



A comparison of performance between KVM and Docker instances in OpenStack

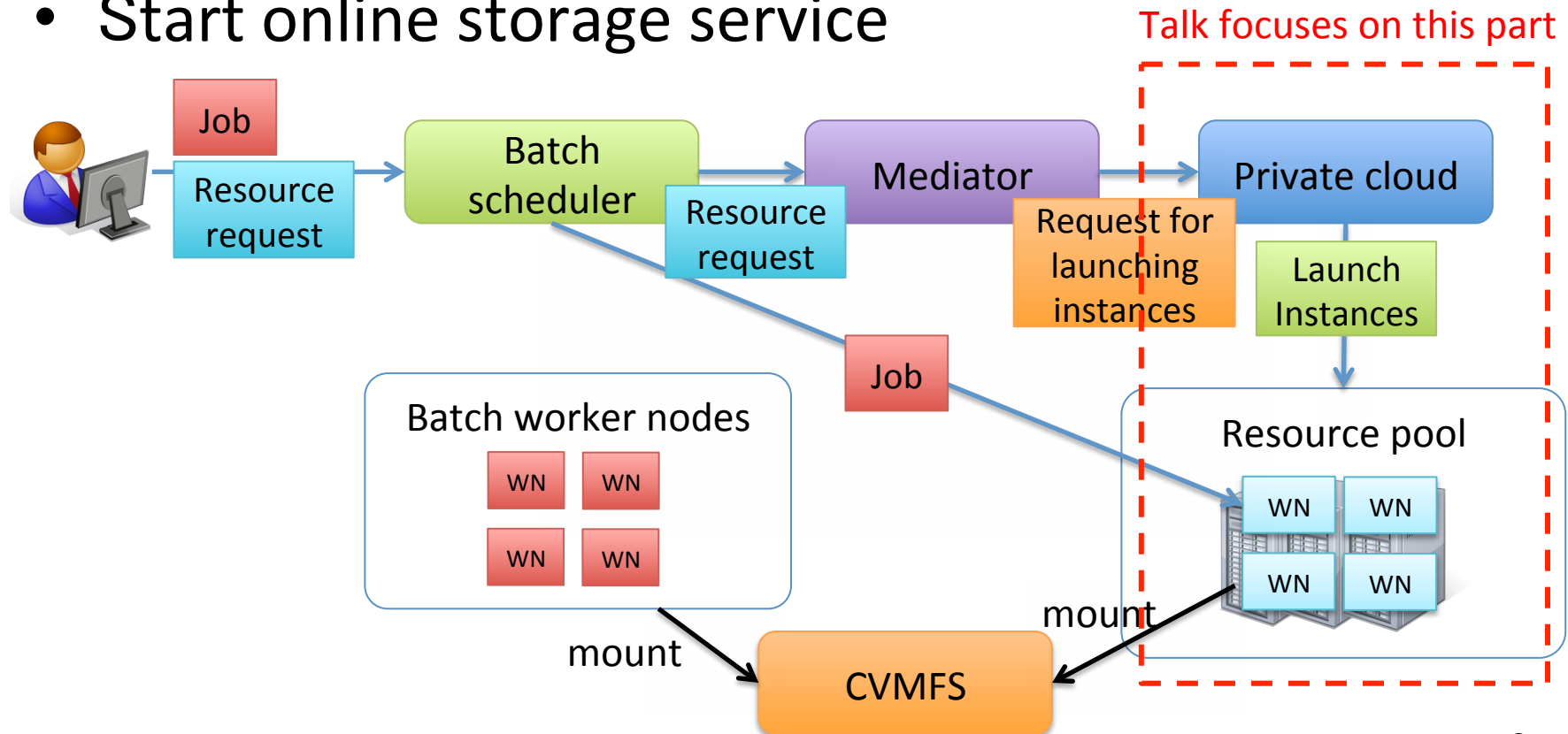
Wataru Takase

High Energy Accelerator Research Organization (KEK), Japan

HEPiX Fall 2015 Workshop at BNL

KEK site will become Cloudy

- Integrate private cloud into batch service
- Deploy CVMFS Stratum 0 and 1
- Start online storage service



Performance Investigation of Cloud

- Is virtual machine performance good?
- What about container technology?
- What about concurrency impact on performance?



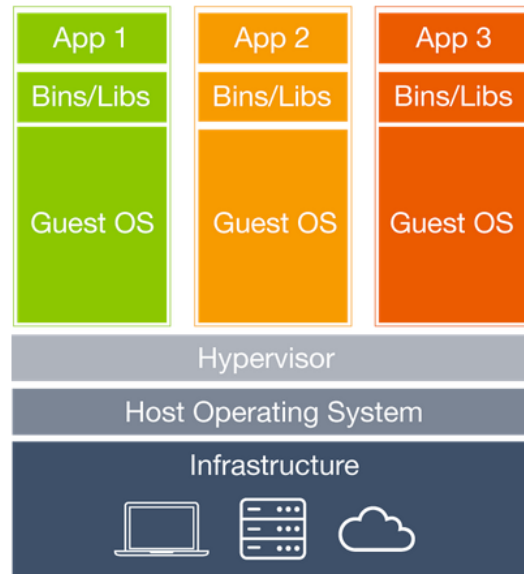
Measured KVM and Docker performance in OpenStack by using Rally

