CERN

European Organization for Nuclear Research
Organisation Européenne pour la Recherche Nucléaire

Large Storage Systems Present & Future

Dr. Andreas-Joachim Peters

GridKA 2012 - Karlsruhe

andreas.joachim.peters@cern.ch

CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

Outline

- Why are large storage systems relevant?
- What are the challenges?
- How large are storage systems today?
- Which technologies are used?
- Which technologies are emerging?
- What can we expect from current and future technology?
- Summary and Conclusions

Why are large storage systems relevant?

According to International Data Corporation, the total amount of global data is expected to grow to **2.7 zettabytes** during 2012.

This is **48%** up from 2011.

The storage industry has sold more than **350 exabyte** of storage in 2011.

Multi-PB storage systems are a **norm** and available by many vendors!

Storage Market 2011

Technology	Situation	Units Sold [Million]	Volume Sold [Exa Byte]
DRAM	4 companies have 90% of market share: Samsung, Samsung, Hynix, Micron, Elpida (bancrupt)	800	2
NAND	4 companies have 99% market share: Samsung, Toshiba, Micron, Hynix	4000	19
SSD	> 50 companies	17	3*
HDD	3 companies only: Western Digital 50%, Seagate 39%,Toshiba 11%	630	330
TAPE	3 technologies: IBM, Oracle, LTO---consortium LTO has 90% market share	27	20

90% of capacity sold!

* included in NAND numbers

From Bernd Panzer Steindel/CERN

LSS Challenges and Implications

- large volume & meta data
  - PB Stores with millions to billions of objects
    - can a hierarchical namespace be kept?
- new scale of hardware failures
  - e.g. 12k disks ~ 1 failing disk per day
  - multi-PB RAID-6 probability of data loss within 5y at *n*-% level
- flexible, sizable and administrable
  - capacity growth and life-cycle management in production
- manifold requirements and tuning parameters
  - bandwidth, IOPS, latencies, costs, volume/object/user scalability
  - interfaces for objects, files, block devices, cloud infrastructure
  - technology evolution decreases performance:capacity ratio

scale-out storage systems, aggregation & federation

new **redundancy** methods - RAIN, eventual consistency with quorum on object versions

elastic block storage w/o DHTs/dynamo principle

virtualization

storage tiering

virtualization

volume grows faster than **bandwidth** - MB/s per GB

Challenge

finally: costs matter more at large scale!

Some Challengers ...

whamcloud lustre

panasas

OrangeFS

Cleversafe LIMITLESS DATA STORAGE

ceph PETABYTE SCALE STORAGE

amazon web services

XTREEMFS

HUAWEI

SCALITY

GLUSTER

hadoop HDFS

openstack

BLUEARC PART OF HITACHI DATA SYSTEMS

dCache

XRootD

IBM Systems > Software > SONAS IBM

General Parallel File System

Efficient storage management for big data applications

RACKSPACE CLOUD FILES On-Demand Storage & Content Delivery

CASTOR CERN Advanced STORage manager

DPM

EOS

The world is divided

POSIX
Cluster Filesystems

Commercial Products

Hardware Solutions

"Storage in a Box"

