Feedback on BeeGFS

A Parallel File System for High Performance Computing

Philippe Dos Santos et Georges Raseev

FR 2764
Fédération de Recherche LUmière MATière

December 13 2016

Four laboratories in Université Paris Saclay

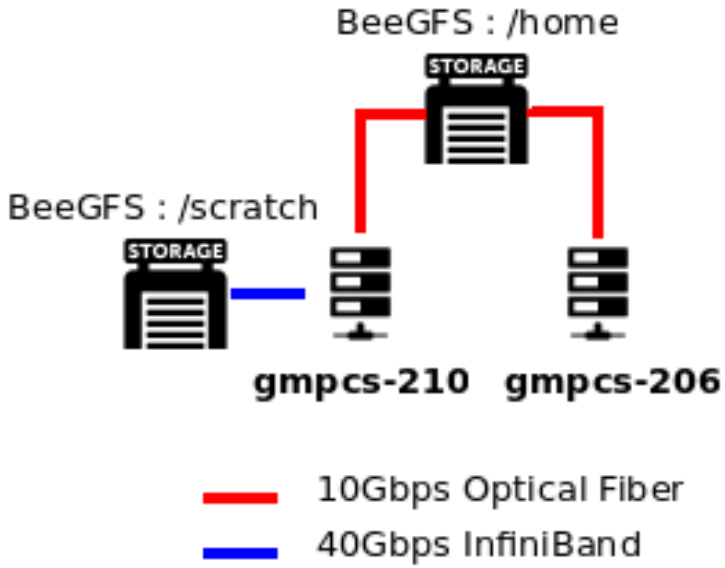
(Laser electron interaction : molecular physics, surfaces and nanophysics. Chemistry and biology interface)

CEMOX GMPCS CLUPS DTPI

- Several shared plateforms
  - experiments
  - HPC cluster
- Computing
  - workstations spread in labs
  - national computing centers
- Shared cluster
  - "Mésocentre"
  - Working since 2008 : 1 Tflops
  - Two branches in 2016 : 15 Tflops

Service continuity

- Two branches (and more ?)
  - gmpcs-210 et gmpcs-206
    (Datacenters in ISMO, VirtualData et DI)
  - Continuously available
    (Maintenance, power cut, ...)
  - Single point of access
    (Virtual IP + Keepalived)
  - Job distribution
    (Slurm + Multi Cluster Operation)
- BeeGFS storage
  - 1x common BeeGFS storage
    (/home - 6 To - user files and programs)
  - 1x BeeGFS in gmpcs-210
    (/scratch - 40 To - temporary big files)



Educated guess : hardware is the limiting factor

- Standard cluster stalled from time to time
  - Related to the master
  - Only happened when too many IO on /home
  - /home NFS was shared among computing nodes
- Disk IO related ?
  - Read / Write Speed
  - SATA
  - HDD (Hard Disk Drive) < 200 MB/s
  - SSD (Solid State Drive) < 550 MB/s
- Network related ?
  - Latency and Bandwidth
  - Ethernet 1Gb/s limiting ?
  - Switch to InfiniBand 40Gb/s ?

In real life : the file system is the limiting factor

- Network File System
  - Slows down when multiple clients read/write simultaneouly
  - Data is located on one data server
- Parallel file system
  - metadata server(s) for data placement
  - data server(s) for the data storage
  - allows many IO from many clients
- Improved throughput with same hardware
  - From NFS to BeeGFS
  - From 3.2Gbps to 9.6Gbps (x3 !)

IO on the shared file system = bottleneck

- NFS known limitations
  - NFS (IP) over Ethernet (1Gbps)
    11/2008 : 800Mbps max throughpupt / whole cluster slowdown
    Solution : File staging on compute nodes for IO / MPI ?
  - NFS (IP over InfiniBand - IPoIB) QDR interconnect (40Gbps)
    08/2013 : 3.2Gbps max throughput (x4) / cluster ok
    Jobs are slow when lot of simultaneous IO
- Which parallel file system ?
  - Production ready
  - Good use of network bandwidth
  - Easy to manage
  - Price quality ratio (low entry price)
- BeeGFS parallel filesystem
  - For a start 6TB (expandable to 40TB)
  - Standard cluster = 9.6 Gbps with InfiniBand (40Gbps RDMA)
  - Two branches = 2.4 Gbps with Optical Fiber (10Gbps IP)

