UKAI: Centrally Controllable Distributed Local Storage for Virtual Machine Disk Images

Keiichi SHIMA <keiichi@iijlab.net>

IEEE Globecom 2013 Workshop on
Cloud Computing Systems, Networks, and Applications

Background

- Virtualization technology deployed widely
- Many services are running on virtual infrastructure
- Continuing demand for cost reduction leads to a larger scale infrastructure operation

Operation Requirements

- Consolidation
  - Accommodate virtual machines working for a same service in nearby hypervisors
- Relocation
  - Push out running virtual machines to upgrade the hosting hypervisor
  - Move virtual machines to newly installed racks or datacenters

What we can/cannot do

- What we can do now
  - CPU and memory migration
  - Networking functions migration
- The harder part is,
  - Storage functions migration

Technical Requirements

- Controllability
- Redundancy
- Performance

What is Controllability?

- Location controllability
  - I don’t want to place pieces of a virtual disk content to somewhere mathematically calculated location
- Granularity
  - I want to control the location of a virtual disk per virtual disk (per virtual machine)
- Migration schedule
  - I don’t want to generate heavy traffic for storage migration when we have a lot of user traffic in daytime

What is Redundancy?

- A virtual disk should have multiple replicas

- Requirements for the number of replicas may be different per virtual machine

- Dynamic control of the number of replicas

Performance

- A virtual disk substance should be placed or replaced on a storage node near the hypervisor on which the virtual machine is running

- Replication operations should not disrupt the virtual machine operation

Existing Techs

- Live Storage Migration [Hirofuchi2009]
- DRBD [Ellenberg2008]
- Distributed Filesystems [Weil2006, Gluster2010, Morita2010]

[Hirofuchi2009] T. Hirofuchi, H. Ogawa, H. Nakada, S. Itoh, and S. Sekiguchi, “A Live Storage Migration Mechanism over WAN for Relocatable Virtual Machine Services on Clouds,” in Proceedings of the 2009 9th IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGRID’09), 2009, pp. 460–465.
[Ellenberg2008] L. Ellenberg, “DRBD(R)9 & Device-Mapper Linux(R) Block Level Storage Replication,” in Proceedings of Linux-Kongress 2008, October 2008.
[Weil2006] S. A. Weil, S. A. Brandt, E. L. Miller, D. D. E. Long, and C. Maltzahn, “Ceph: A Scalable, High-Performance Distributed File System,” in Proceedings of the 7th symposium on Operating systems design and implementation (OSDI’06). USENIX, 2006, pp. 307–320.
[Gluster2010] Gluster Inc., “Gluster File System Architecture,” Gluster Inc., Tech. Rep., 2010 .
[Morita2010] K. Morita, “Sheepdog: Distributed Storage System for QEMU/KVM,” Linux.conf.au 2010, January 2010.

Live Storage Migration

- Copy on demand
  - When a virtual machine has migrated, it points the old storage location
  - When access to some part of the storage, the related block data will be transferred from the original location to the local storage
- Drawbacks
  - No controllability and redundancy

DRBD

- A kind of RAID0 system over a network
- Drawbacks
  - Difficult to control the replica level
  - No per virtual machine control mechanism
  - Relocation

Distributed Filesystems

- Ceph, Gluster, and Sheepdog
  - Data will be autonomously distributed to a set of storage nodes
- Drawbacks
  - Using a consistent hash mechanism causes issues on data placement/replacement
  - That also causes performance issue when delay between storage nodes are large

Do we need a Super-duper Filesystem?

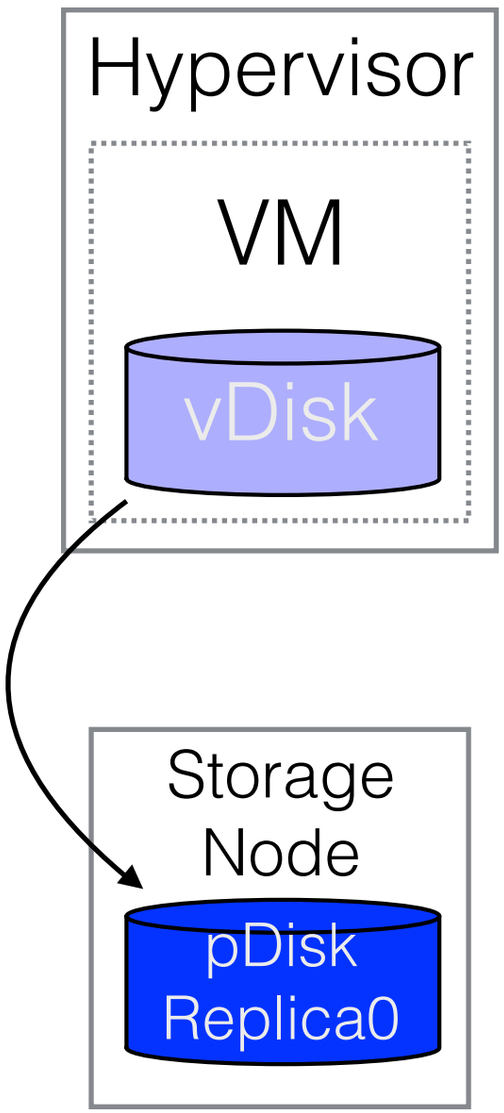
- No we don't

- We have different focuses from existing file/storage systems

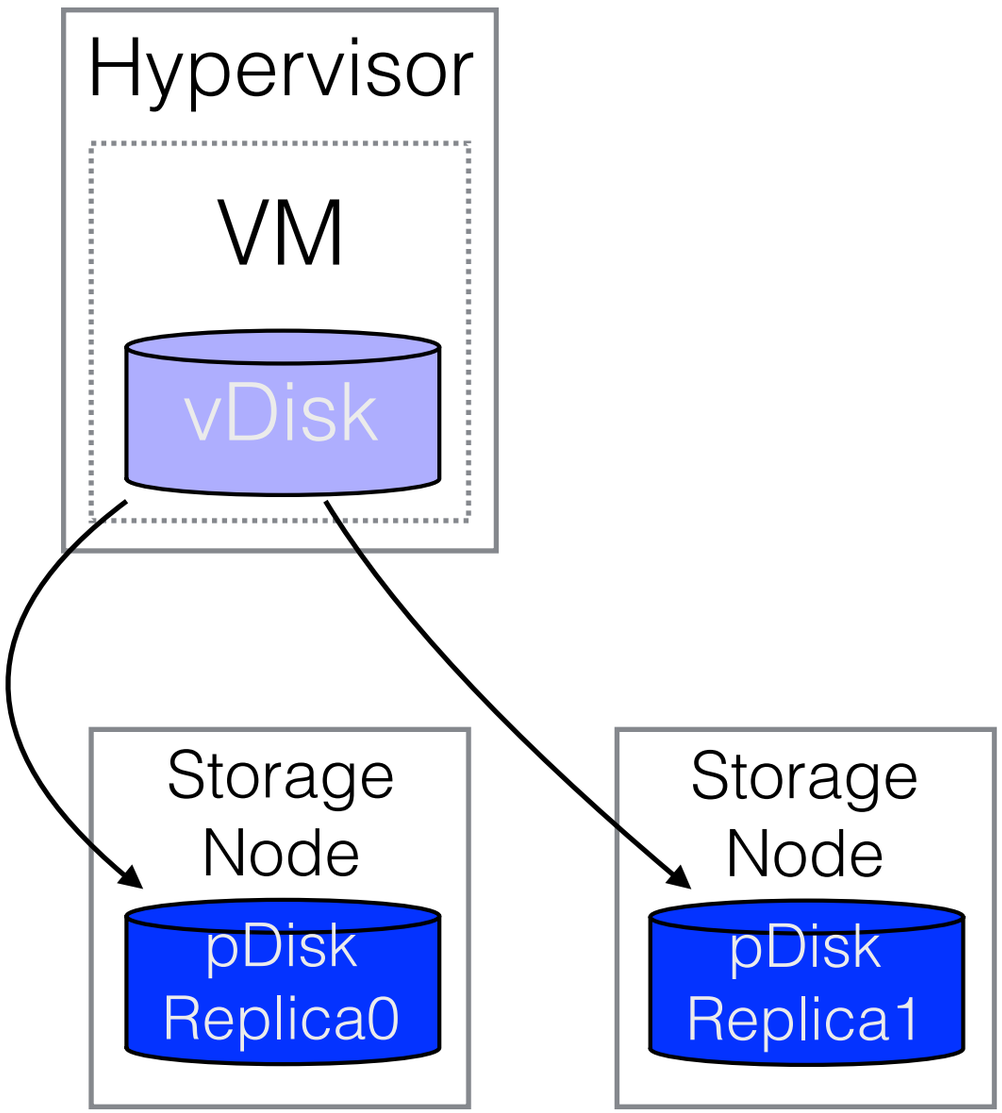
Our Field

- A virtual disk image is used only by one virtual machine at one time
  - No concurrent access from multiple users
  - Required metadata is simple and updated only by known hypervisors using the virtual disk
  - No need for a distributed resource locking mechanism