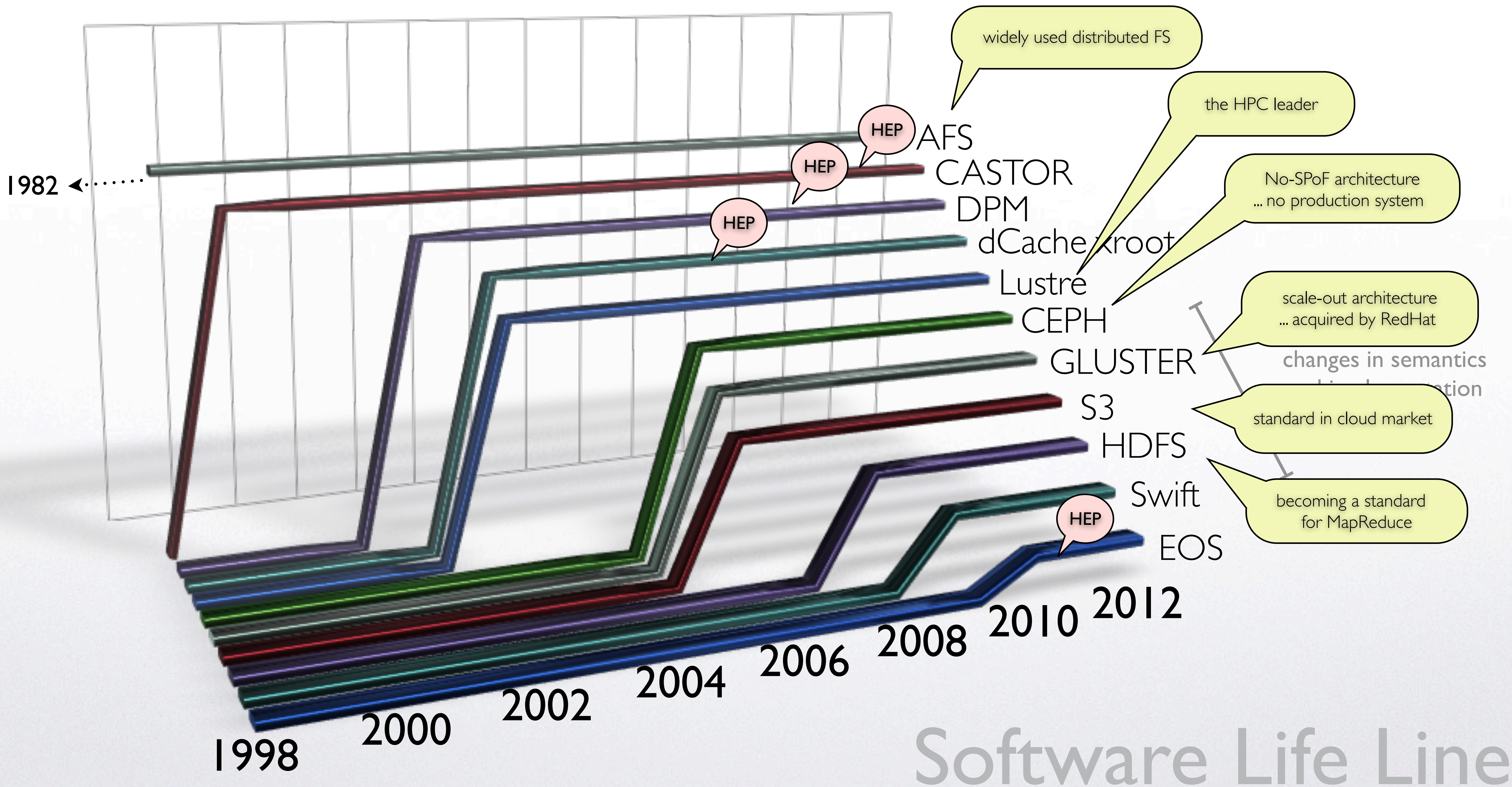
Cloud/MapReduce
Storage

Open Source

Software Solutions
"Buy & Build"

... but also coalescing
into many hybrid storage systems ...

