Multi-tenancy: a real-life implementation

April, 2018

Sebastien Buisson
Thomas Favre-Bulle
Richard Mansfield

© 2018 DDN Storage

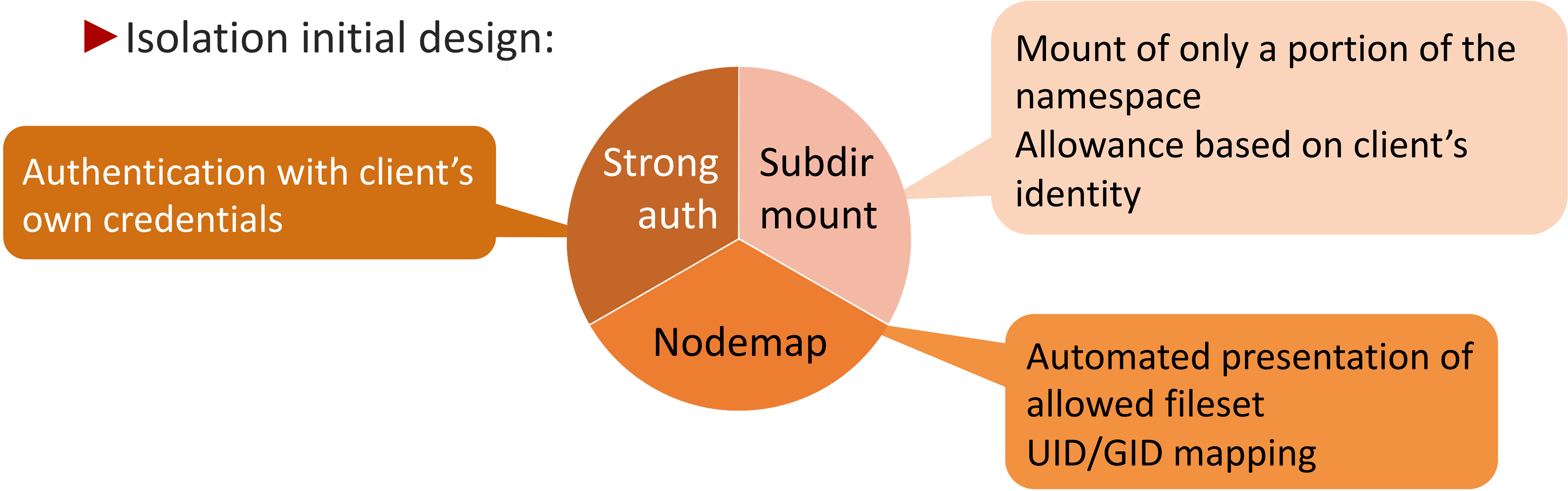
Multi-tenancy: a real-life implementation

- The Multi-Tenancy concept
- Implementation alternative: Uppsala real-life use case
  - Customer requirements
  - Cluster architecture
  - Software implementation
  - Performance evaluation

© 2018 DDN Storage

The Multi-Tenancy concept

- Isolation initial design:



Authentication with client's own credentials

Mount of only a portion of the namespace
Allowance based on client's identity

Automated presentation of allowed fileset
UID/GID mapping

- Isolation enables Multi-tenancy:
  - Different populations of users on the same file systems
  - Isolation of these different populations of users

The Multi-tenancy concept

▶ What if strong authentication not possible?

- Need to find another way to trust client's ID

▶ Reasons for not having strong authentication

- Not implemented on-site for user authentication
    - Too difficult starting to use strong authentication with Lustre
- Not adapted to application workflows
    - Too complex to deploy credentials for VMs or Containers