- Cloud performance
- Instance performance

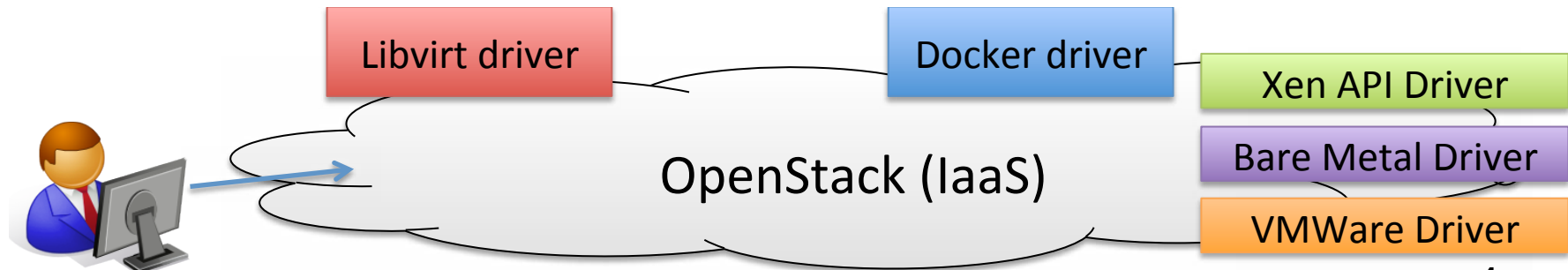
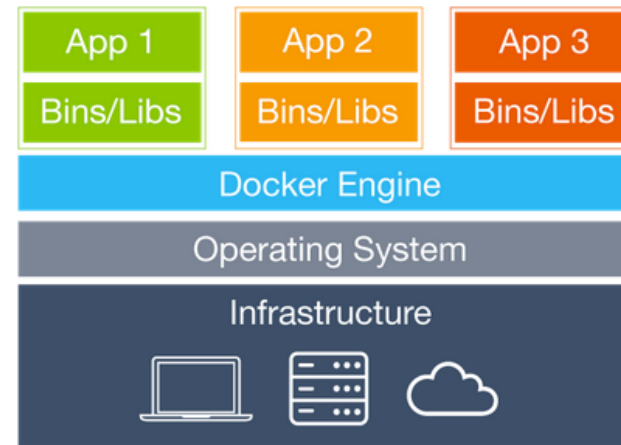
KVM? Docker?

- KVM : VM hypervisor
- Docker : Container manger

Each VM runs on virtual hardware



Containers share host kernel and hardware



What is Rally?

- Benchmarking tool for OpenStack
- Generates real workload
- Provides more than 100 test scenarios:
 - Boot server and migrate it
 - Create image and boot server
 - Create volume and attach to server
 - ...

Example_of_rally_benchmark_input.yaml

```
---
NovaServers.boot_and_delete_server:
  -
    args:
      image:
        name: "centos-cloud:7.1"
      flavor:
        name: "m1.xsmall"
      min_sleep: 60
      max_sleep: 60
    runner:
      type: "constant"
      times: 32
      concurrency: 32
    context:
      users:
        tenants: 1
        users_per_tenant: 32
```

Test environment

- OpenStack Kilo (RDO)
- 1 controller + 1 compute node
- nova-network
- Rally (2d874a7)
- Sysbench 0.4.12

Physical server	OS	Kernel	CPU	CPU cores	RAM (GB)	Disk (GB)
OpenStack controller	CentOS 7.1.1503	3.10.0-229	Intel(R) Xeon(R) CPU E5649 x2	24*	32	300
OpenStack compute	CentOS 7.1.1503	3.10.0-229	Intel(R) Xeon(R) CPU E5-2630 v3 x2	32*	64	3800
Rally	CentOS 7.1.1503	3.10.0-229	AMD Opteron(TM) Processor 6212	8	16	1700

* HT is enabled

Instance image and flavor

OS	Kernel	vCPU	RAM (GB)	Disk (GB)
CentOS 7.1.1503	3.10.0-229	1	1.8	10

Test environment

Compute node

File system	XFS on LVM on hardware RAID 5
IO scheduler	Deadline
Clocksource	TSC
QEMU	1.5.3
libvirt	1.2.8
Docker	1.6.2
Nova Docker driver	nova-docker stable/kilo (d556444)

KVM

Image format	qcow2
Block device driver	VirtIO
Cache mode	none
File system	XFS
Clocksource	TSC

Docker

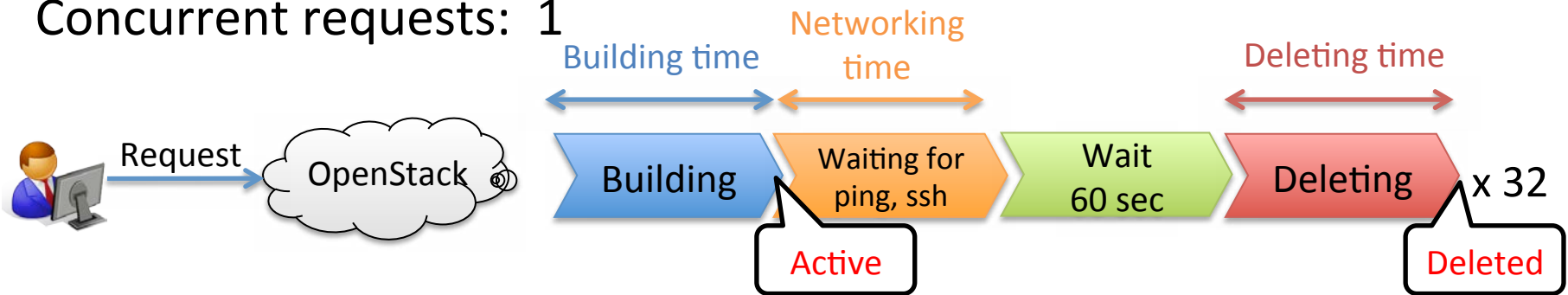
Storage driver	OverlayFS
-----------------------	-----------

