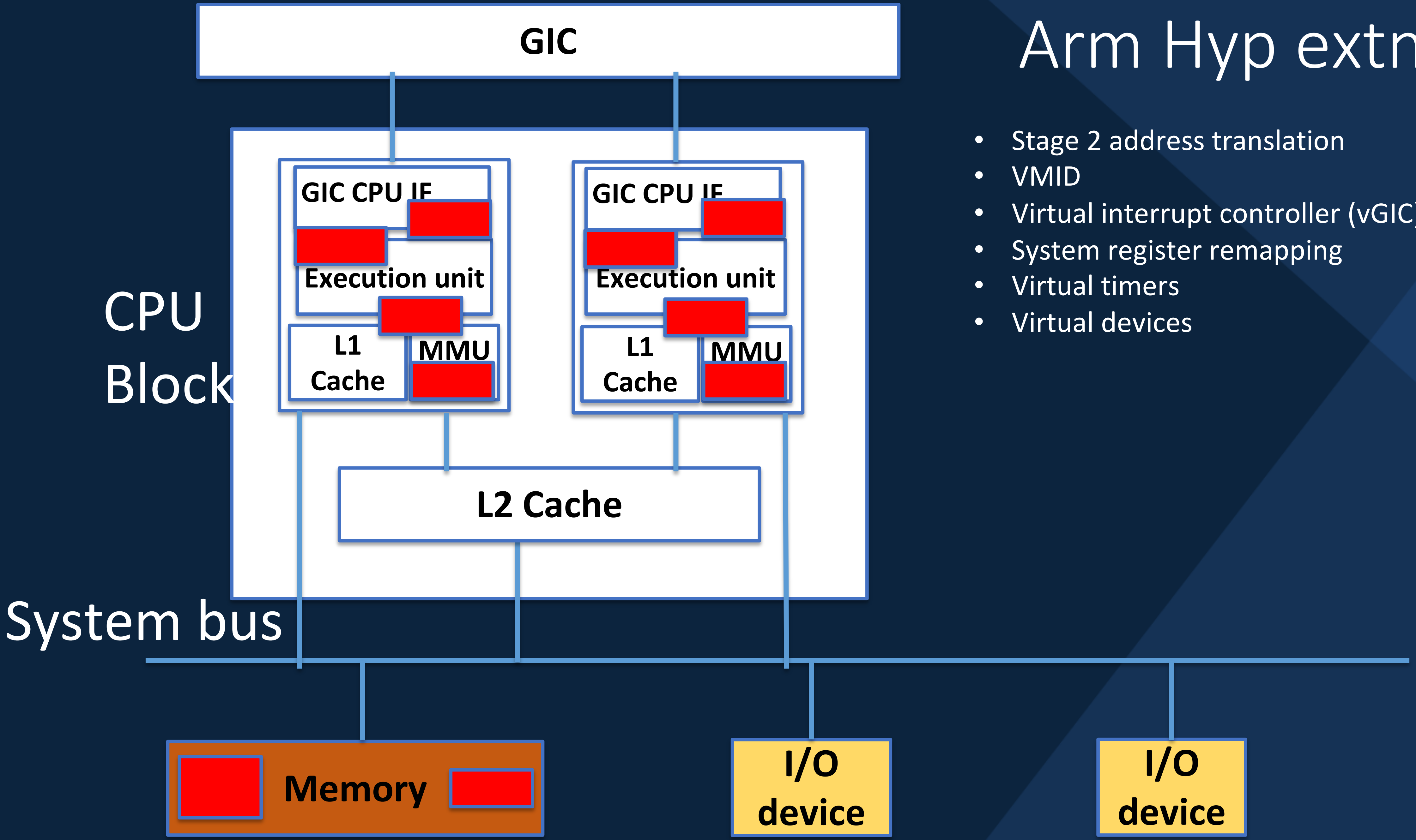
SMP overview



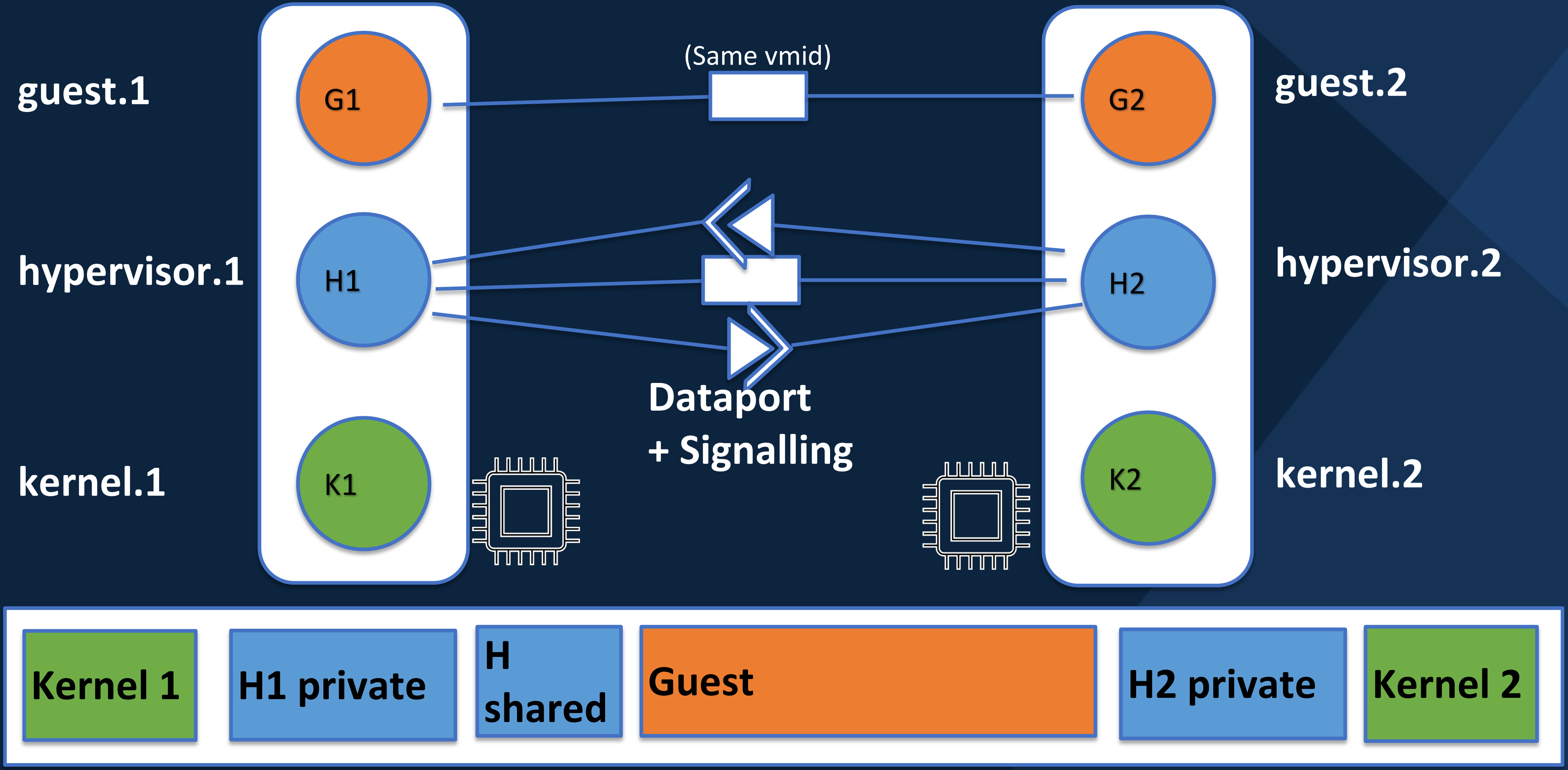
- Parallel execution units
- Memory caching hierarchy
- MMU needed for each core
- Interrupt controller can interrupt per core
- Hardware peripherals on system bus
- Synchronization needs

Arm Hyp extns.

- Stage 2 address translation
- VMID
- Virtual interrupt controller (vGIC)
- System register remapping
- Virtual timers
- Virtual devices



Like a distributed system (with shared mem)



guest.1

hypervisor.1

kernel.1

G1

H1

K1

(Same vmid)

