EVALUATION OF GLUSTER AT IHEP

CHENG Yaodong
CC/IHEP
2011-5-4

中国科学院高能物理研究所
Institute of High Energy Physics
Chinese Academy of Sciences

Motivation

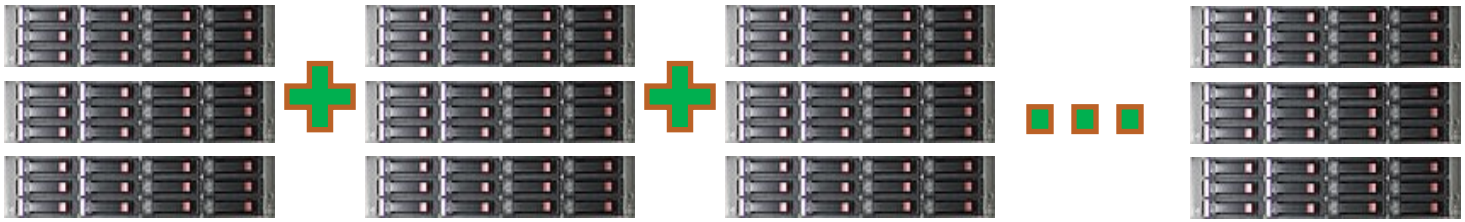
- Currently we deploy LUSTRE (1.7PB, version1.8.1.1) for local experiments, which present good performance, however also have disadvantages, for example:
  - No replication support
  - No data rebalance support
  - Bad metadata performance of a large number of small files
  - Difficult to remove and add OSTs elastically
  - Tightly coupled with Linux kernel, hard to debug
  - …
- Actually, it is difficult to find a file system to meet all our requirements, and we will continue to use Luster for a period of time in the future, but…
- It is necessary to check requirement lists and find potential alternatives

Checklist

- Performance
  - High aggregate bandwidth; reasonable I/O speed for each task
- Scalability
  - Near linear scalability in both capacity and performance
  - Scale to a large number of files (metadata scalability)
- Stability/robustness
  - Run unattended for long periods of time without operational intervention
- Reliability
  - Make data reliable in case of hardware/software/network failure
- Elasticity
  - Flexibly adapt to the growth (or reduction) of data and add or remove resources to a storage pool as needed, without disrupting the system
- Easy to use and maintenance
- Security
  - Authentication & authorization
- Monitoring and alerting

What is Gluster

- GlusterFS is a scalable open source clustered file system
  - It aggregates multiple storage bricks over Ethernet or Infiniband RDMA interconnect into one large parallel network file system
  - offers a global namespace
  - Focus on scalability, elasticity, reliability, performance, ease of use and manage, …
- Website: http://www.gluster.org/