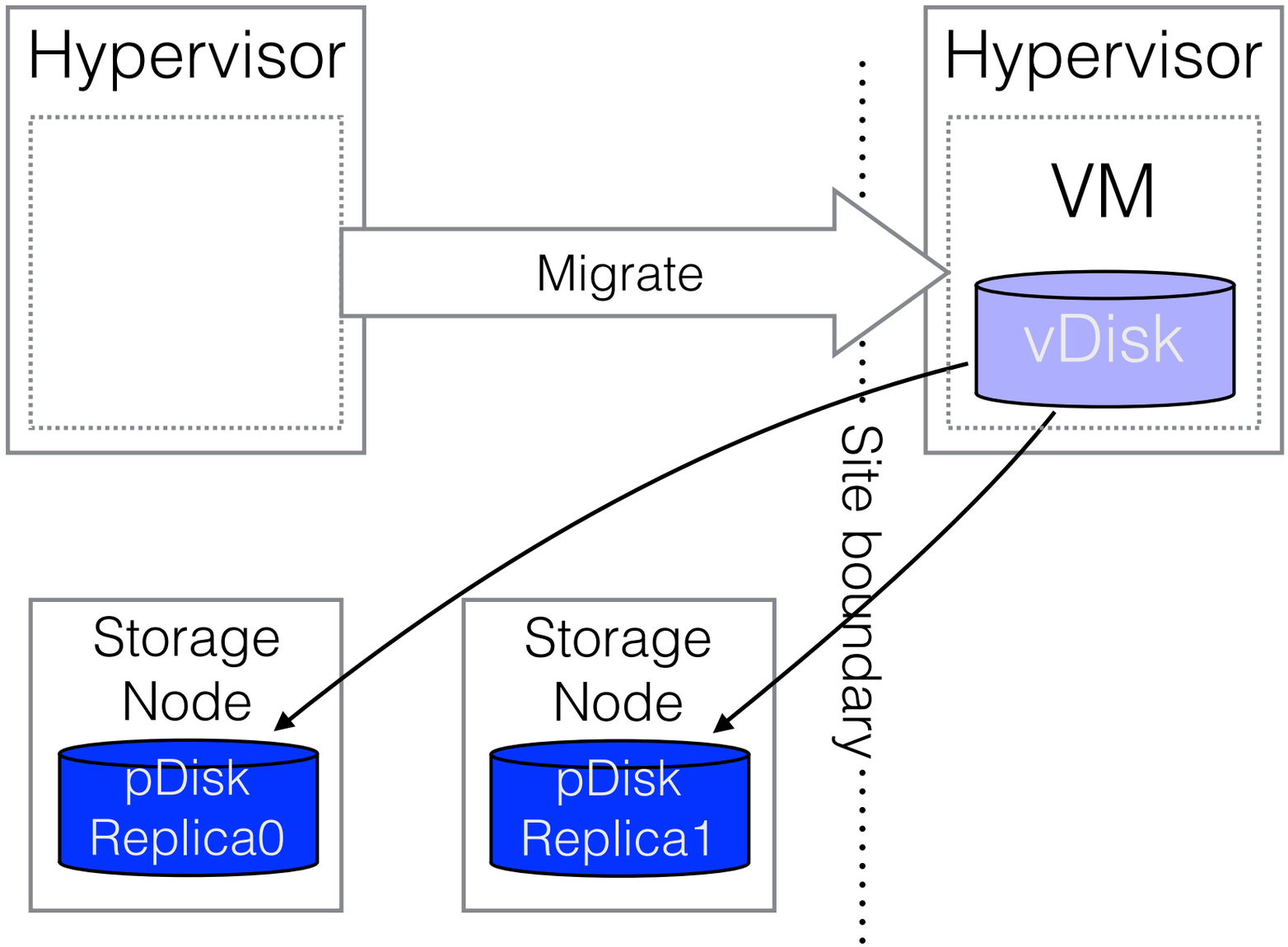
Operation Image



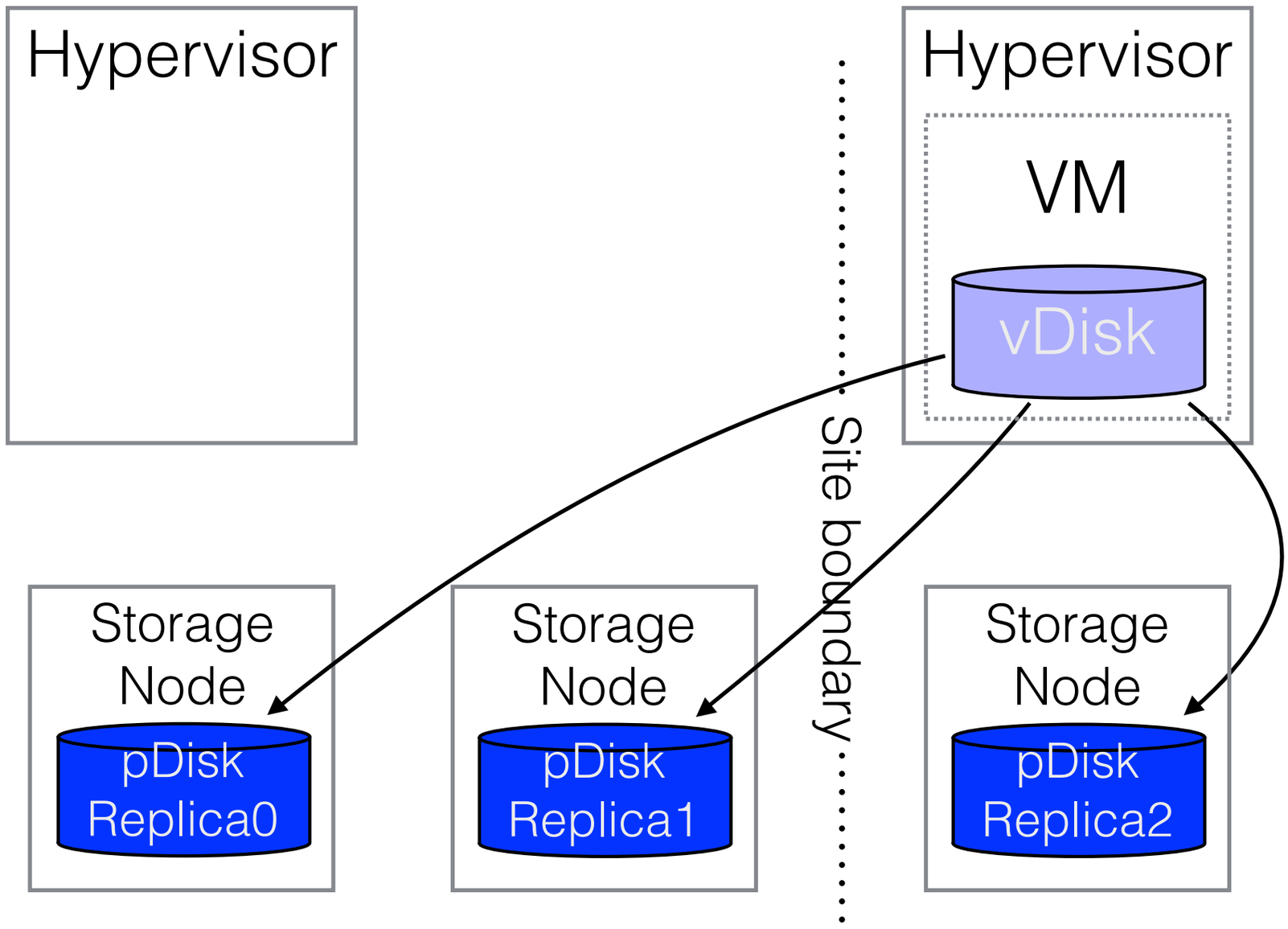
Operation Image



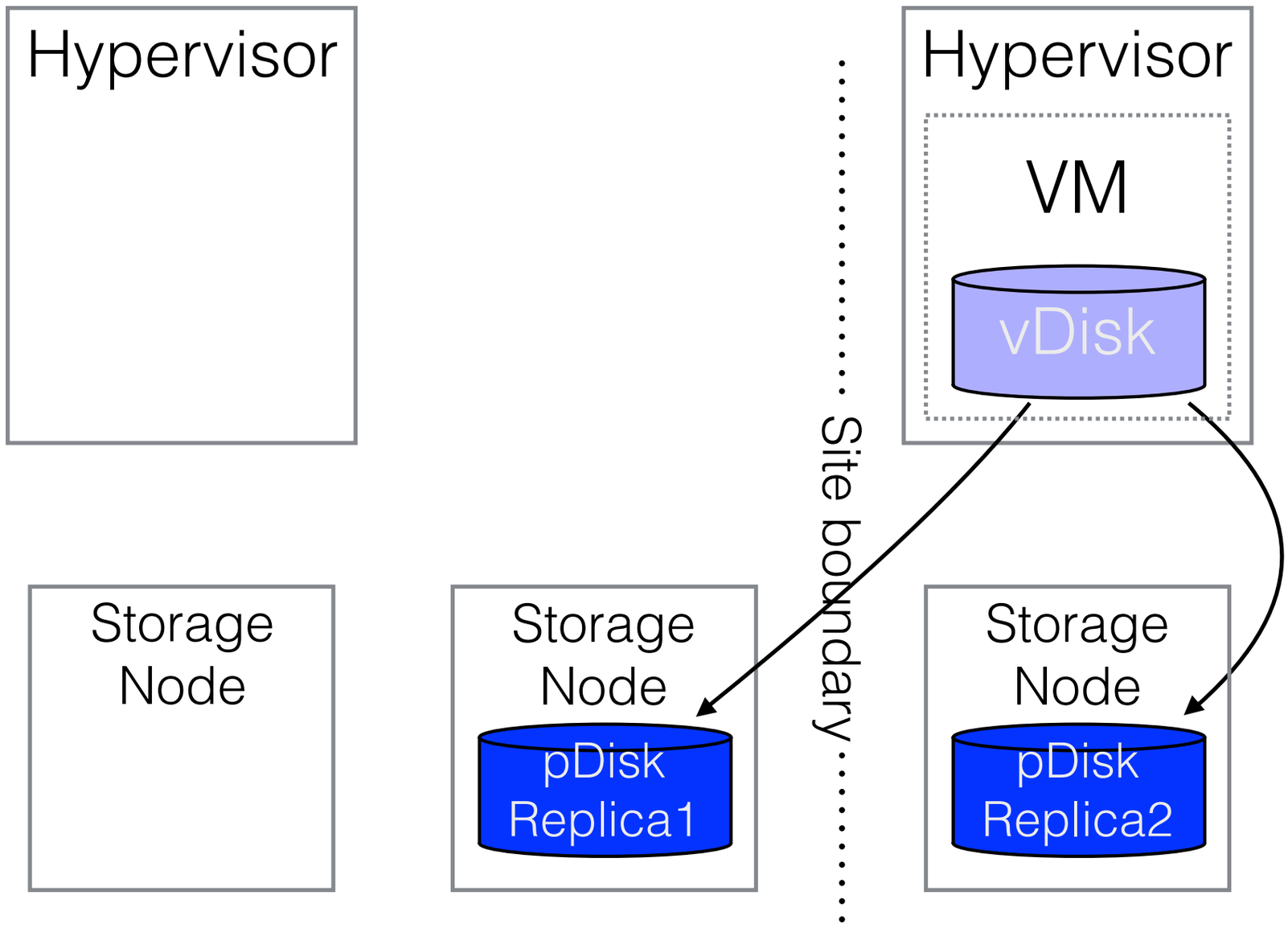
Operation Image



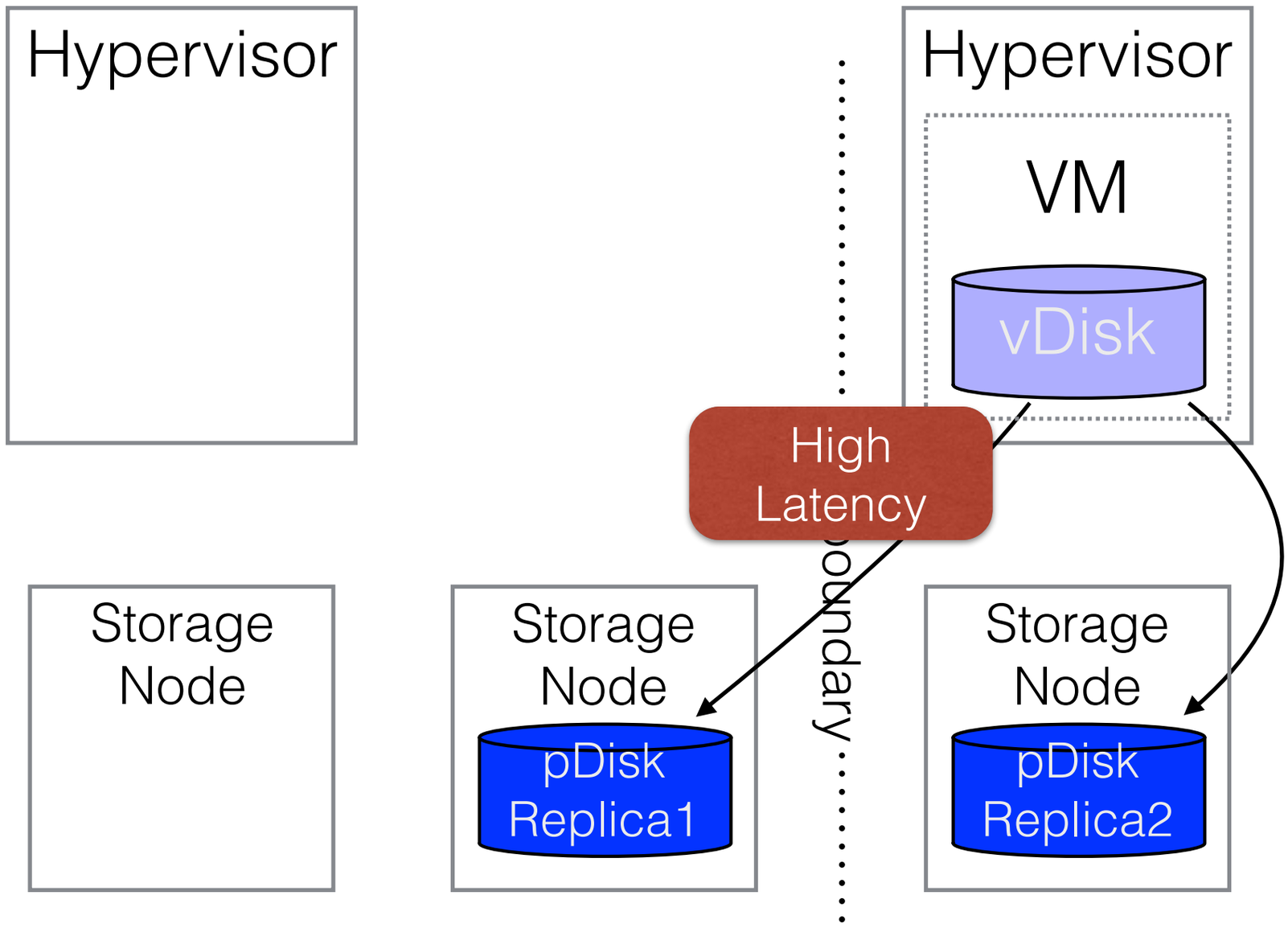
Operation Image



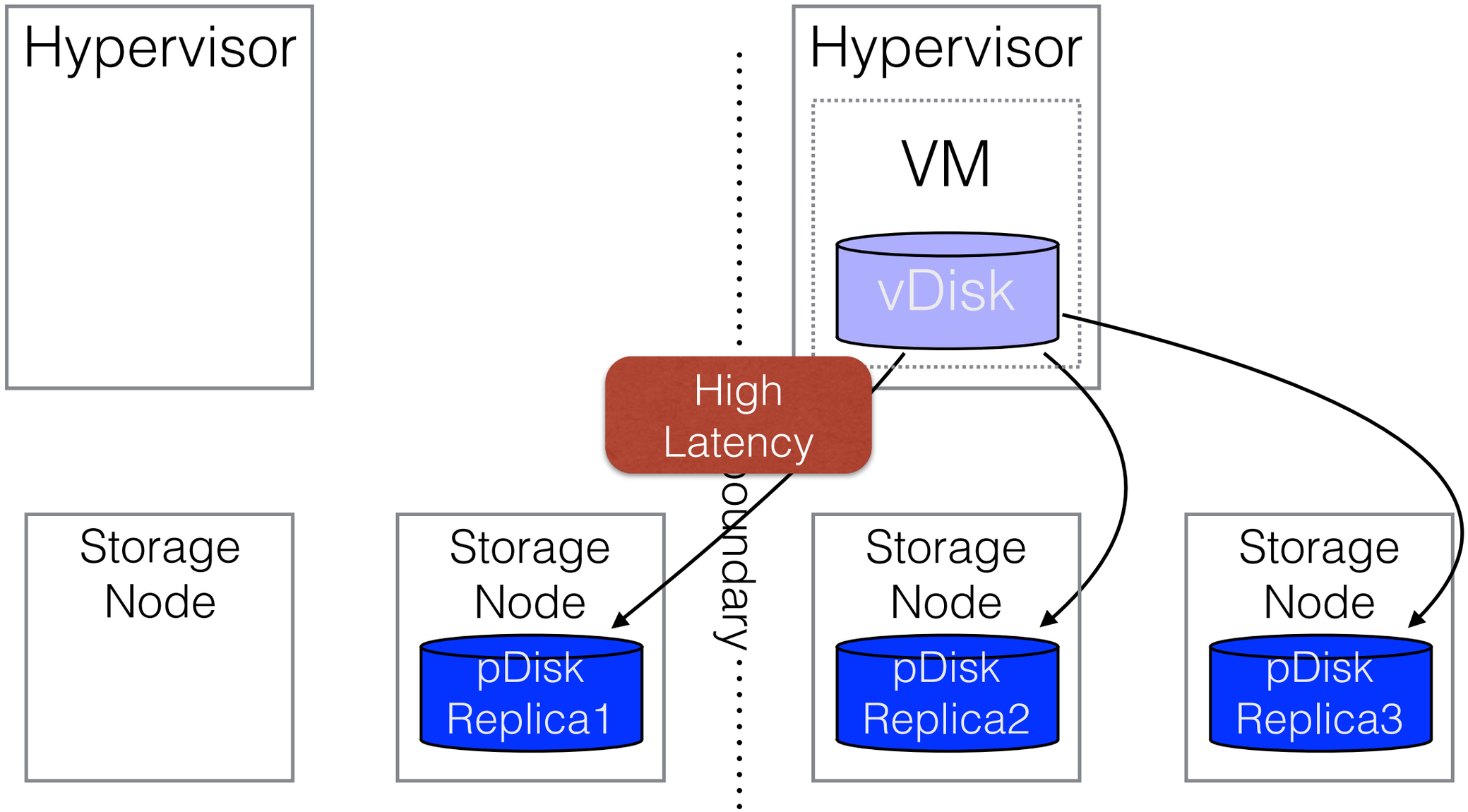
Operation Image



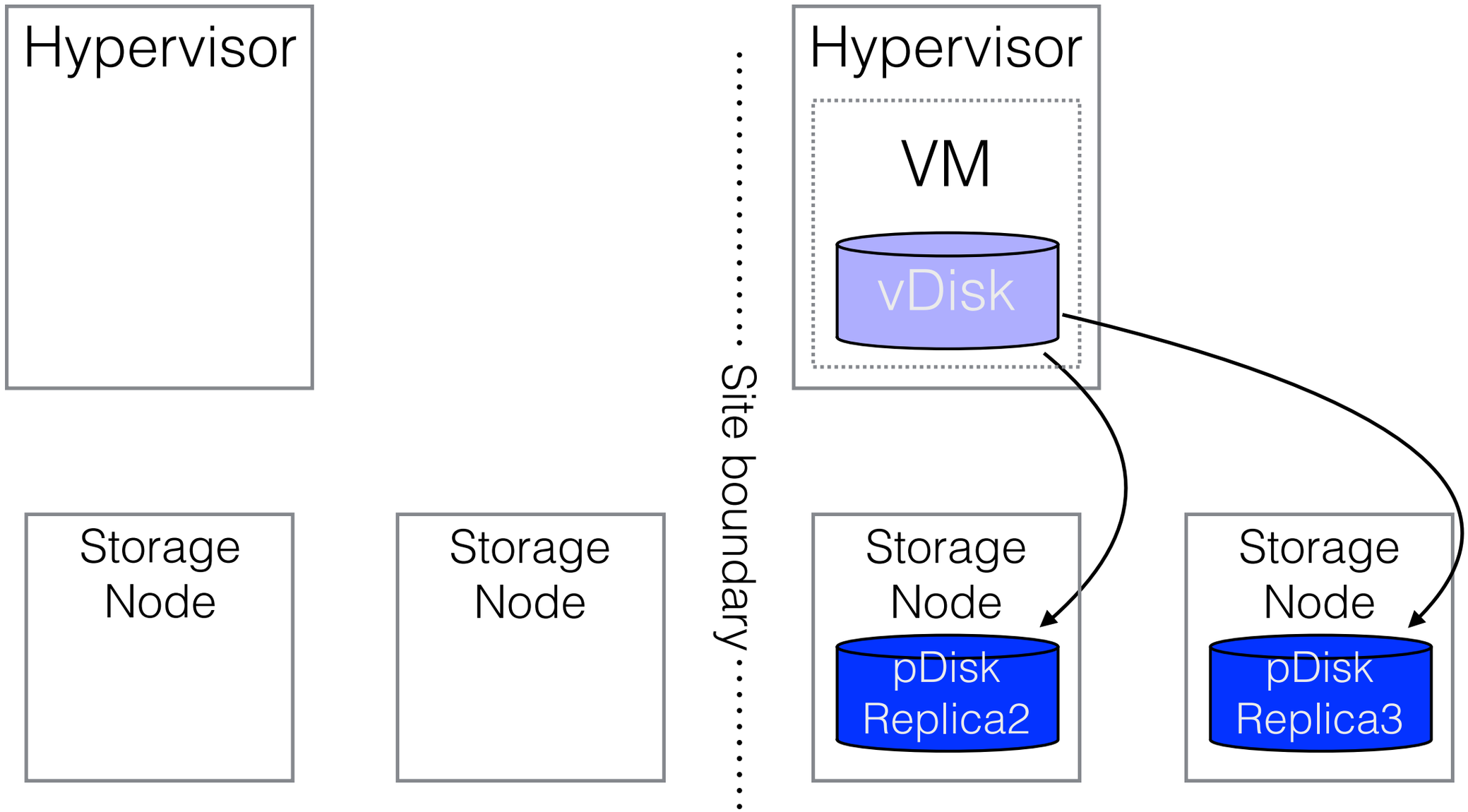
Operation Image



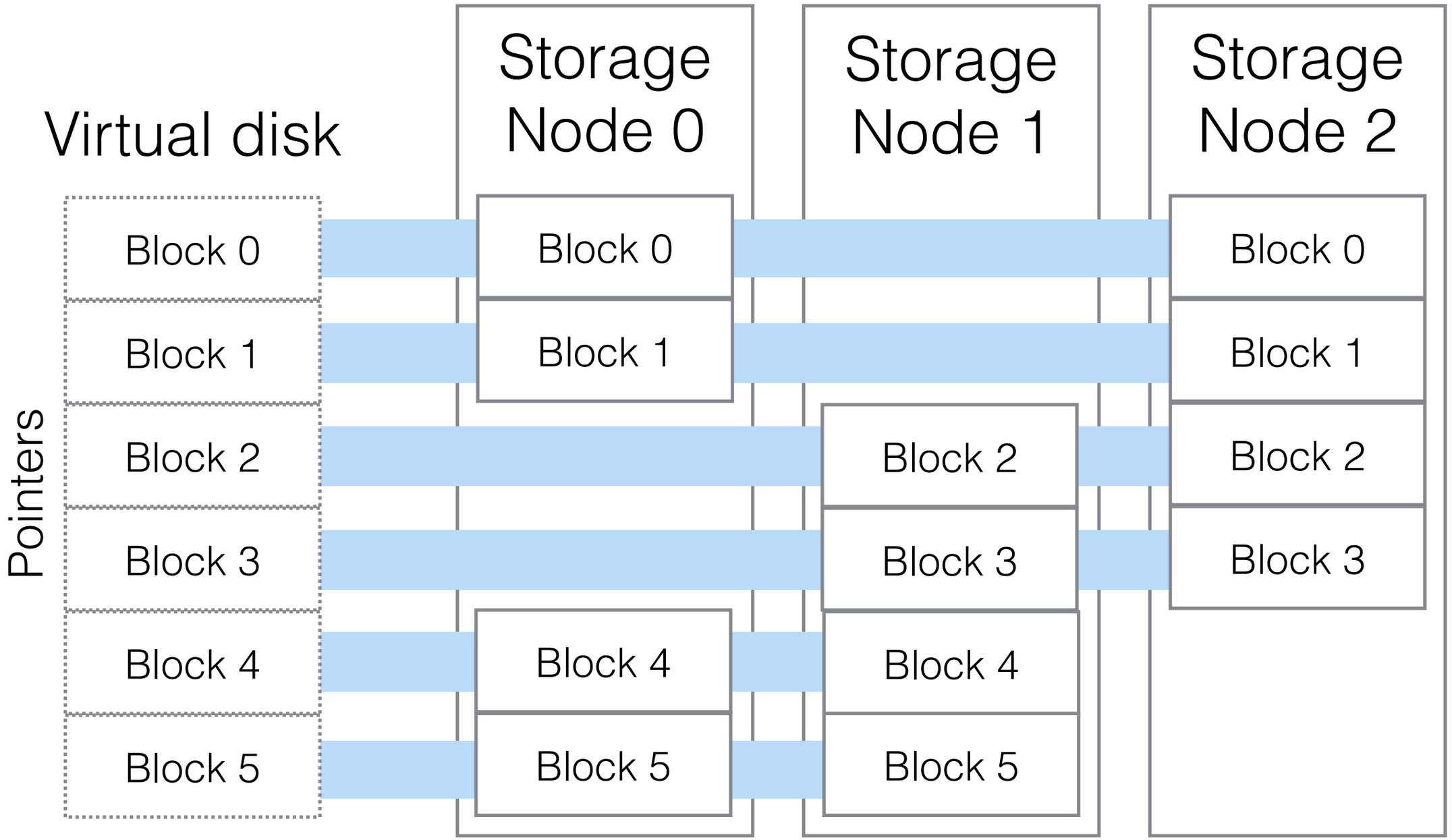
Operation Image



Operation Image



Disk Image Structure



Metadata Structure

Name, Size, Blocksize

