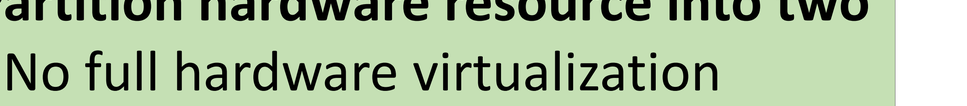
Toward On-demand Nested Virtualization for Live-Refreshing Cloud Systems

Ryosuke Yasuoka (University of Tsukuba), Takaaki Fukai, Takahiro Shinagawa (The University of Tokyo)

Background

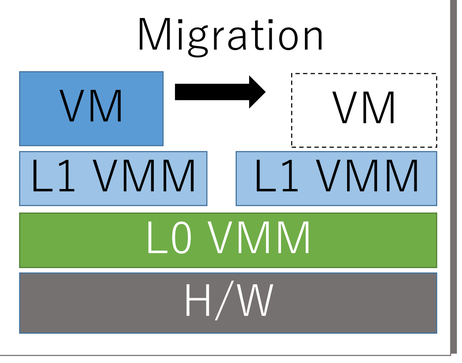
- VMM needs to be refreshed (=rebooted) for applying patches, upgrading, rejuvenation
- VMs live migrated avoids stopping VMs
- Problem: Heavy network load



