NUTANIX™

Storage Performance Challenges *in Virtualisation*

Dr Felipe Franciosi
AHV Engineering Lead

8 August 2018

Disclaimer

This presentation and the accompanying oral commentary may include express and implied forward-looking statements, including but not limited to statements concerning our business plans and objectives, product features and technology that are under development or in process and capabilities of such product features and technology, our plans to introduce product features in future releases, the implementation of our products on additional hardware platforms, strategic partnerships that are in process, product performance, competitive position, industry environment, and potential market opportunities. These forward-looking statements are not historical facts, and instead are based on our current expectations, estimates, opinions and beliefs. The accuracy of such forward-looking statements depends upon future events, and involves risks, uncertainties and other factors beyond our control that may cause these statements to be inaccurate and cause our actual results, performance or achievements to differ materially and adversely from those anticipated or implied by such statements, including, among others: failure to develop, or unexpected difficulties or delays in developing, new product features or technology on a timely or cost-effective basis; delays in or lack of customer or market acceptance of our new product features or technology; the failure of our software to interoperate on different hardware platforms; failure to form, or delays in the formation of, new strategic partnerships and the possibility that we may not receive anticipated results from forming such strategic partnerships; the introduction, or acceleration of adoption of, competing solutions, including public cloud infrastructure; a shift in industry or competitive dynamics or customer demand; and other risks detailed in our Form 10-Q for the fiscal quarter ended April 30, 2017, filed with the Securities and Exchange Commission. These forward- looking statements speak only as of the date of this presentation and, except as required by law, we assume no obligation to update forward- looking statements to reflect actual results or subsequent events or circumstances. Any future product or roadmap information is intended to outline general product directions, and is not a commitment, promise or legal obligation for Nutanix to deliver any material, code, or functionality. This information should not be used when making a purchasing decision. Further, note that Nutanix has made no determination as to if separate fees will be charged for any future product enhancements or functionality which may ultimately be made available. Nutanix may, in its own discretion, choose to charge separate fees for the delivery of any product enhancements or functionality which are ultimately made available.

Certain information contained in this presentation and the accompanying oral commentary may relate to or be based on studies, publications, surveys and other data obtained from third-party sources and our own internal estimates and research. While we believe these third-party studies, publications, surveys and other data are reliable as of the date of this presentation, they have not independently verified, and we make no representation as to the adequacy, fairness, accuracy, or completeness of any information obtained from third-party sources.

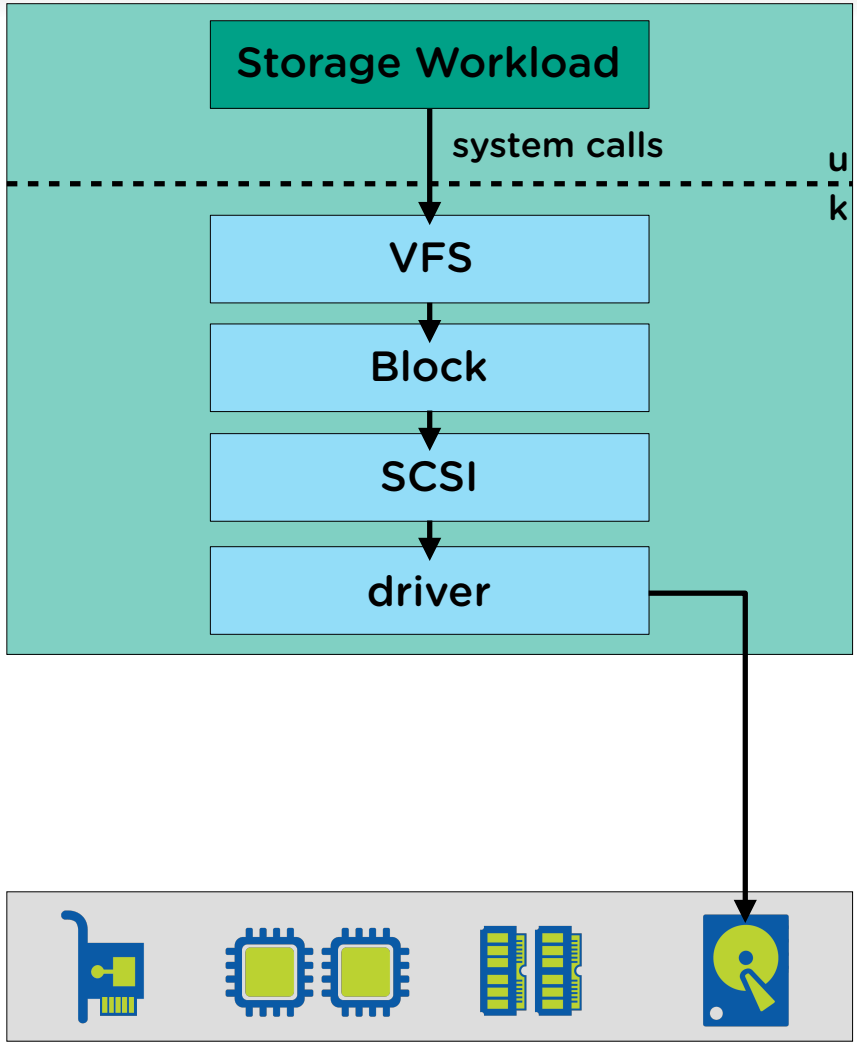
NUTANIX

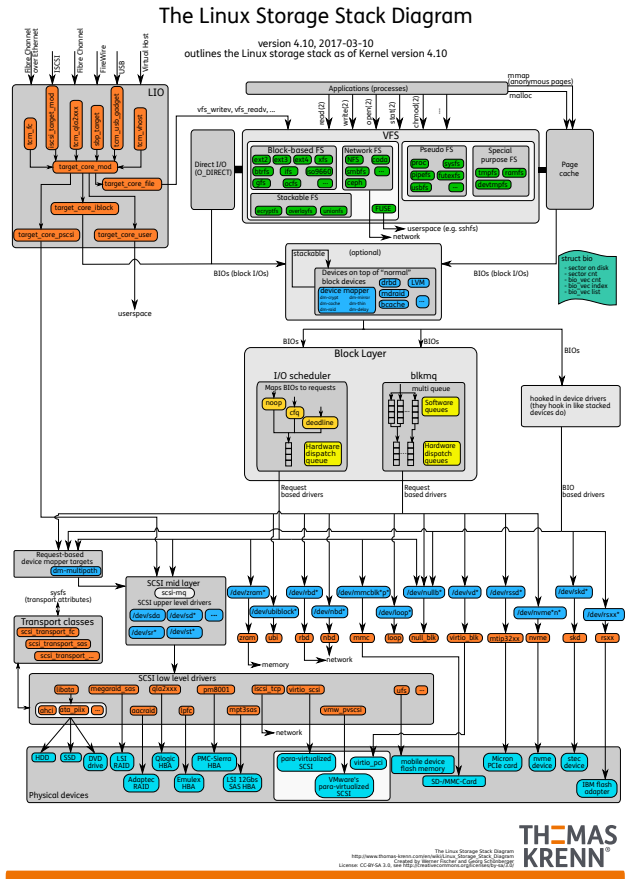
Agenda

- Virtualisation overhead for storage workloads
  - Storage performance challenges for virtual machines
  - Understanding the virtualisation overhead
- Hypervisor Analysis
  - Review of how hypervisors virtualise storage
  - Nutanix AHV: Leveraging storage multi-queue and SPDK
- Userspace FTW
  - Leaner software means better performance
  - Making the most of NVMe and 3DXP

NUTANIX

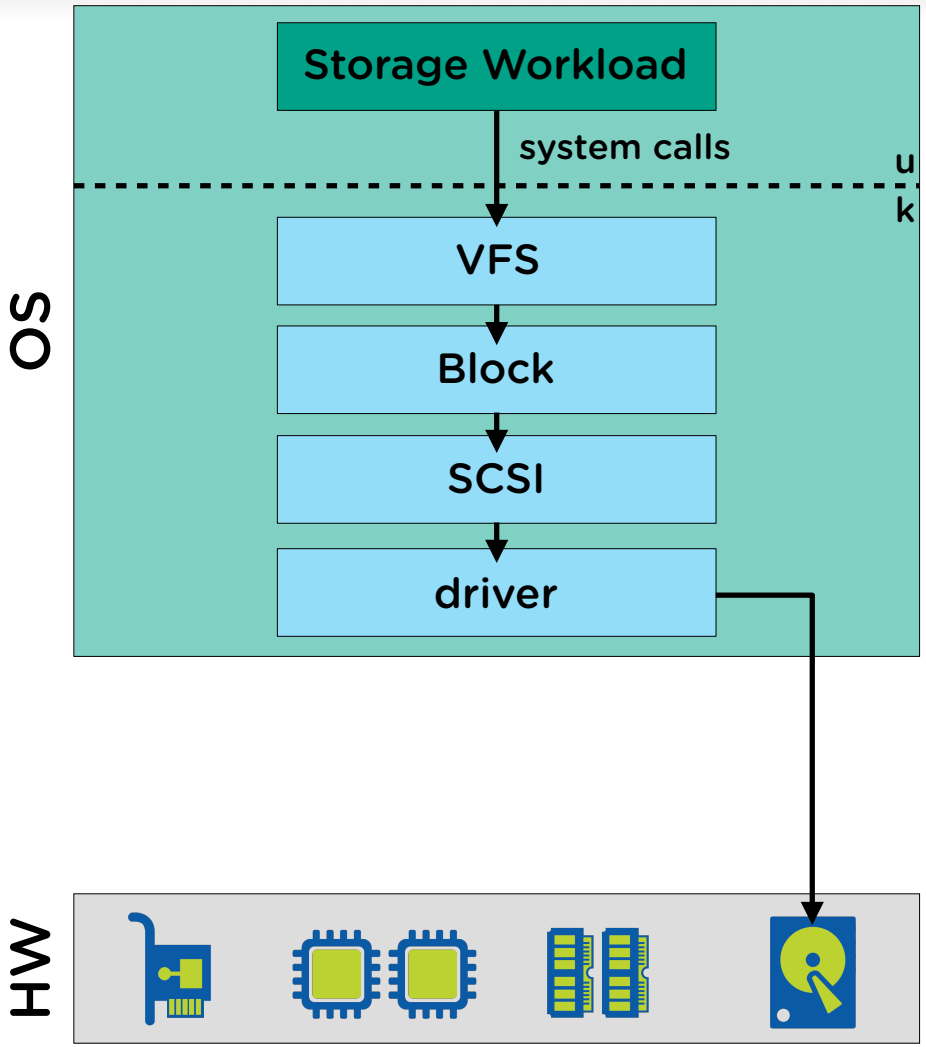
Storage Access and Performance





Strongly encouraged read:
https://www.thomas-krenn.com/en/wiki/Linux_Storage_Stack_Diagram