List of hypervisors = metadata loc.

Block 0

Block 1

Block 2

Block 3

Block 4

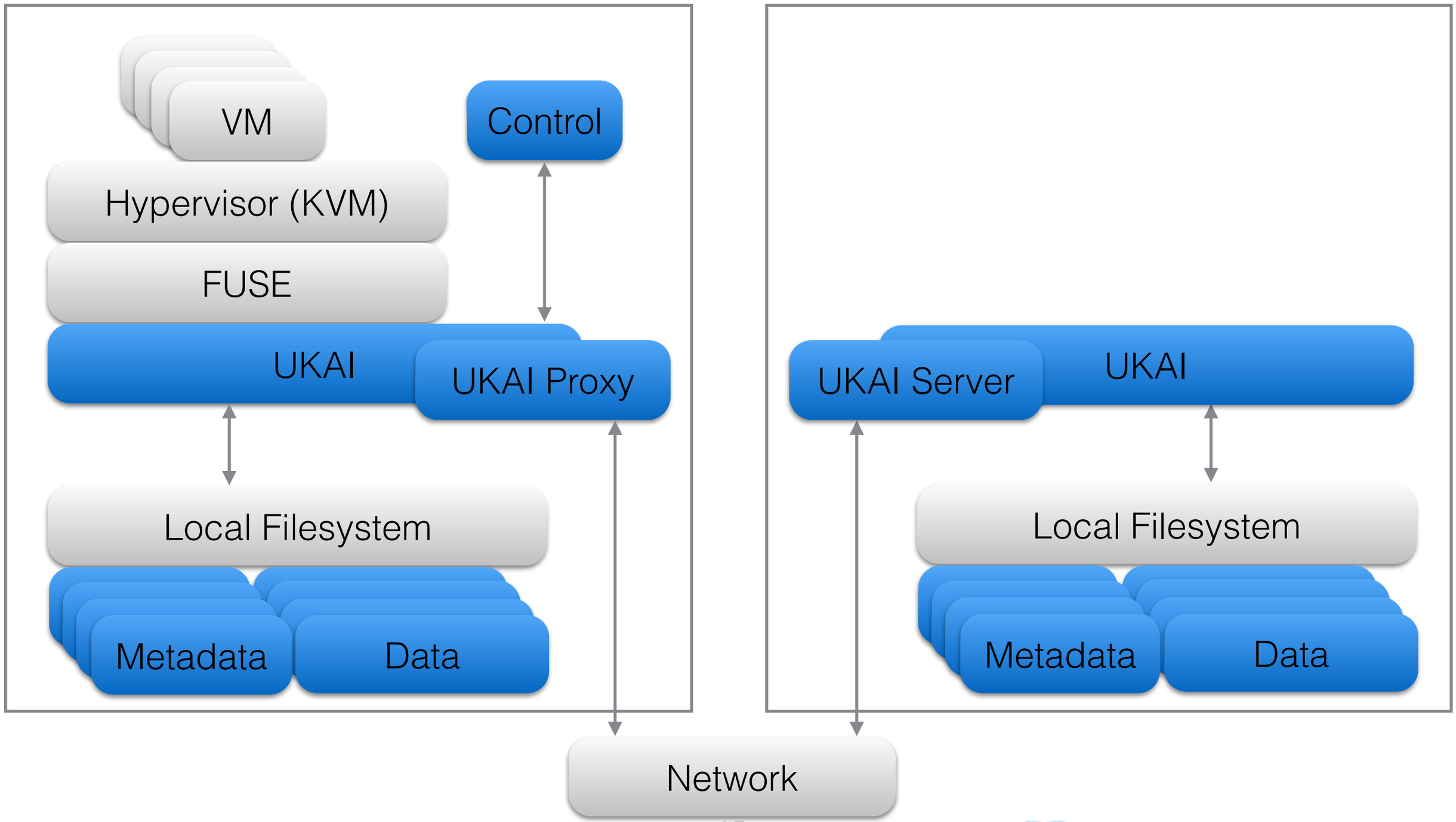
Block 5

Sync status for each block

Storage node location

```
{"name": "image01", "size": 1200000, "block_size": 200000,
 "hypervisors": [ "192.0.2.100", "192.168.2.110" ],
 "blocks": [
  { "192.0.2.100": { “in_sync": 0 },
    "192.0.2.101": { “in_sync": 2 },
    "192.0.2.102": { “in_sync": 0 }},
  { "192.0.2.100": { "in_sync": 0 },
    "192.0.2.101": { "in_sync": 2 },
    "192.0.2.102": { "in_sync": 0 }},
  { "192.0.2.100": { "in_sync": 2 },
    "192.0.2.101": { "in_sync": 0 },
    "192.0.2.102": { "in_sync": 0 }},
  { "192.0.2.100": { "in_sync": 0 },
    "192.0.2.101": { "in_sync": 0 },
    "192.0.2.102": { "in_sync": 0 }},
  { "192.0.2.100": { "in_sync": 0 },
    "192.0.2.101": { "in_sync": 2 },
    "192.0.2.102": { "in_sync": 2 }},
  { "192.0.2.100": { "in_sync": 0 },
    "192.0.2.101": { "in_sync": 0 },
    "192.0.2.102": { "in_sync": 2 }}]
}
```