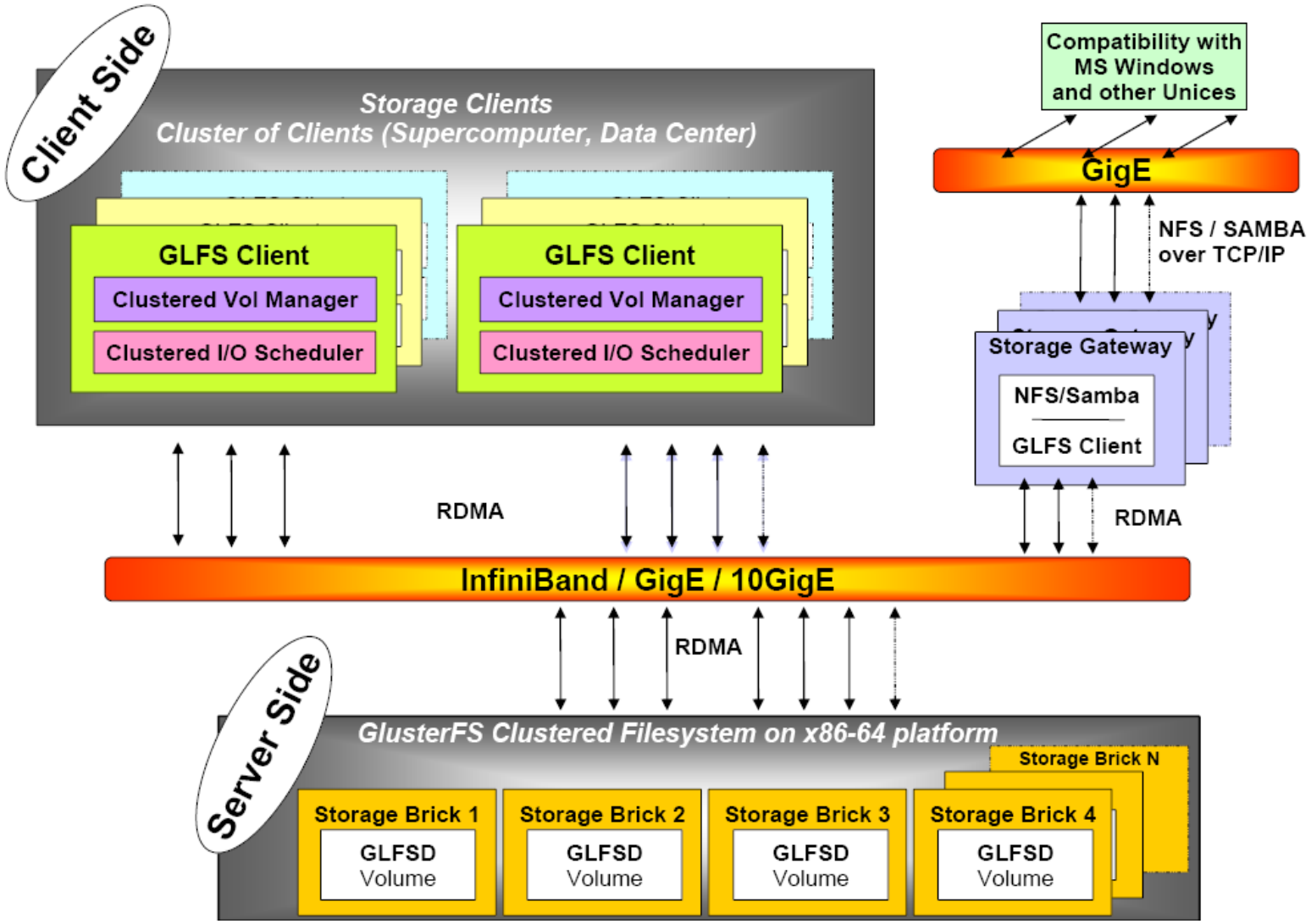


N × Performance & capacity

GLUSTER FEATURES

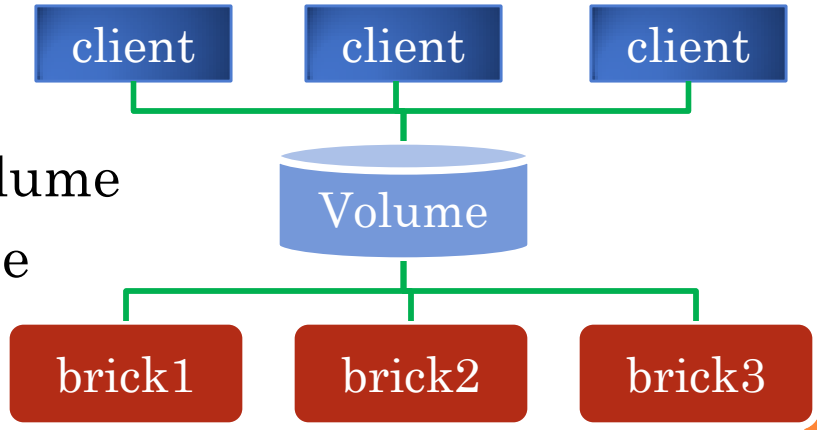
- More scalable, reliable
  - No Metadata server with elastic HASH Algorithm
- More flexible volume management (stackable features)
  - Elastic to add, replace or remove storage bricks
- Application specific scheduling / load balancing
  - round robin; adaptive least usage; non-uniform file access
- Automatic file replication, Snapshot, and Undelete
- Good performance
  - Striping
  - I/O accelerators - I/O threads, I/O cache, read ahead and write behind !
- Fuse-based client
  - Fully POSIX compliant
  - Unified VFS
- User space
  - Easy to install, update, or debug