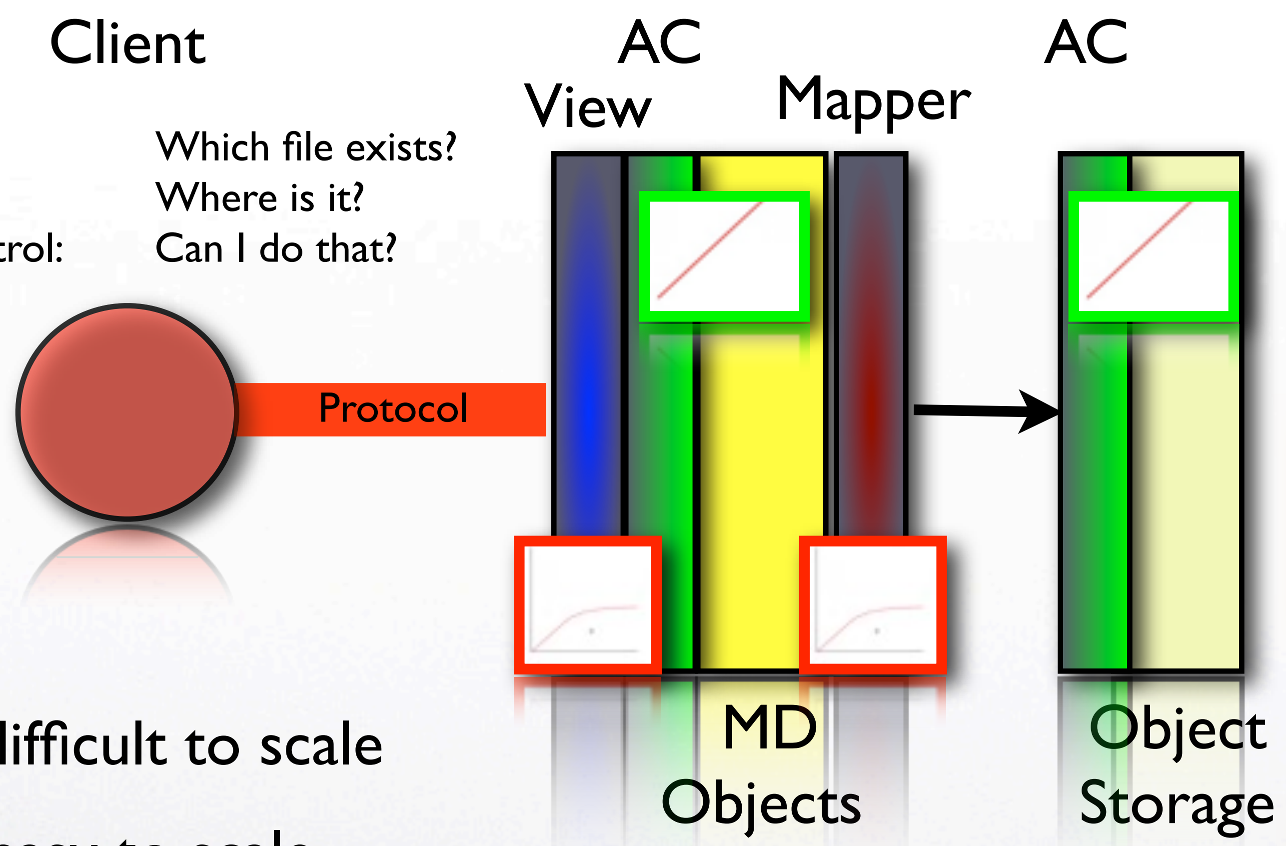
(Open) Solutions have a long way in(-to) production ...