Dataport
+ Signalling

G2

H2

K2

guest.2

hypervisor.2

kernel.2

Kernel 1

H1 private

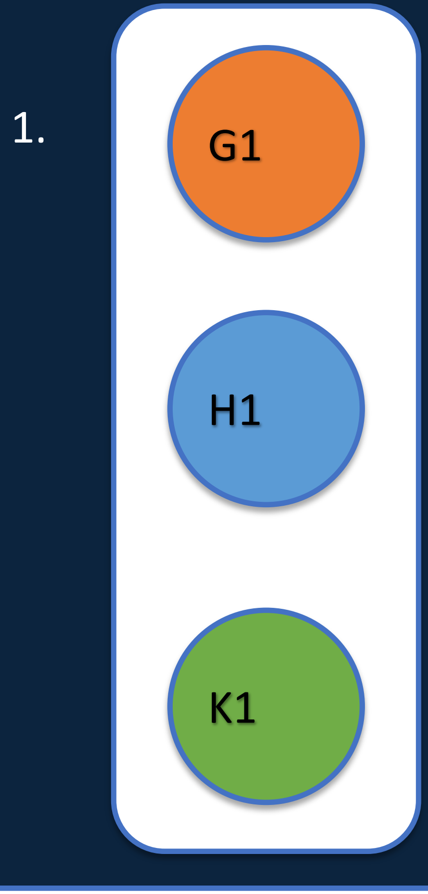
H
shared

Guest

H2 private

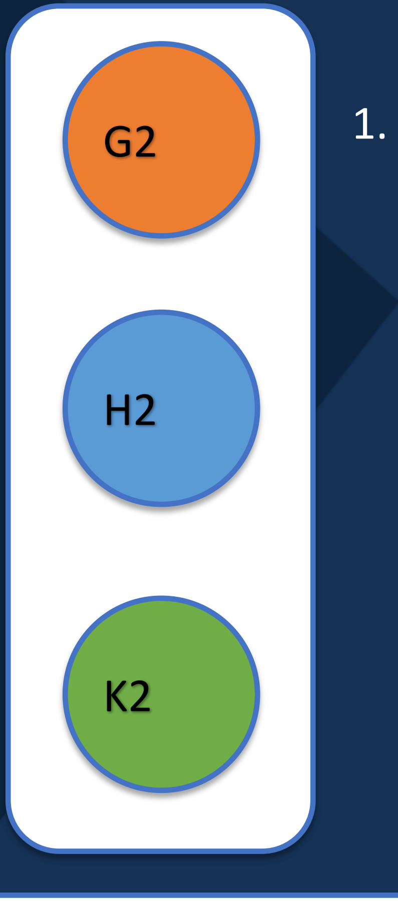
Kernel 2

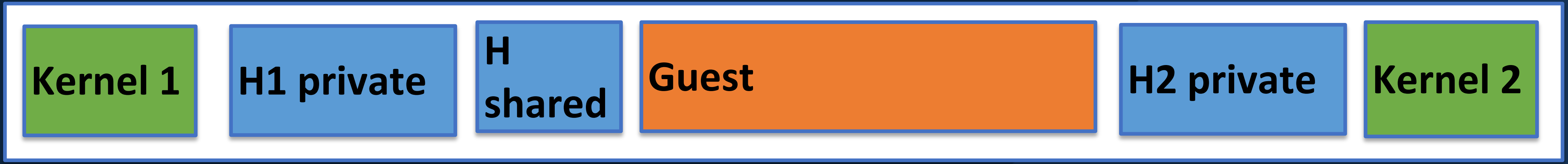
Regular operation



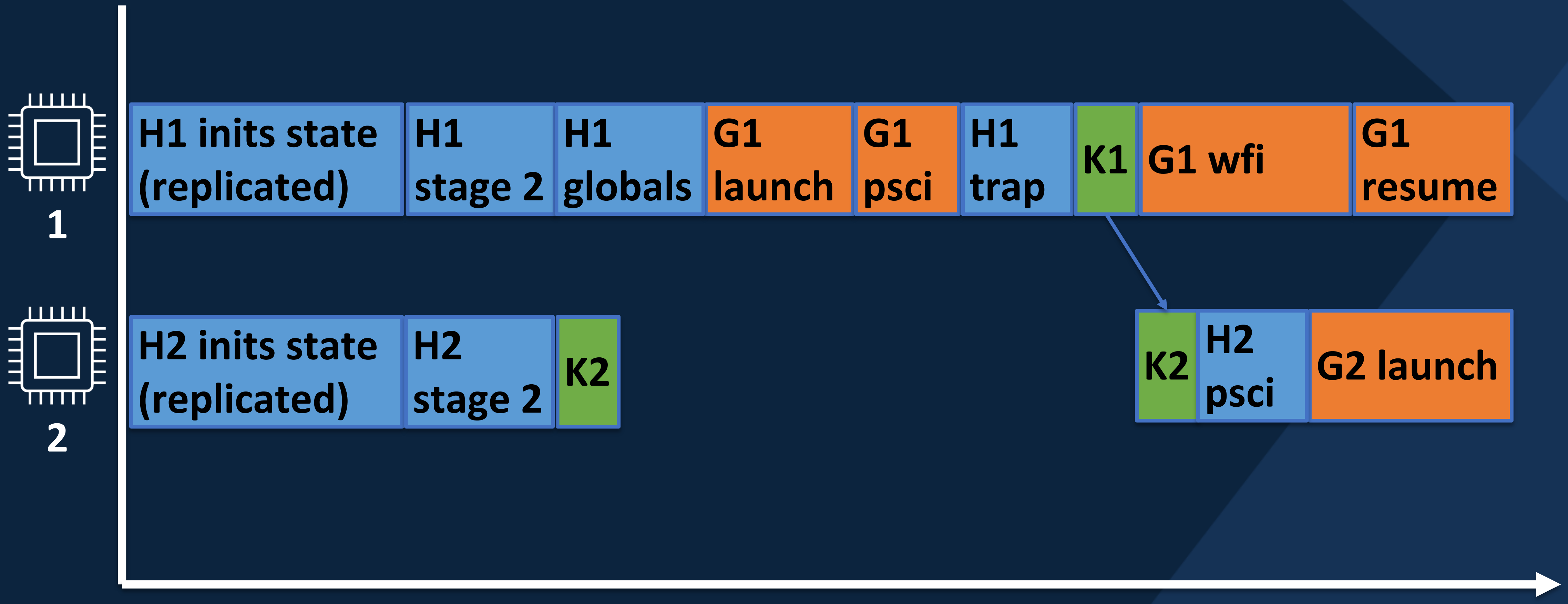
1. Both cores are running in the guest

- Regular ISA operations are largely the same
- Memory operations work the same way
- Process context switching
- TLB shootdown
- Cache invalidation
- Spin lock synchronisation