Benchmark Scenarios

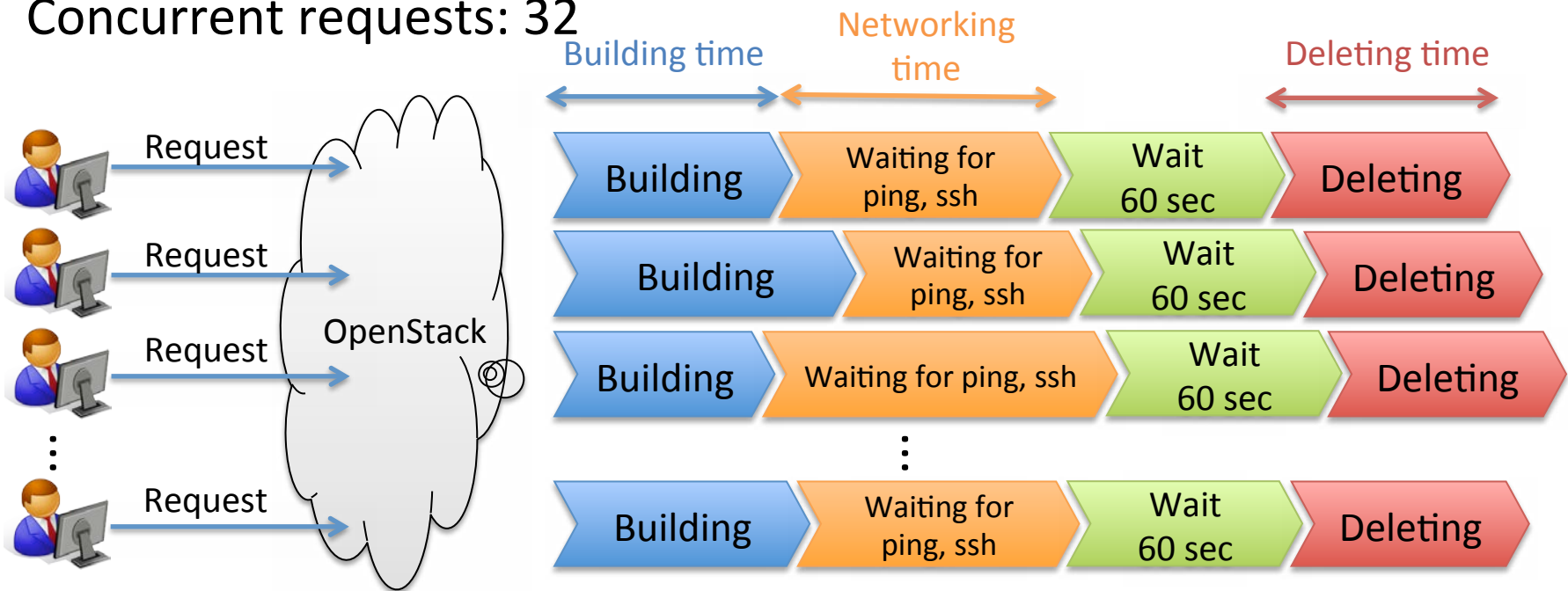
1. Measure cloud performance
 - Boot a server and then delete
2. Measure instance performance
 - Boot a server and run Sysbench (test=cpu)
 - Boot a server and run Sysbench (test=memory, memory-oper=read)
 - Boot a server and run Sysbench (test=memory, memory-oper=write)
 - Boot a server and run Sysbench (test=fileio, file-test-mode=seqrd)
 - Boot a server and run Sysbench (test=fileio, file-test-mode=rndrd)
 - Boot a server and run Sysbench (test=fileio, file-test-mode=seqwr)
 - Boot a server and run Sysbench (test=fileio, file-test-mode=rndwr)
- Each scenario launches 32 instances
 - Change number of concurrent requests from 1 to 32