© 2018 DDN Storage

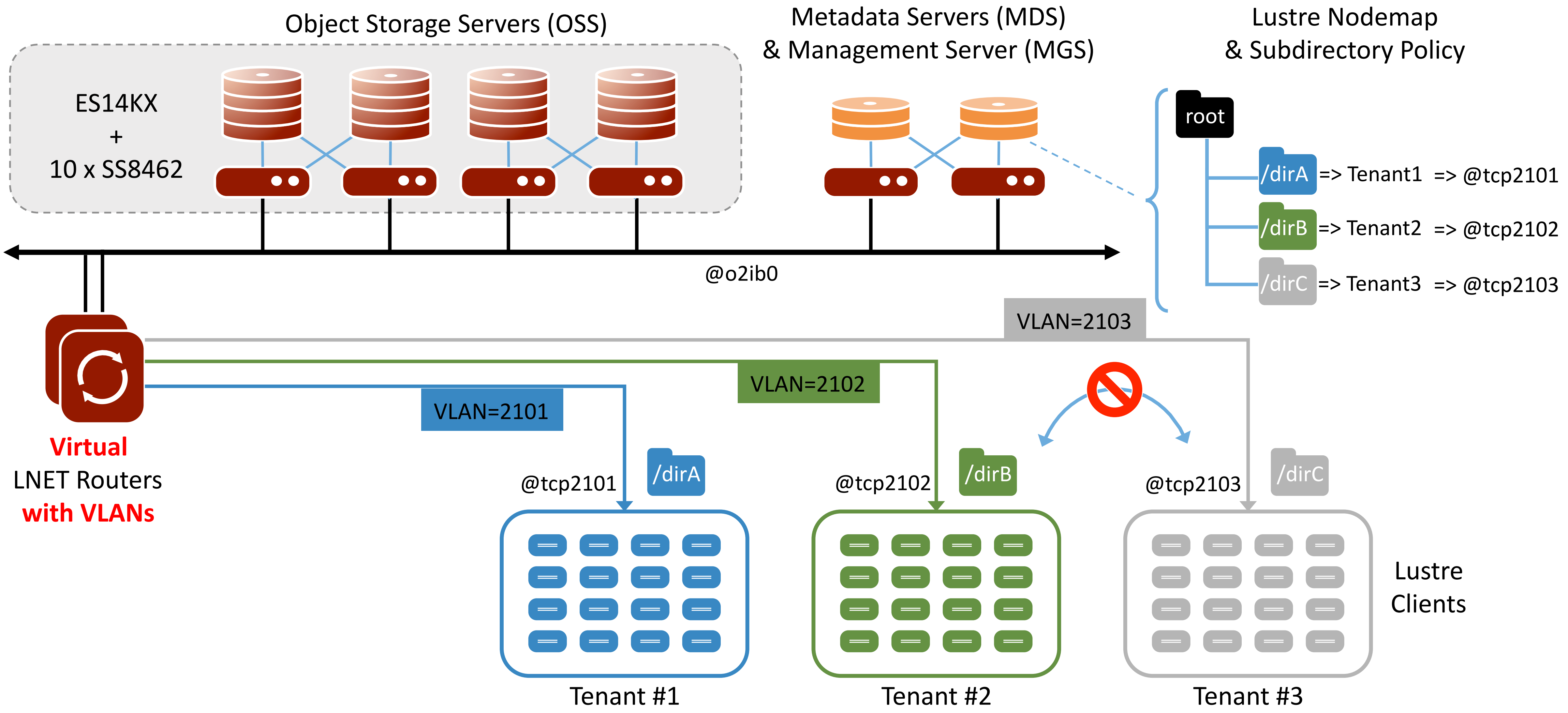
Uppsala real-life use case

UPPSALA
UNIVERSITET

- UPPMAX requirements:
  - 4 PiB usable
  - Target Lustre bandwidth
    - 24 GB/s = 22,35 GiB/s read/write minimum from clients
  - Isolation for up to 200 tenants
    - minimum 50 in parallel
    - heterogeneous bandwidth usage
  - No strong authentication available

- UPPMAX workflow
  - OpenStack environment
    - login & compute nodes dynamic instantiation
  - Ethernet network