Storage Access and Performance

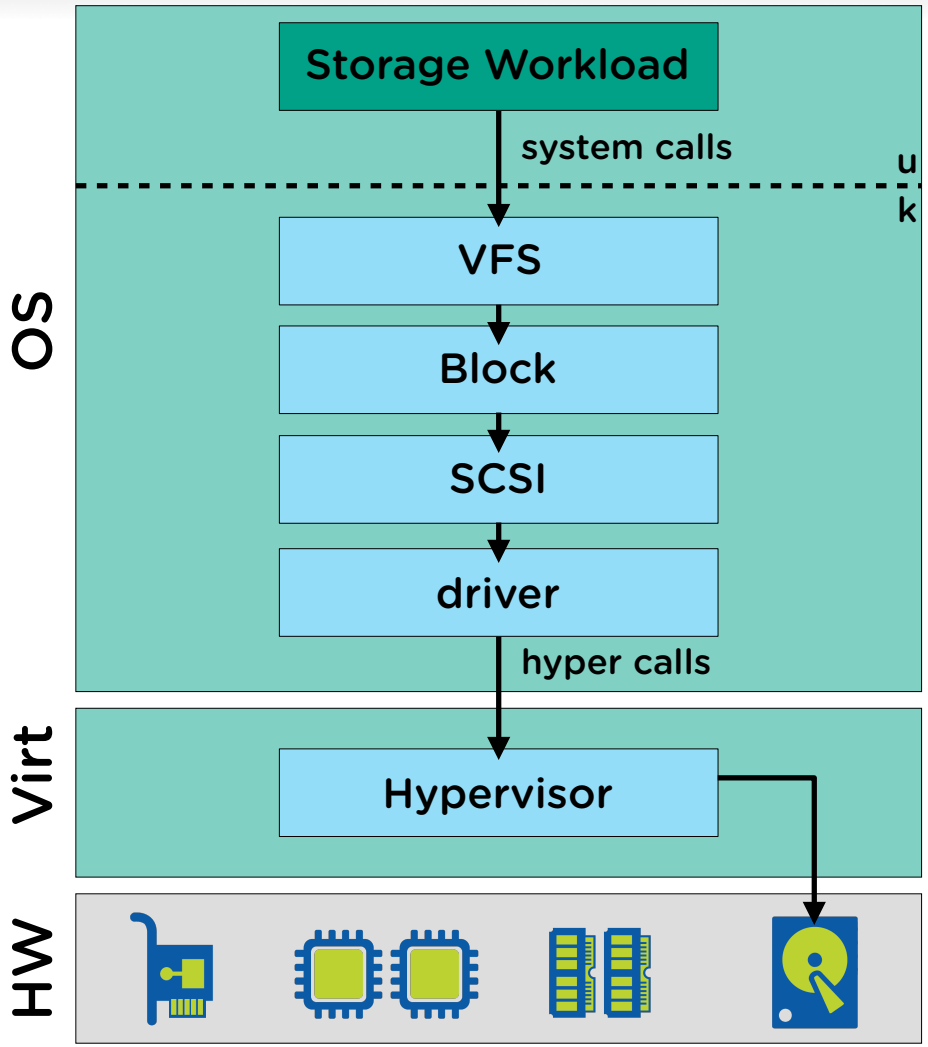


Where did time go?

Time spent on CPU is in order of microseconds.

Time spent on disks is in order of milliseconds.

Storage Access and Performance



~ µs

~ µs

~ ms

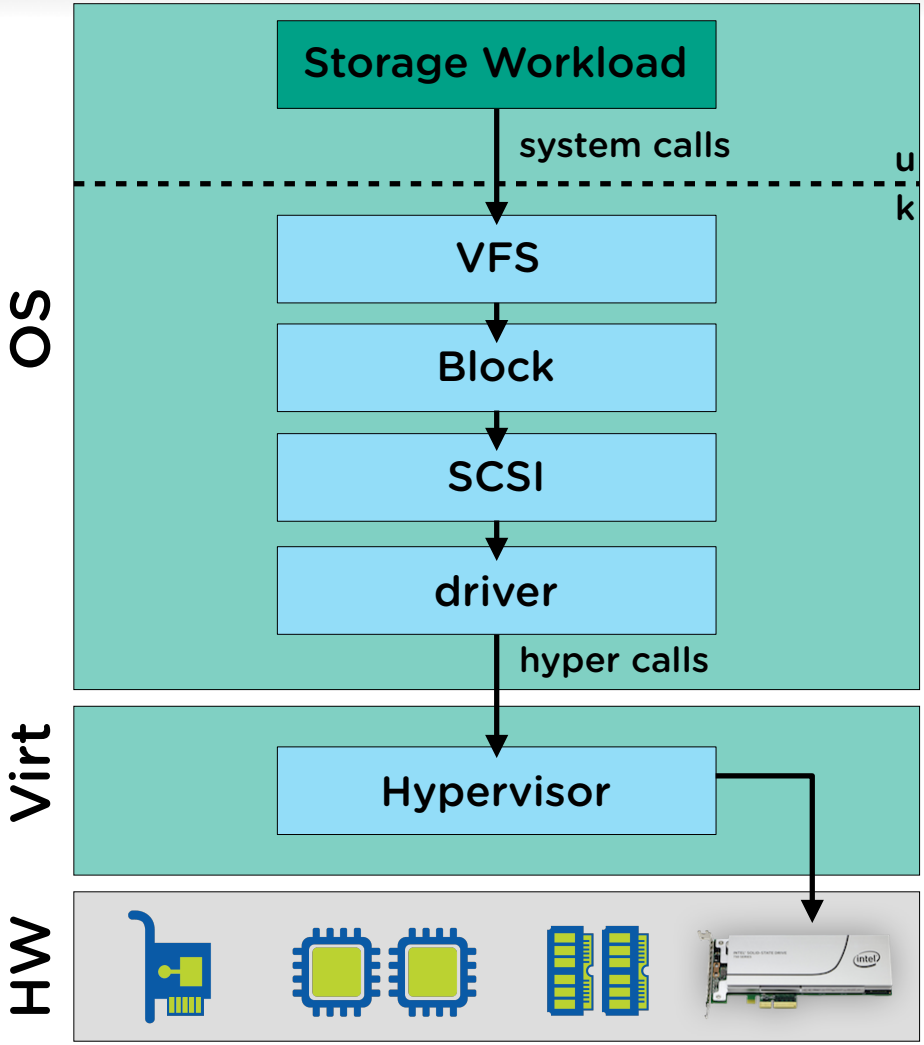
Where did time go?

Time spent on CPU is in order of microseconds.

Hypervisor adds some more microseconds.

Time spent on disks is in order of milliseconds.

Storage Access and Performance



~ µs

~ µs

~ µs

Where did time go?

Time spent on CPU is in order of microseconds.

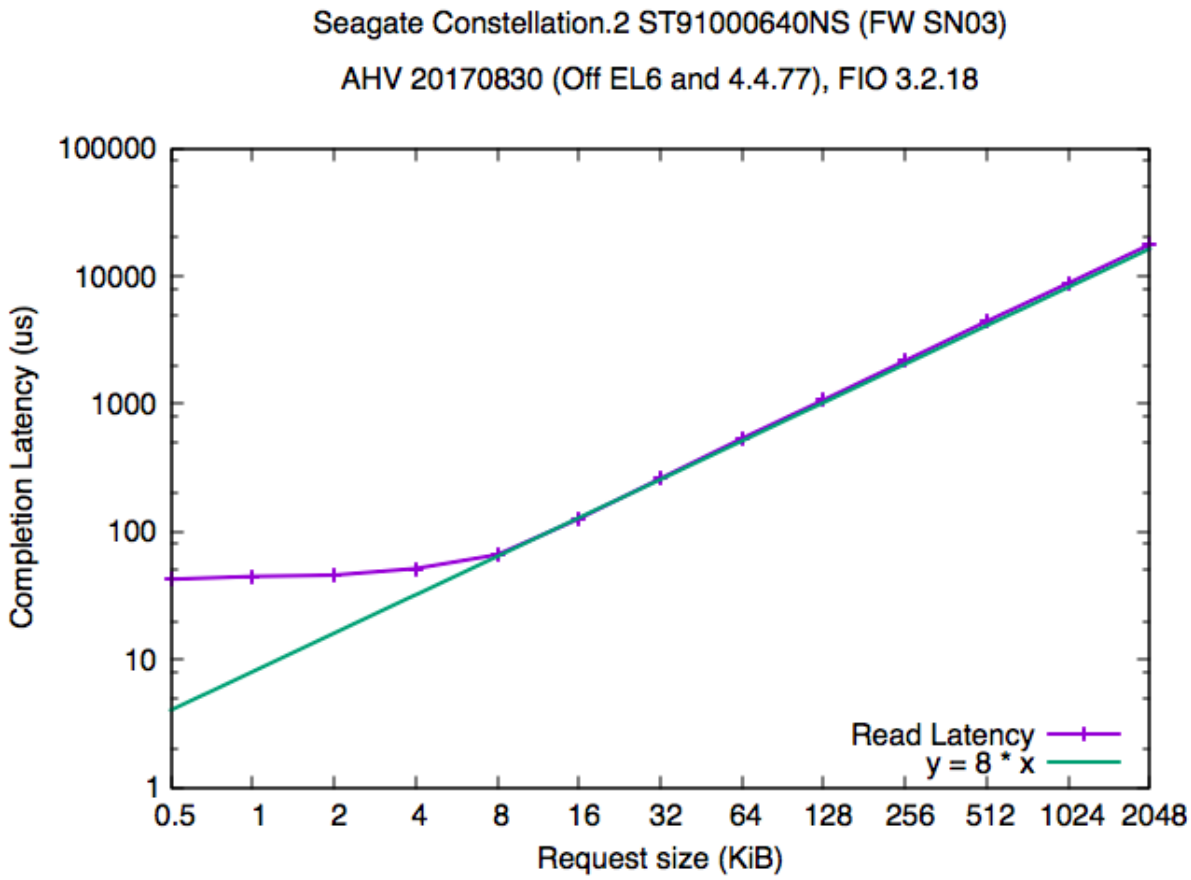
Hypervisor adds some more microseconds.