GLUSTER ARCHITECTURE



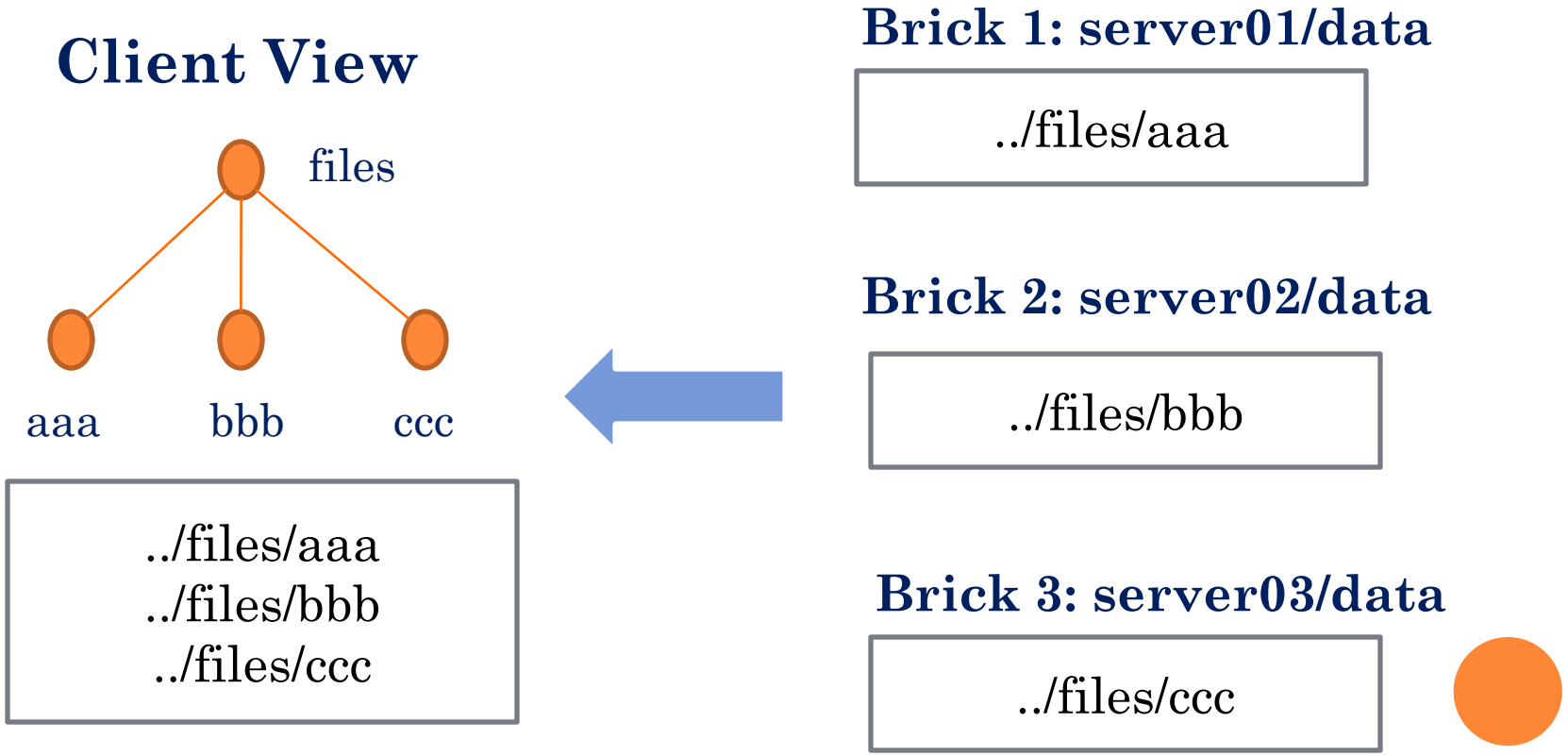
ELASTIC VOLUME MANAGEMENT

- Storage volumes are abstracted from the underlying hardware and can grow, shrink, or be migrated across physical systems as necessary.
- Storage servers can be added or removed on-the-fly with data automatically rebalanced across the cluster. Data is always online and there is no application downtime.
- Three types of volume
  - Distributed Volume
  - Distributed Replicated Volume
  - Distributed striped Volume