Uppsala real-life use case: solution based on Lustre 2.10

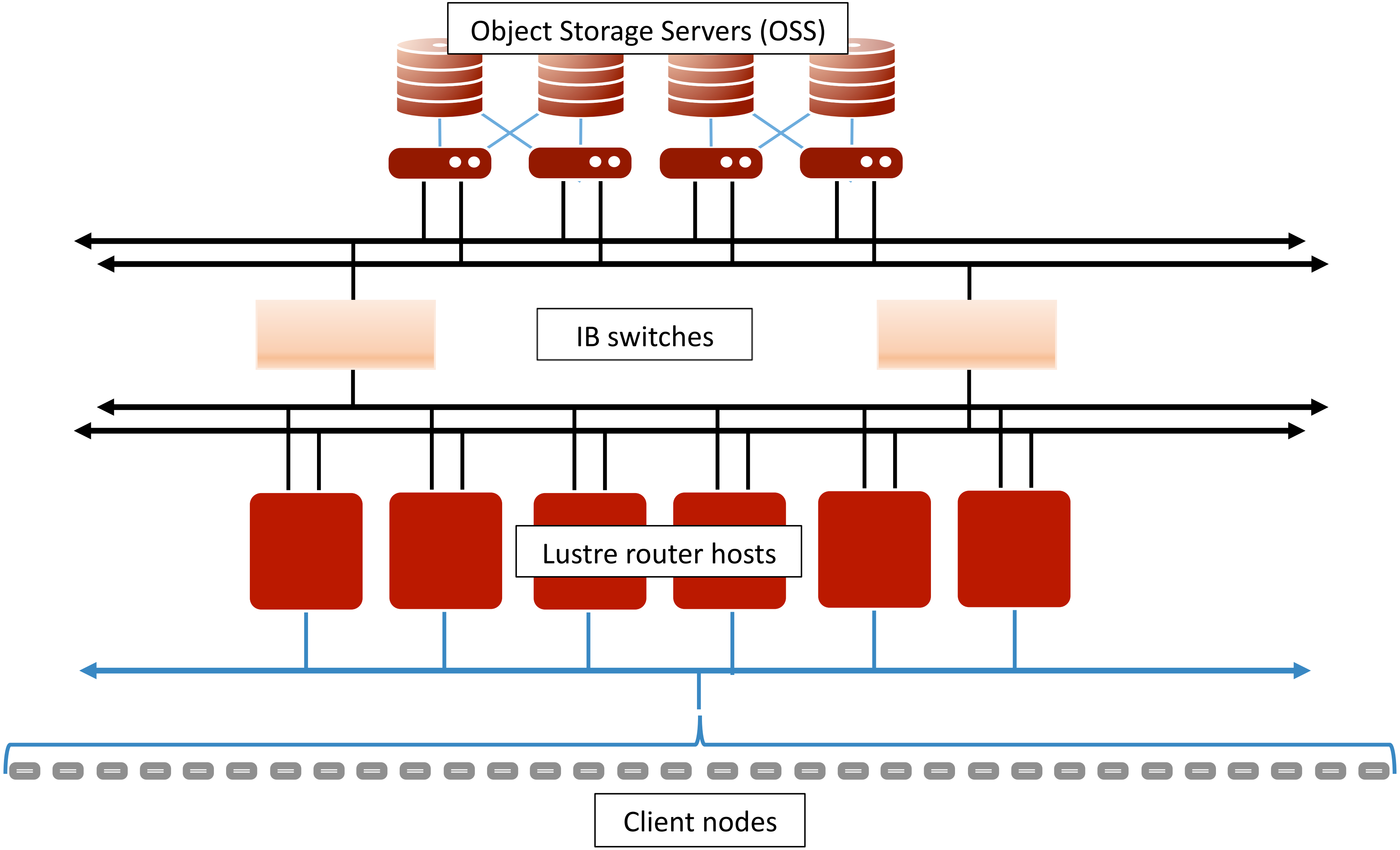


© 2018 DDN Storage

Uppsala real-life use case

- Idea to achieve multi-tenancy: LNet routers
  - 1 tenant == 1 LNet network
    - 1 LNet == 1 nodemap entry
    - 1 LNet == 1 routing rule to reach servers from Eth to IB
- But users can be root inside OpenStack VMs
  - To prevent tenant impersonation (”NID spoofing”):
    - tenant A == VLAN A in OpenStack
    - router A == Tag A on network interface
  - Enhanced workflow
    - Instantiate vRouters along with compute nodes

Uppsala real-life use case: routing + multi-rail



Object Storage Servers (OSS)

IB switches

Lustre router hosts

Client nodes

Uppsala real-life use case: routing + multi-rail

▶ With Lustre 2.10, use with caution:

- LNet routing problem on TCP: LU-10707
  - Workaround: **`options ksocklnd peer_timeout=0`**

- Routing + multirail corner case in 2.10:
  - No automatic peer discovery
  - Need to declare peers beyond routers

© 2018 DDN Storage

Uppsala real-life use case: performance evaluation

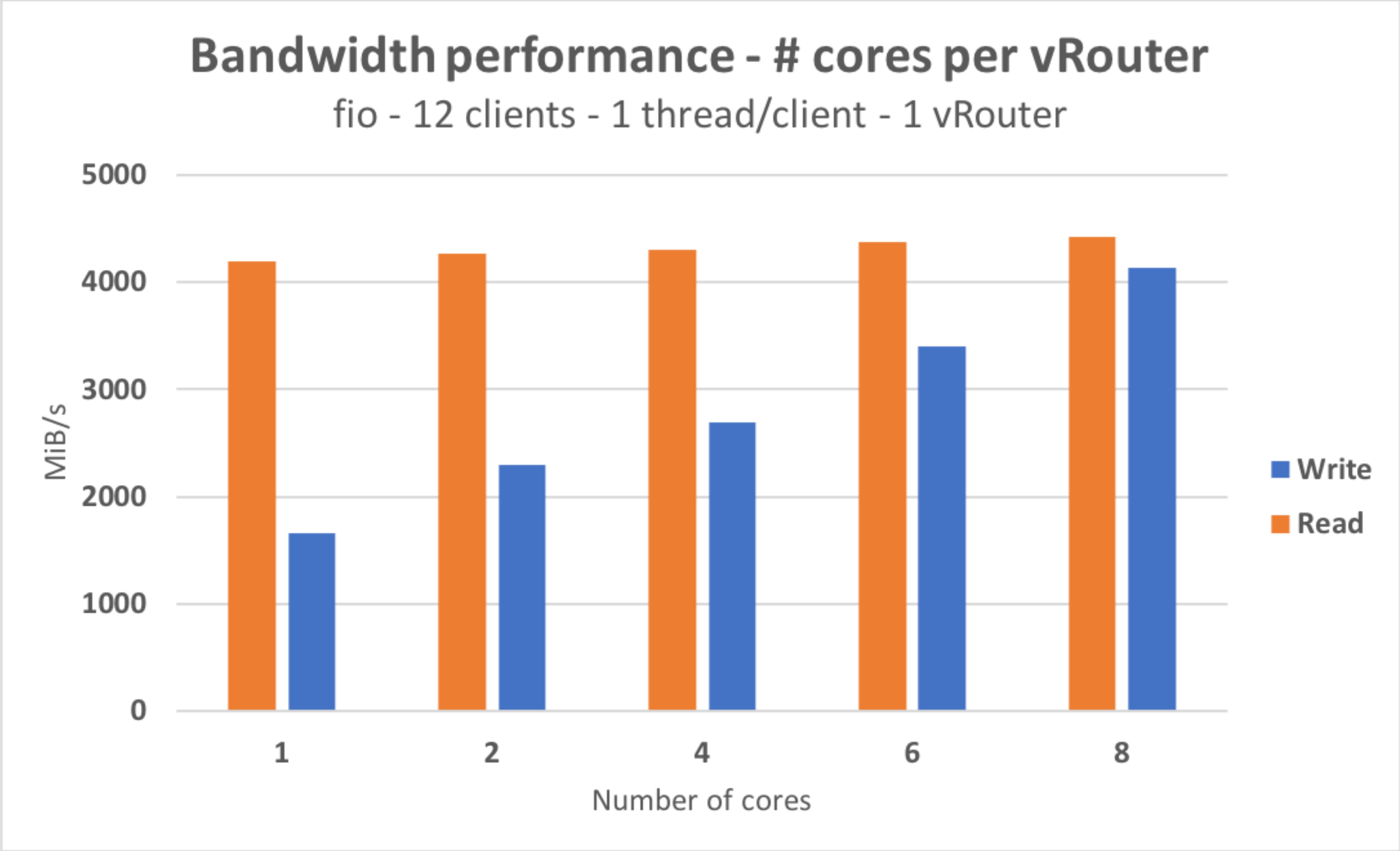
Bandwidth	Write GiB/s	Read GiB/s
Raw storage: 72 pools, RAID 6 8+2	43,5	63,4
↓	*~ 10 %*	*~ 30 %*
Obdfilter-survey: 72 OSTs	38,6	41,8
↓	*~ 20 %*	*~ 20 %*
fio from Lustre clients, no routing	31,1	33,8
↓	*~ 15 %*	*~ 20 %*
fio from Lustre clients, through routers*	26,7	26,3
↓	*~ 0 %*	*~ 0 %*
fio from Lustre clients, through routers and nodemap enabled	26,6	26,3