Best known parallel filesystems and Top 500

- Best known parallel filesystems for HPC
  - Lustre - Open source
    (50% Top 500 : Titan #2, Sequoia #3, K Computer #4, CEA/TGCC-GENCI #44, ...)
  - GPFS - IBM license
    (50% Top 500 : Mira #5, CNRS/IDRIS-GENCI #60, ...)
  - PanFS - Panasas license
    (Cielo #57)
- Top 500 (top500.org) du 26/07/2015
  - Top 500 supercomputers worldwide
  - BeeGFS in use on 6 supercomputers
  - Mainly used in german speaking countries
- Same concepts
  - IO in parallel on multiple disks
  - Metadata server(s)
  - Data servers

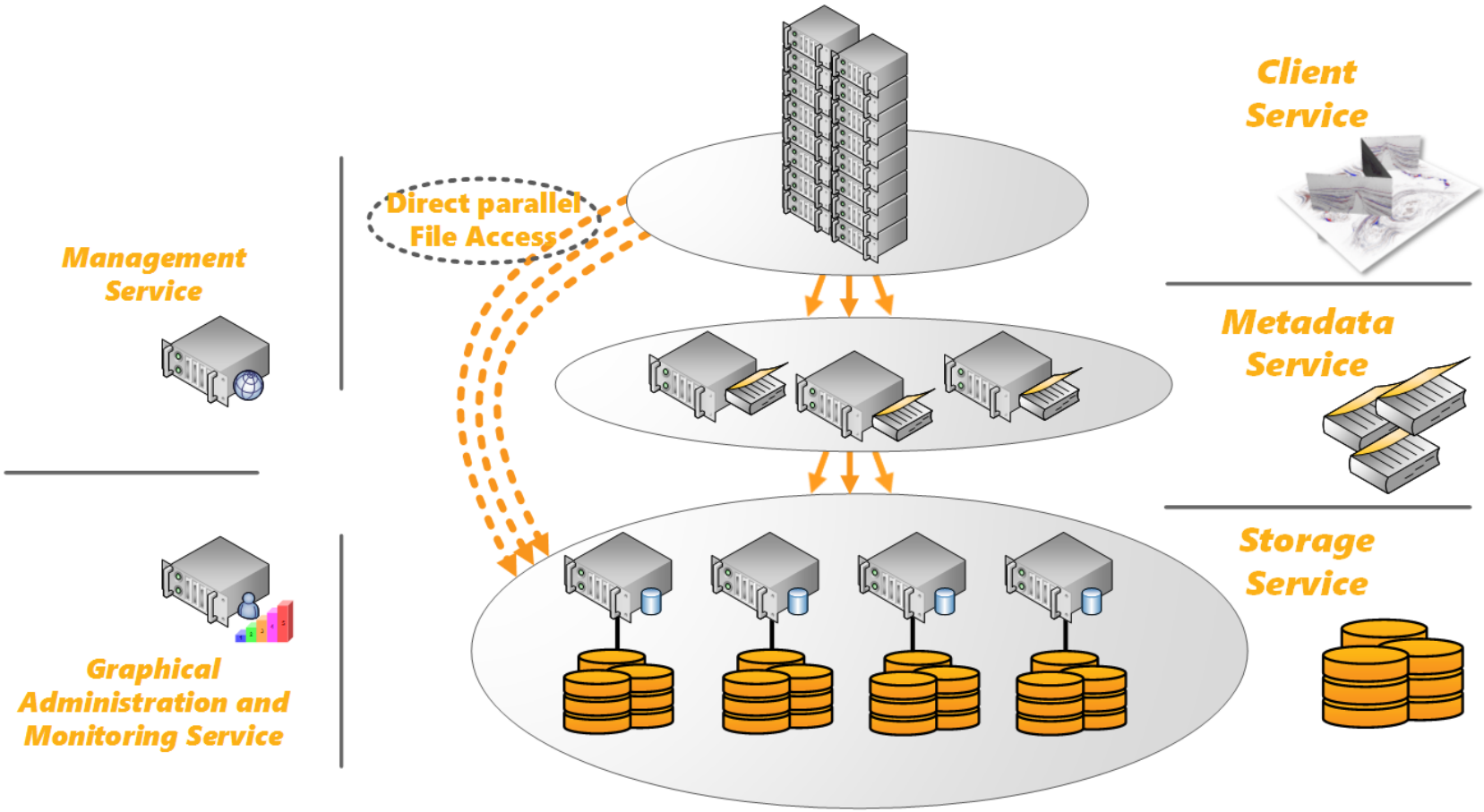
BeeGFS in a few words

- BeeGFS has two main roles
  - To organize namespace of the files
  - To store attributes of the files and their content
- To organize with metadata server(s)
  - Data position on disks
  - File size
  - Owners and permissions
  - ...
- To store file content with storage server(s)
  - What users are interested in
  - The content of the file
- Who's in charge of data integrity ?
  - File cut into pieces (chunks or stripes)
  - Each metadata and data server manages several drives
  - RAID ensures data integrity in case of a drive failure