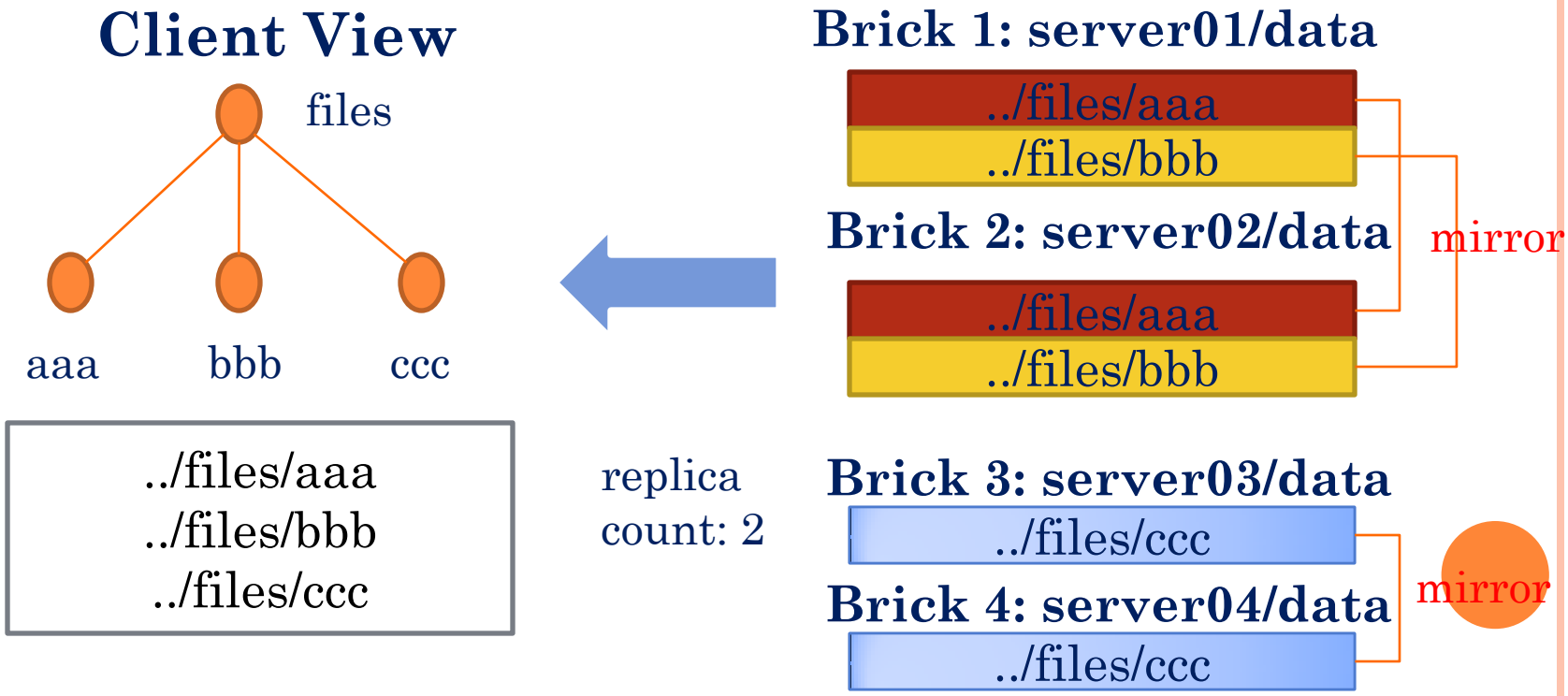
DISTRIBUTED VOLUME

- Files are assigned into different bricks in multiple servers
- No replication, No stripe, files are kept in original format



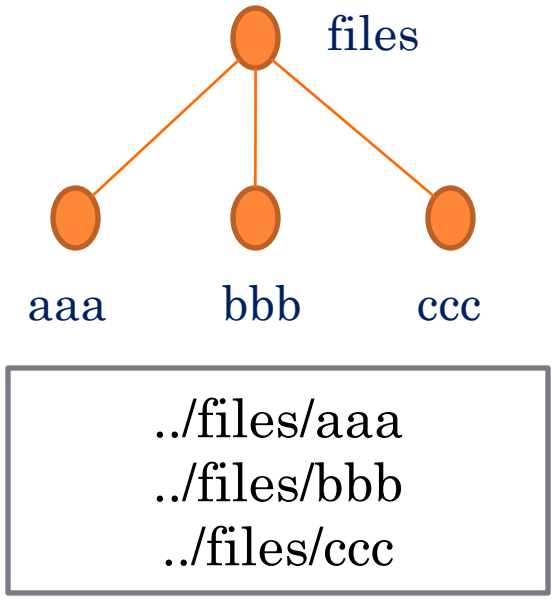
Distributed Replicated Volume

- Replicate (mirror) data across two or more nodes
- Used in environments where high-availability and high-reliability are critical
- Also offer improved read performance in most environments
- But two or more times of raw disk capacity are needed



Distributed Striped Volume

- Stripe data across two or more nodes in the cluster.
- Generally only used in high concurrency environments accessing very large files
- File will be inaccessible if one brick fails



Stripe count: 3

Brick 1: server01/data

../files/aaa
../files/bbb
../files/ccc

Brick 2: server02/data

../files/aaa
../files/bbb
../files/ccc

Brick 3: server03/data

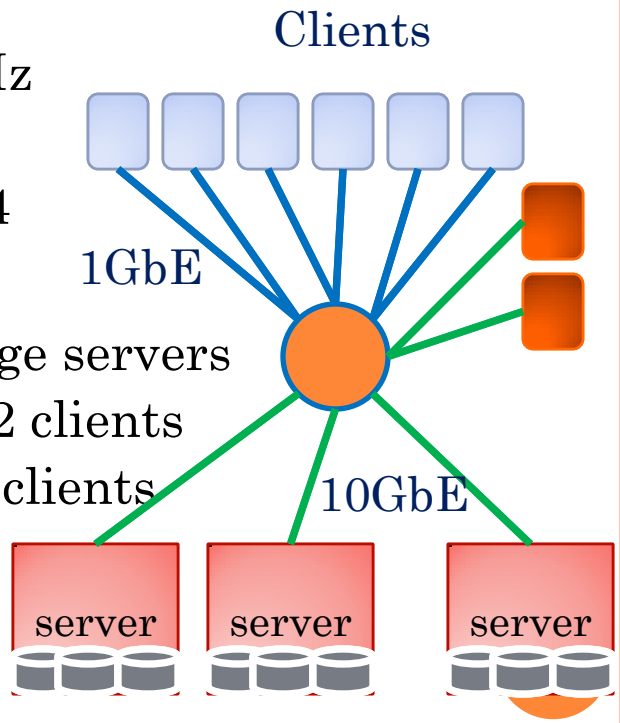
../files/aaa
../files/bbb
../files/ccc

Evaluation of Gluster

- Evaluation Environments
- Function Testing
  - Reliability
  - …
- Performance Testing
  - Single client R/W
  - Aggregate I/O
  - Stress testing
  - Metadata testing