Boot a server and then delete

- Concurrent requests: 1



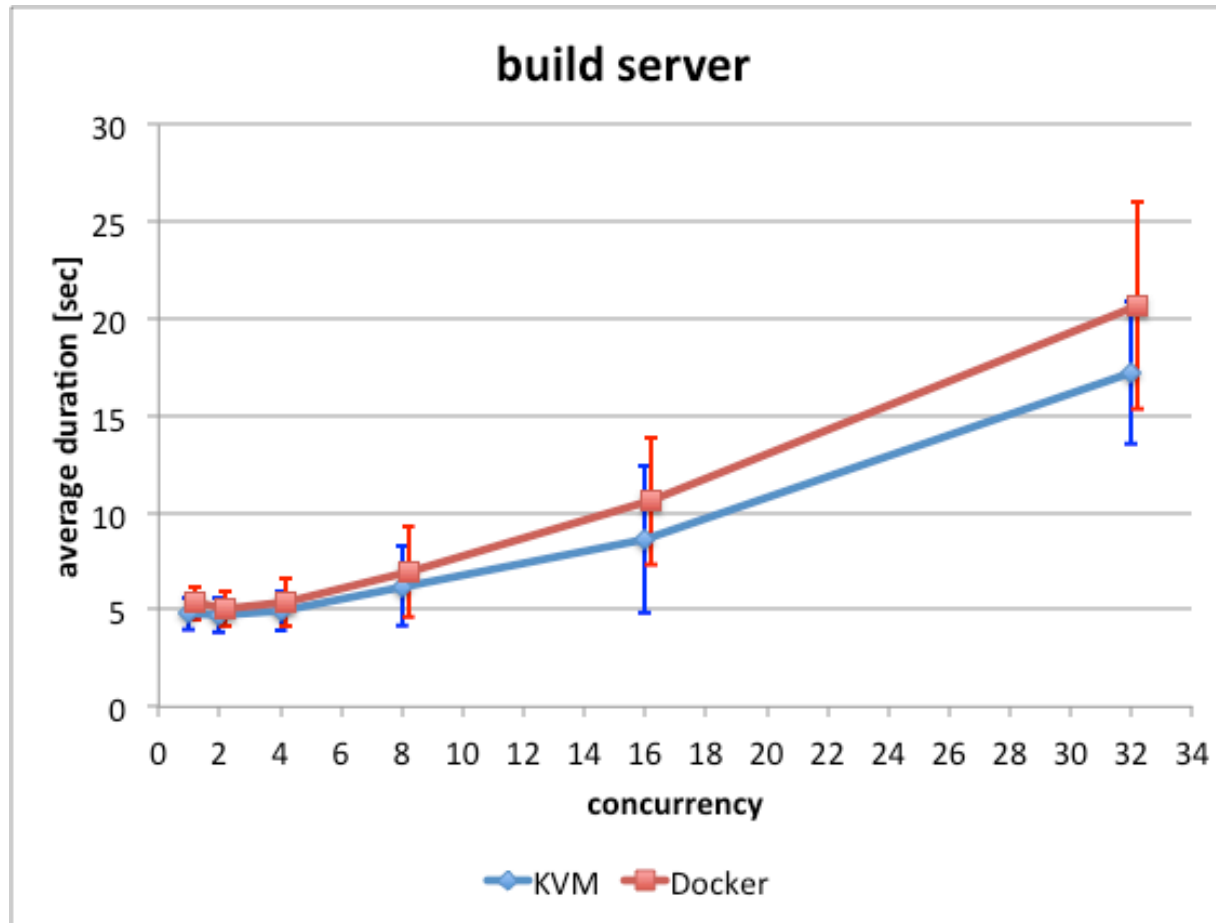
- Concurrent requests: 32



Build a server

N: 96

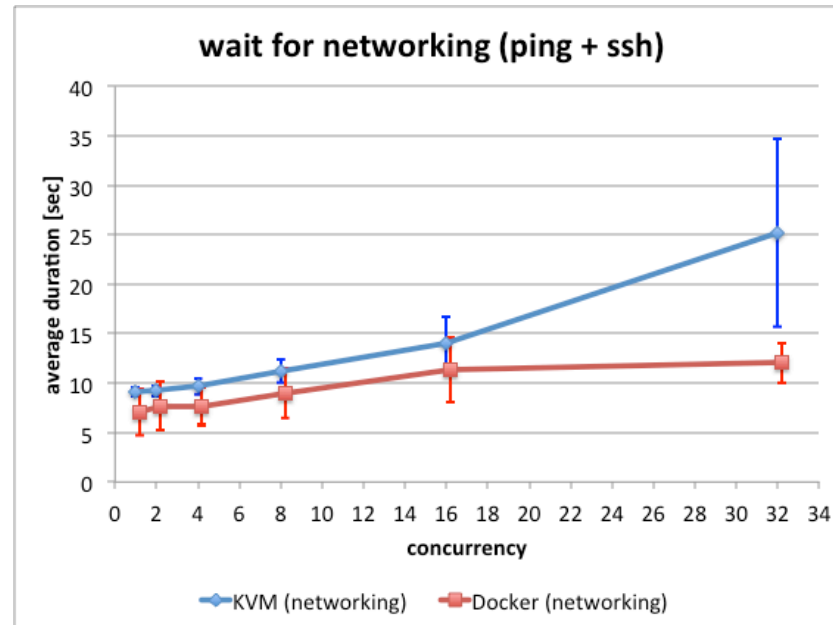
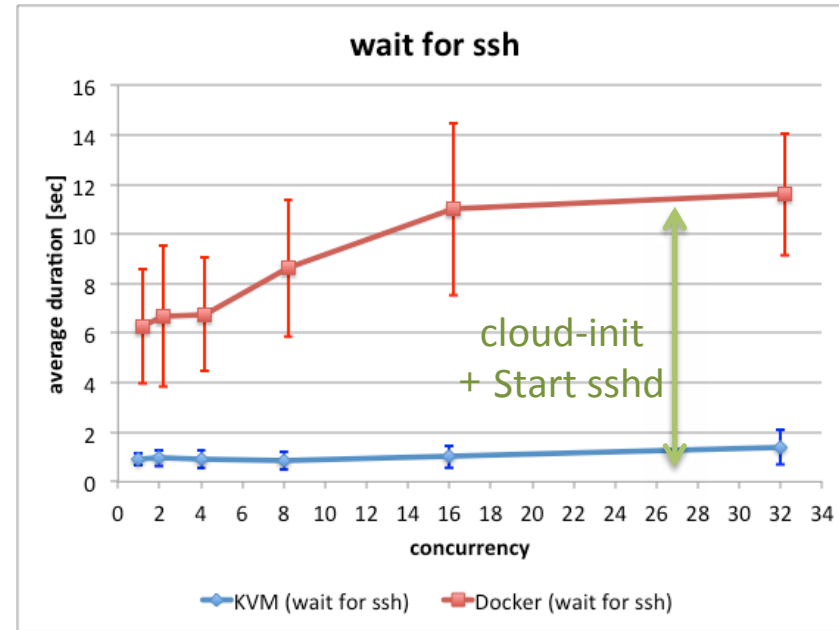
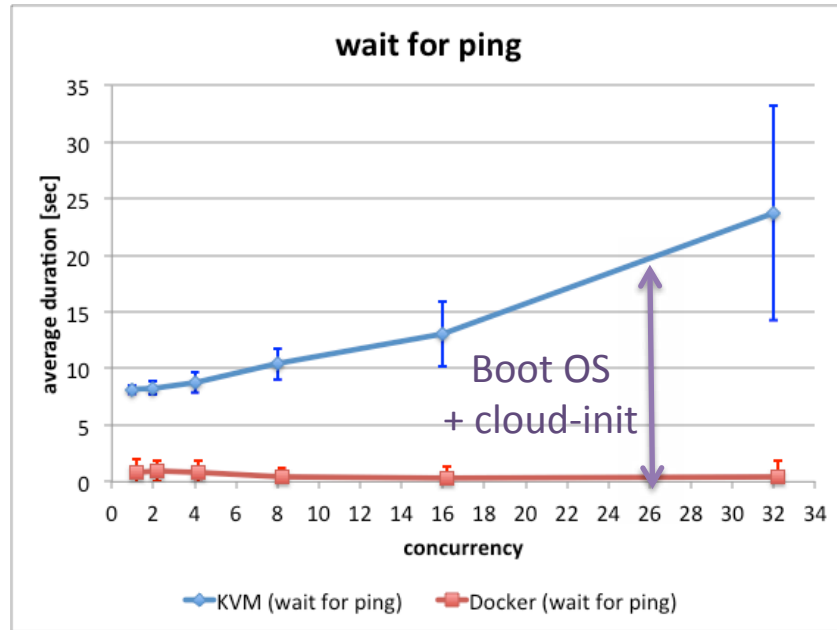
Error bar: SD