Four main services for management of metadata, data and client

- GNU/Linux only
- On top of existing filesystem : ext4, xfs, zfs, ...
- Management Server (MS)
  - Knows every service (metadata, data, client)
  - Not critical
- MetaData Server (MDS)
  - Metadata management
  - One MDS has only one MetaData Target (MDT)
- Object Storage Server (OSS)
  - Data management
  - One OSS may have many ObjectStorage Targets (OST)
- Client : compile and loads the **beegfs** kernel module

Four main services for management of metadata, data and client



BeeGFS courtesy

Scalability

- Metadata Target (MDT) performance
  - One directory per MDS
    (MDS randomly chosen => load shared on all MDS)
  - Dedicated SSD disks on RAID1 / RAID10
    (Mirroring or striping + mirroring / since random access, avoid RAID5 and RAID6)
  - ext4 filesystem
    (Efficient with small files)
  - Large inodes stores metadata
    (ext4 inode extended attribute = 512 bytes)
- Object Storage Server (OSS)
  - Striping : numtargets + chunksize
    (How many OST + file size chunk)
  - 1x OST 40TB at 500MB/s => 4x OST 160TB at 2GB/s
    (More OST => more storage space and more throughput)
  - Typical 6 to 12 HDD on RAID6
    (RAID 6 ensures data integrity)
  - On top of existing filesystem : xfs, ext4, zfs, ...

All-in-one BeeGFS node (mgmtd, meta and storage)

1x Application server

1x JBOD

- CPU : 16 cores at 2.4GHz
  (2x Intel Xeon E5-2630 v3)
- RAM : 64GB
  (8x 8Go DDR4 at 2133MHz)
- Metadata : 4x SSD 200GB
  (1x MDT / RAID10 => 400GB)
  (Rule = 0.5% of storage space)
  (Metadata on a dedicated RAID controller)
- Data : 12x HDD 4TB
  (1x OST / RAID6 => 40TB)
  (Data on a dedicated RAID controller)
- Infiniband QDR (40 Gbit/s)
  (Intel True Scale HCA, 1x QSFP port)
- 1Gbps Ethernet NIC
  (Intel I350 on motherboard)

Benchmark using 8 compute nodes

2x Twinsquare (8x nodes)

- CPU : 20 cores at 2.5GHz
  (2x Intel Xeon E5-2670 v2)
- RAM : 128GB
  (8x 16Go DDR3 at 1866MHz)
- HDD : 2TB
  (1x per node, 7200 RPM, SATA-3, 3.5")
- Infiniband QDR (40 Gbit/s)
  (Intel/QLogic QLE7340, 1x QSFP port)
- 1Gbps Ethernet NIC
  (Intel I350 on motherboard)