Most NVMe: latency is in order of microseconds.

Storage Access and Performance

What does it mean to saturate the storage?

- Mechanical drive
- Sequential reads
- Queue depth = 1
- Varying request size



Storage is saturated.

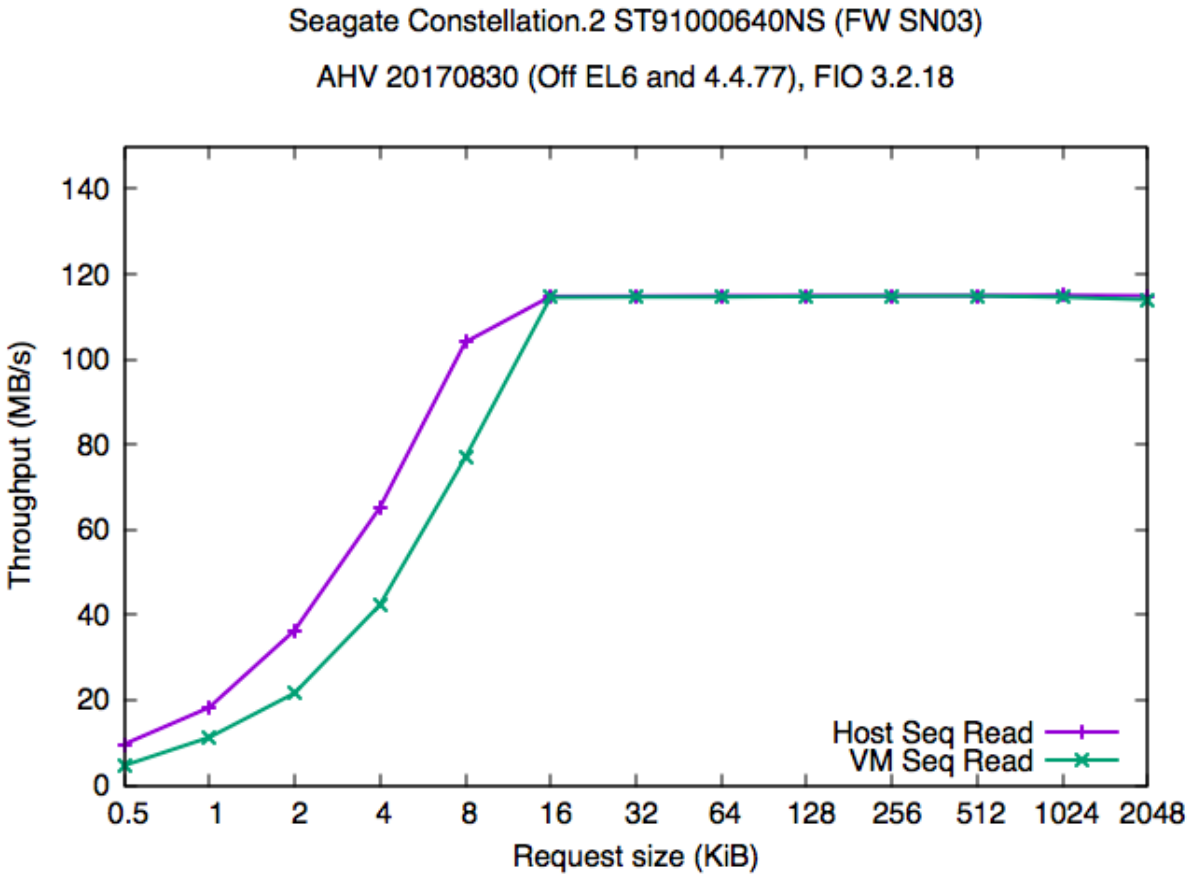
Storage Performance and Virtualisation

How does that translate to throughput?

- Mechanical drive
- Sequential reads
- Queue depth = 1
- Varying request size

And from a VM ?

- Debian 9.4 VM (FIO 3.2.18)
- Host with Qemu 2.6
- Disk over virtio-scsi

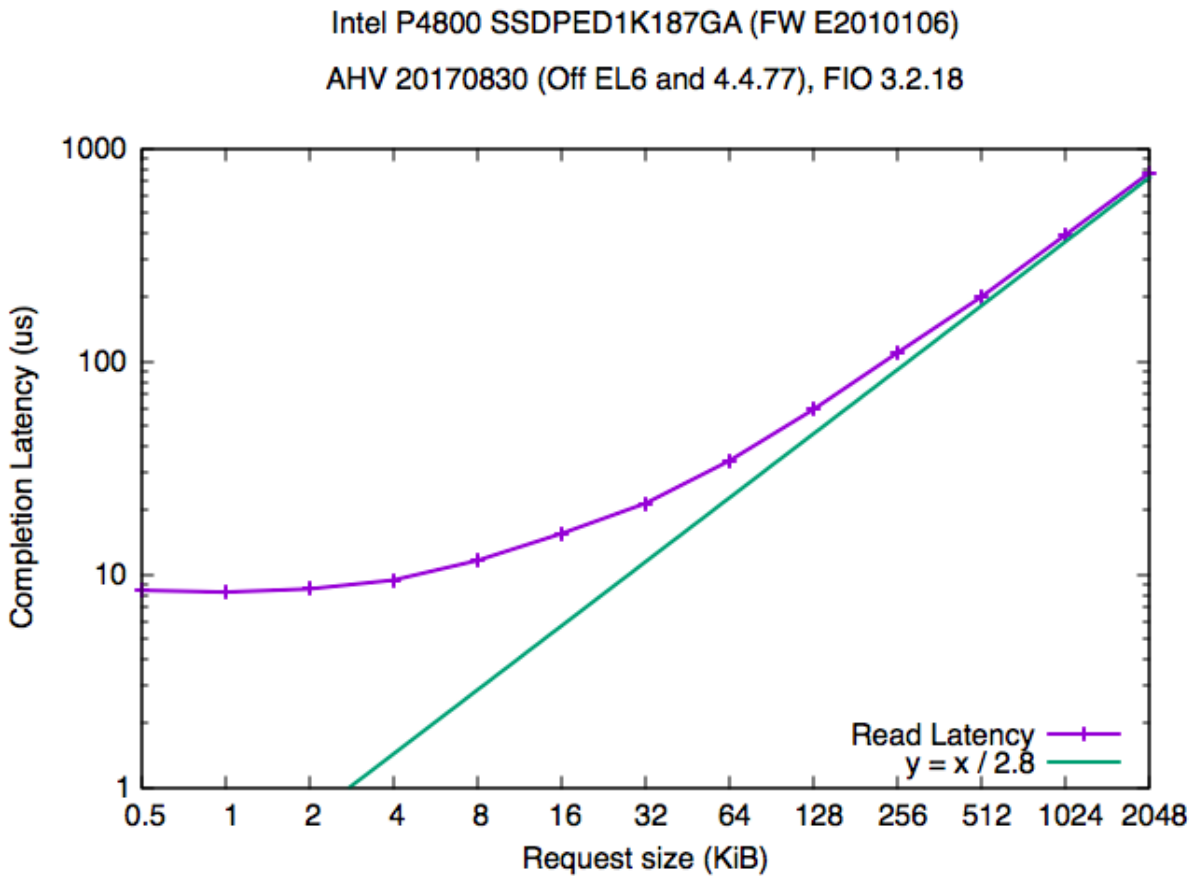


Storage Performance and Virtualisation

What about modern storage devices?

- NVMe w/ 3DXP
- Sequential reads
- Queue depth = 1
- Varying request size

Storage is NOT saturated



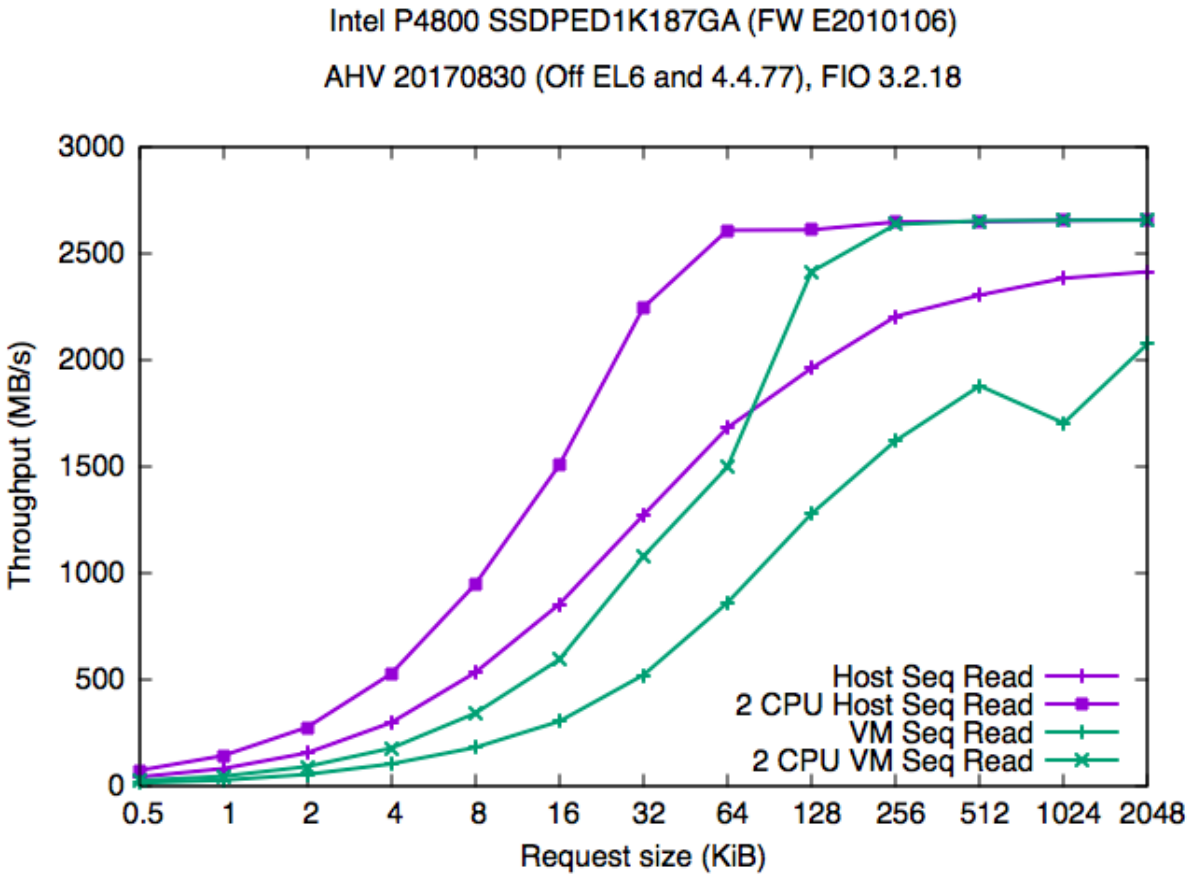
Storage Performance and Virtualisation

How does that translate to throughput?