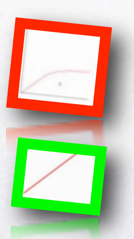


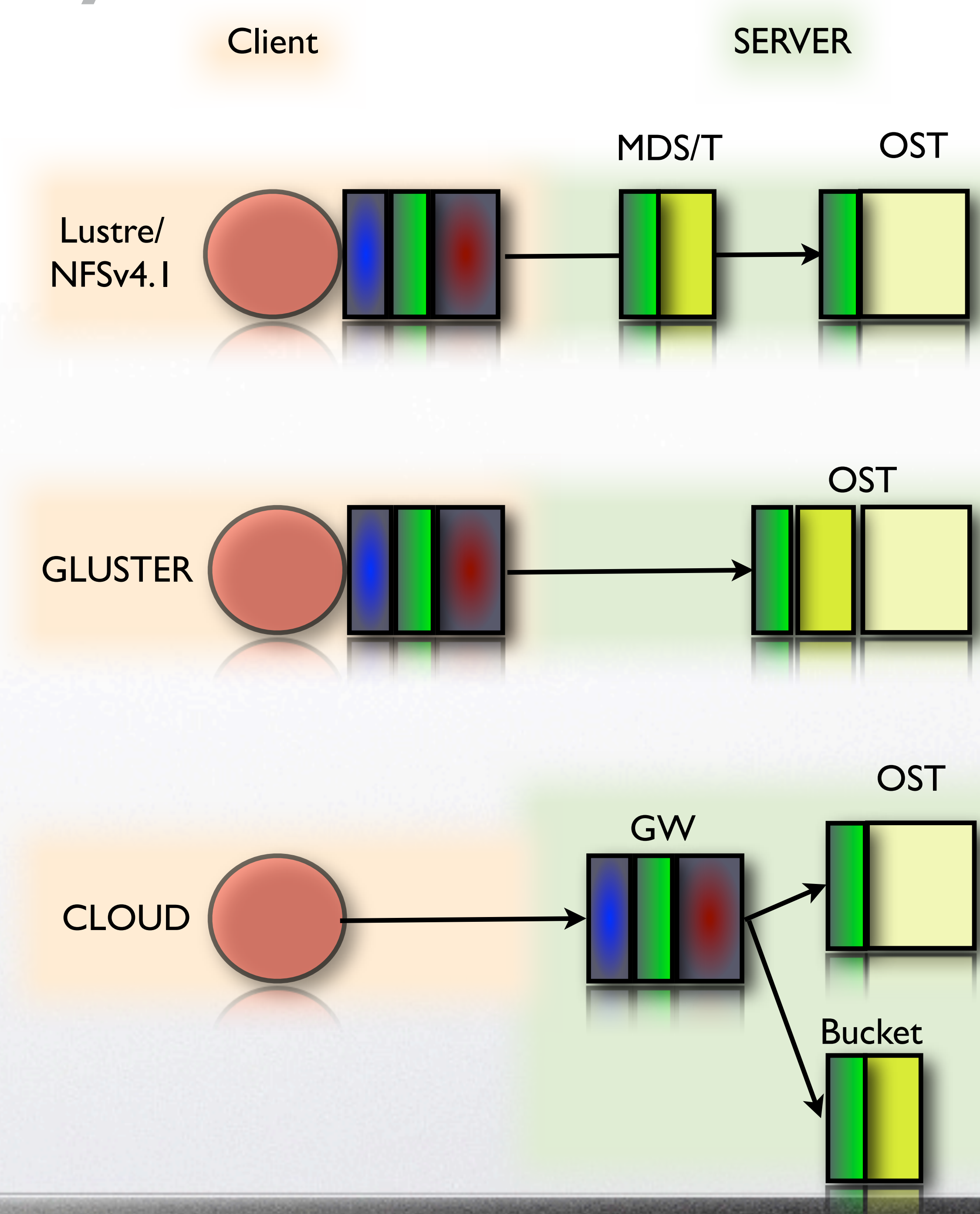
Scalable Storage Systems

Standard Model

View:	Which file exists?
Mapper:	Where is it?
Access Control:	Can I do that?