Prototype Implementation



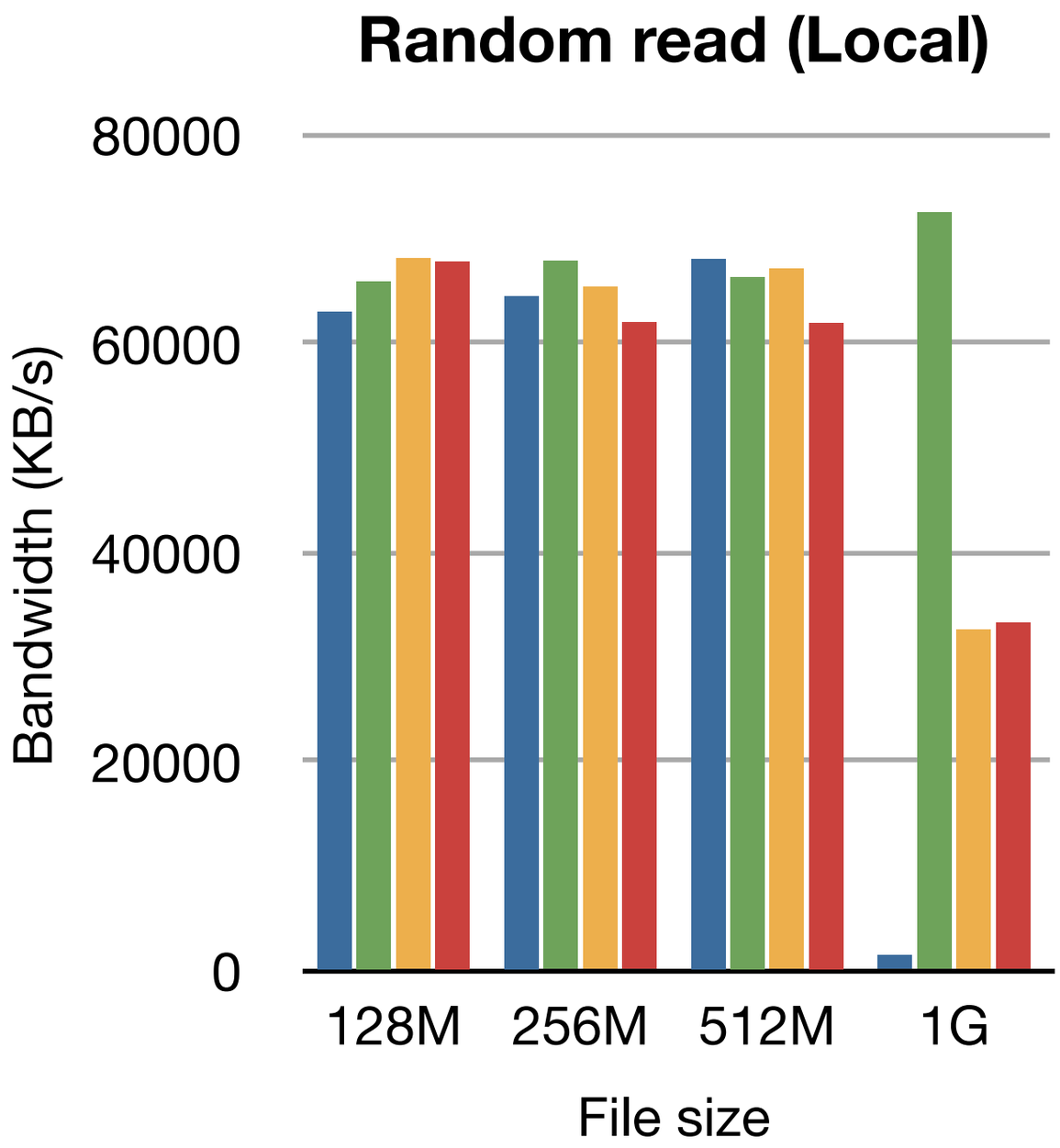
Preliminary Measurement

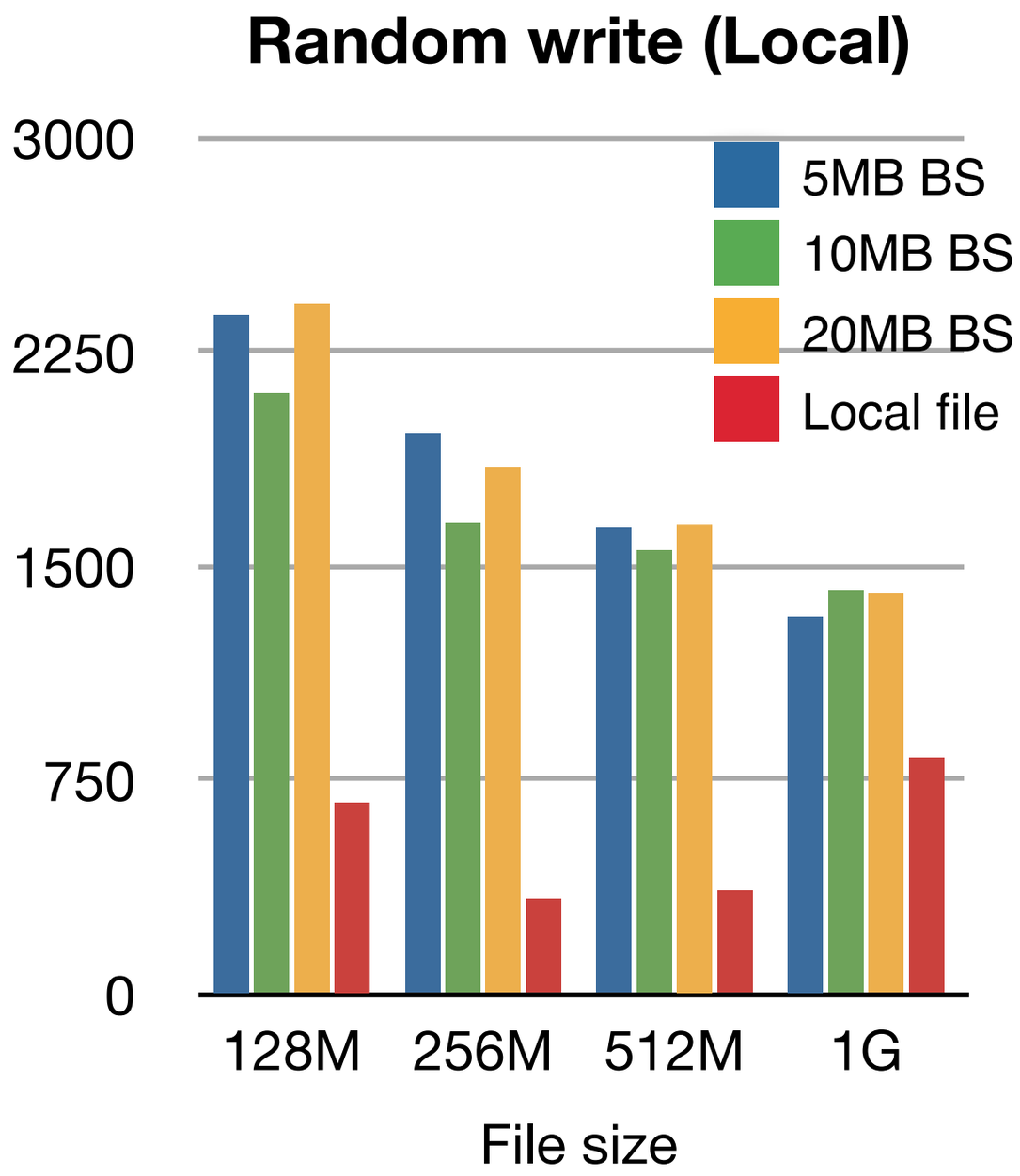
- Virtual disk on a local file v.s. UKAI local access
- Virtual disk on NFS v.s. UKAI remote access
- UKAI mirror (one in local, one in remote)

Preliminary Measurement

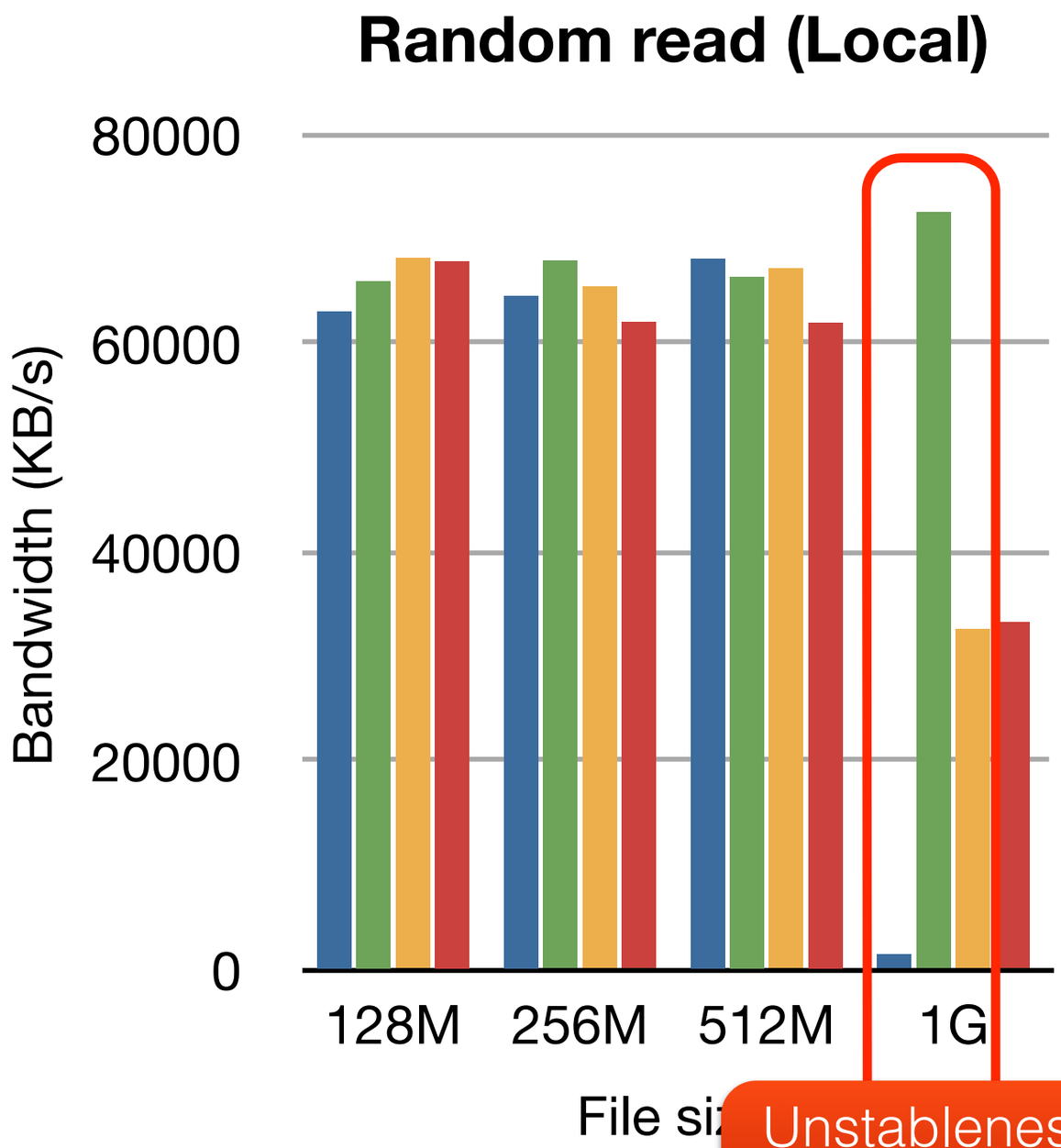
- Hypervisor specification
  - Intel Core2 Duo E8400 @ 3GHz
  - Intel E1000 Gigabit Ethernet
  - 250GB SATA HDD
  - Ubuntu 12.04 + KVM
- Virtual machine specification
  - 1 vCPU + 1GB memory
  - 8GB virtual disk
  - Ubuntu 12.04
- Measurement software was `fio` (http://freecode.com/projects/fio)