- At high concurrency KVM is around 20% better

Wait for Networking

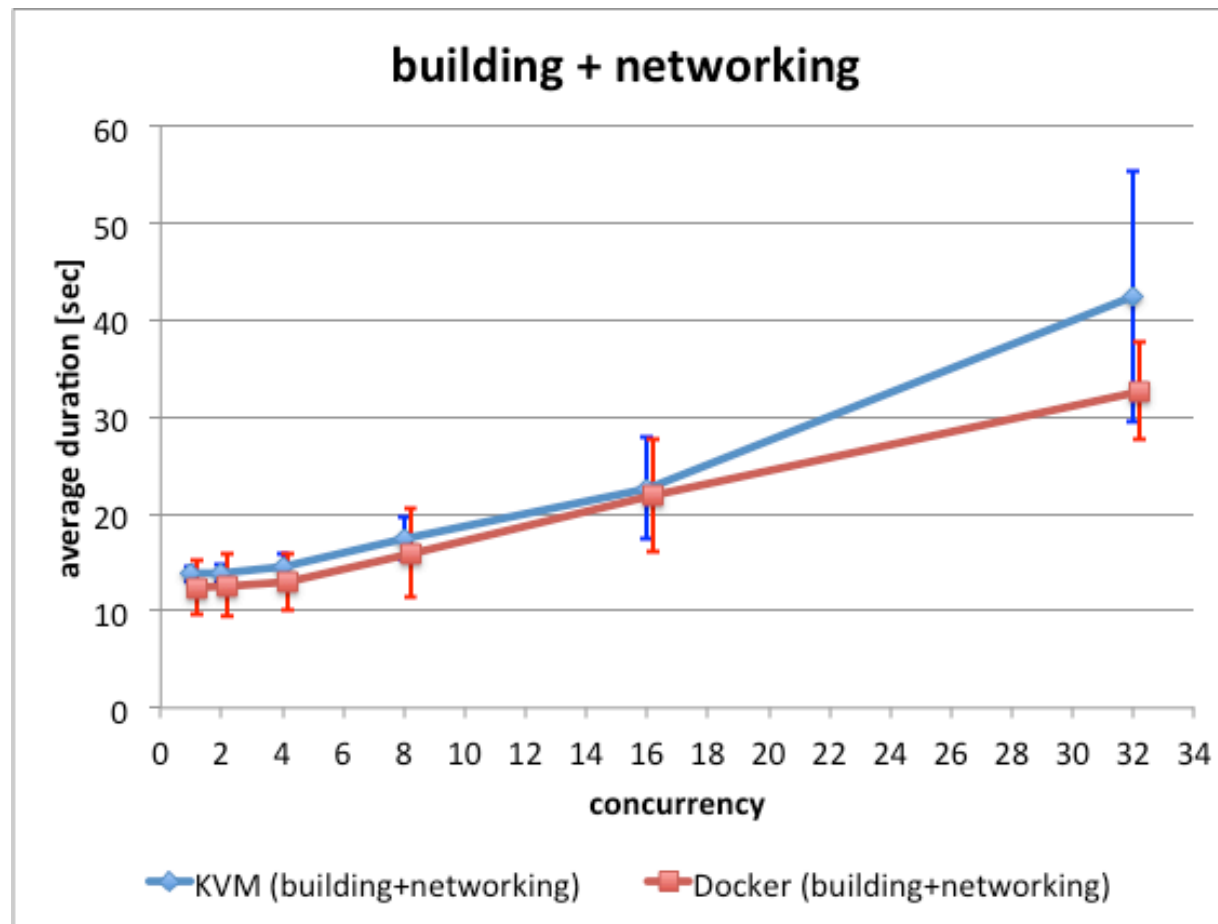
N: 96
Error bar: SD



Building + Networking

N: 96

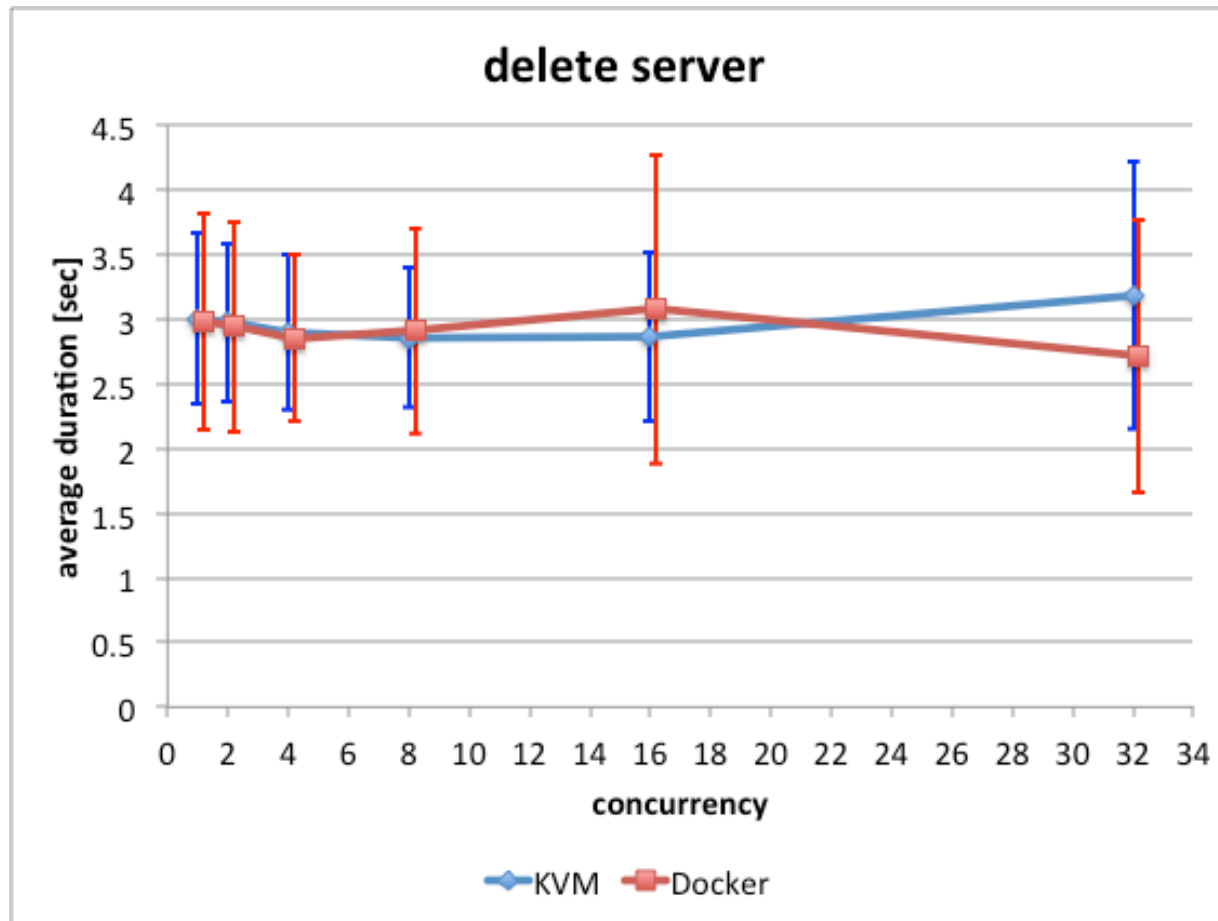
Error bar: SD



Delete a server

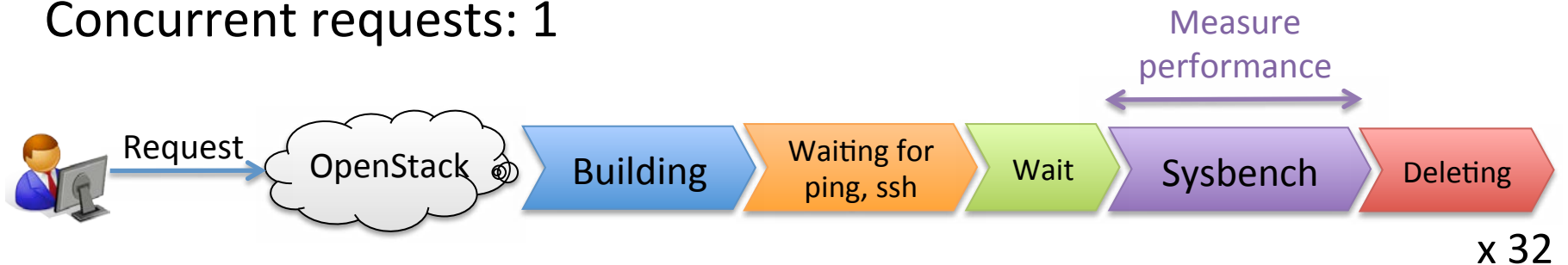
N: 96

Error bar: SD