Requirement:

22+ GiB/s

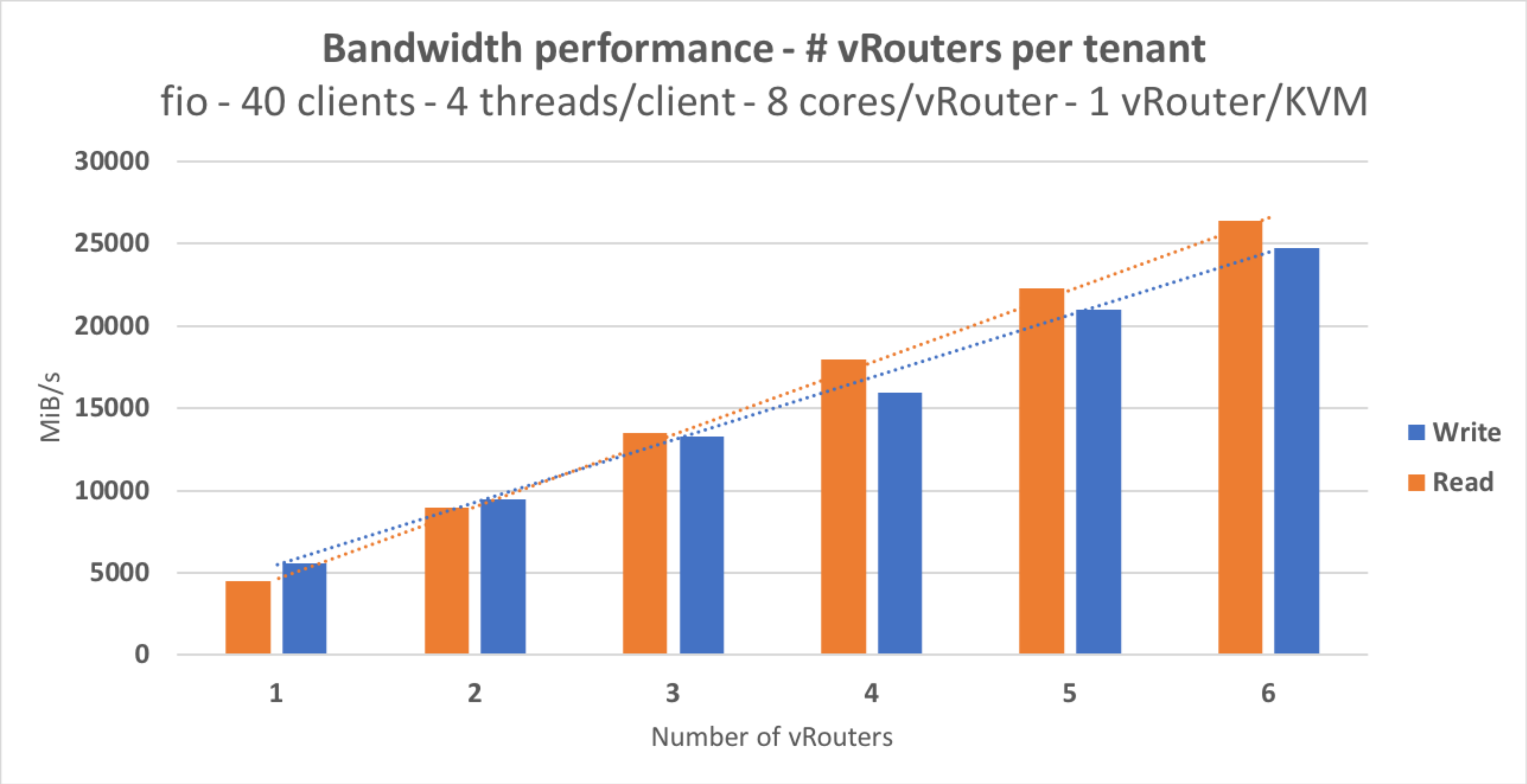
* Bottleneck is KVM hosts bandwidth on Eth network: 6 x 40 Gb/s ≈ 28 GiB/s