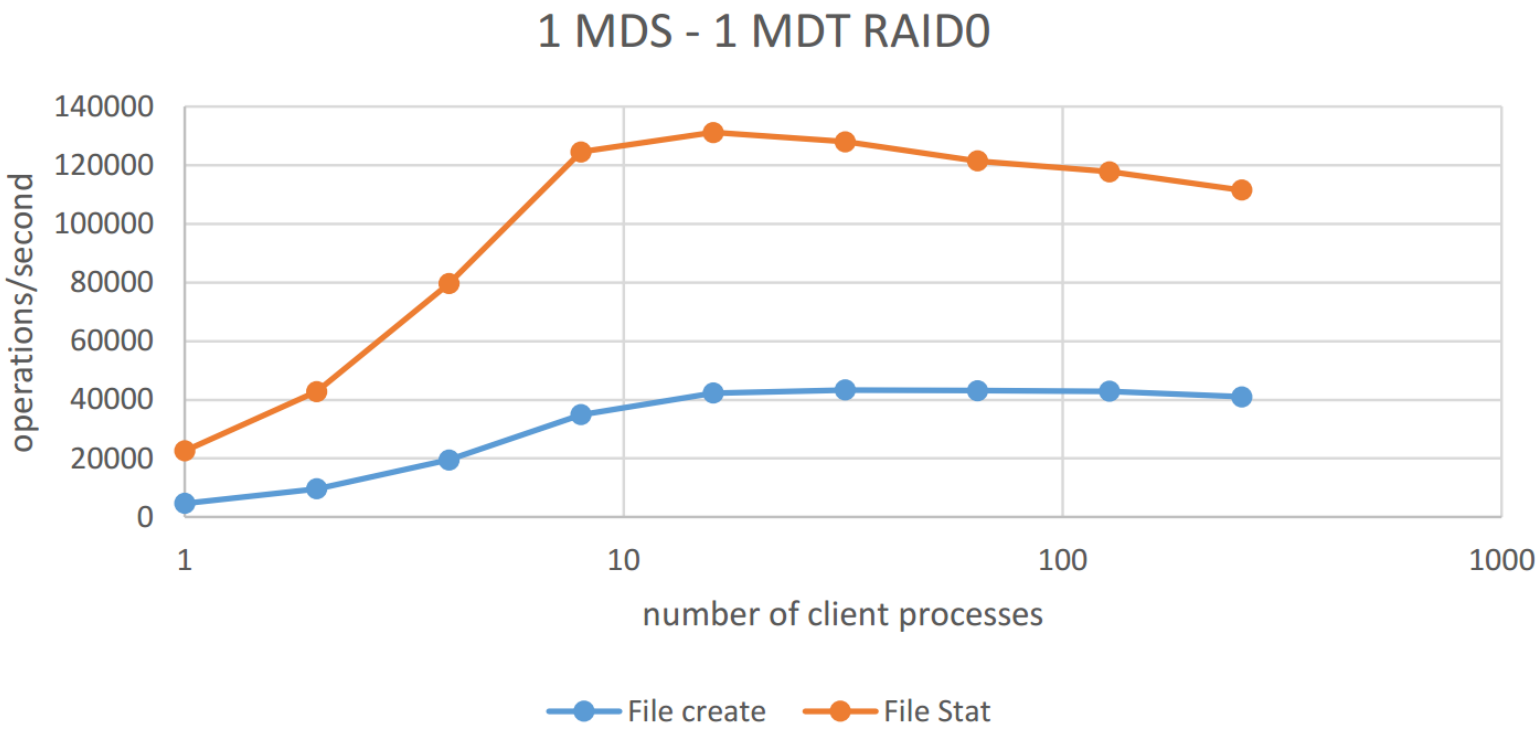
How to measure metadata performance ?

- Open source mdtest tool
  (http ://sourceforge.net/projects/mdtest/)
- Performs open/stat/close operations on files and directories
  (MPI coordinated)
- Progressive increase of IOs
- Creates directory tree with files
  (Reports number of IOPS)
- Metadata Performance Evaluation of BeeGFS
  (http ://www.beegfs.com/docs/Metadata_Performance_Evaluation_of_BeeGFS_byThinkParQ.pdf)