- NVMe w/ 3DXP
- Sequential reads
- Queue depth = 1 (per CPU)
- Varying request size

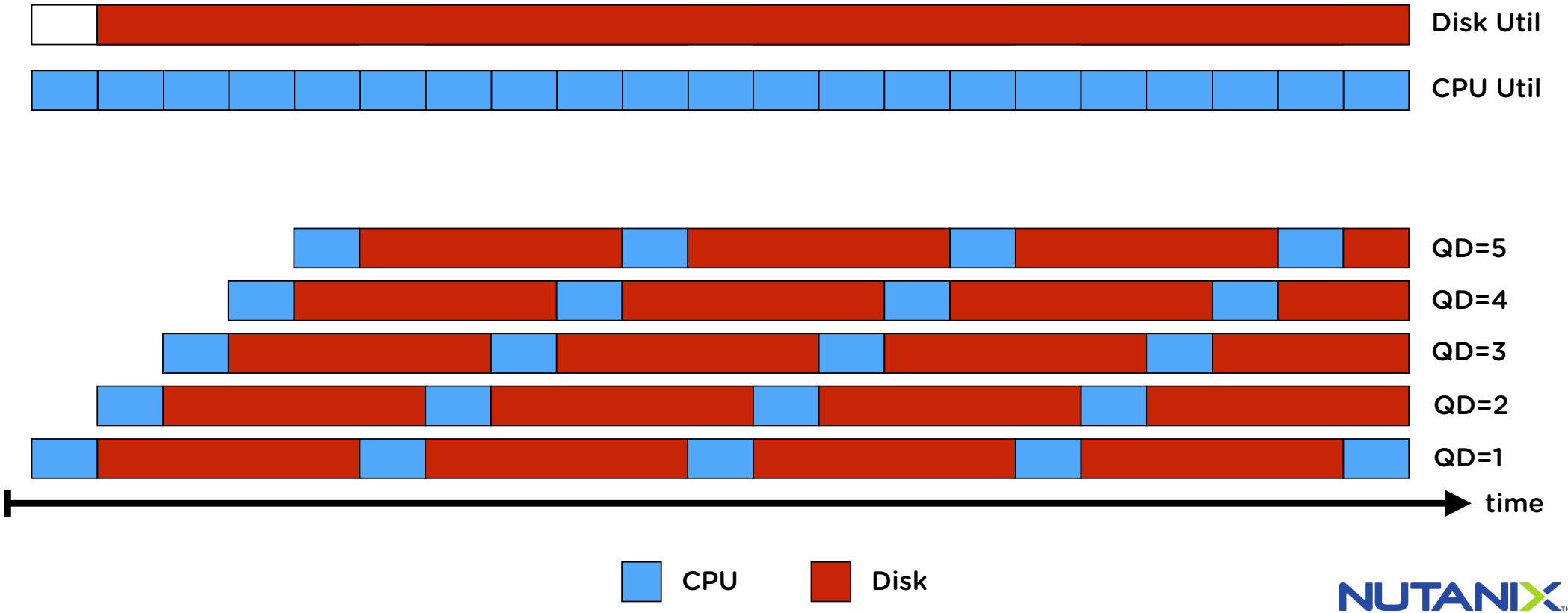
And from a VM ?

- Debian 9.4 VM (FIO 3.2.18)
- Host with Qemu 2.6
- Disk over virtio-scsi



Saturating CPUs and Storage

NVMe is "parallel", a single CPU is not.



NUTANIX

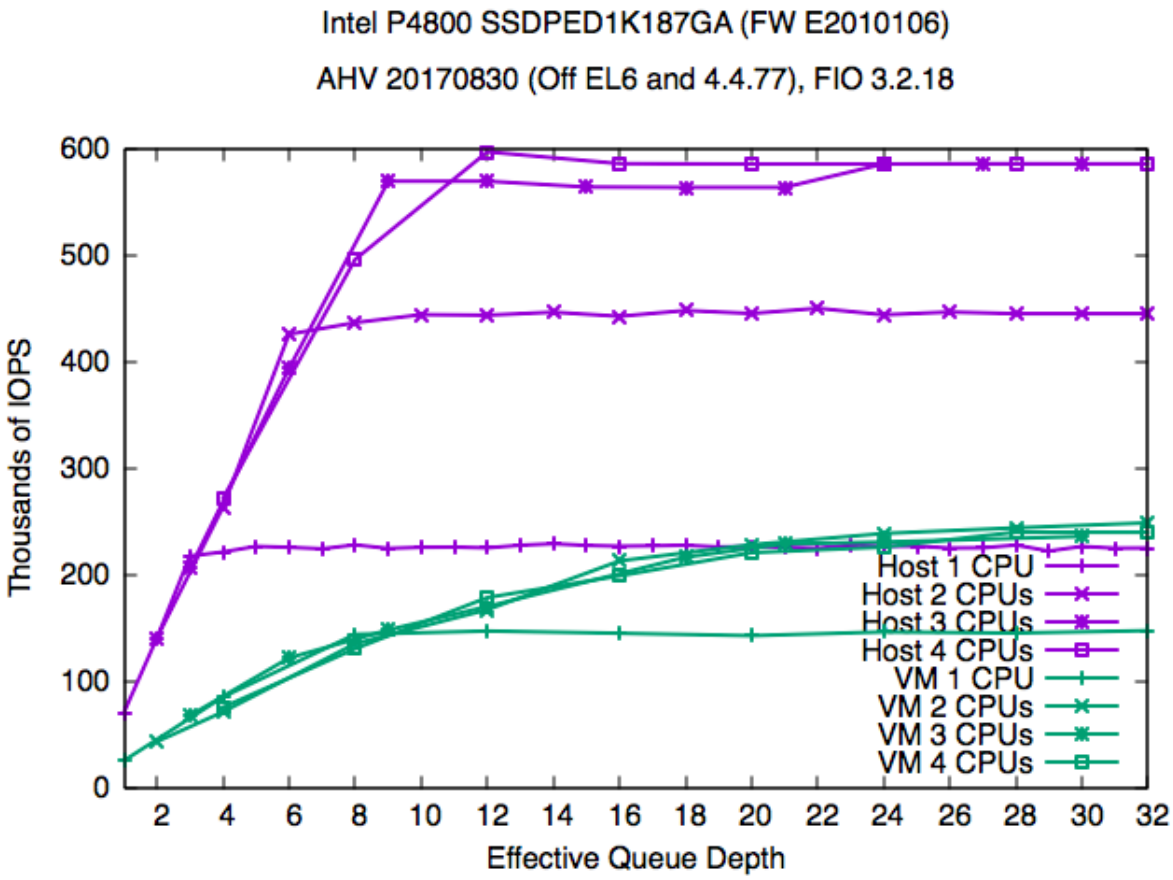
Storage Performance and Virtualisation

What about IOPS?

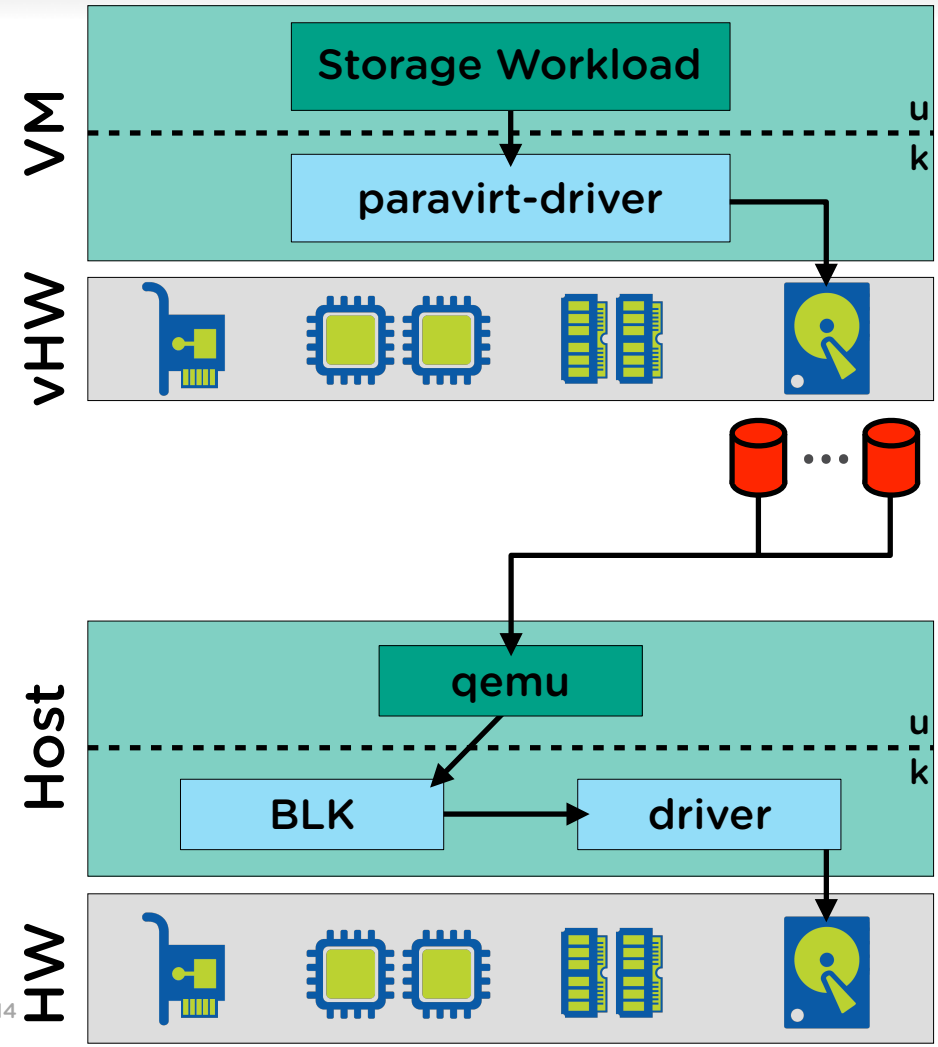
- NVMe w/ 3DXP
- Random reads
- Varying queue depth
- 4 KiB request size

And from a VM ?