Uppsala real-life use case: throughput evaluation



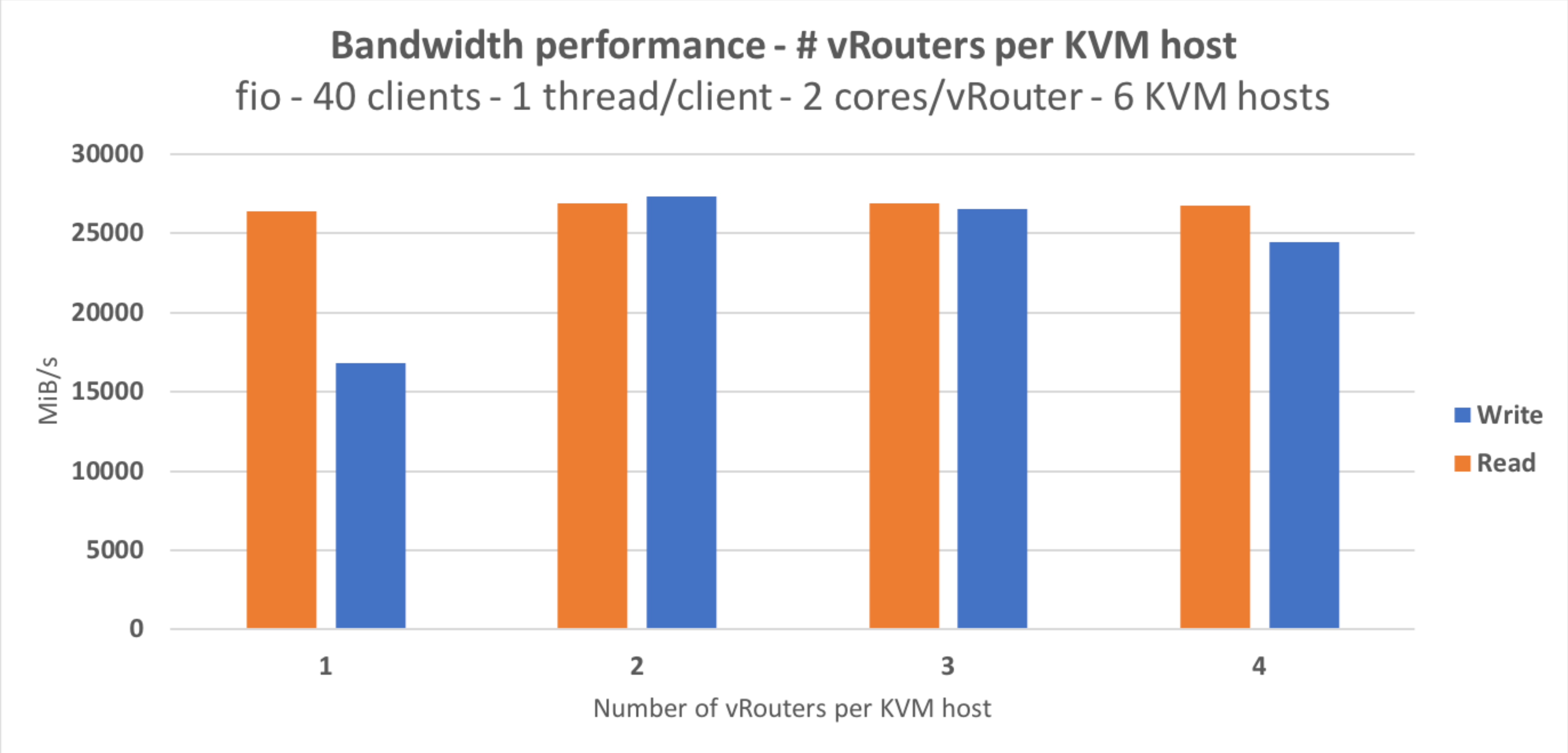
© 2018 DDN Storage

Uppsala real-life use case: performance evaluation



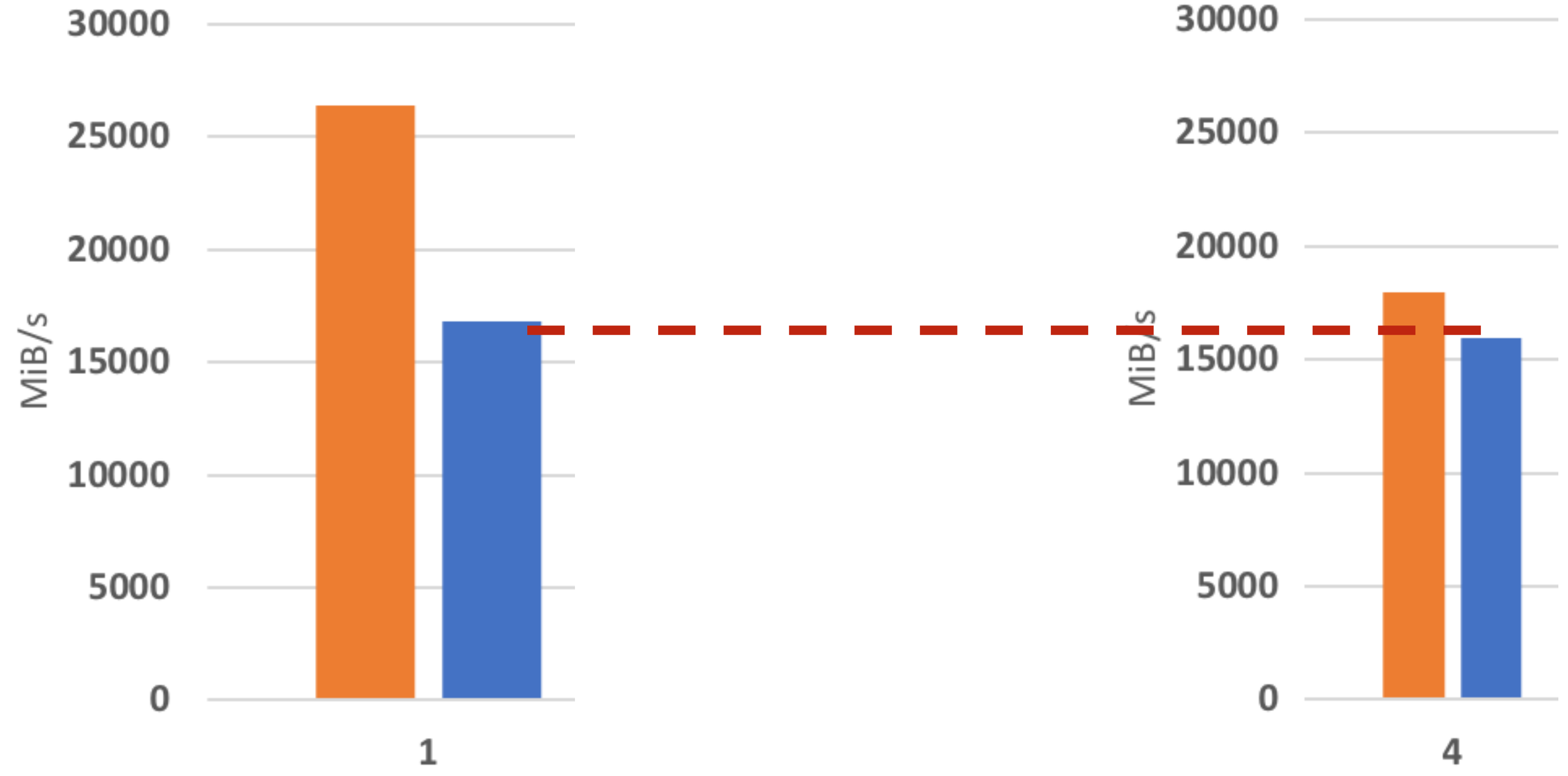
© 2018 DDN Storage

Uppsala real-life use case: performance evaluation



© 2018 DDN Storage

Uppsala real-life use case: vRouter sizing rationale



2-core vRouters, 1 per KVM host

⇒ 12 cores

8-core vRouters, on 4 KVM hosts

⇒ 32 cores

© 2018 DDN Storage

Uppsala real-life use case: performance evaluation

Metadata	create op/s	stat op/s	remove op/s
Raw storage: 4 LUNs, RAID 10 SAS drives	N/A	N/A	N/A
⇣			
mds-survey: 4 MDTs	N/A	N/A	N/A
⇣			
mdtest from Lustre clients, no routing	62100	277800	149200
⇣	*~ 30 %*	*~ 25 %*	*~ 25 %*
mdtest from Lustre clients, through routers*	42700	202900	111300
⇣	*~ 0 %*	*~ 1 %*	*~ 0 %*
mdtest from Lustre clients, through routers and nodemap enabled	42800	201000	111800

* IB-TCP routing adds latency, negatively impacting metadata performance.

Uppsala real-life use case: performance evaluation

- ► Choice: only 2-core vRouters == smaller, more numerous
  - Better request parallelization
  - Better flexibility
  - More tenants in parallel

- ► Resources available
  - 13 vRouters per KVM server (28 cores in total, 2 cores left for hypervisor)
  - 78 vRouters in total
  - Depending on bandwidth needs
    - 1 or several vRouters per tenant, on multiple KVM hosts

Conclusion

- We are able to provide isolation feature for Lustre

- By enforcing security thanks to a combination of:
  - Virtualized LNet routers
  - VLANs
  - Subdirectory mount
  - Nodemap
  - UID/GID mapping

© 2018 DDN Storage

Conclusion

- Happy with all the new technologies employed:
  - Lustre 2.10
  - Multi-Rail
- And with previously released features as well:
  - LNet routers
  - Subdirectory mount
  - UID/GID mapping
- Use more features in the future
  - Project Quota

Thank You!

Keep in touch with us.