All-in-one BeeGFS node benchmark

Metadata performance

All-in-one BeeGFS node metadata performance



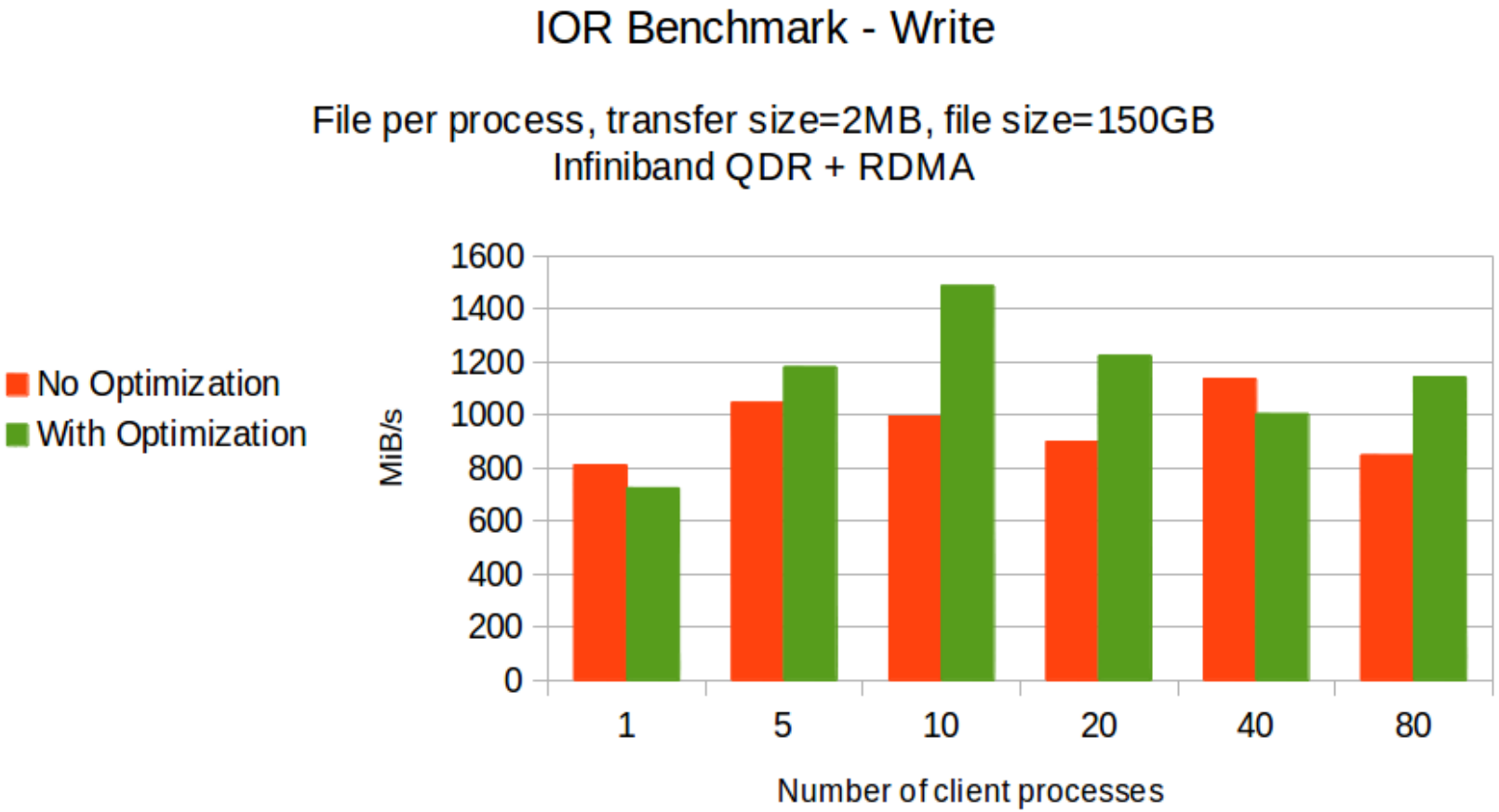
Metadata Performance Evaluation of BeeGFS
(http ://www.beegfs.com/docs/Metadata_Performance_Evaluation_of_BeeGFS_byThinkParQ.pdf)

How to measure data performance ?

- Open source IOR tool
  (http ://sourceforge.net/projects/ior-sio/)
- This parallel program performs writes and reads
  (MPI coordinated)
- Progressive increase of IOs
- Measure parallel file system I/O performance
  (Reports throughput at both the POSIX ans MPI-IO level)
- Picking the right number of targets per storage server for BeeGFS
  (http ://www.beegfs.com/docs/Picking_the_right_Number_of_Targets_per_Server_for_BeeGFS_by_ThinkParQ.pdf)