Evaluation Environments

- Server configuration: three storage servers
  - Processor: 2 Intel XEON E5430 @ 2.66GHz
  - Memory: 24GB
  - DISK: each server attach 12 SATA disks
  - OS Kernel: SLC4 2.6.9-78.0.1.EL.cernsmp x86_64
- Client configuration: 8 clients
  - Processor: 2 Intel XEON E5430 @ 2.66GHz
  - Memory: 16GB
  - OS Kernel: SL55 2.6.18-194.3.1.el5 x86_64
- Network link
  - 10Gbps link interconnected between storage servers
  - 10Gbps link between storage servers and 2 clients
  - 1Gbps link between storage servers and 6 clients
- Software version
  - GlusterFS 3.1.2



THREE VOLUMES

- Distributed volume
  - Name: disvol
  - Bricks: server1:/data02/gdata01 (11TB)
    server2:/data05/gdata01 (2TB)
    server3:/data06/gdata01 (2TB)
  - Total capacity: 15TB
- Distributed replicated volume
  - Name: repvol
  - Bricks: server1:/data01/udata (863GB)
    server2:/data06/gdata01 (2TB)
  - Total capacity: **863**GB
- Distributed striped volume
  - Name: stripevol
  - Bricks: server1:/data08/udata (2TB)
    server2:/data08/udata (2TB)
  - Total capacity: 4TB

Reliability testing

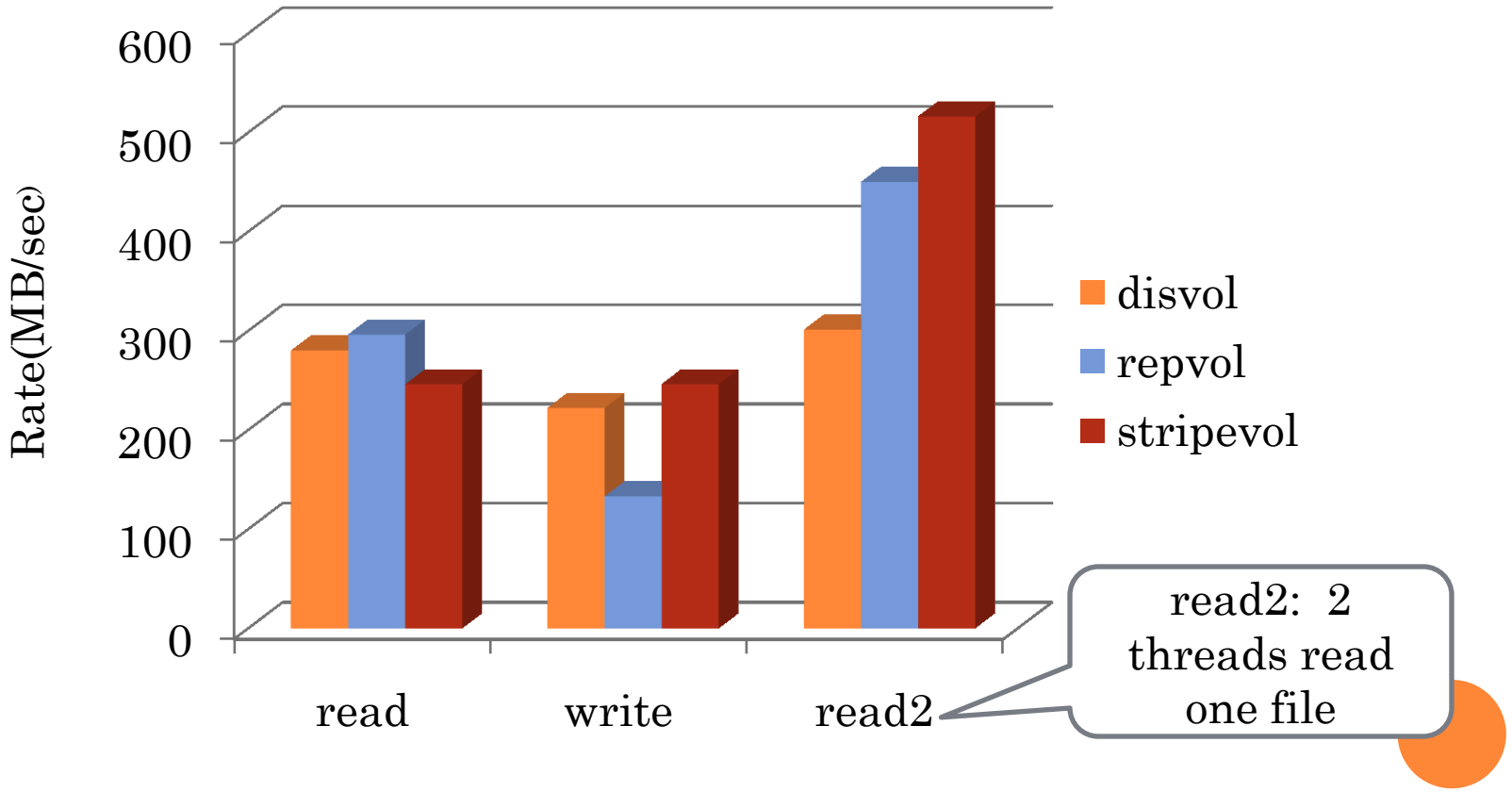
- Two cases
  - Storage server or network fails for one moment, then recovers
  - Disk is destroyed and all data in the disk is lost permanently
- Different types of volume (distributed, striped, replicated volumes) and running operations (read, write) have different affects in the two cases
  - Running operations mean that one is reading or writing files when storage server or network fails