sales@ddn.com

@ddn_limitless

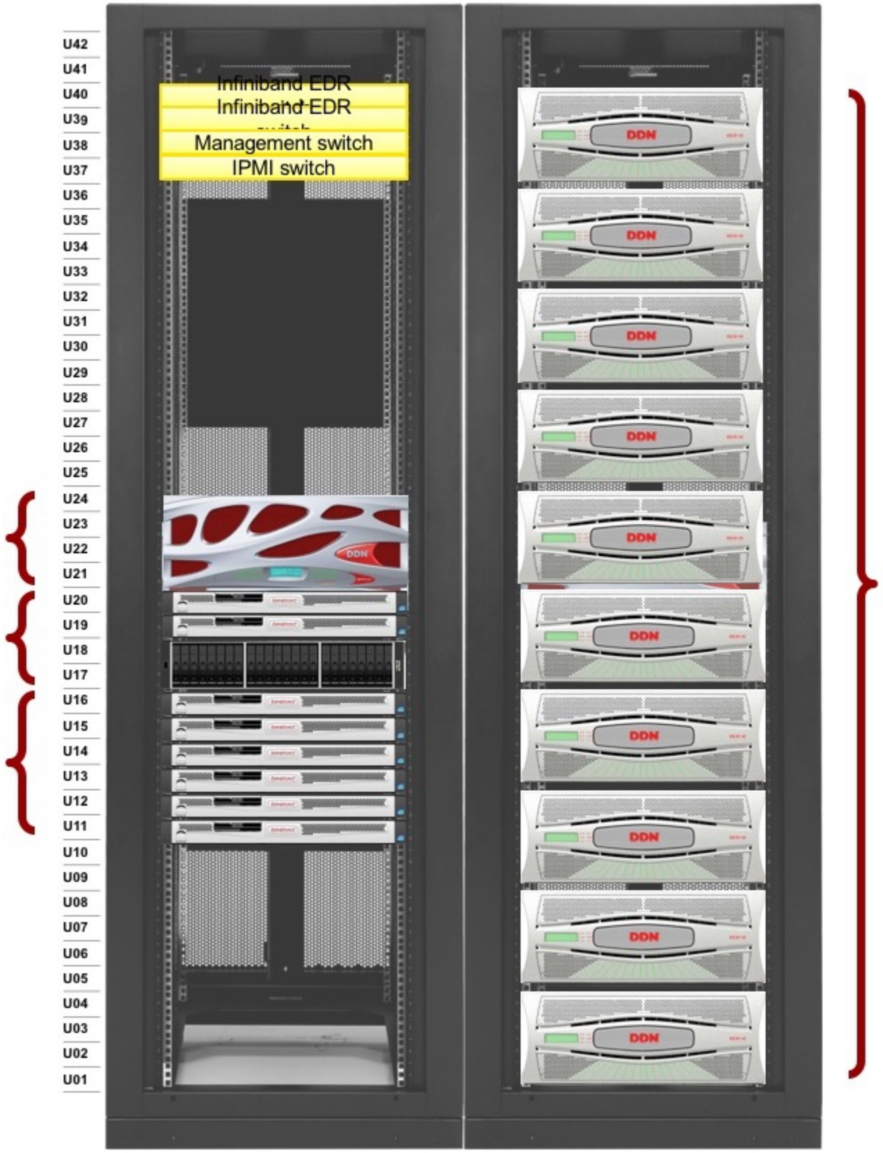
company/datadirect-networks

9351 Deering Avenue
Chatsworth, CA 91311

1.800.837.2298
1.818.700.4000

©2018 DataDirect Networks, Inc. *Other names and brands may be claimed as the property of others. Any statements or representations around future events are subject to change.

Uppsala Secure Lustre architecture: storage



ES14KX Lustre Appliance
- 4 x virtual Object Storage Servers
- 8 x EDR host ports

2 x Metadata Servers, each
- Dual 2.4GHz 10-core, 256GB
- 2 x EDR host ports

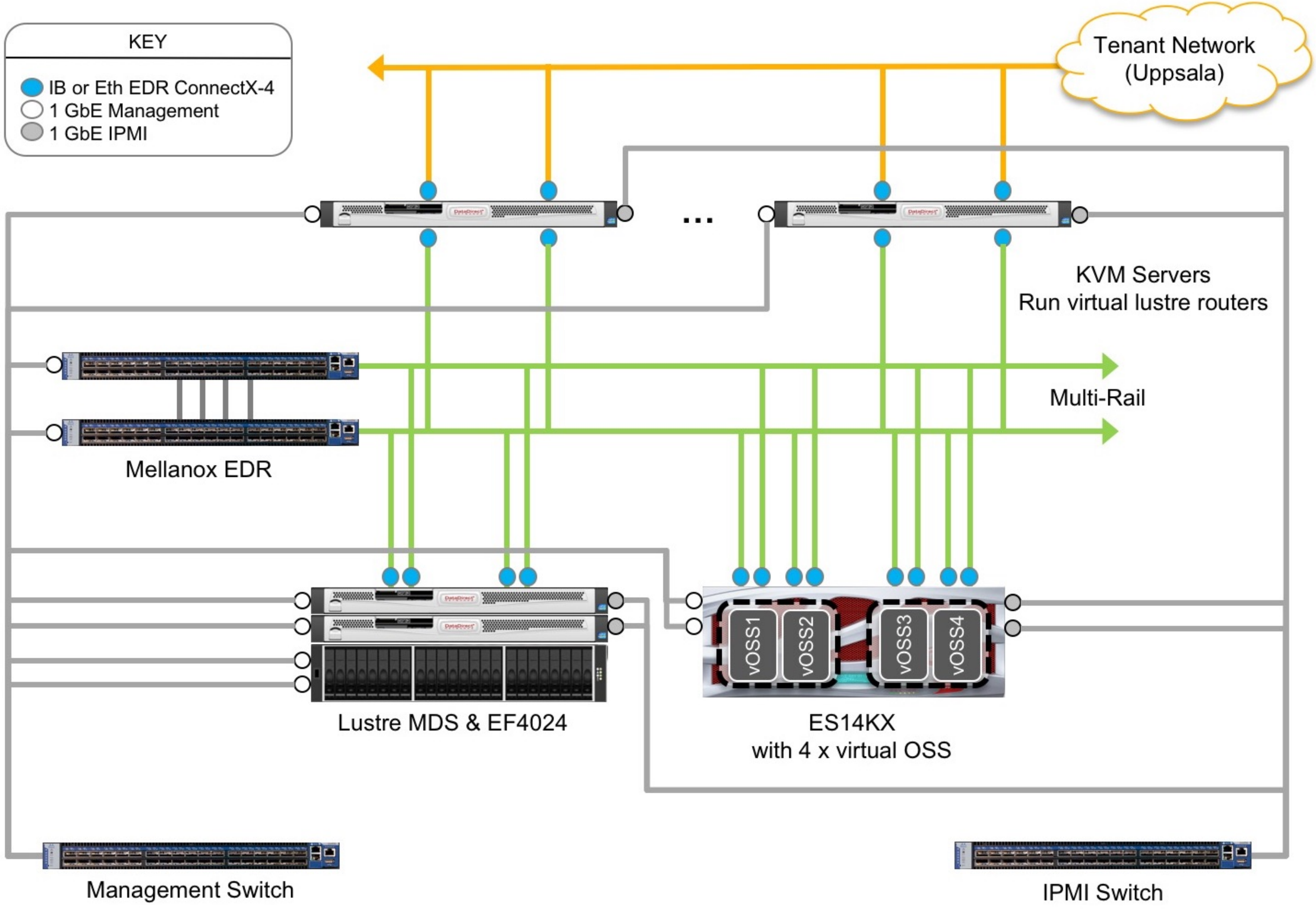
1x EF4024 Metadata Storage
- 12 x 600GB SAS