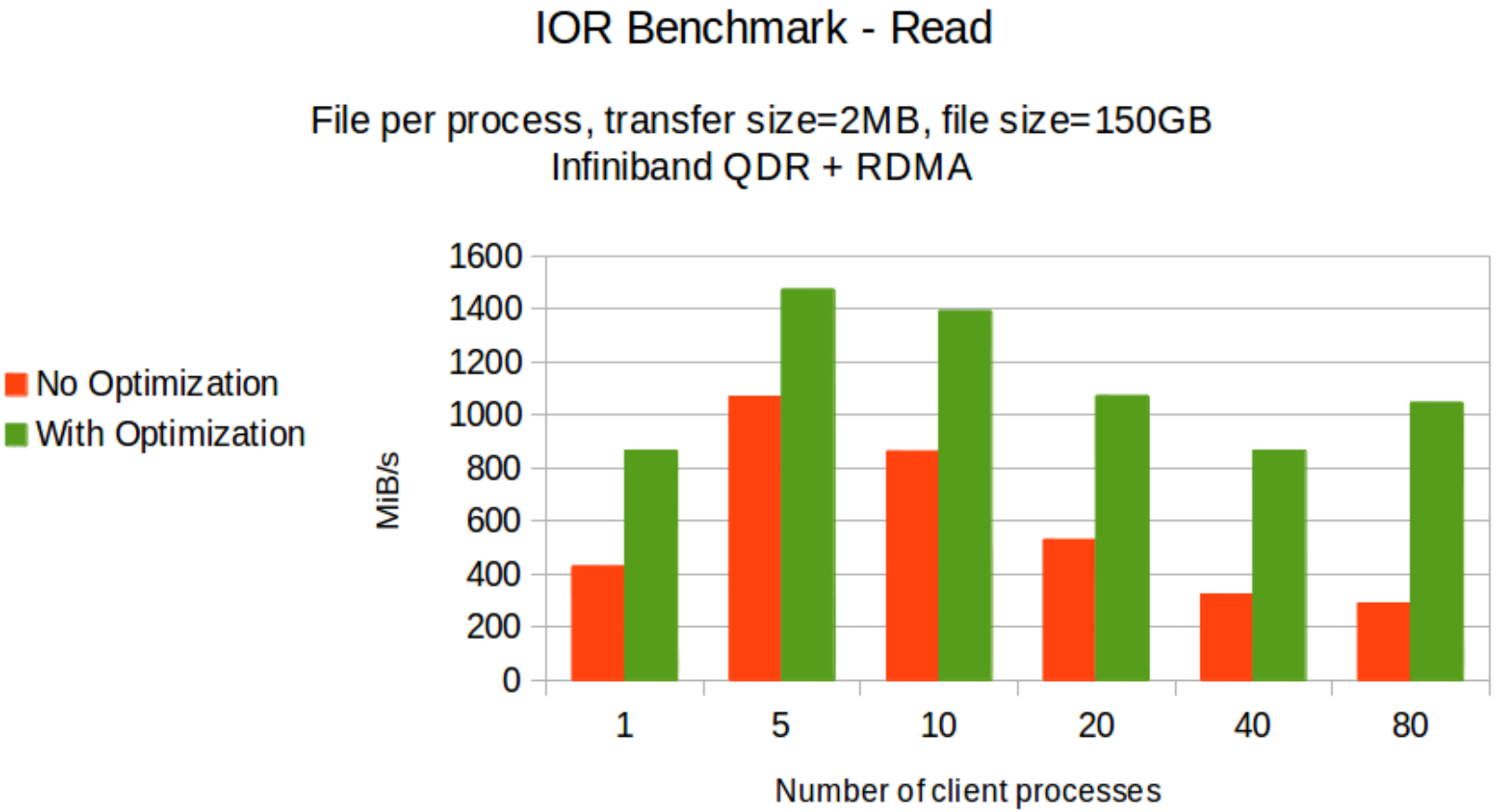
Tuning is the key

(Formatting and mounting options, partition alignement, sysctl, pinning BeeGFS processes, ...)



Tuning is the key

(Formatting and mounting options, partition alignement, sysctl, pinning BeeGFS processes, ...)