RESULTS

		disvol	repvol	stripevol
First case: Server 1 fails for one moment	capacity	Shrink to 4TB	Expand to 2TB	Error, can't display
	File accessibility	Files on server1 disappeared	All files can be accessible.	Transport endpoint is not connected
	Running read	Files on server1: Error, exit Files on other server: ok	No any break, all is right	Read Error
	Running write	the same as read	After short break, then continue.	Write Error
Second case: Disks on server1 destroyed	capacity	Shrink to 4TB	Expand to 2TB	Error, can't display
	File	Files on server1 Lost	All files can be accessible	File Lost

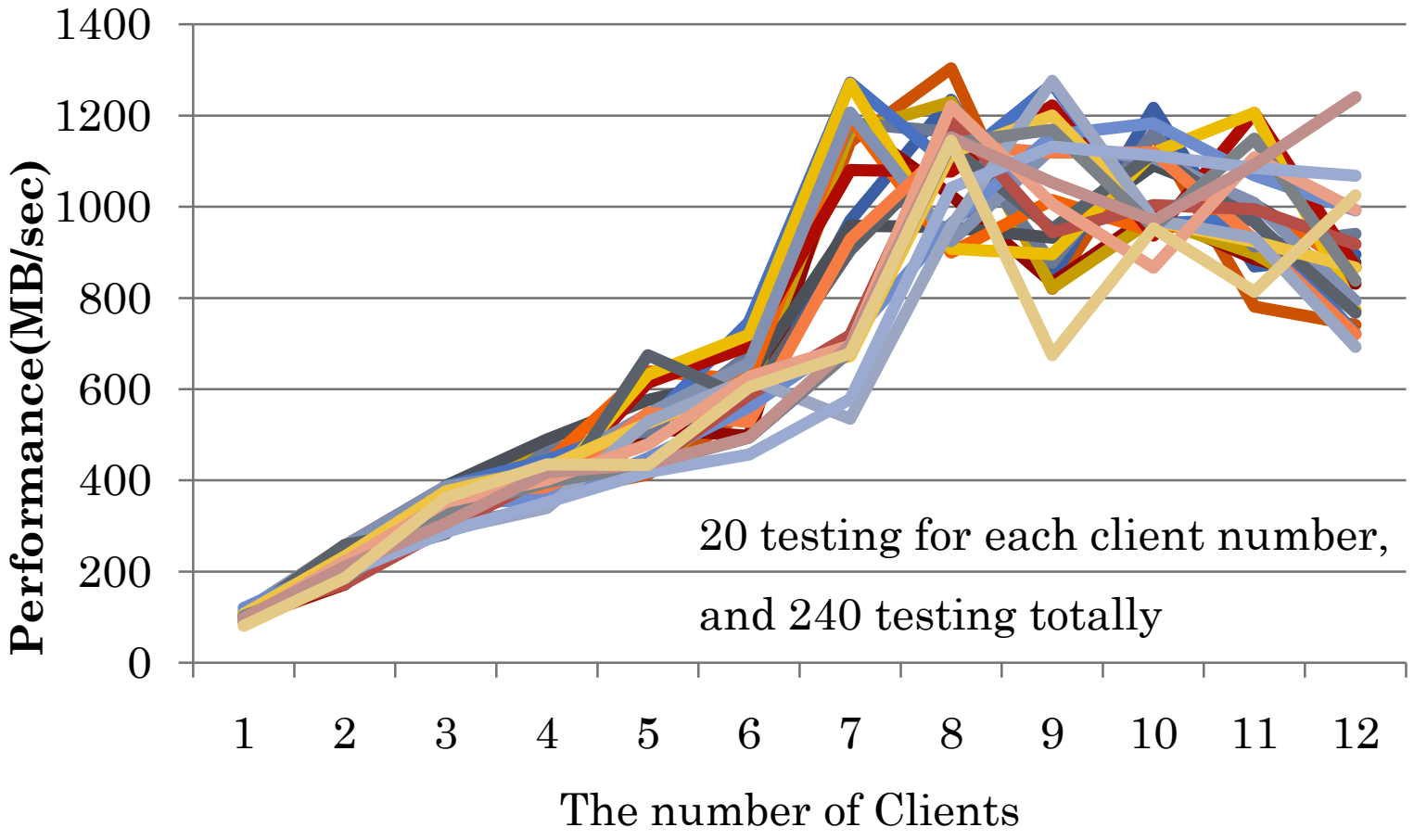
Performance Test

- Single client read/write
  - Method: using "dd" write or read a 10GB file from one client

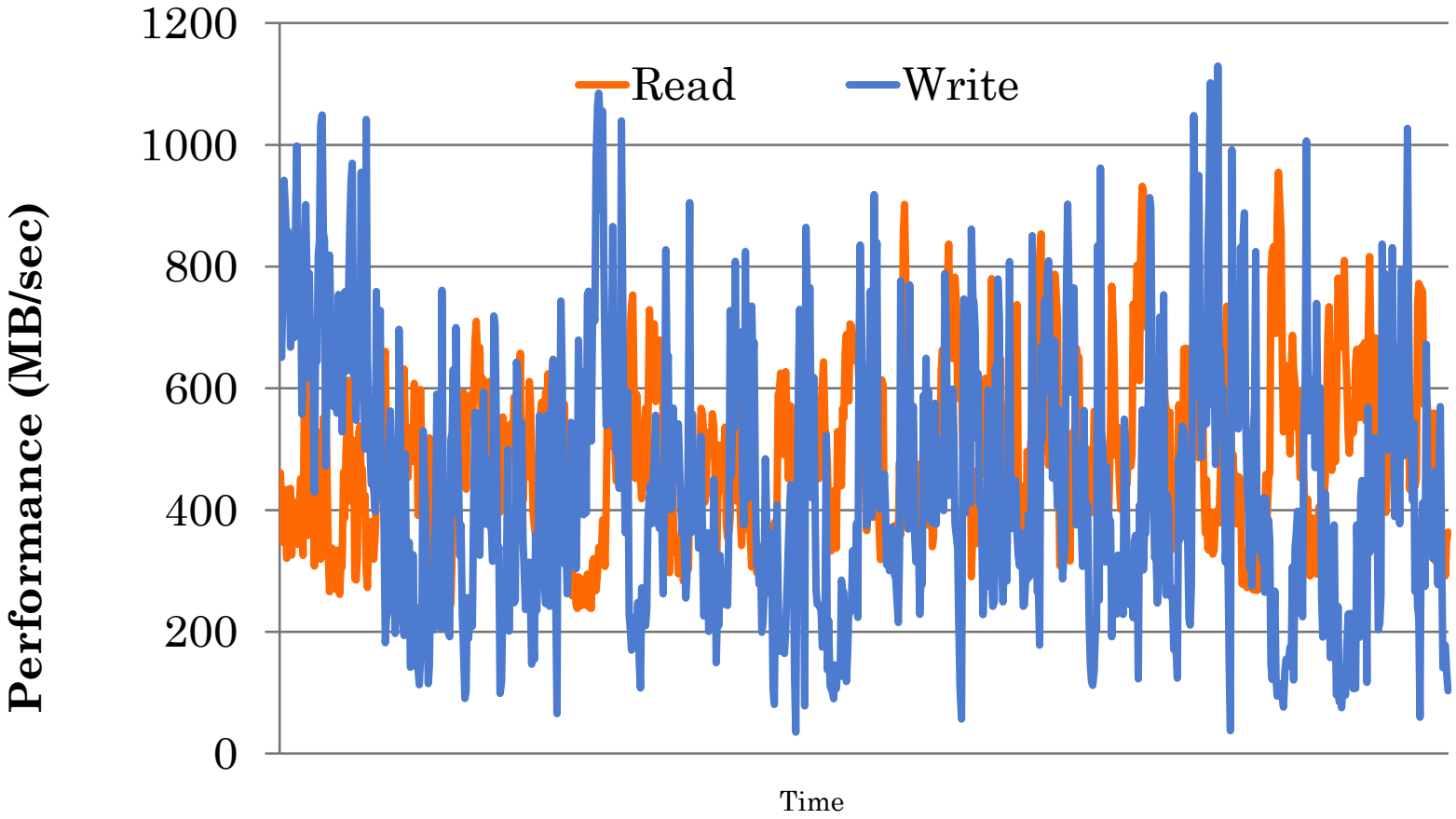


Multiple clients

- Method: Multiple ‘dd’ of different files are read and written from multiple clients simultaneously