Traditional nested virtualization

Migrate VMs in the same machine

- ⭕ No network load
- ✖ Overhead by nested virtualization



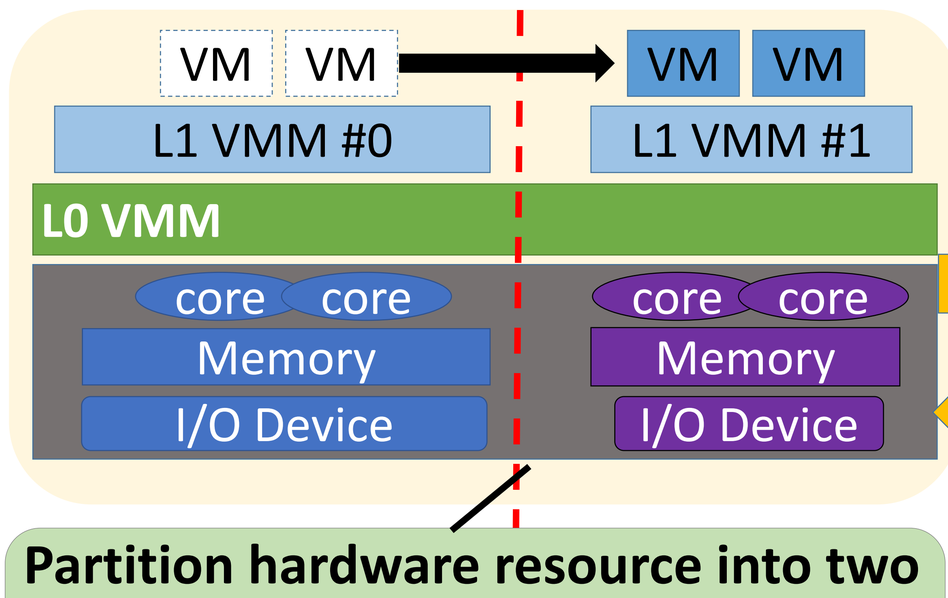
Our goal

(1) Keep the advantages of nested virtualization

(2) Eliminate the nested virtualization overhead during normal runtime

Proposal: **On-demand Nested Virtualization with hardware partitioning**

In refreshing the VMM



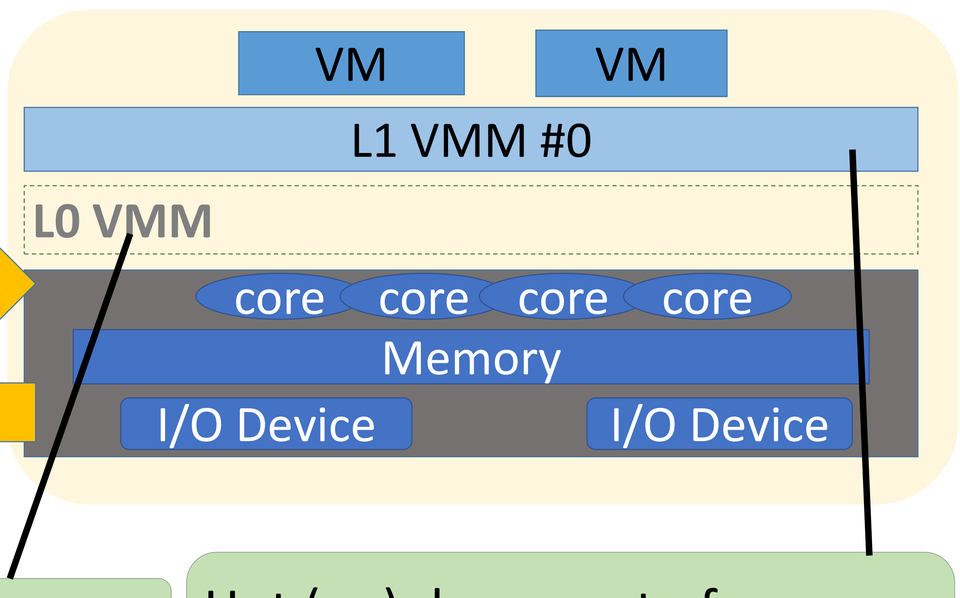
Partition hardware resource into two

- No full hardware virtualization
- **Keep the same hardware interface**

De-Virtualization

virtualization

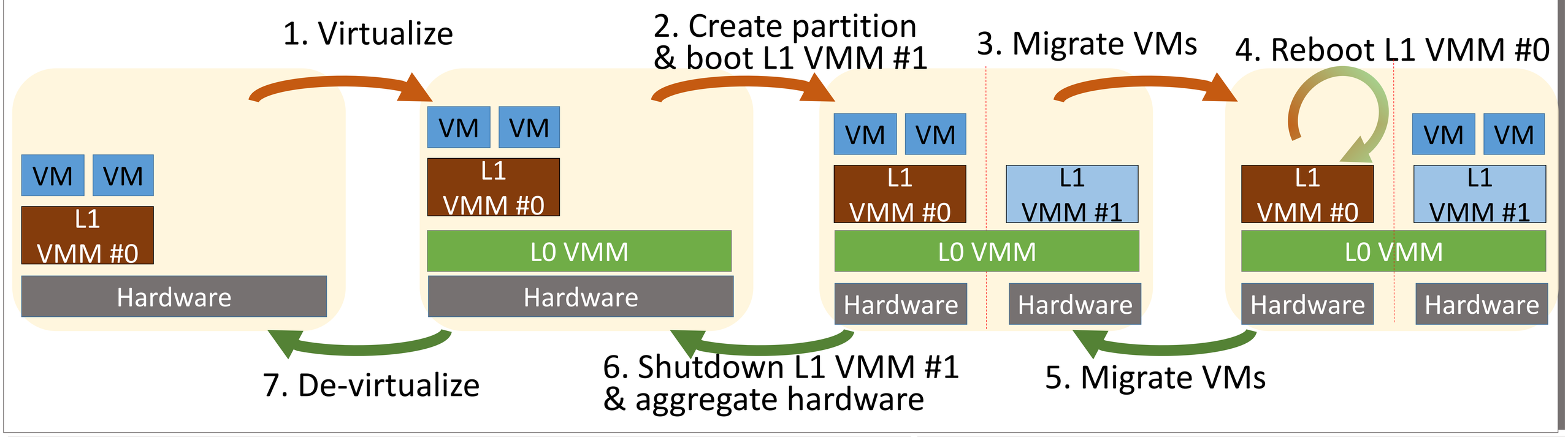
During normal runtime



Completely stops

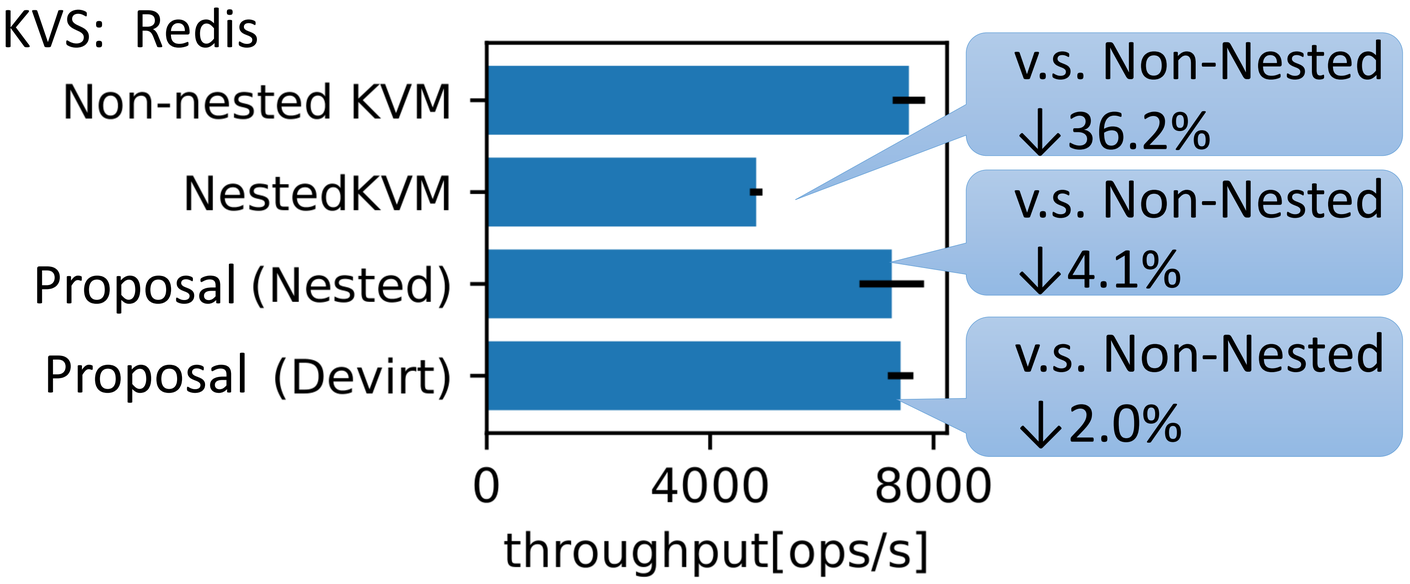
Hot (un)plug a part of hardware on (de-)virtualizing

Refreshing operation



Performance evaluation: KVS throughput

- Workload: YCSB benchmark (Read : Update = 50%:50%)
- KVS: Redis



Implementation

- L0 VMM: Based on TinyVisor [1] (diff: +3318, -48)
- L1 VMM #0: Custom KVM (diff: +292, -0)
- L1 VMM #1 and L2 guest OS: no change

[1] https://osdn.net/projects/tinyvisor/

Future work

- Performance measurement on VMM refreshing
- Gradual resource reallocation
- Zero-copy migration