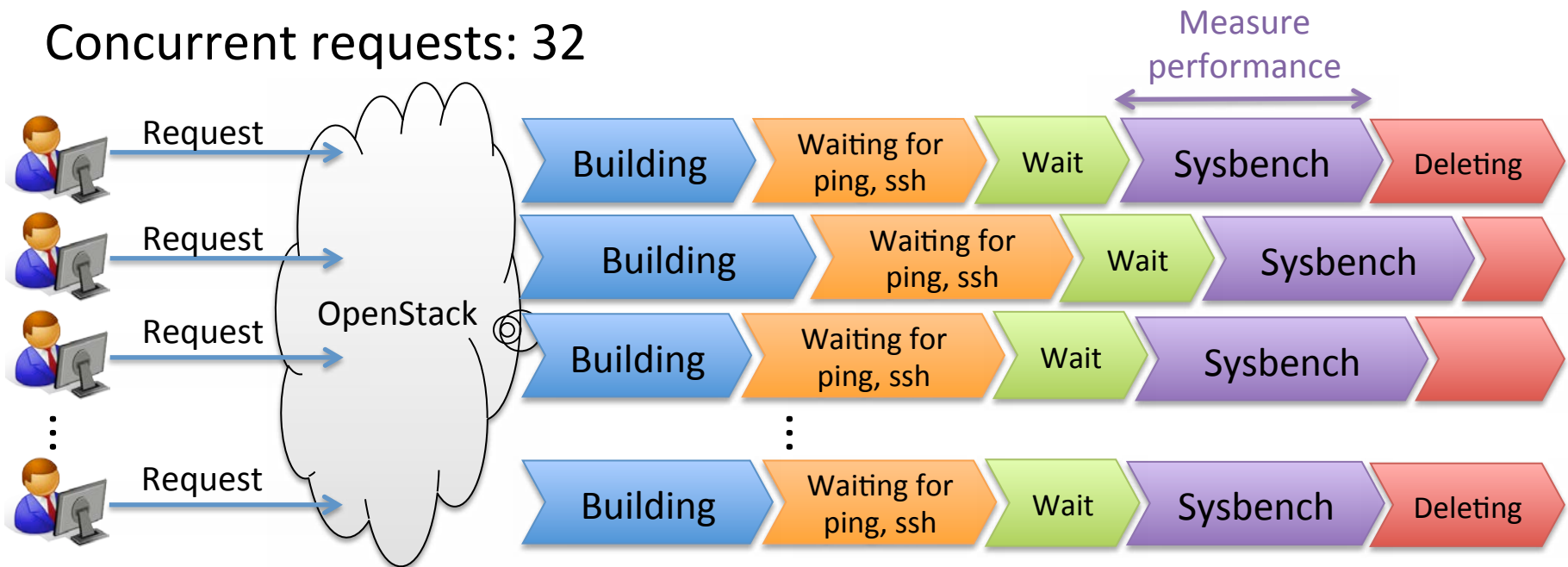


Instance Performance Comparison

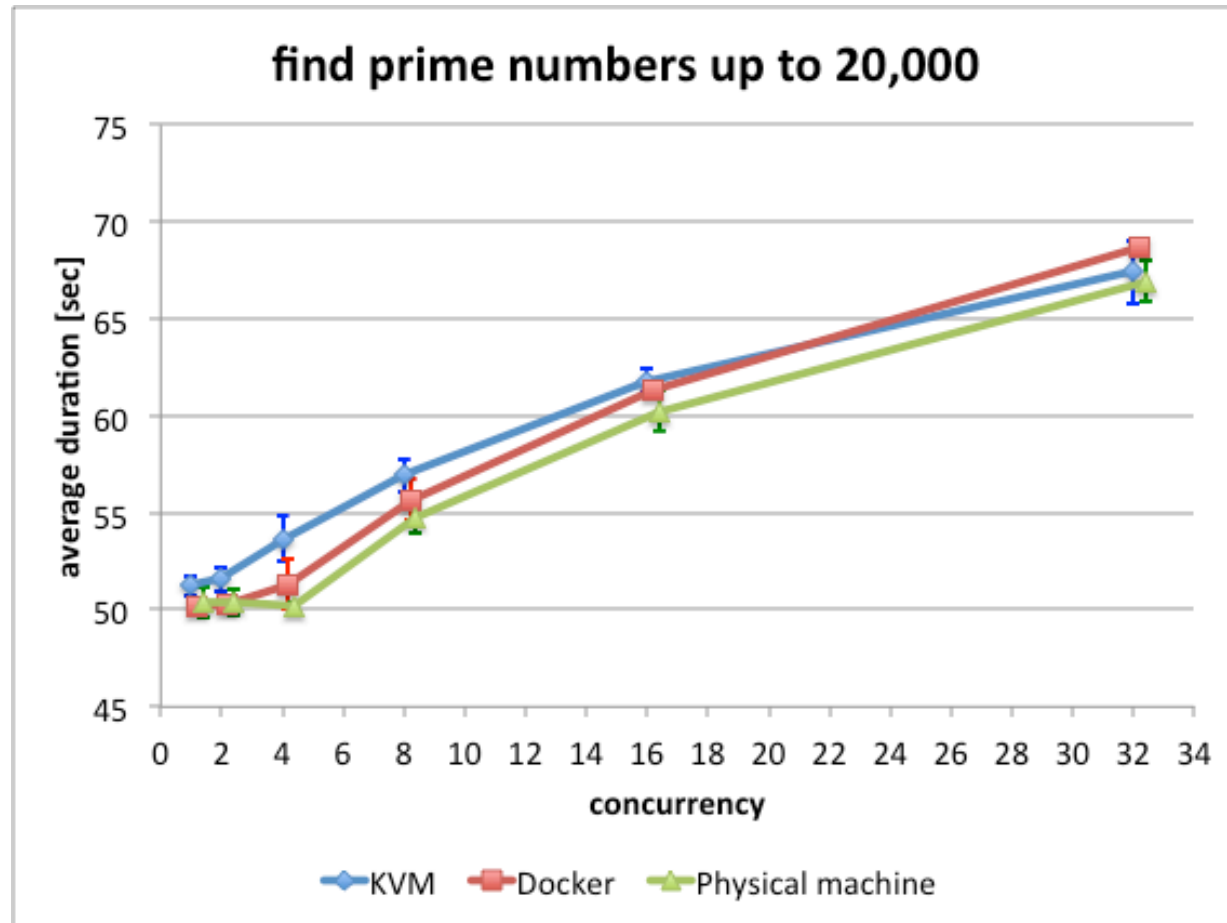
- Concurrent requests: 1



- Concurrent requests: 32



test=cpu, cpu-max-prime=20000, num-threads=1



N: 32

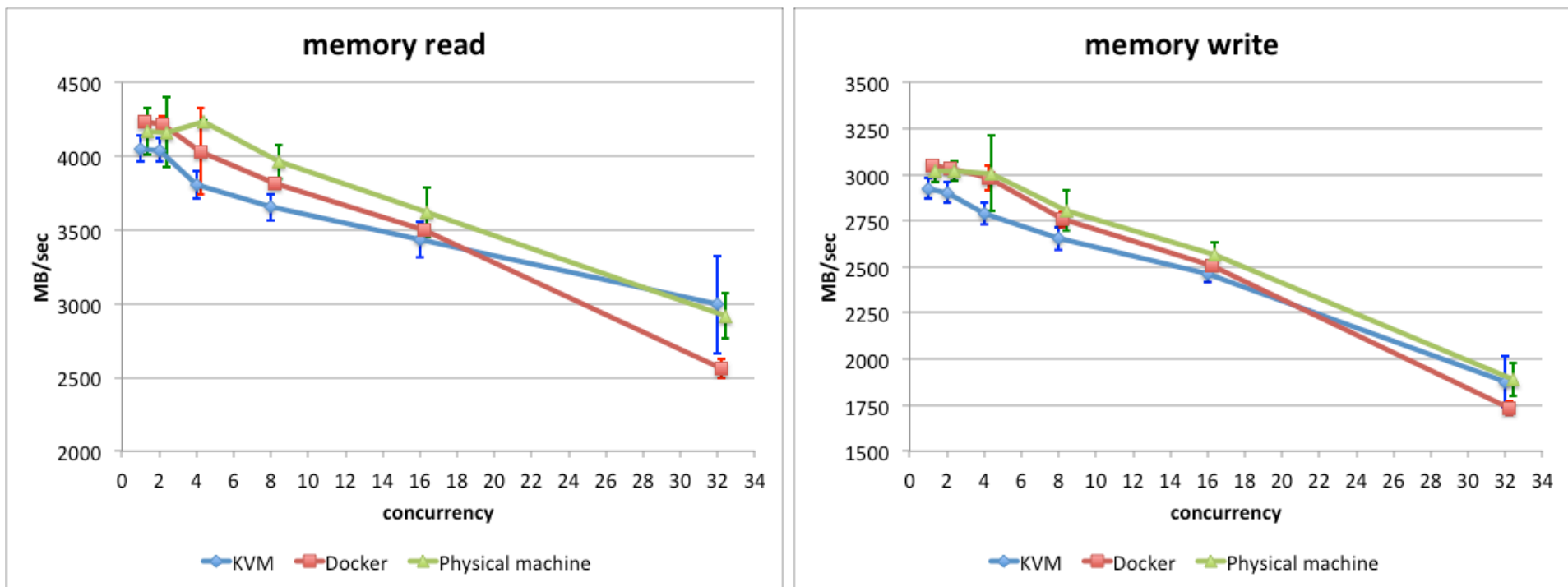
Error bar: SD

- At low concurrency KVM is 2-7% worse than native
- If No. of concurrent requests > 2 , Docker is 2% worse than native

- test=memory, memory-oper=read, memory-block-size=1K, memory-total-size=100G, max-time=300, num-threads=1
- test=memory, memory-oper=write, memory-block-size=1K, memory-total-size=100G, max-time=300, num-threads=1