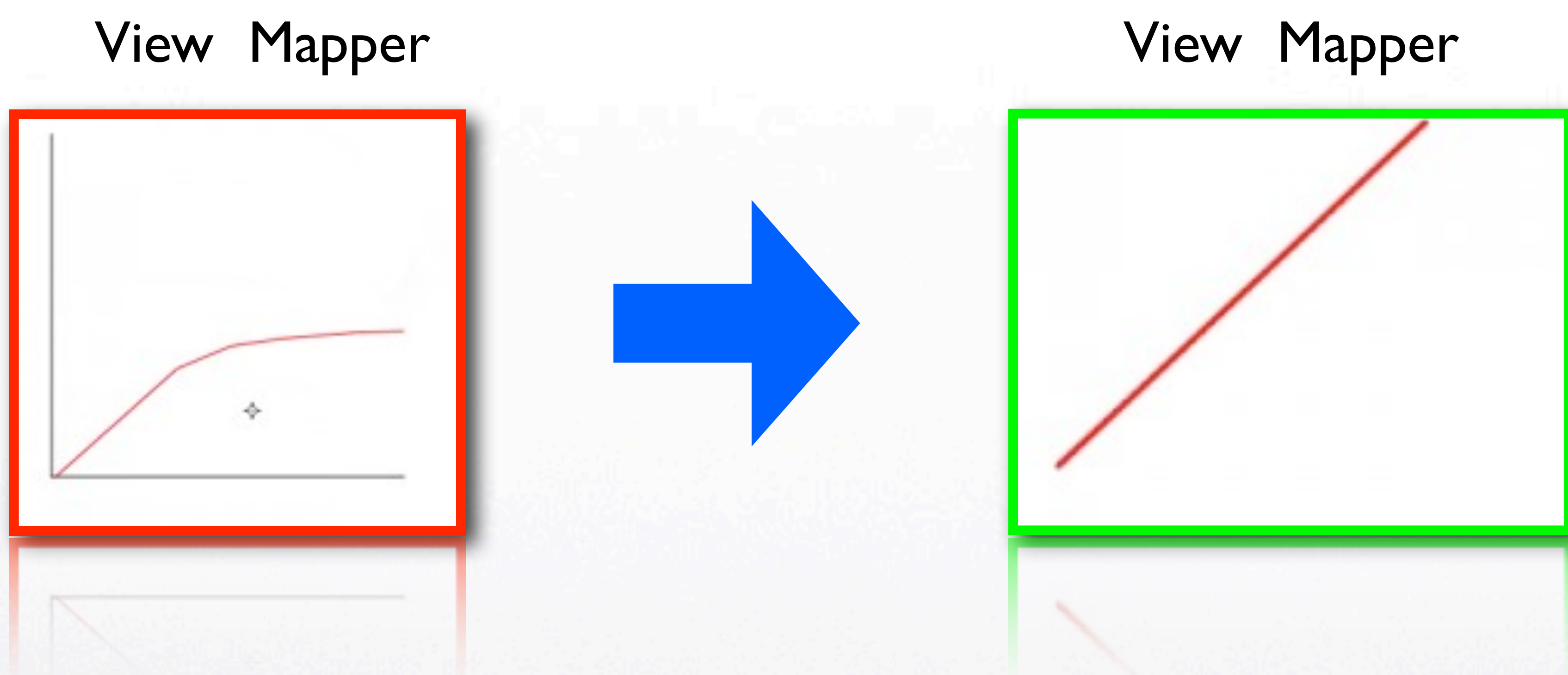




Difference in implementations

- Location of components (client/server side,in-/out-of-band)
- Implementation of View, Access Control & Mapper
- MD Content, Object Content

How to scale?



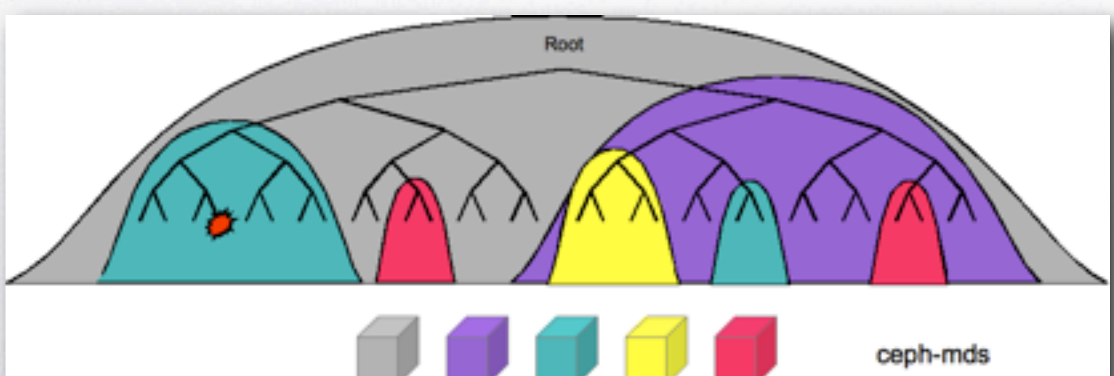
Scalability of Storage View

- Options
  - remove view - pure key-value store
    out-source 'problem' into NOSQL solution
  - split view by 'directories(-parts)' or 'buckets'
    - distribute pieces over many machines and make redundant