- Debian 9.4 VM (FIO 3.2.18)
- Host with Qemu 2.6
- Disks over virtio-scsi



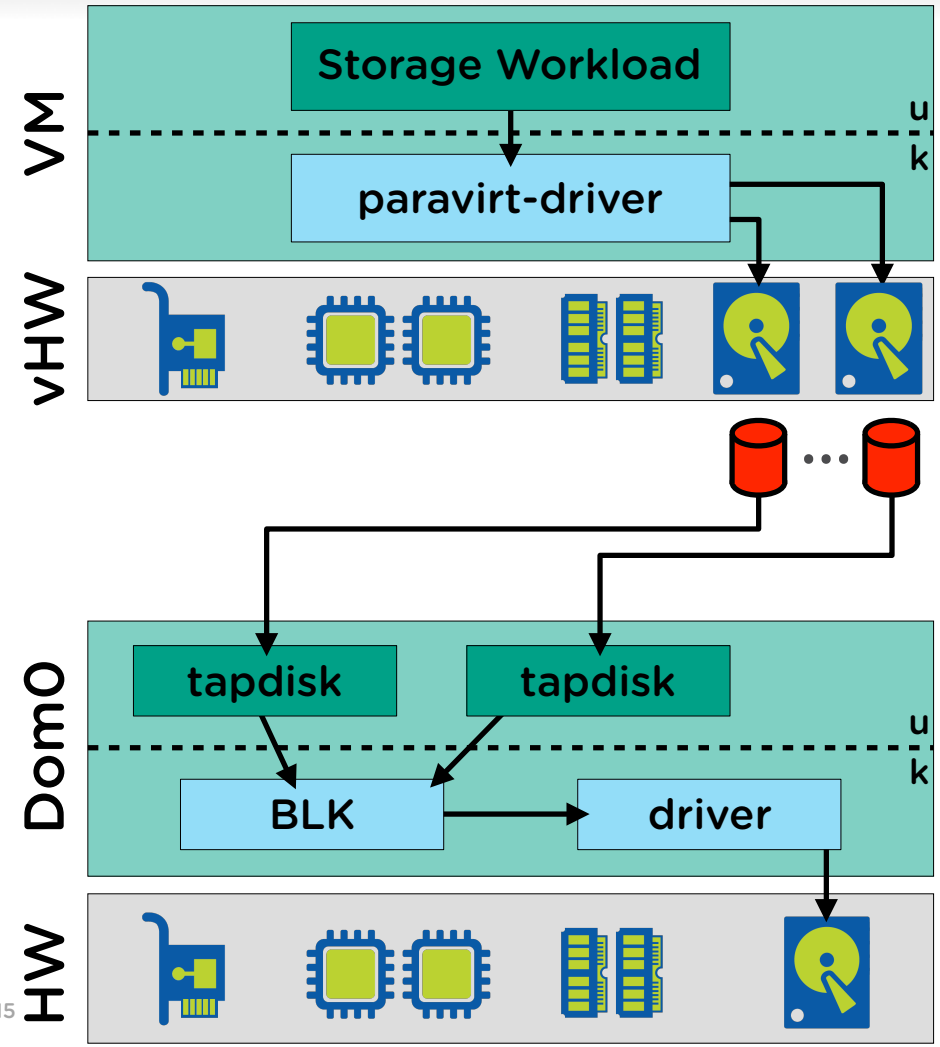
Storage Performance and Virtualisation



Typical virtio-scsi deployment

- One controller presented to VM
- Disks are luns under targets
- One qemu thread handles ctrl
- Qemu bottlenecks on CPU
- Adding more disks won't help
- Adding more ctrls won't help

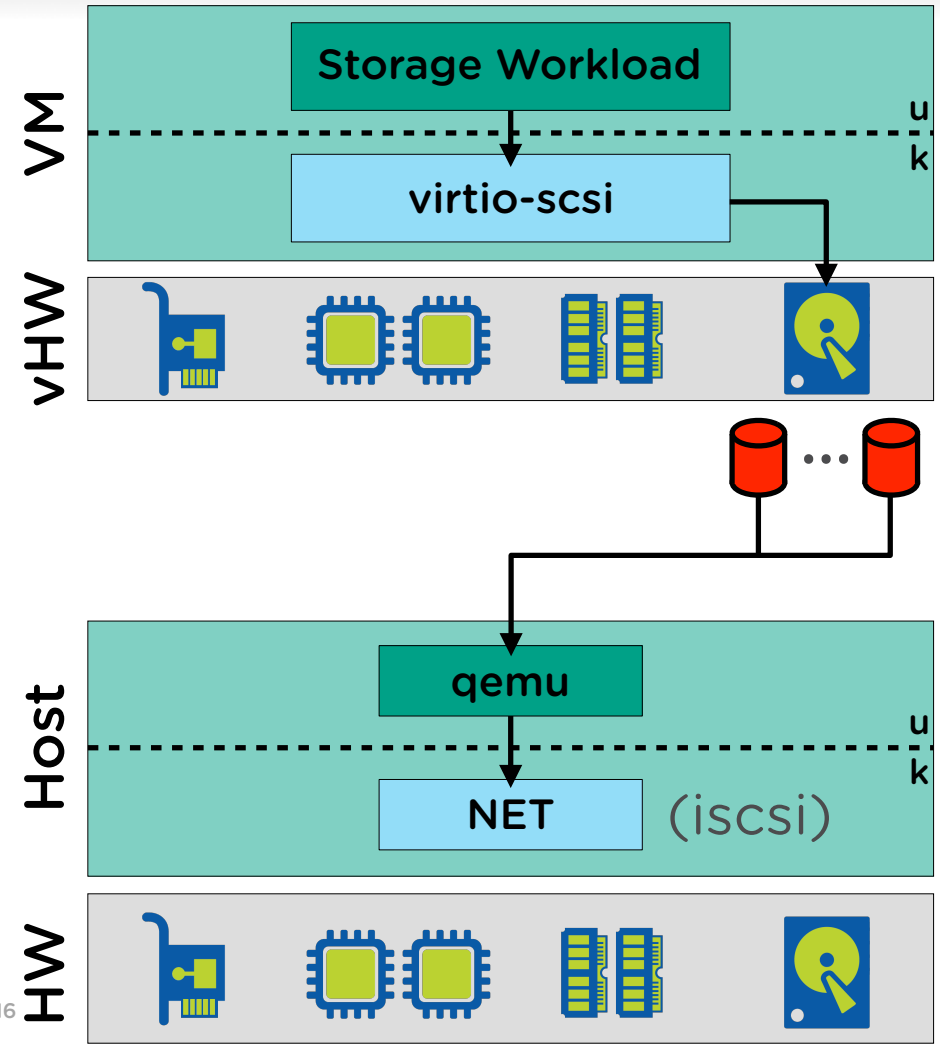
Storage Performance and Virtualisation



Typical XenServer deployment

- Each vdisk is a block device
- Each vdisk backed by a tapdisk
- Tapdisk bottlenecks on CPU
- Bad scalability:
  - Require more vdisks
  - Too much CPU consumption
  - Doesn't scale with VM size
  - Incompatible with workloads

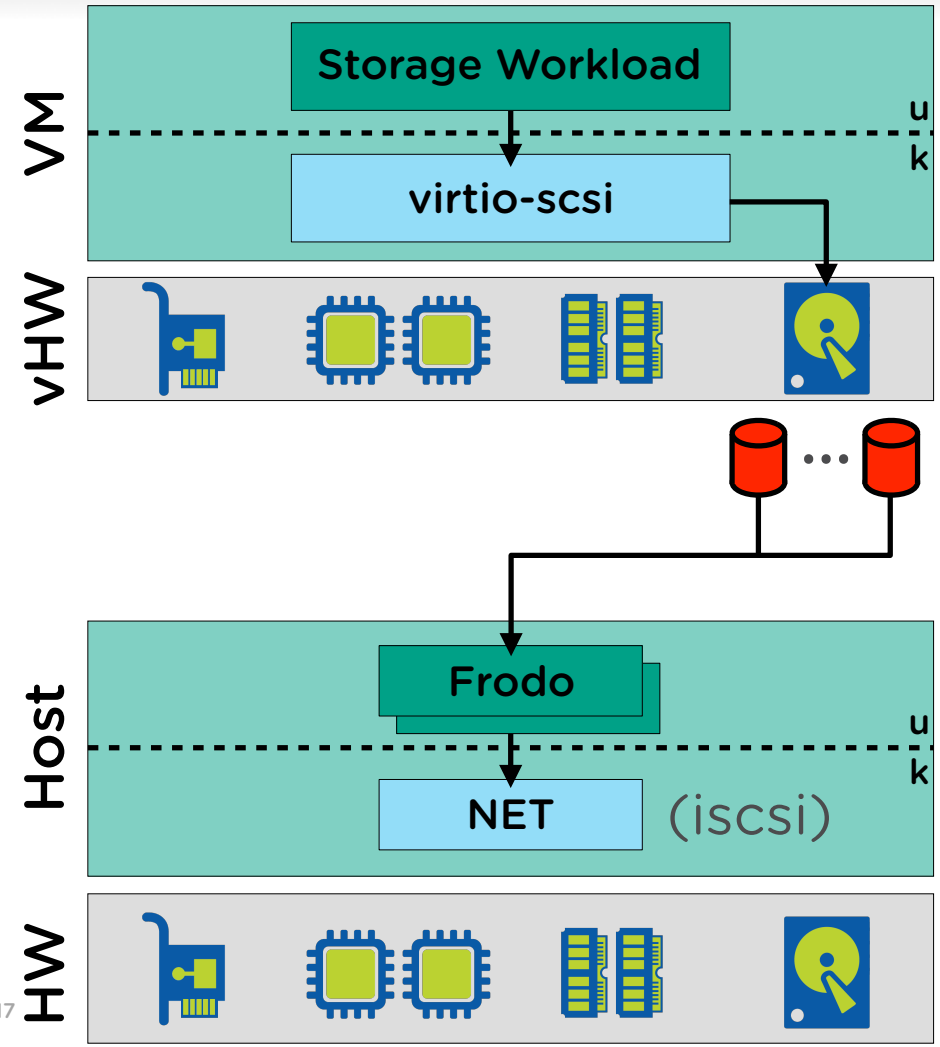
Nutanix AHV



Nutanix AHV up to 5.1

- Qemu handles storage datapath
- With fast devices, Qemu bottlenecks on CPU
- Qemu dataplane meant to provide more threads
- Some hypervisors recommend more controllers (similar to XS)