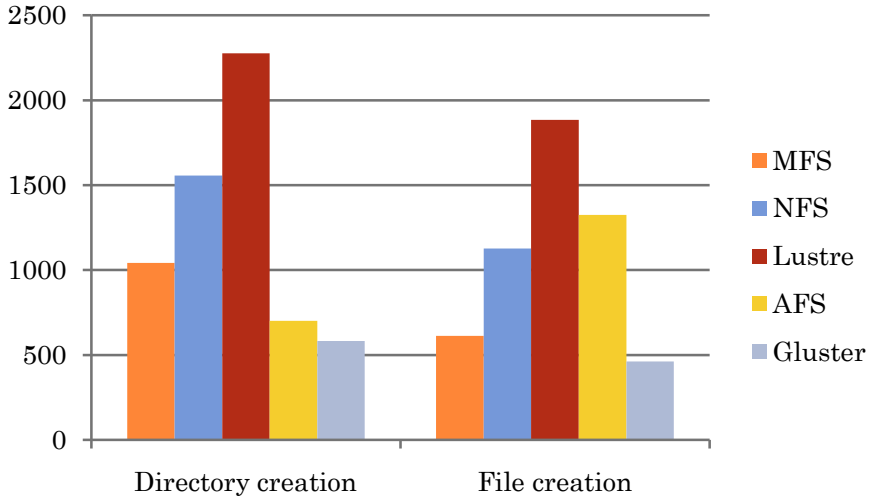
STRESS TESTING

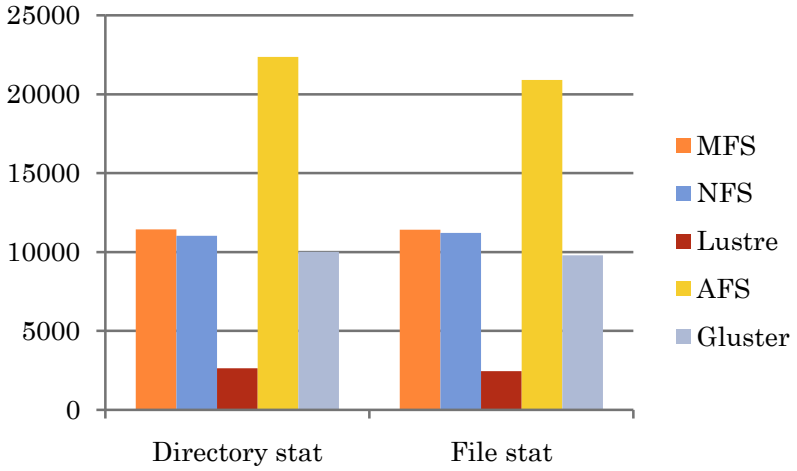


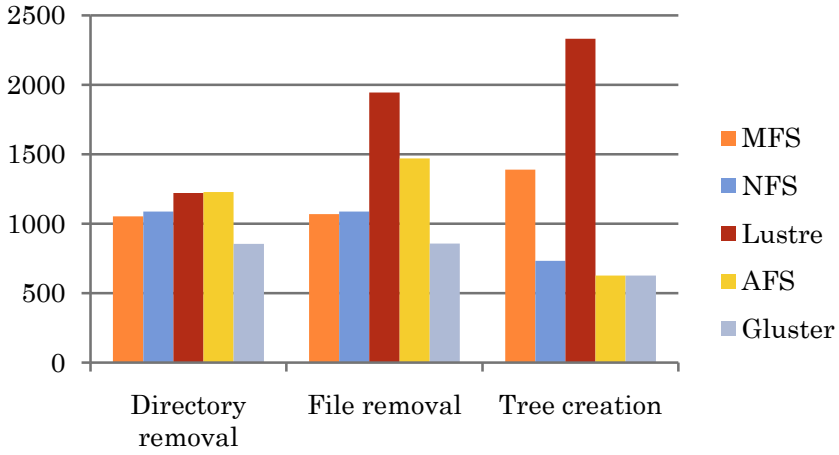
8 Clients, 8 threads per client read/write from 2 servers attached 12 SATA disks for 10 hours

METADATA TESTING

- mpirun -n 4 mdtest -d /data03/z -n 10000 -i 10
  4 tasks, 40000 files/directories (10 iterations)







Conclusions

- In the test bed, gluster showed good scalability, performance, reliability, elasticity and so on.
- But,
  - Not yet stable enough
    - For example, lock problems occurred many time during the testing
  - Not support file-based replication
    - Currently, it only support volume-based replication, and replication policy can't be changed after volume is created
  - Lots of work, for the maintenance of global tree view is done by client, which overload the client, then such operations as 'ls' are very slow
  - …
- Some other considerations are also taken into account when selecting storage
  - Cost of administration, ease of deployment, capacity scaling, support for large name spaces
- Generally, gluster is a good file system, and we will continue to pay attention to it.

Thanks!!

chyd@ihep.ac.cn