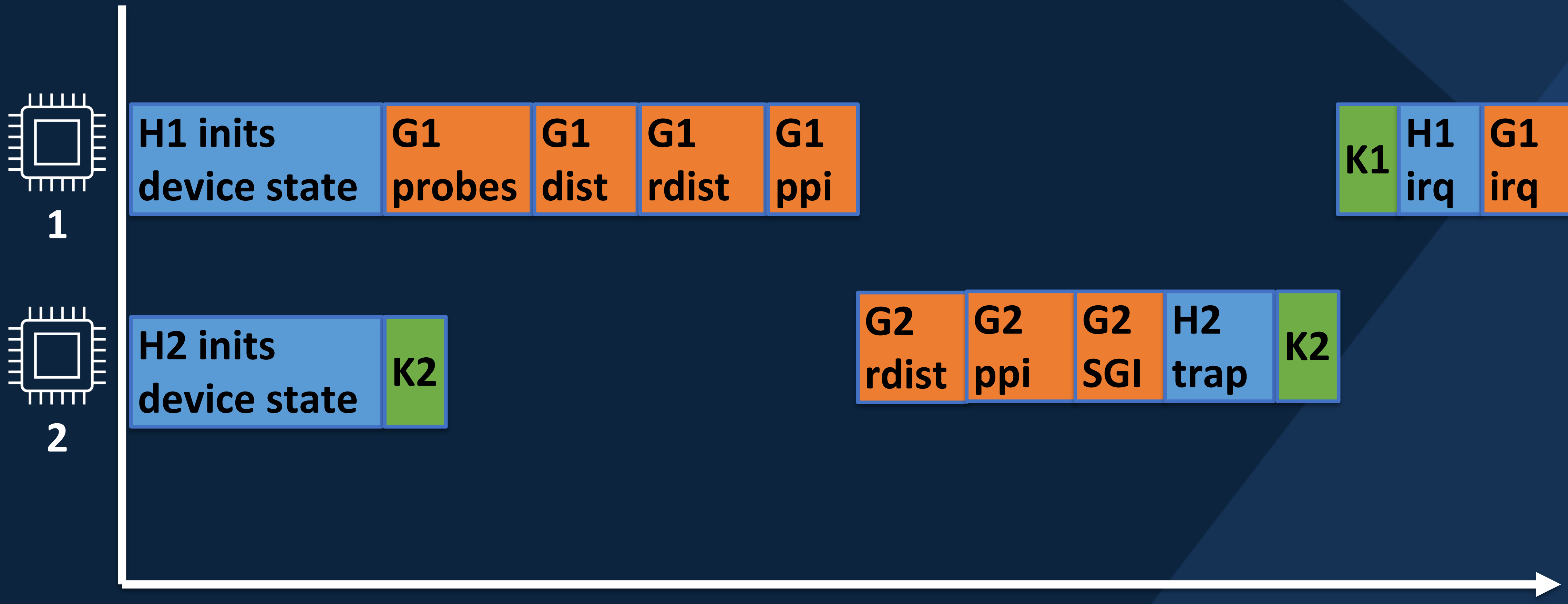


Initialization



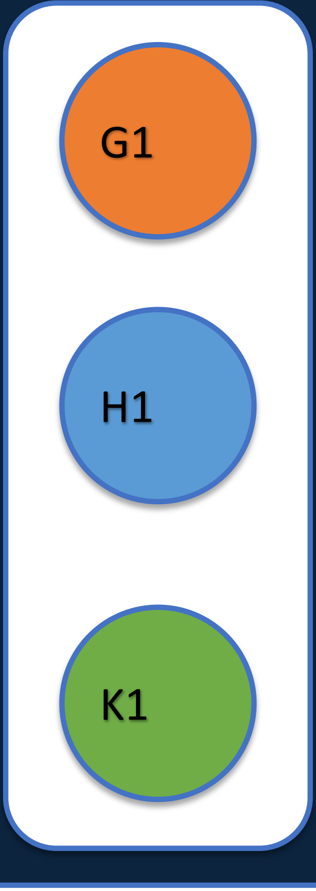
Time since boot

Virtual Interrupt controller (vGIC)