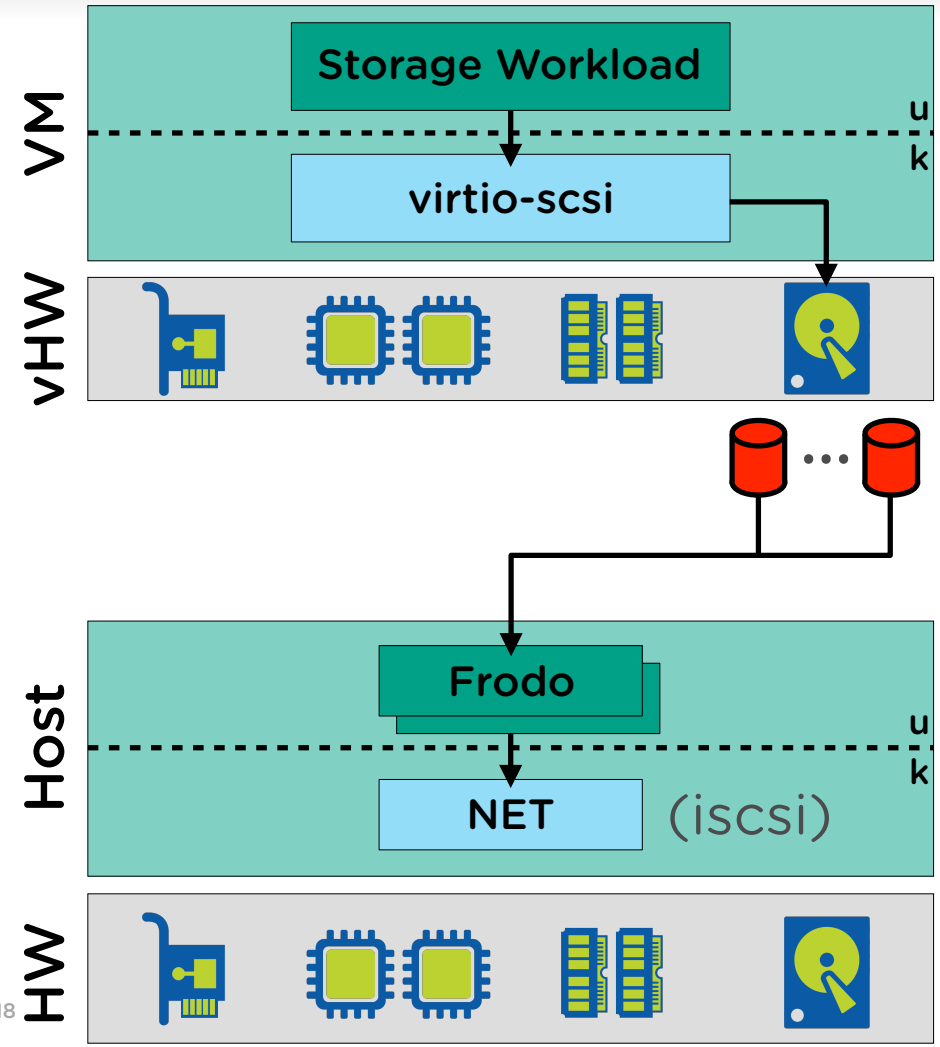
Nutanix AHV



Nutanix AHV 5.5 onwards

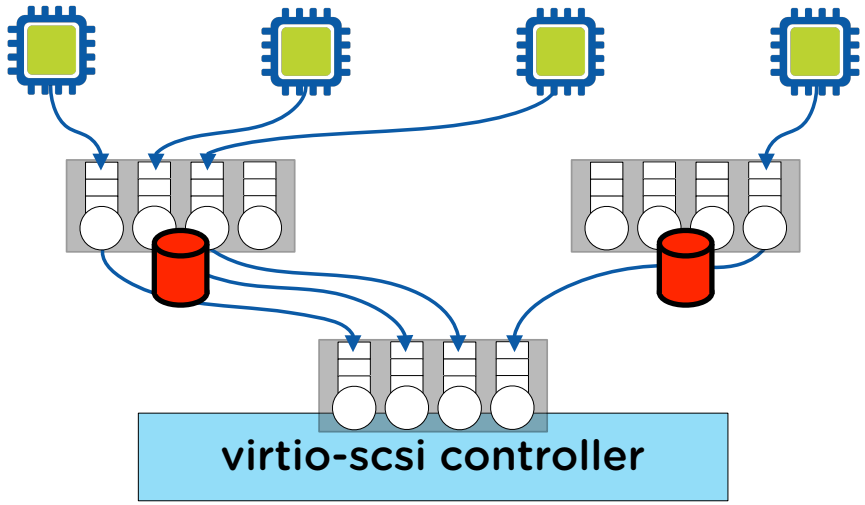
- Frodo handles storage datapath (offloaded by Qemu: vhost-user)
- Frodo presents a MQ controller
- Frodo is multi-threaded, using different threads for different VQs
- Frodo's code is very lean, each thread performs better than Qemu (160k+ IOPS/thread vs 80k IOPS @4k Random Reads on NTNX)