6 x KVM Servers
- Hosts virtual LNET Routers
- Dual 2.4GHz 14-core, 256GB

10 x SS8462 Disk Enclosures
- 720 x 8TB NL-SAS
- 4 PiB usable (5.7 PB raw)

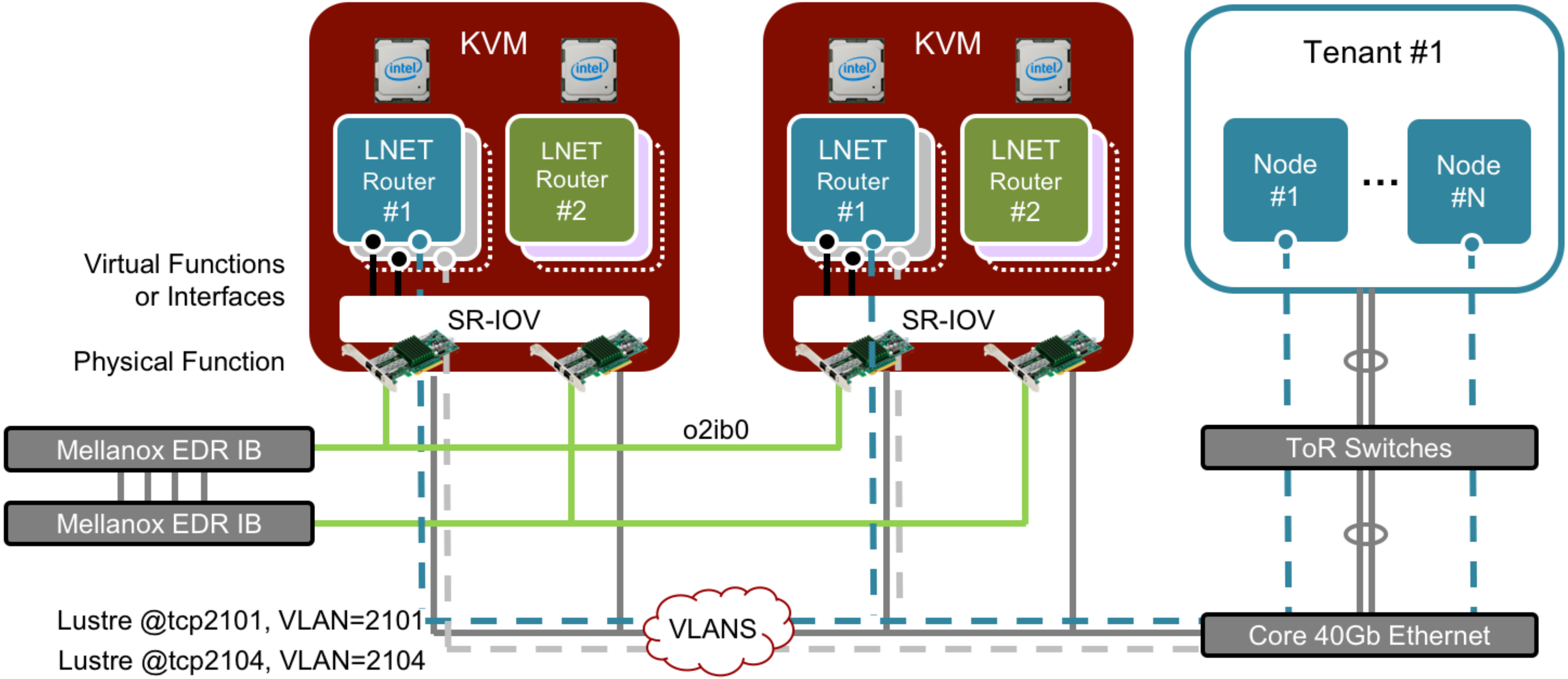
© 2018 DDN Storage

Uppsala Secure Lustre architecture: network



© 2018 DDN Storage

Uppsala Secure Lustre architecture: router



© 2018 DDN Storage