Complex Solution

dynamic subtree partitioning providing full hierarchy

ceph



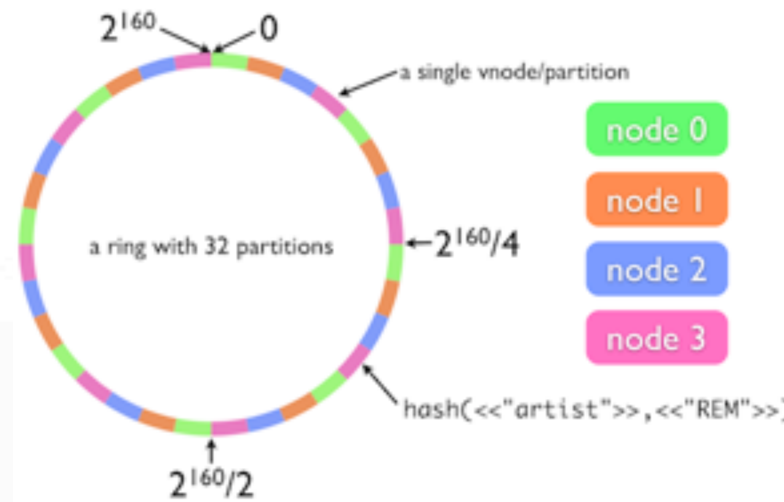
Trivial Solution

flat bucket space with internal pseudo hierarchy

Bucket Name	X	Y	Z	A	B
S3	/A B C D E F G	/X Y Z	/1 /2 /3 /4/1 /4/2	/A B C	/A /B ?C /D/1 /E/2 /E/5

Scalability of Storage Mapping

- Options
  - Algorithmic mapping
    - hash(key) => Lookup-Table => [Locations]
      - Consistent hashing **Dynamo** principle



- Location can be computed on client and server side without MD lookup
- Files have to be chopped into small pieces for good balancing - homogeneous pool segmentation needed

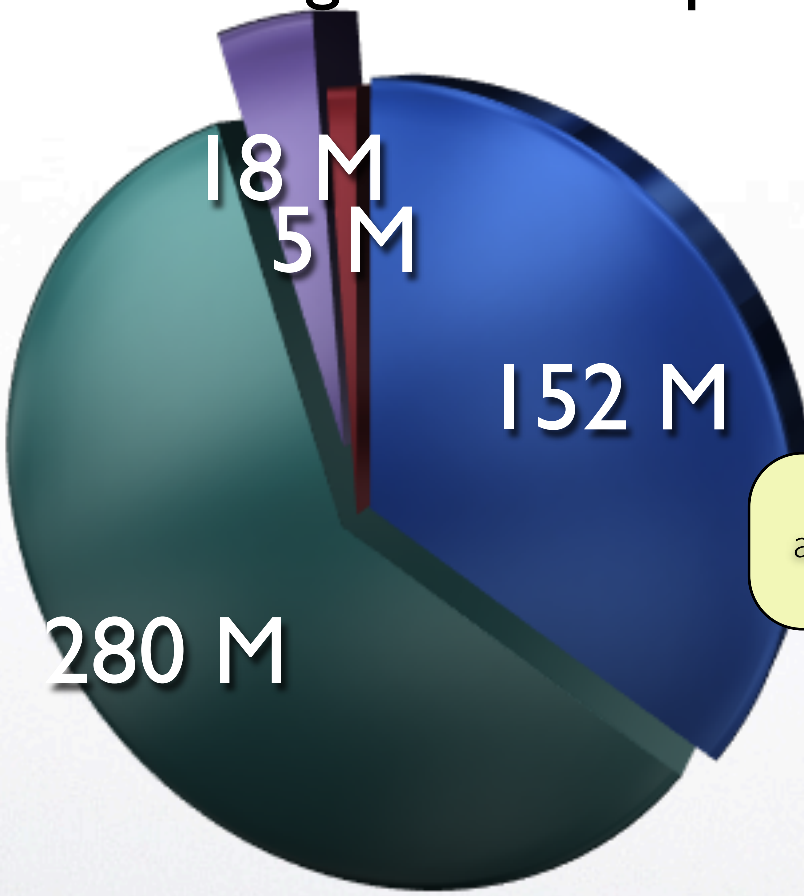
  - Location Index (Cache)
    - stored along with meta data or cached
- Location needs external lookup on clients and storage servers
- More flexibility in placement policies - ‘simpler to look at’

What is a large storage system today?

Is LHC Storage large?