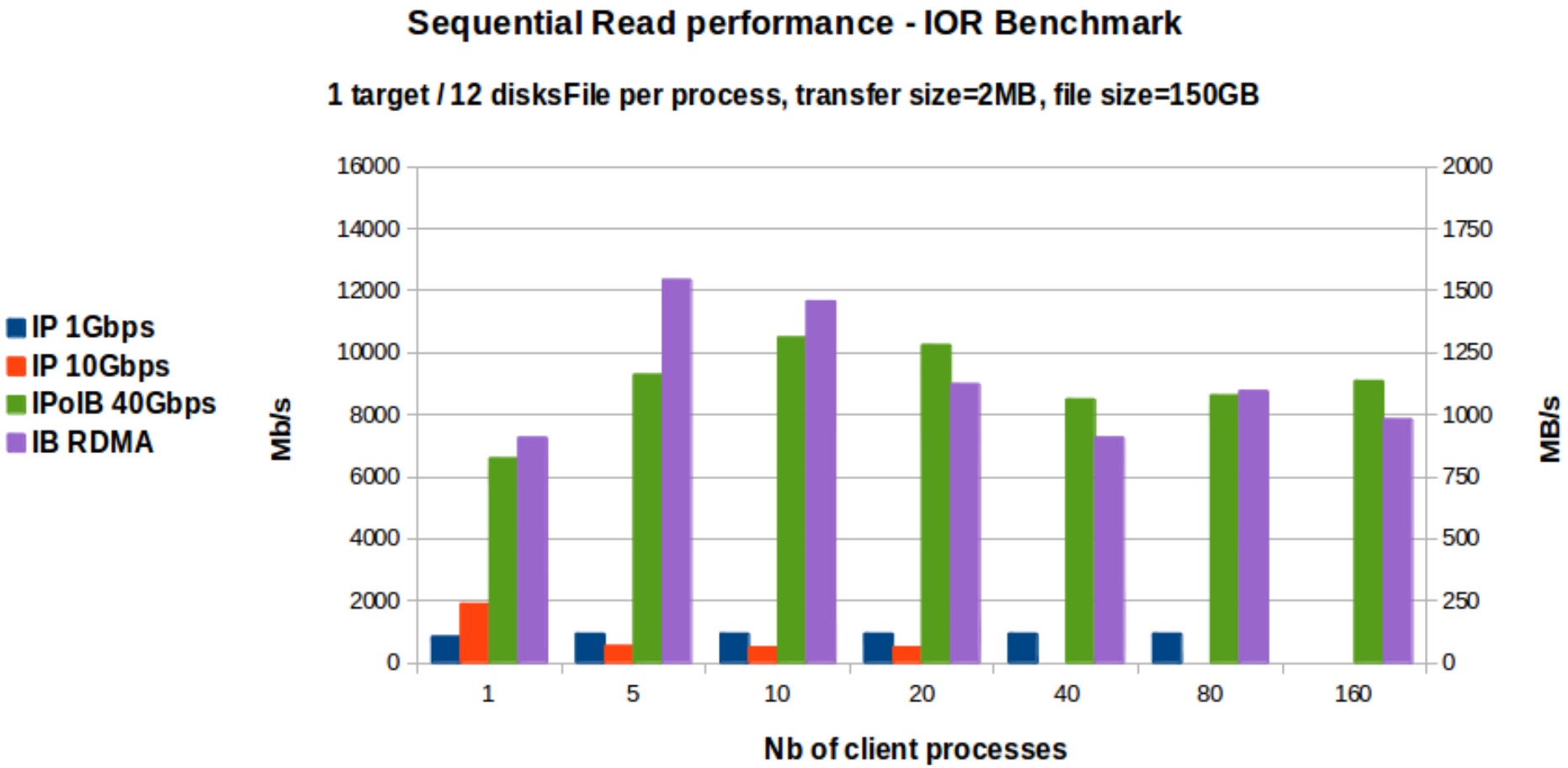


Best to have high bandwidth network

1Gbps Ethernet vs 10Gbps Ethernet vs 40Gbps Infiniband (IPoIB vs RDMA)

Best to have high bandwidth network

1Gbps Ethernet vs 10Gbps Ethernet vs 40Gbps Infiniband (IPoIB vs RDMA)



All-in-one BeeGFS node in the end

Good working and scalable

1x Application server

1x JBOD

- Tune to achieve best performance
  (Know how users access data)
- Keep high level IO and throughput
  (Keep on adding compute nodes)
- 40TB scalable to 160 TB
  (Adding JBODs on existing application server)
- Throughput increase
  (Adding application servers and JBODs)
- Data close to compute nodes
  (Better to improve throughput)
- RAID6 against drive failure
  (But no data replication)

All-in-one BeeGFS node in the end

Throughput increase

Write throughput using all-in-one BeeGFS node

(8x RAID6 arrays with 6 disks each / 4x RAID6 arrays with 12 disks each / 1x RAID60 array with 48 drives)