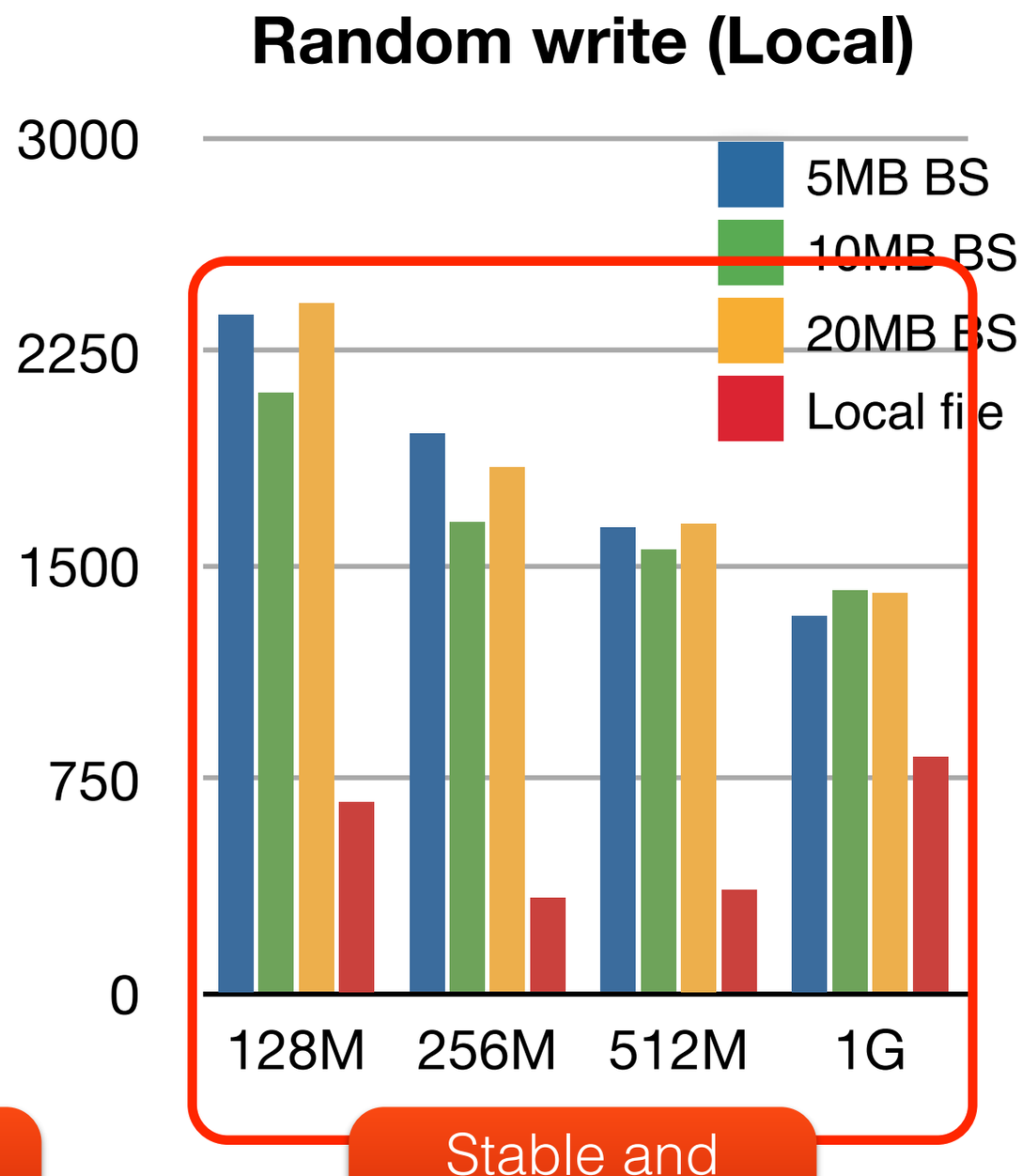
File v.s. UKAI local





File v.s. UKAI local

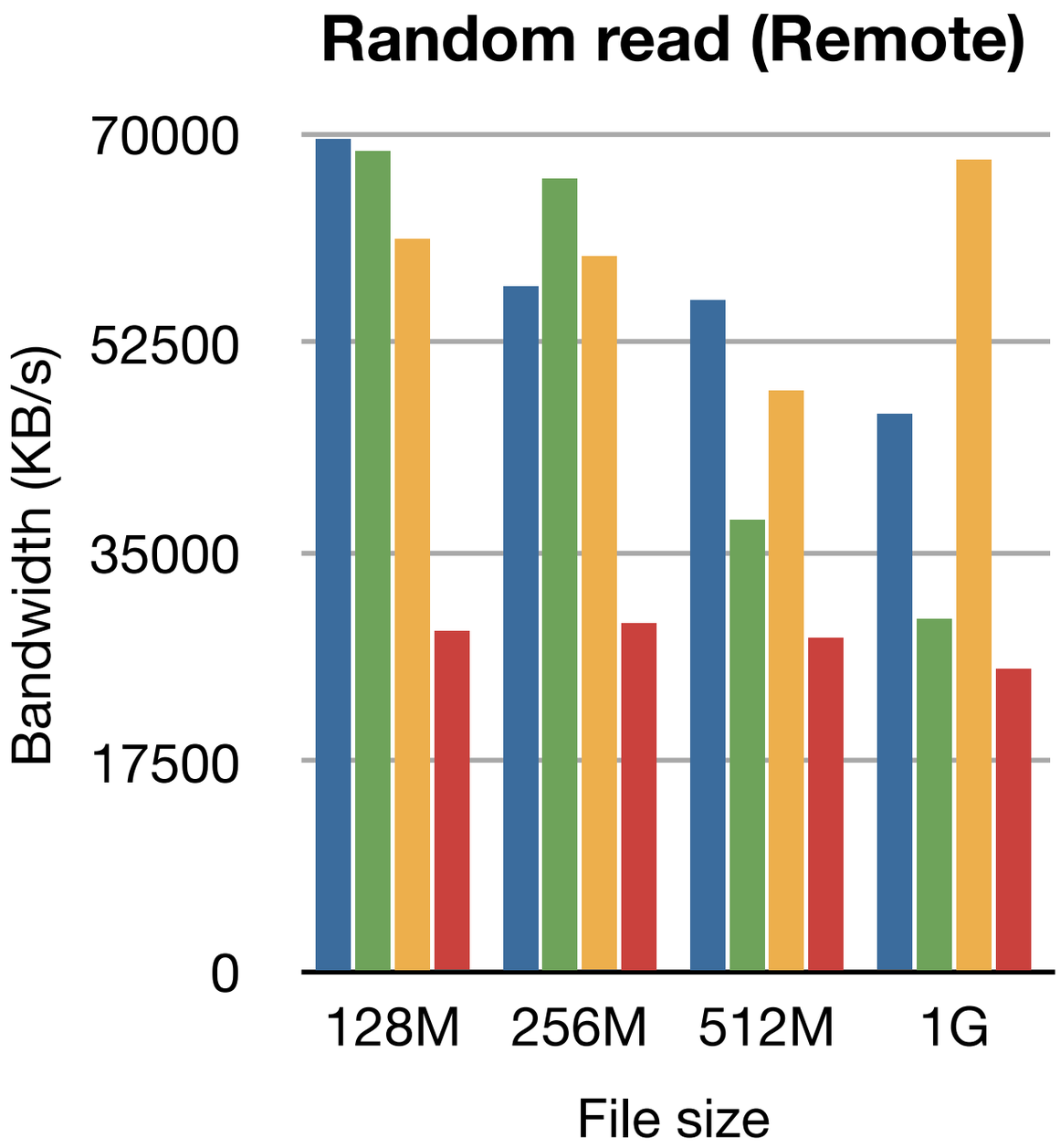


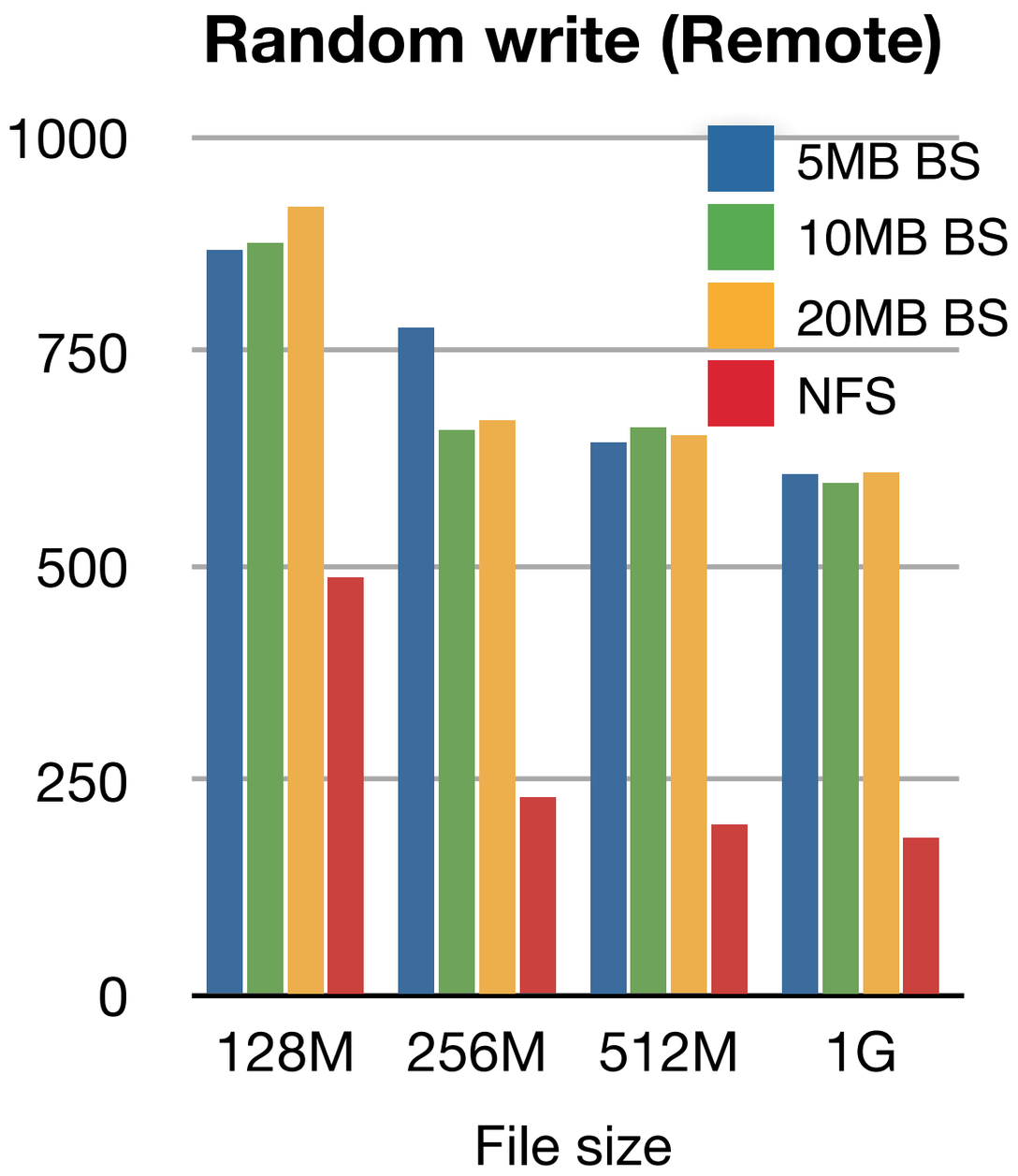


Unstableness Observed

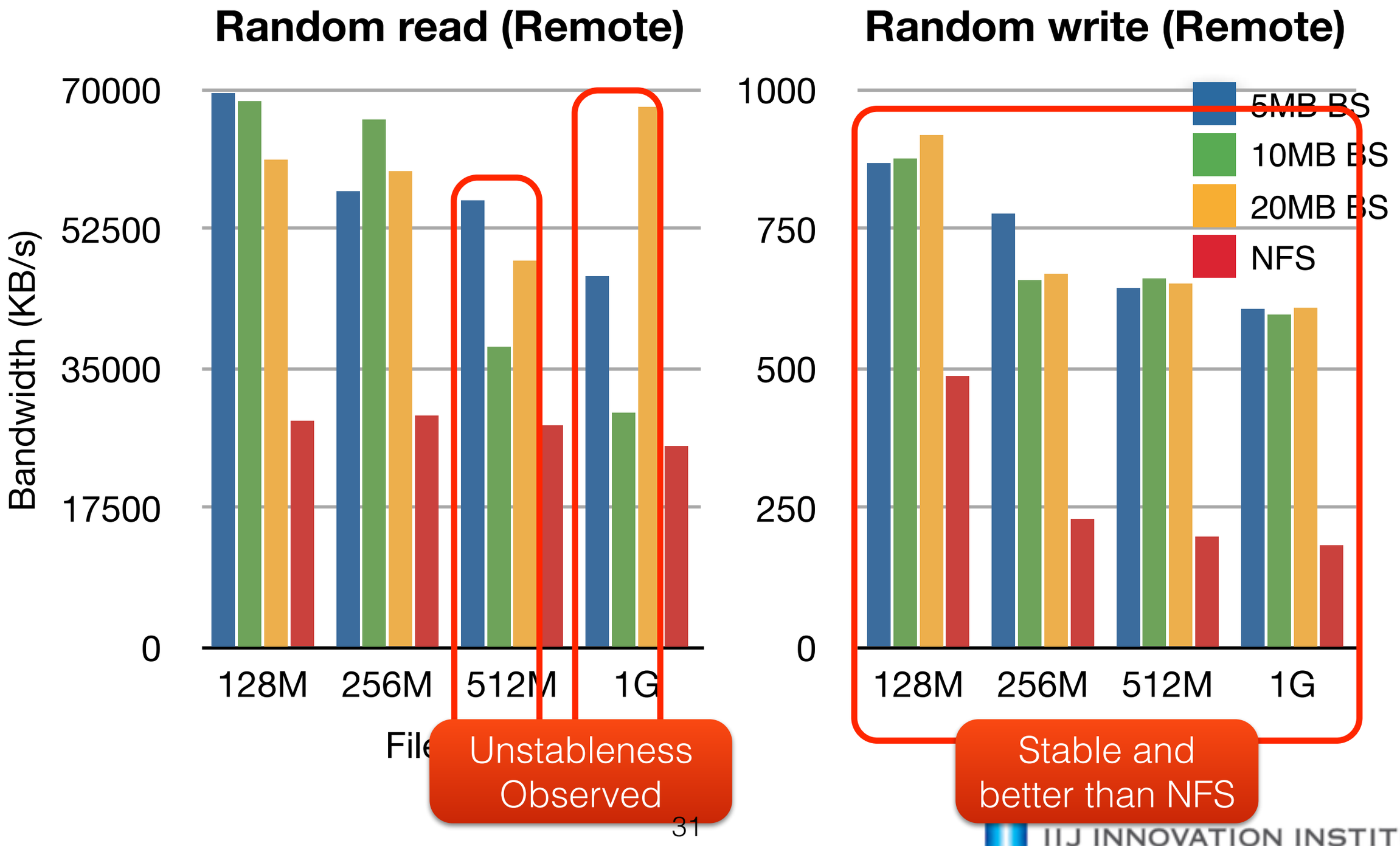
Stable and better than local

NFS v.s. UKAI remote

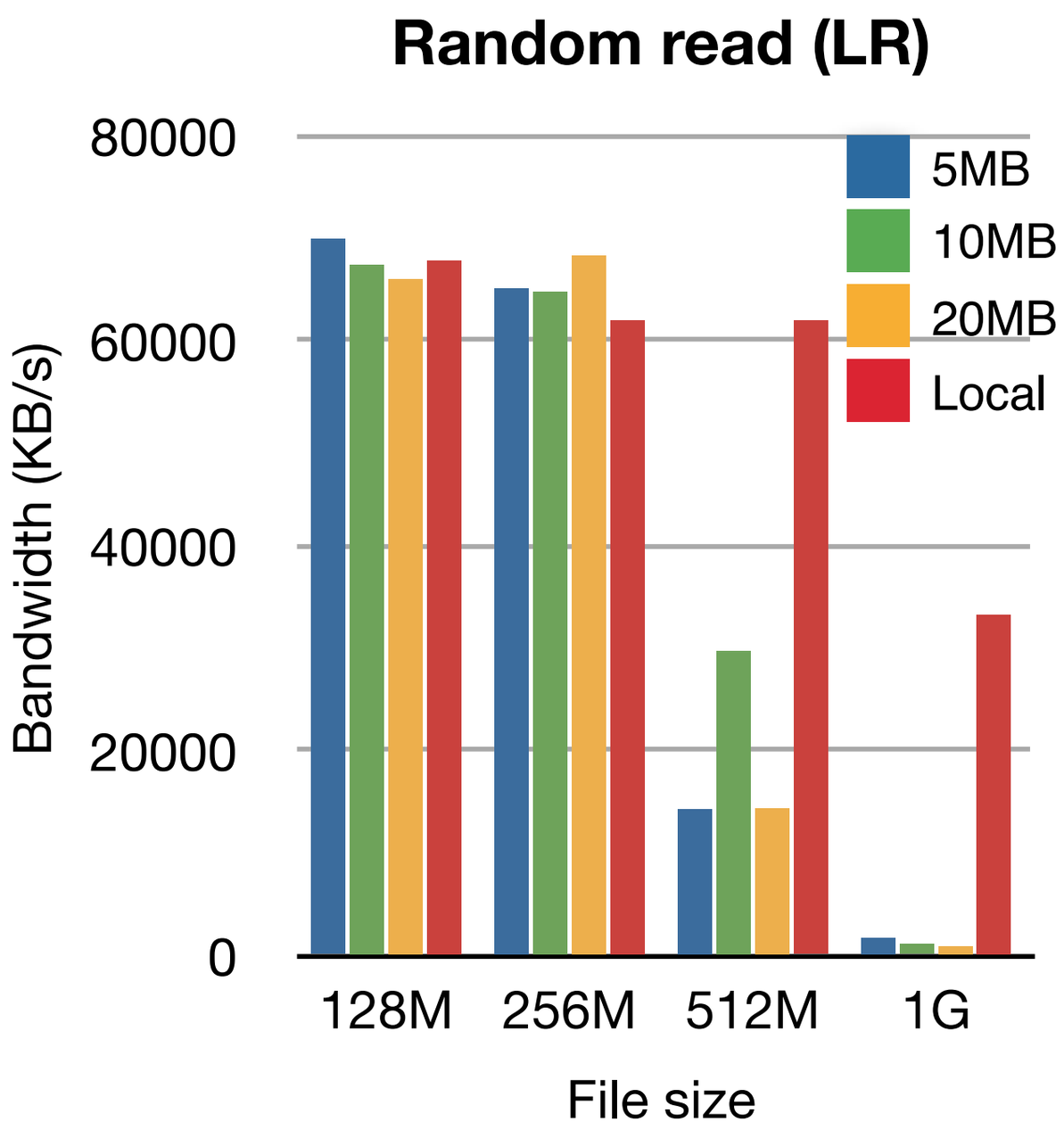


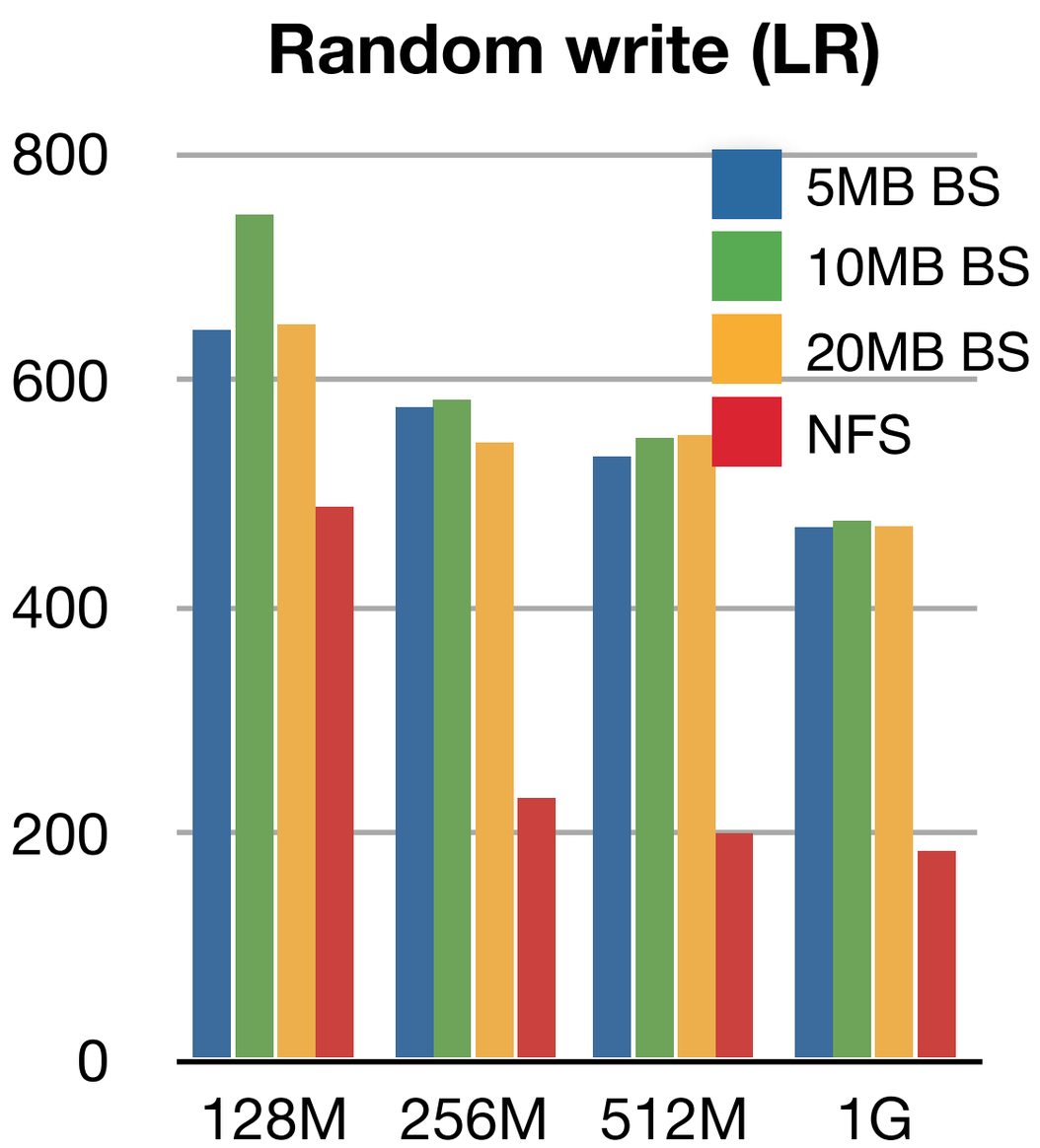


NFS v.s. UKAI remote

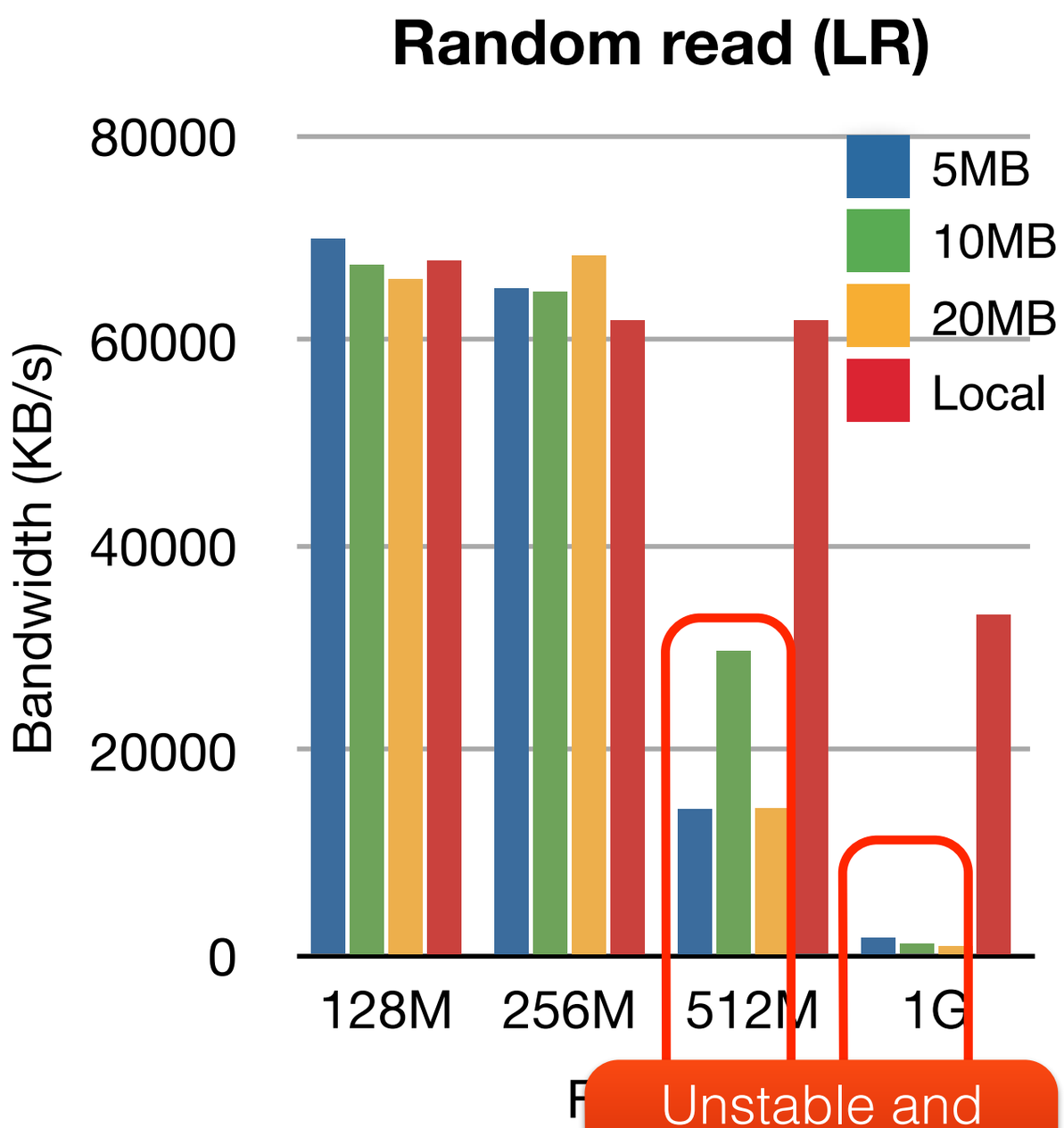


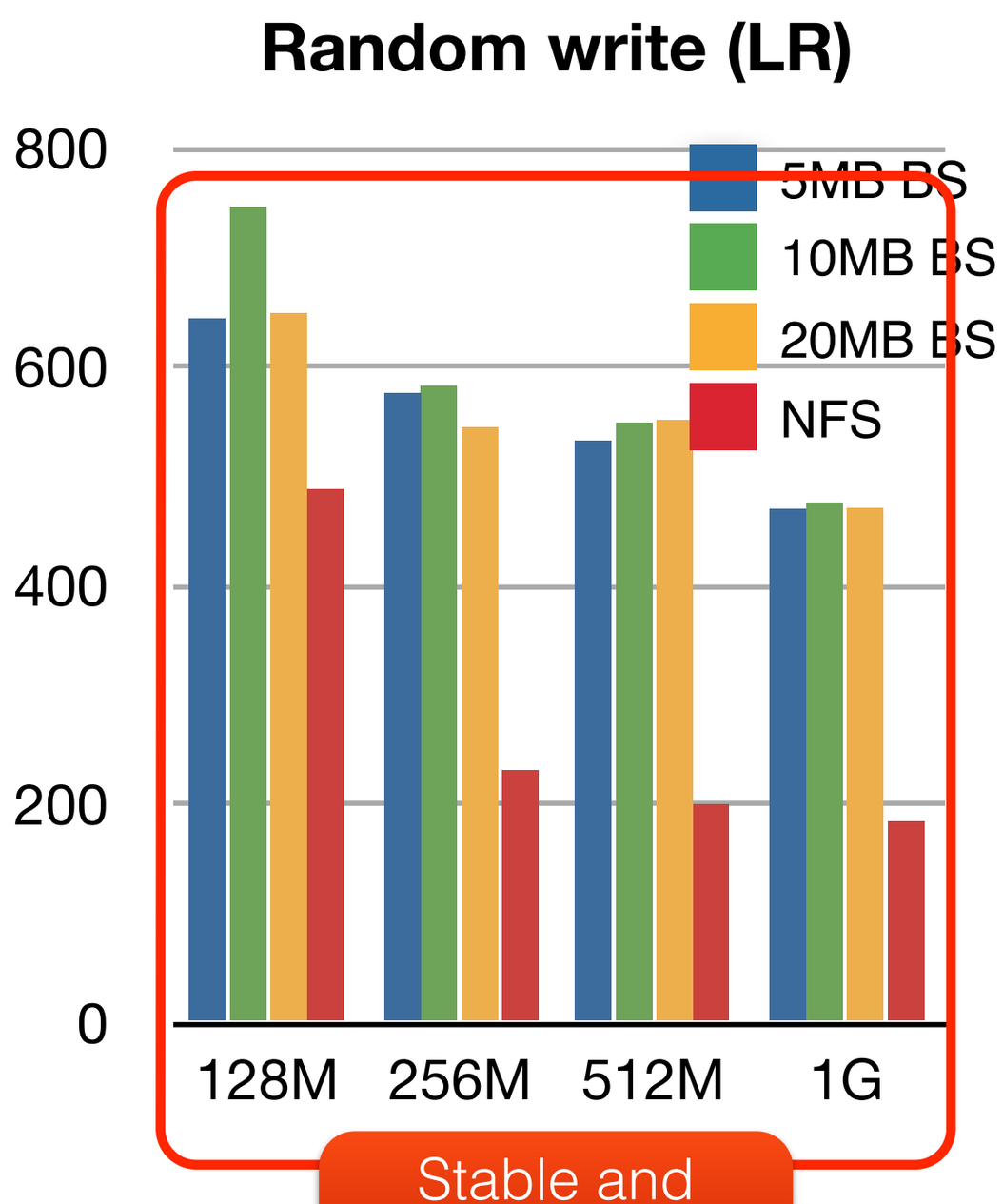
UKAI mirror


Random read (LR)
80000
60000
40000
20000
0
Bandwidth (KB/s)
5MB
10MB
20MB
Local
128M
256M
512M
1G
File size