LHC Storage

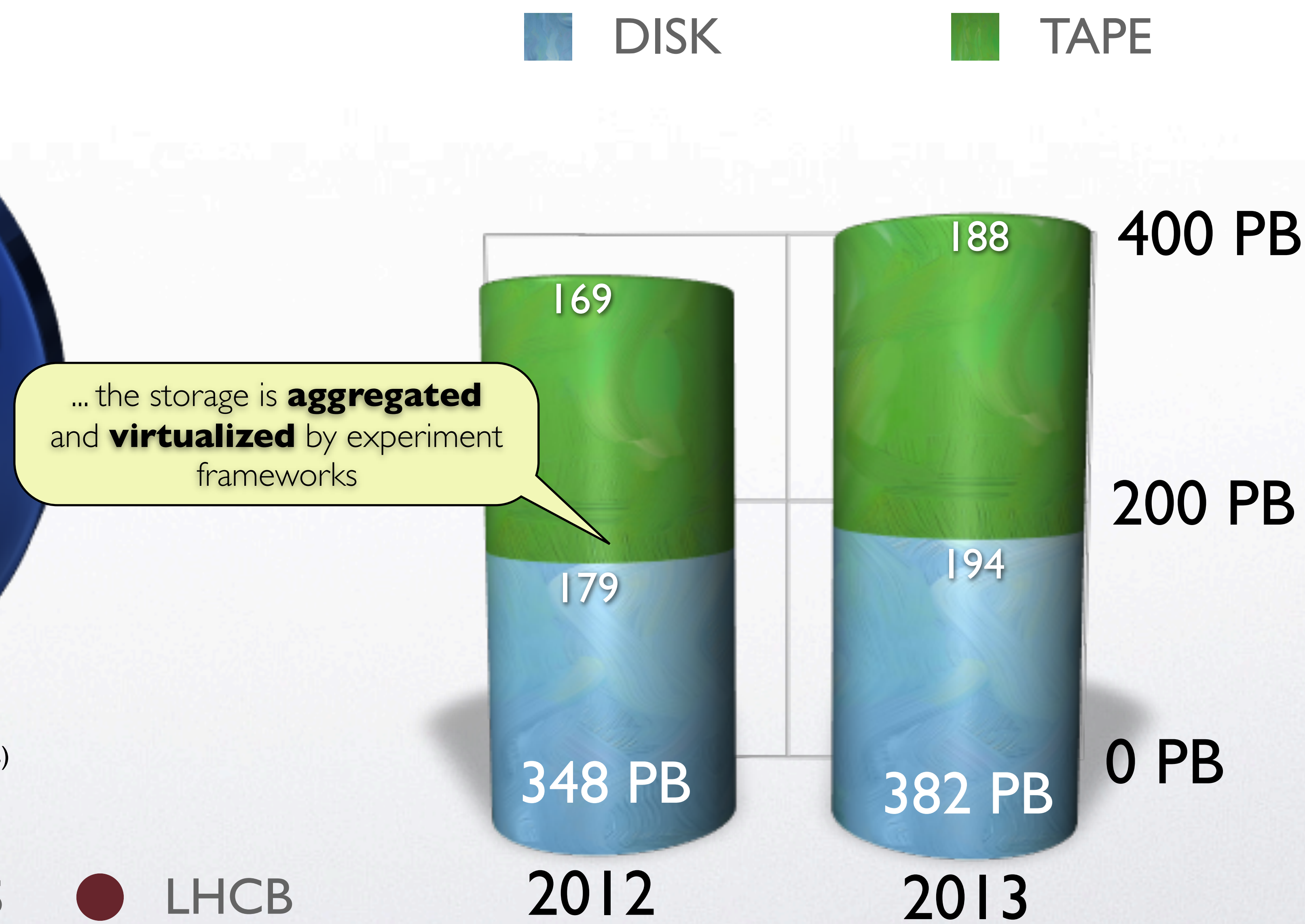
Number of managed files - April 2012



This are only managed files (there are more user files)

