Geographically Distributed Software Defined Storage (proposal)

*Sergey Khoruzhnikov[1], Vladimir Grudinin[1], Oleg Sadov[1], **Andrey Shevel[1,2]**, Arsen Kairkanov[1], Oleg Lazo[1], Anatoly Oreshkin[2]*

22nd International Conference on Computing in High Energy and Nuclear Physics, hosted by SLAC and LBNL, Fall 2016

Presenter: Andrey Y Shevel

1 ITMO University, S.Petersburg (Russia)

2 National Research Centre "Kurchatov Institute"
PETERSBURG NUCLEAR PHYSICS INSTITUTE

Scientific sources of Big Data

- **Scientific experimental installations**
  - http://www.iter.org - **International Thermonuclear Experimental Reactor** *(coming)*
    - **~1 PB/year**
  - http://www.lsst.org - **Large Synoptic Survey Telescope**
    - **~10 PB/year**
  - http://www.cern.ch — CERN, http://www.fair-center.eu - FAIR, http://www.cta-observatory.org - CTA — The Cherenkov Telescope Array
    - **~20+ PB/year (each site)**
  - **https://www.skatelescope.org/ - Square Kilometre Array**
    - **~300-1500 PB/year**
  - **Many other aspects of big data: https://www.nist.gov/el/cyber-physical-systems/big-data-pwg**
- **Marginal remark: total volume of data in the World grows two times an year, i.e. around 75% of data were written last two years.**

Storage

- All types of storage are distributed (depends on the scale of distribution: among disk drives, servers in Data Center, or amongst Data Centers (large RTT => 5 msec).
- Several of storage systems for science are proposed and many running.
- Commercial companies suggest distributed data storage solutions: Google (Mesa: GeoReplicated, Near RealTime, Scalable Data Warehousing), Dropbox, Box, Adrive, Amazon, DDN Storage, …
- Which are appropriate solutions for globally distributed data storage in scientific research and education ?
  - Obviously we need for *software defined* solutions.

Main features of SDS

Software Defined Storage should include:

- Automation – Simplified management that reduces the cost of maintaining the storage Infrastructure.
- Standard Interfaces – APIs for the management, provisioning and maintenance of storage devices and services.
- Virtualized Data Path – Block, File and Object interfaces that support applications written to these interfaces.
- Scalability – Seamless ability to scale the storage infrastructure without disruption to availability or performance.

Technical details of GDSDS

- Important features:
  - Data storing and Data transfer
    - Reliability: *data replication, erasure coding.*
    - Reduce the volume: *Data compression.*
    - Security: *Data encryption, ACL.*
  - GDSDS Web portal and GDSDS CLI.
  - Network architecture.
  - Caching, Tiering.
  - Automatic storage deployment by user request.

Network aspects on GDSDS

- First of all we have to keep in mind the CAP theorem:
  - Theoretically NOT possible to guarantee all below requirements at the same time.
    - Consistency
    - Availability
    - Partitioning

Similar (in some aspects) developments

- Project OsiRIS at University of Michigan - https://indico.cern.ch/event/466991/contributions/1143627/
- http://eos.cern.ch
- Owncloud.org

Basic assumptions on GDSDS

- It is assumed
  - GDSDS consists of several groups of storage servers located in geographically different regions.
  - Groups of servers are connected by a number of parallel virtual data links.
    - Data links might have different features: speed, price, encryption type (including *quantum encryption), etc.*
  - Data links are to be configured with SDN.
  - Client can ask to perform a number of operations:
    - Create, Upgrade, Downgrade, Delete, Replicate, Migrate, etc an instance of Virtual Storage allocated on GDSDS. The instance might be created with different SLA
    - Write/Read data to/from the instance of Virtual Storage.