Nutanix AHV



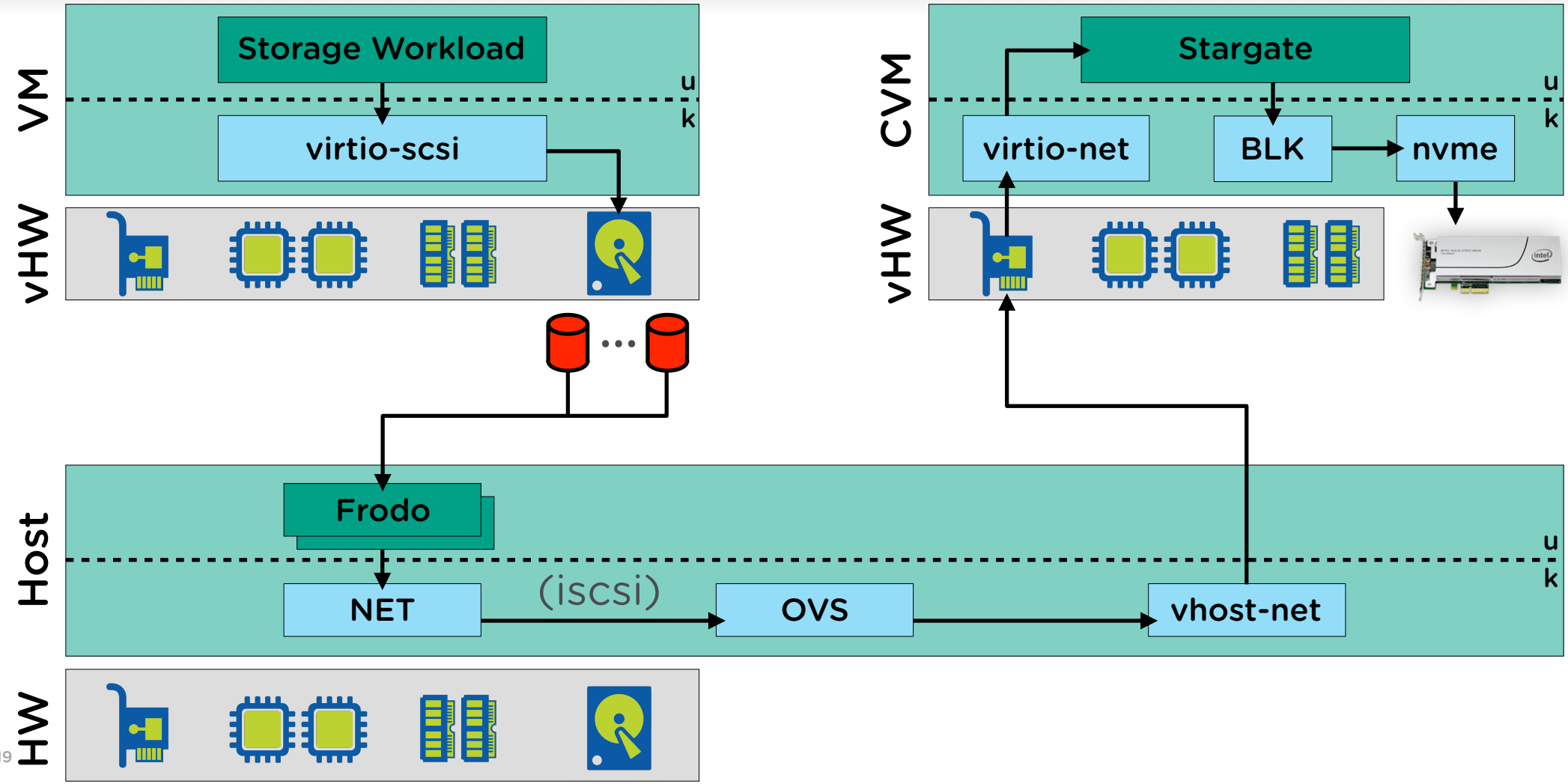
Nutanix AHV 5.5 onwards

- VM gets 1 (vHW) VQ per vCPU
- OS creates 1 (SW) VQ/vCPU/vDisk

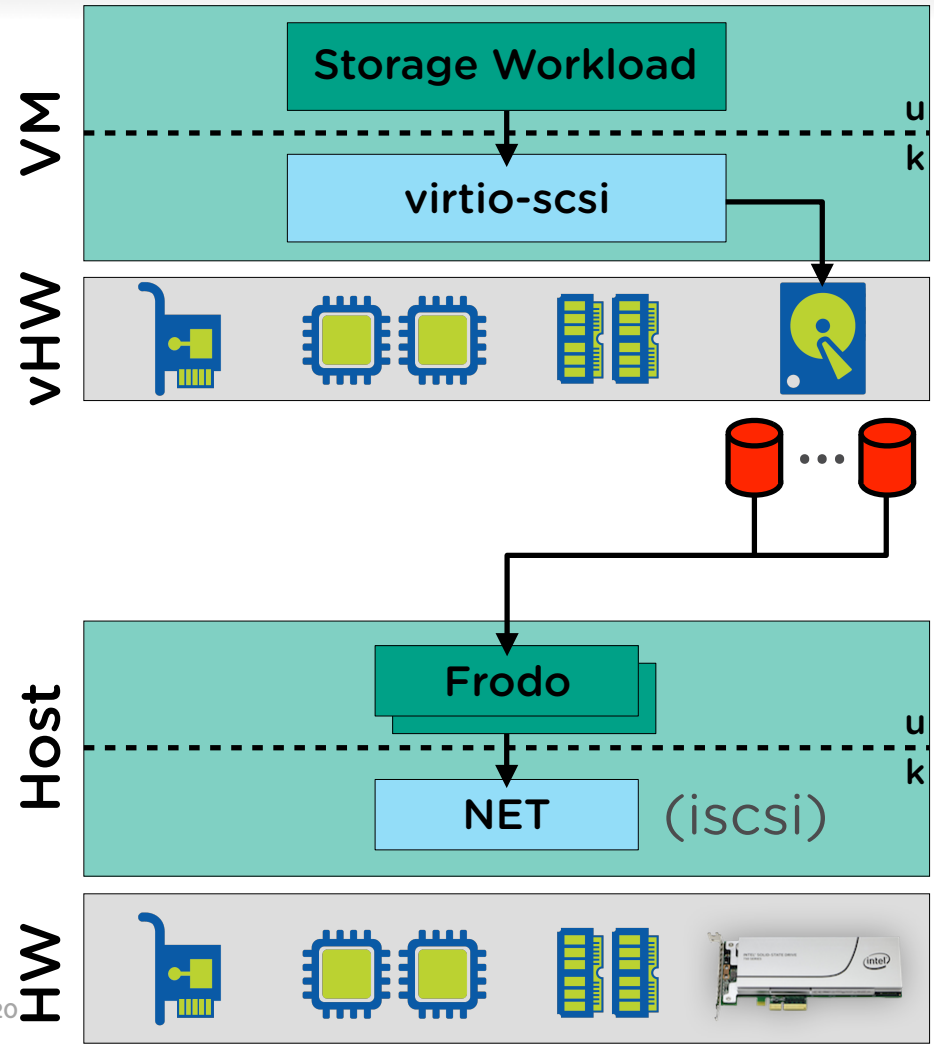


- Lock-free datapath
- Higher number of inflight requests

Nutanix AHV and NVMe



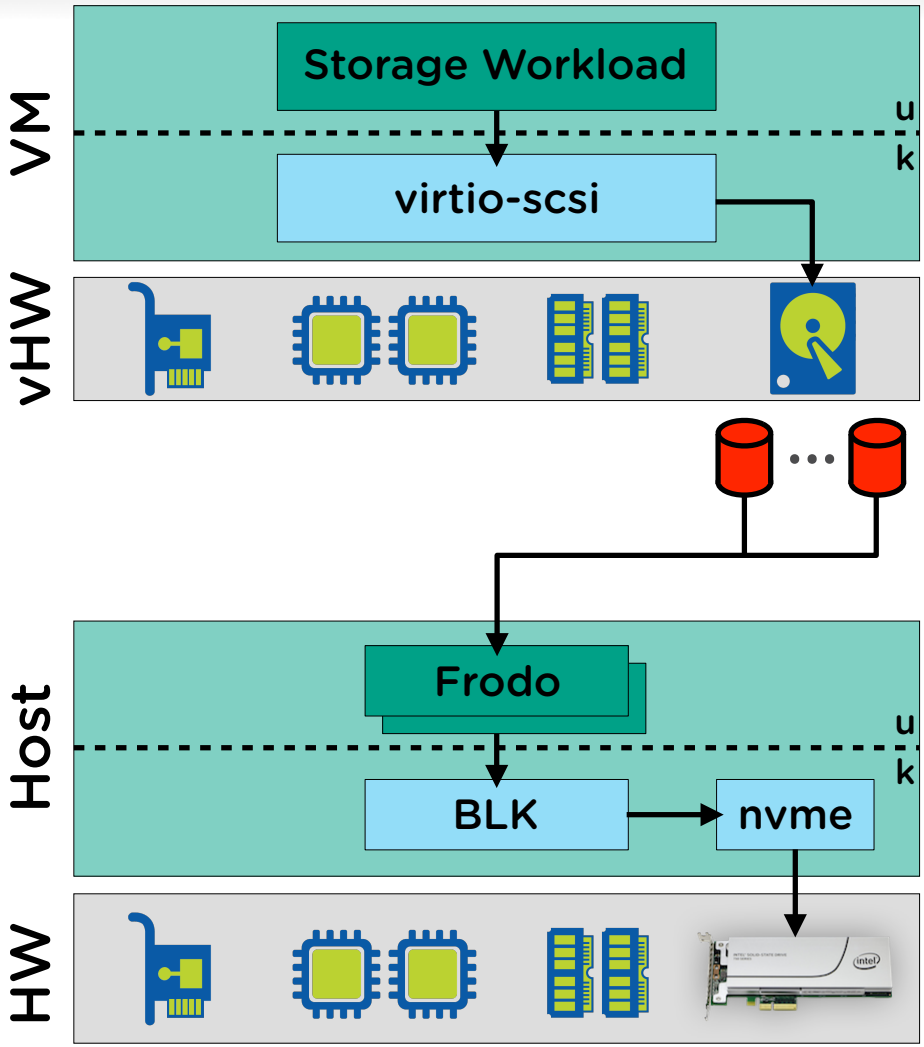
Nutanix AHV and NVMe



Nutanix AHV under devel

- Current datapath too long to fully benefit from NVMe lower latency
- Bring NVMe closer to VM
- Minimise virtualisation overhead

Nutanix AHV and NVMe

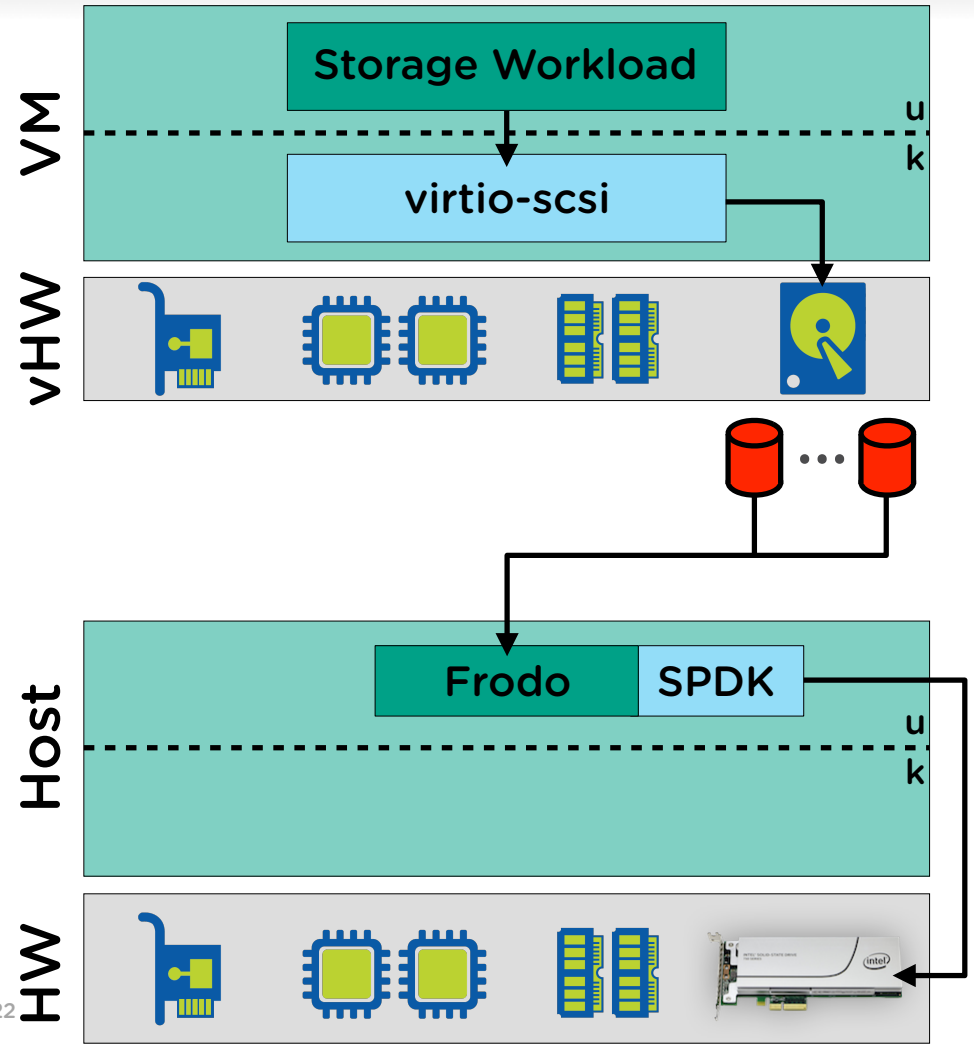


Nutanix AHV under devel

- Current datapath too long to fully benefit from NVMe lower latency
- Bring NVMe closer to VM
- Minimise virtualisation overhead

- One way of doing that is to use libaio and submit requests through the kernel... not.