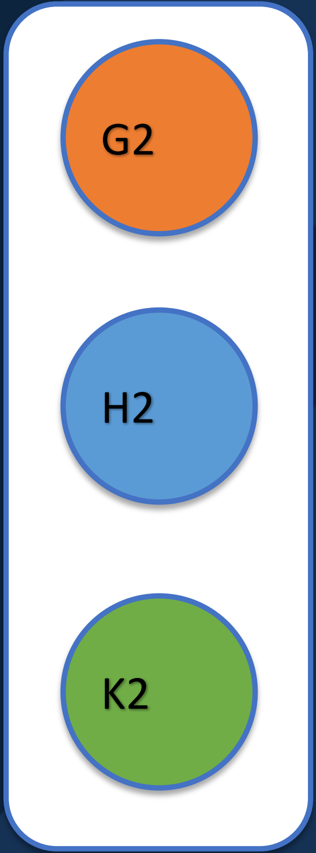


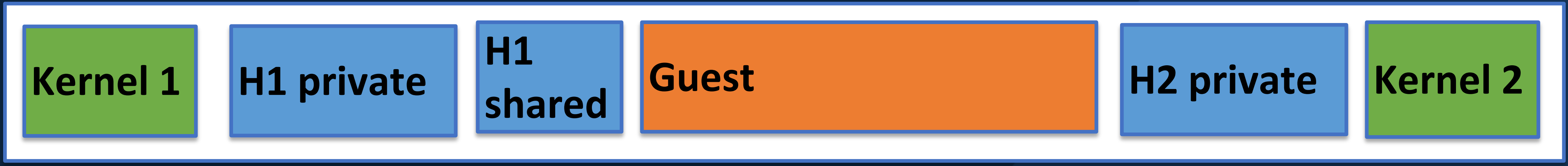
Time since boot

Other Examples



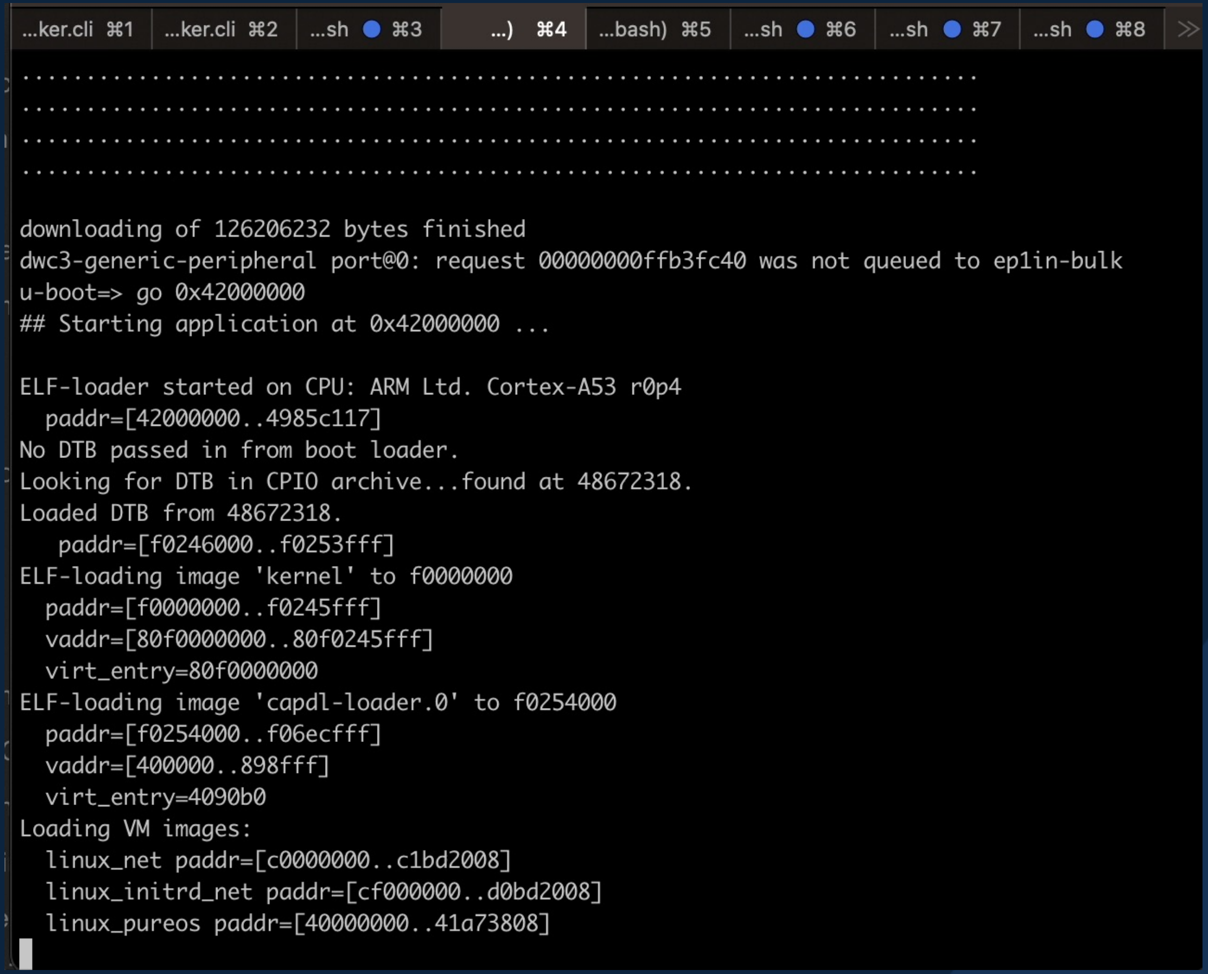
- Virtual devices eg virtio_net
- Cross platform communication
- Privileged operations (SMC calls or protected device access)





Does it work?

Yes



```
..........................................................................
..........................................................................
..........................................................................
..........................................................................


downloading of 126206232 bytes finished
dwc3-generic-peripheral port@0: request 00000000ffb3fc40 was not queued to ep1in-bulk
u-boot=> go 0x42000000
## Starting application at 0x42000000 ...

ELF-loader started on CPU: ARM Ltd. Cortex-A53 r0p4
  paddr=[42000000..4985c117]
No DTB passed in from boot loader.
Looking for DTB in CPIO archive...found at 48672318.
Loaded DTB from 48672318.
   paddr=[f0246000..f0253fff]
ELF-loading image 'kernel' to f0000000
  paddr=[f0000000..f0245fff]
  vaddr=[80f0000000..80f0245fff]
  virt_entry=80f0000000
ELF-loading image 'capdl-loader.0' to f0254000
  paddr=[f0254000..f06ecfff]
  vaddr=[400000..898fff]
  virt_entry=4090b0
Loading VM images:
  linux_net paddr=[c0000000..c1bd2008]
  linux_initrd_net paddr=[cf000000..d0bd2008]
  linux_pureos paddr=[40000000..41a73808]
```

Discussion and extensions

- Dynamic systems
- multiple vms.
- power on and off cores
- Where are scalability issues?
- How trustworthy can VMM layer be?