Random write (LR)
800
600
400
200
0
5MB BS
10MB BS
20MB BS
NFS
128M
256M
512M
1G

UKAI mirror

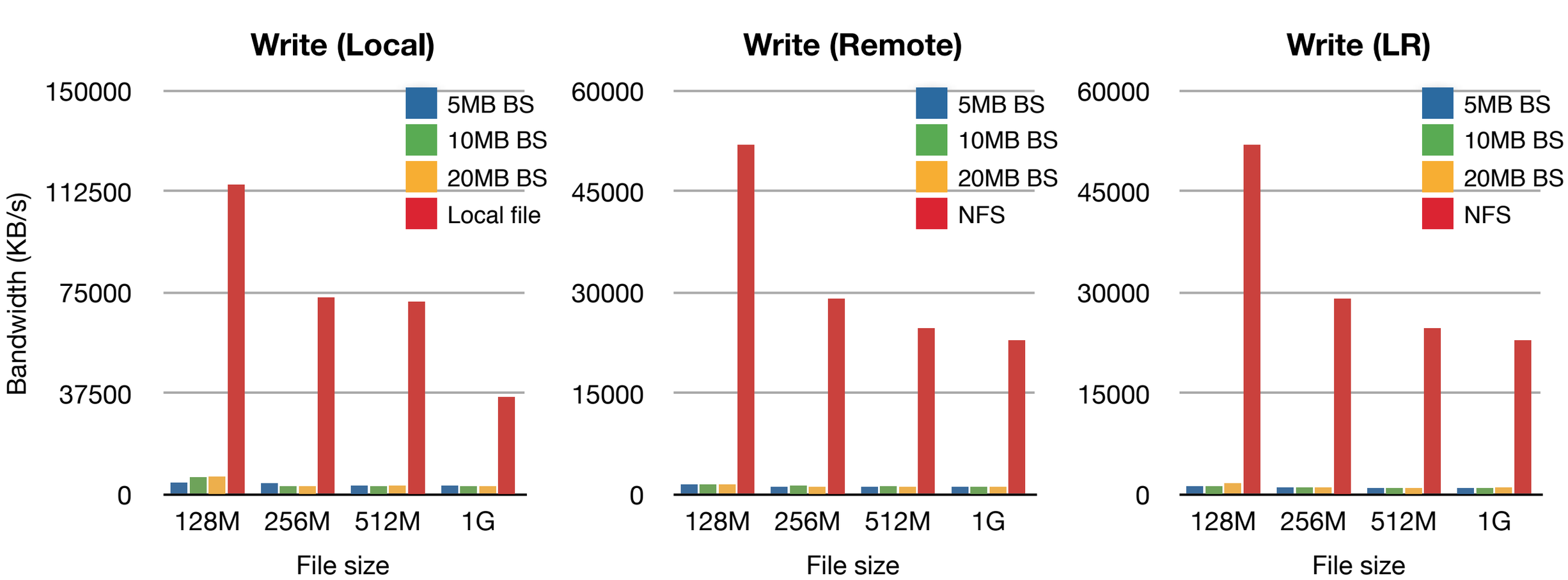

Random read (LR)
Bandwidth (KB/s)
80000
60000
40000
20000
0
128M
256M
512M
1G
5MB
10MB
20MB
Local
Unstable and worse than local


Random write (LR)
800
600
400
200
0
128M
256M
512M
1G
5MB BS
10MB BS
20MB BS
NFS
Stable and better than NFS

But,

Sequential Write

- Quite bad performance observed in sequential write test cases
- We are investigating the problem



Summary

- Defined requirements for the virtual disk operation in a distributed hypervisor operation environment
  - Controllability, Redundancy, and Performance
- Designed a block device architecture to satisfy the requirements
- Implemented prototype software and had a preliminary measurement
  - Almost equal or better in most of random access cases

Future Works

- Need to improve sequential write performance
  - Need more investigation on where is the bottleneck
- Write back cache
- Parallelization
  - Operations can be done in parallel in some cases
- Operation support
  - Need to integrate with cloud orchestration/control systems

Questions?

Prototype software is available from
http://github.com/keiichishima/ukai/