N: 32

Error bar: SD

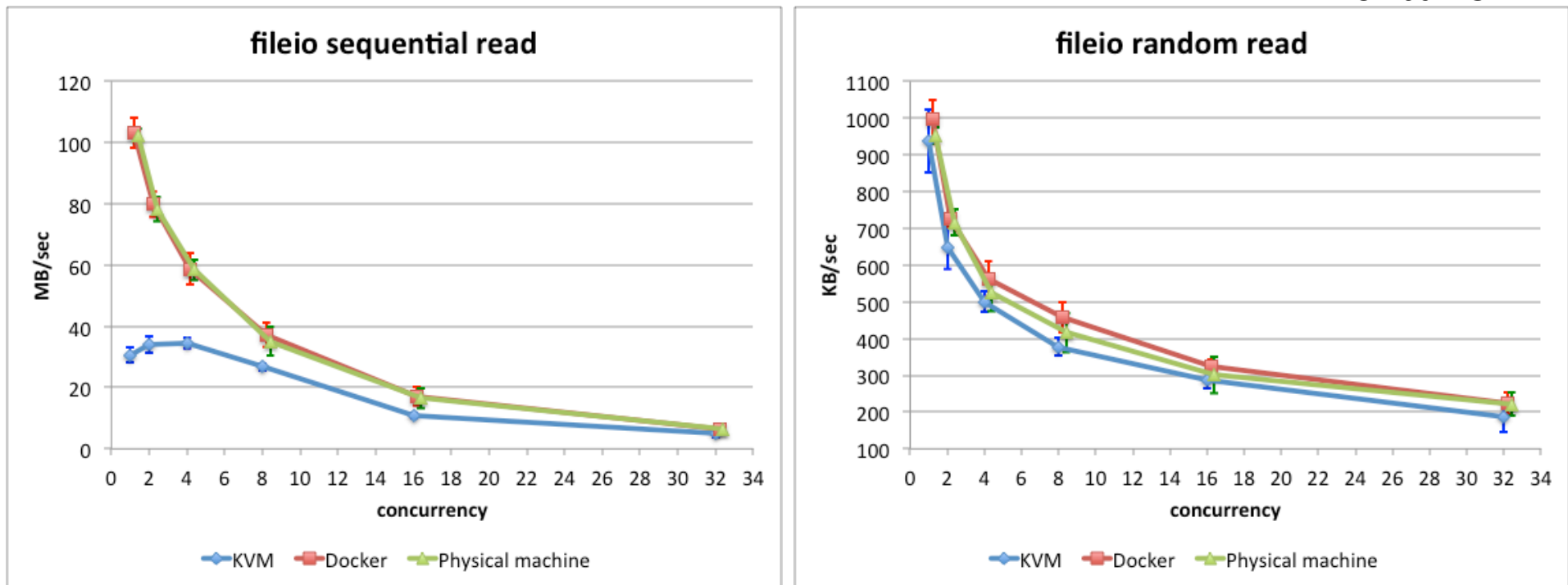


- At low concurrency KVM is 3-10% worse than native
- Docker is 2-5% worse than native (concurrent requests: 1-16)

- test=fileio, file-test-mode=seqrd, file-block-size=4K, file-total-size=8G, file-num=128, file-extra-flags=direct, max-time=300, num-threads=1
- test=fileio, file-test-mode=rndrd, file-block-size=4K, file-total-size=8G, file-num=128, file-extra-flags=direct, max-time=300, num-threads=1

N: 32

Error bar: SD

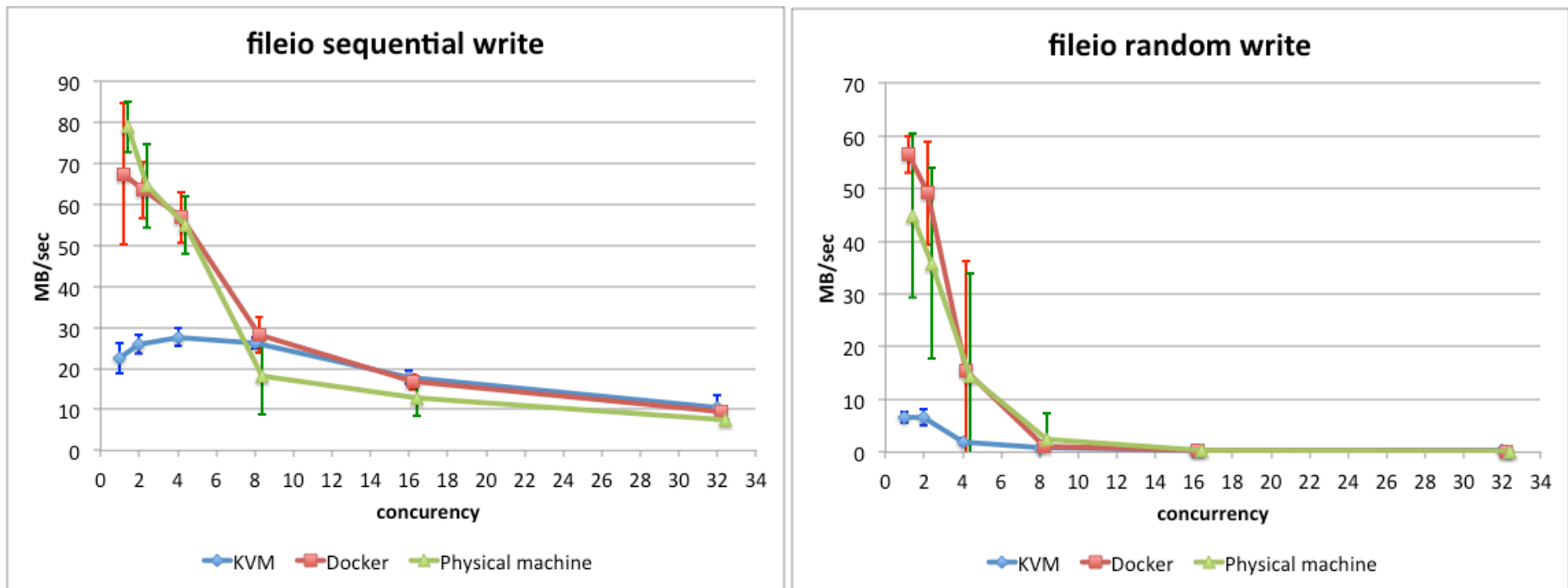


- At low concurrency KVM sequential read is 60-70% worse than native
- KVM random read is several % worse than native
- Docker achieves native performance

- test=fileio, file-test-mode=seqwr, file-block-size=4K, file-total-size=8G, file-num=128, file-extra-flags=direct, max-time=300, num-threads=1
- test=fileio, file-test-mode=rndwr, file-block-size=4K, file-total-size=8G, file-num=128, file-extra-flags=direct, max-time=300, num-threads=1

N: 32

Error bar: SD



- At low concurrency KVM is 70-80% worse than native
- In the case of single request, Docker sequential write is 15% worse than native
- Beside that Docker achieves almost native performance

Summary and Conclusion

- Cloud performance comparison
 - Docker instance becomes ready faster than KVM (building + networking)
- Instance performance comparison
 - CPU and memory performance
 - Native > Docker > KVM
 - KVM
 - File IO performance is poor compared to native
 - Docker
 - Read performance is almost the same as native
 - Write performance is near native
- Docker seems to be a good candidate in the future
 - Nova-docker driver lacks some features and has some bugs
- More investigation is needed
 - Security
 - Stability
 - Other benchmarks (network, volume, tuned KVM)