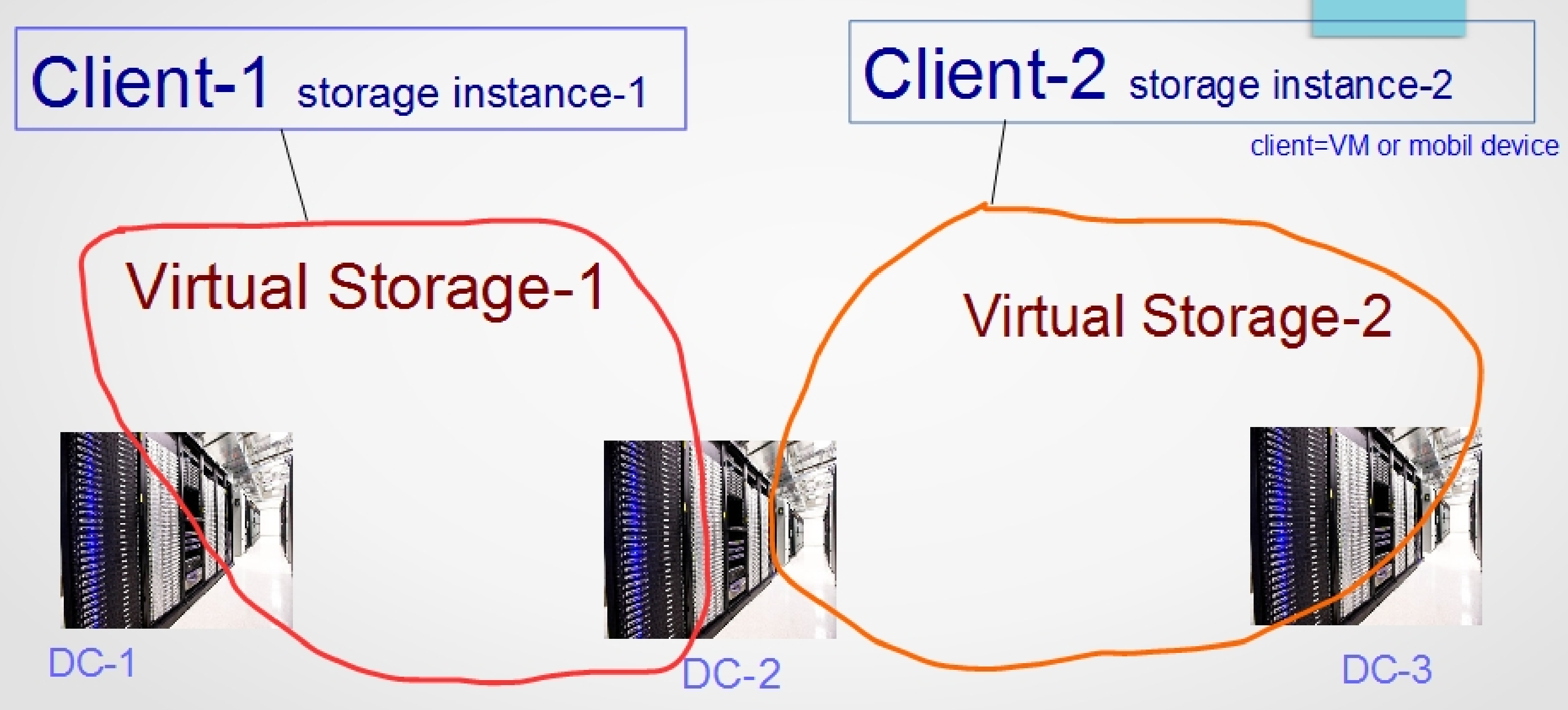
Some details

- It is supposed that command create Storage Instance might be issued by the user from the SGSDS portal. It is not often required operation.
  - In result the user has to receive all information about operation completion code and information how to use created Storage Instance.
- It is planned for each operation create to create new Instance of storage cluster. Separate Instances are completely independent each other.

Examples for SLA

- Specific type of Data Encryption.
- Specific type of Data Compression.
- On one specific Data Center (DC) or on many DCs with specific types of Data Links.
- Type of backend: CEPH, SWFT, EOS, etc

Allocation of instances of Virtual Storage


Client-1 storage instance-1

Client-2 storage instance-2

client=VM or mobil device

Virtual Storage-1

Virtual Storage-2

DC-1

DC-2

DC-3

References

- Jakob Blomer // Survey of distributed file system technology // ACAT 2014, Prague (in references) Also *iopscience.iop.org/article/10.1088/1742-6596/664/4/042004/pdf*
- Why so Sirius? Ceph backed storage at the RAL Tier-1.
  - https://indico.cern.ch/event/466991/contributions/2136880/contribution.pdf
- Analysis of Six Distributed File Systems – HAL-Inria - *https://hal.inria.fr/hal-00789086/file/a_survey_of_dfs.pdf*
- https://en.wikipedia.org/wiki/Comparison_of_distributed_file_systems
- XtreemFS is a fault-tolerant distributed file system for all storage needs http://www.xtreemfs.org/
- Software Defined Storage LizardFS is a distributed, scalable, fault-tolerant and highly available file system - https://lizardfs.com/about-lizardfs/