Picking the right number of targets per storage server for BeeGFS

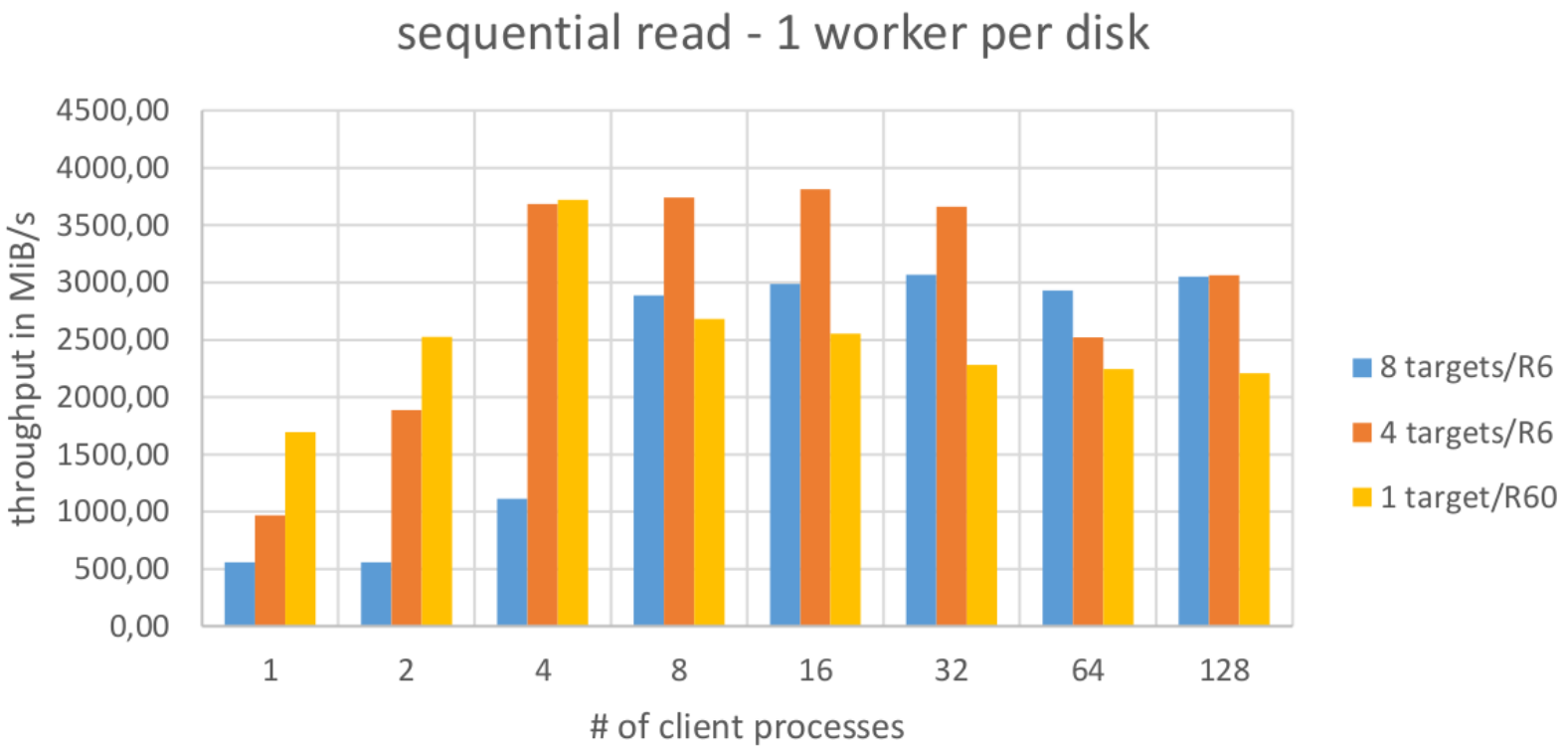
(http ://www.beegfs.com/docs/Picking_the_right_Number_of_Targets_per_Server_for_BeeGFS_by_ThinkParQ.pdf)

All-in-one BeeGFS node in the end

Throughput increase

Read throughput using all-in-one BeeGFS node

(8x RAID6 arrays with 6 disks each / 4x RAID6 arrays with 12 disks each / 1x RAID60 array with 48 drives)



Picking the right number of targets per storage server for BeeGFS

(http ://www.beegfs.com/docs/Picking_the_right_Number_of_Targets_per_Server_for_BeeGFS_by_ThinkParQ.pdf)

In conclusion

Production ready and easy to use

1x Application server

1x JBOD

- **All-in-one BeeGFS node**
  (Metadata and data managed on the same host)
- Avoid commodity hardware
  (2x RAID controllers, SSDs, high speed interconnect, ...)
- Choose class entreprise disks
  (12x 4TB HDD, 7200 RPM, Near Line SAS, 3.5")
- **Tune to achieve best performance**
  (Know how users access data)
- Good performance for the money
  (Low entry price for HPC world !)
- **Easy to administer**
  (Well suited for small IT teams)

Icon made by Freepik from www.flaticon.com