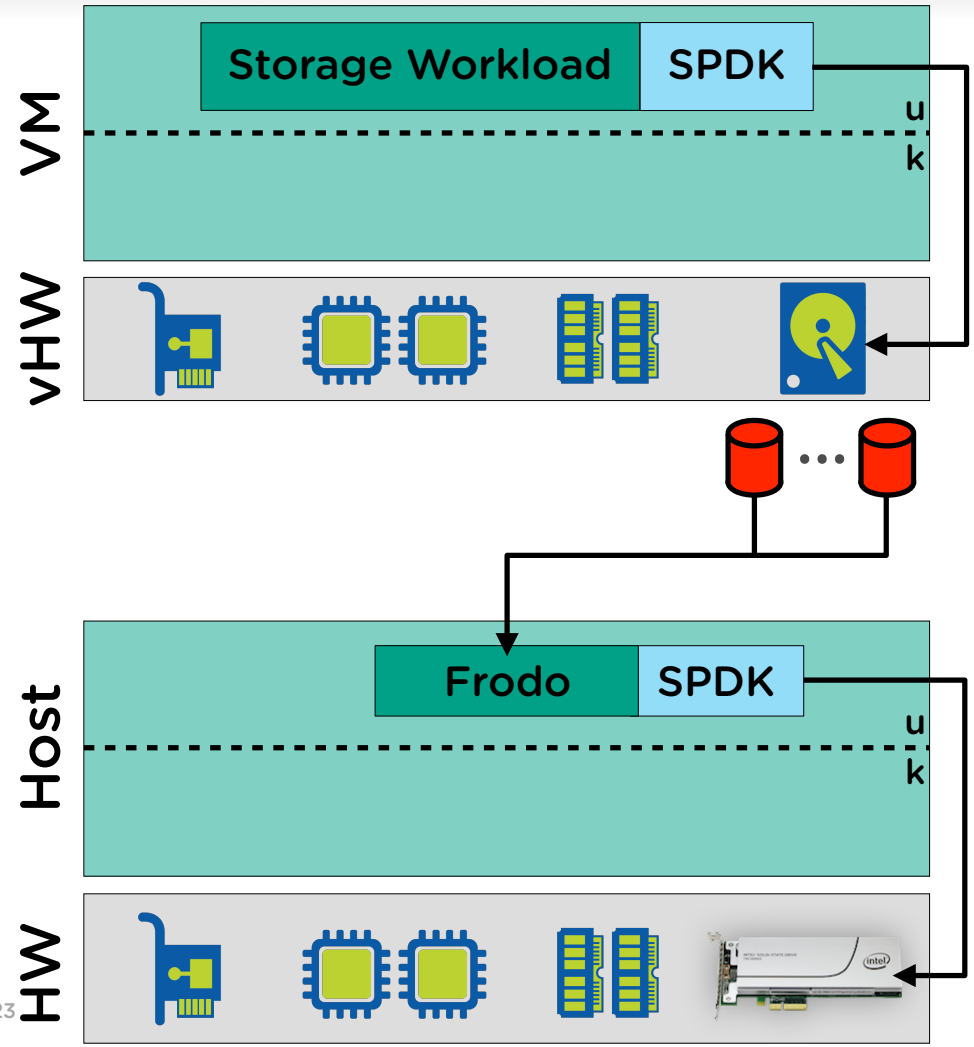
Nutanix AHV and SPDK



Nutanix AHV under devel

- Frodo linked with SPDK for direct access to local NVMe controllers
- Initial consideration for RF1
- Workloads which require high performance, but low resilience

Nutanix AHV and SPDK

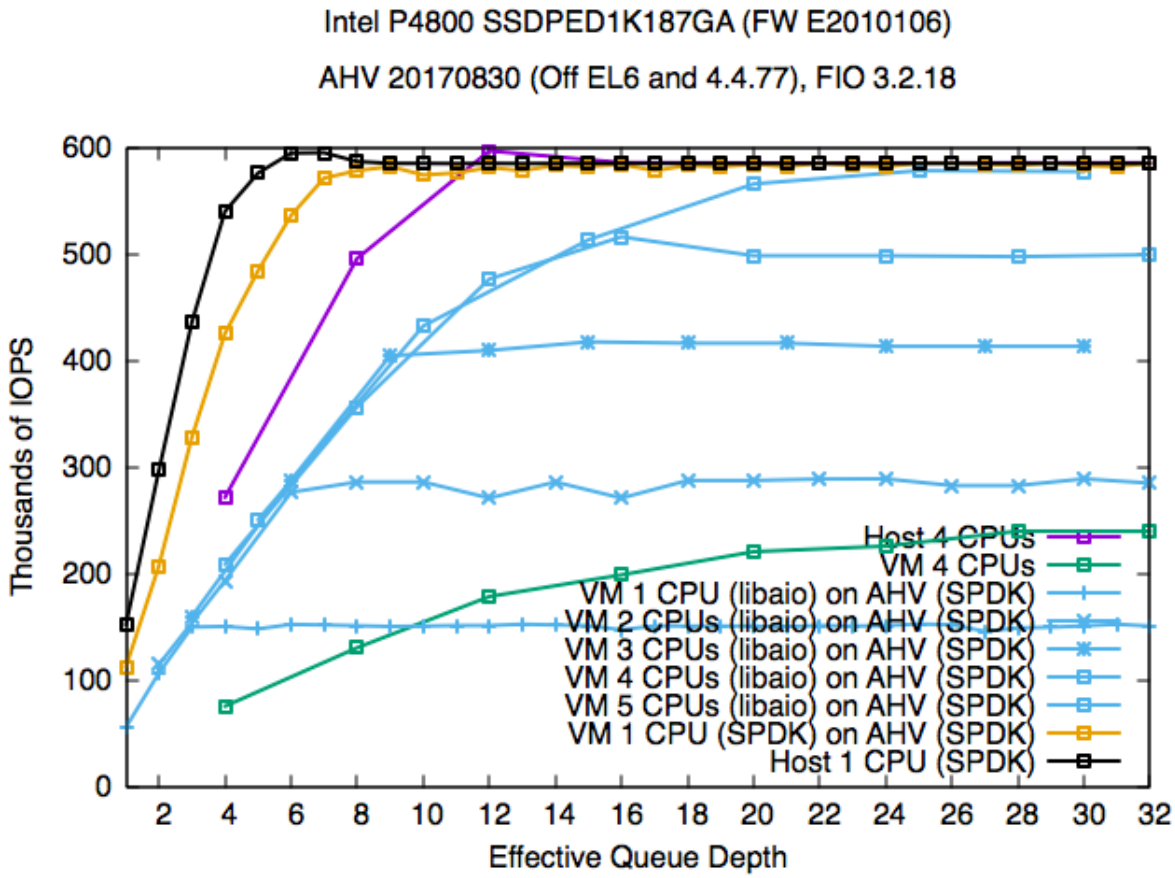


Nutanix AHV under devel

- VMs can also use SPDK!
- On AHV with virtio-scsi PMD
- Spins when reqs are outstanding
- Hypervisor doesn't have to IRQ!

Nutanix AHV and SPDK

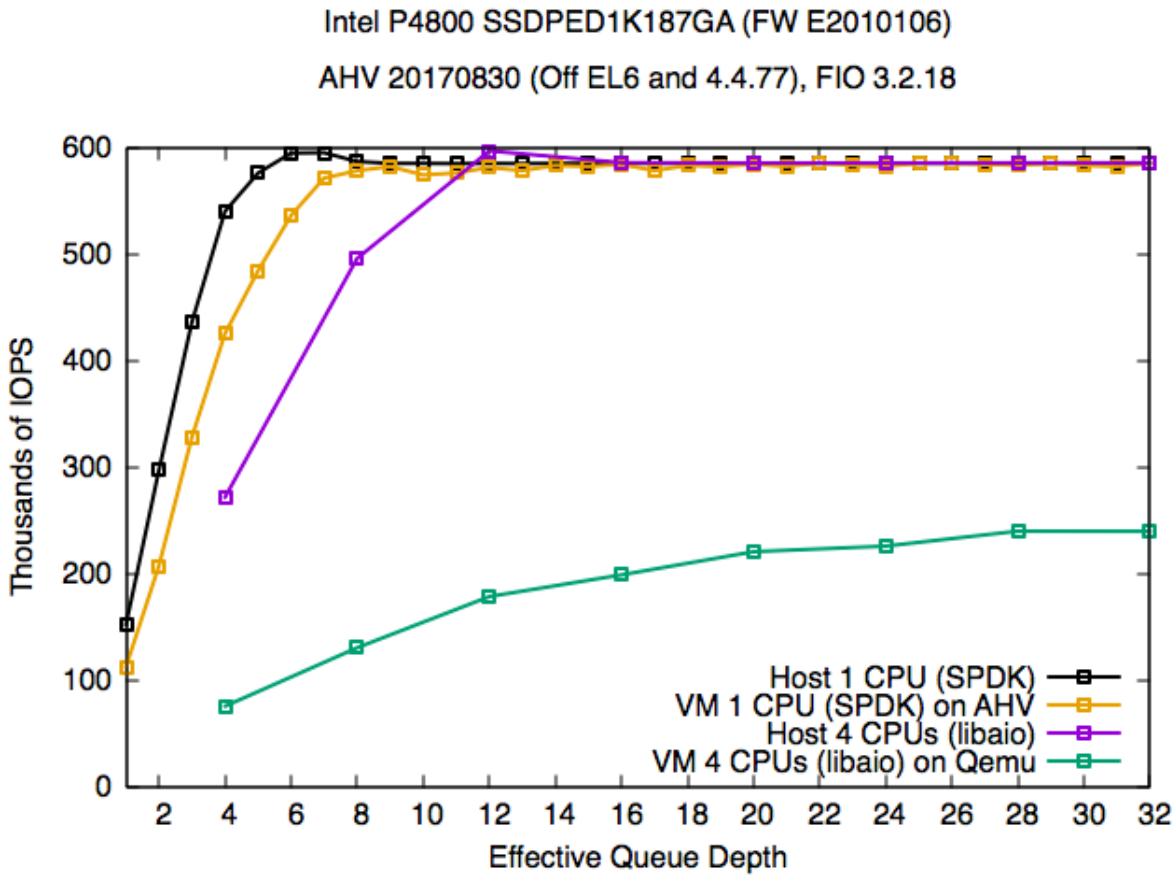
Let's see the numbers!



NUTANIX

Nutanix AHV and SPDK

Let's see the numbers!



NUTANIX

Summary

- Faster storage devices = Harder to virtualise
  - Time spent on CPU more noticeable, results in higher overhead
  - Require careful design for parallel storage access (MQ)
- Userspace-only leaner stack with SPDK
  - Leaner software = lower (CPU) latency
  - Spinning also cuts notification overhead between VM and HOST
- Hypervisors can share NVMe between VMs efficiently
  - Hypervisor uses SPDK for fast and efficient NVMe access
  - VMs can access the same NVMe, using SPDK or not

Thank you! Questions?

felipe@nutanix.com