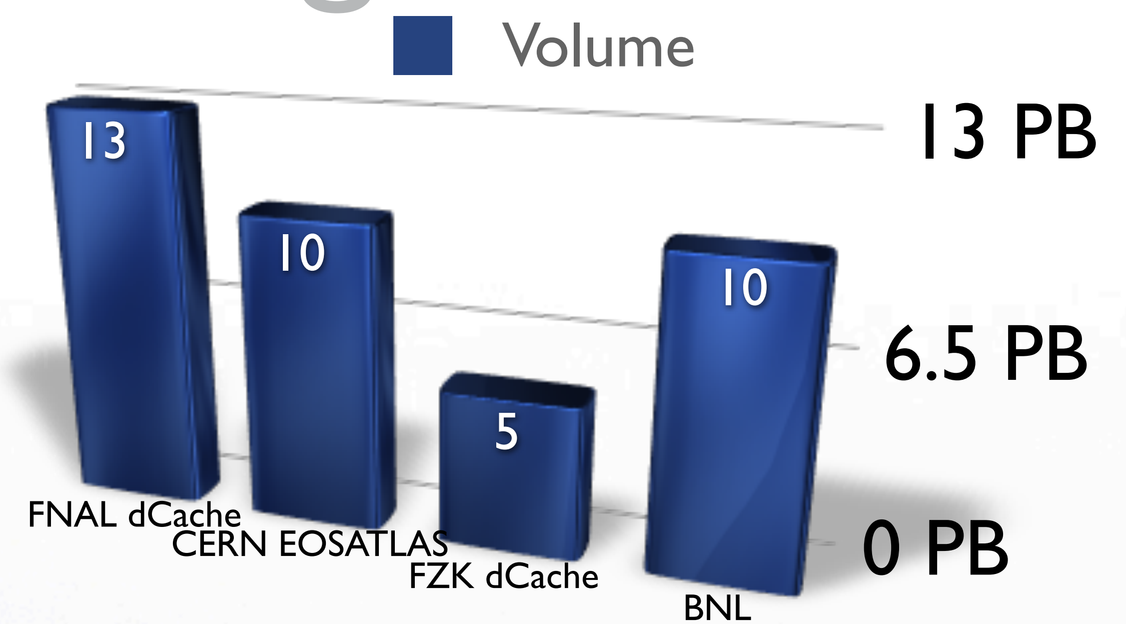
ALICE ATLAS CMS LHCB

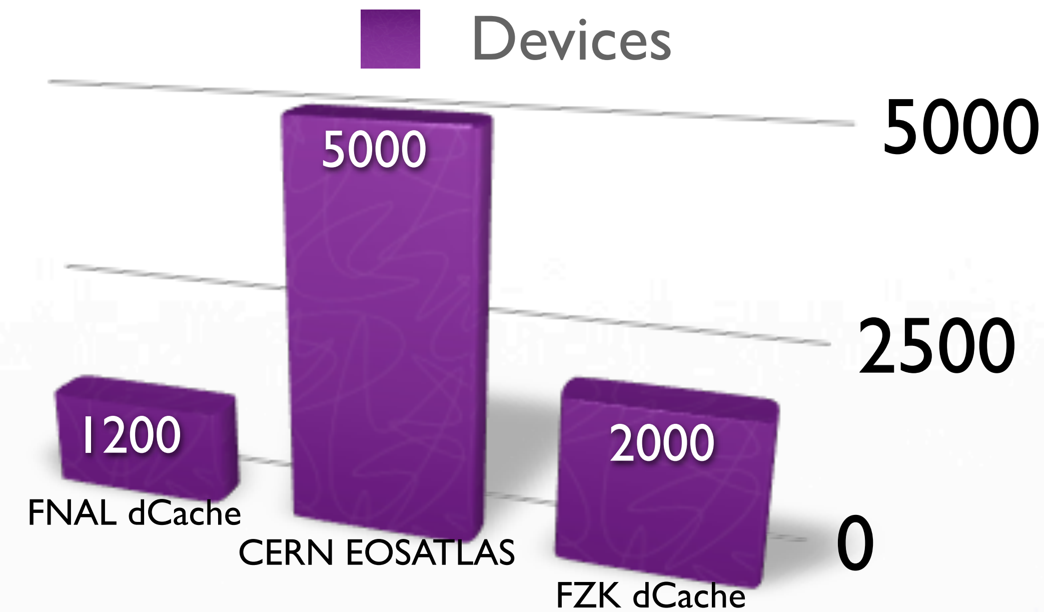
CRSG Recommendations

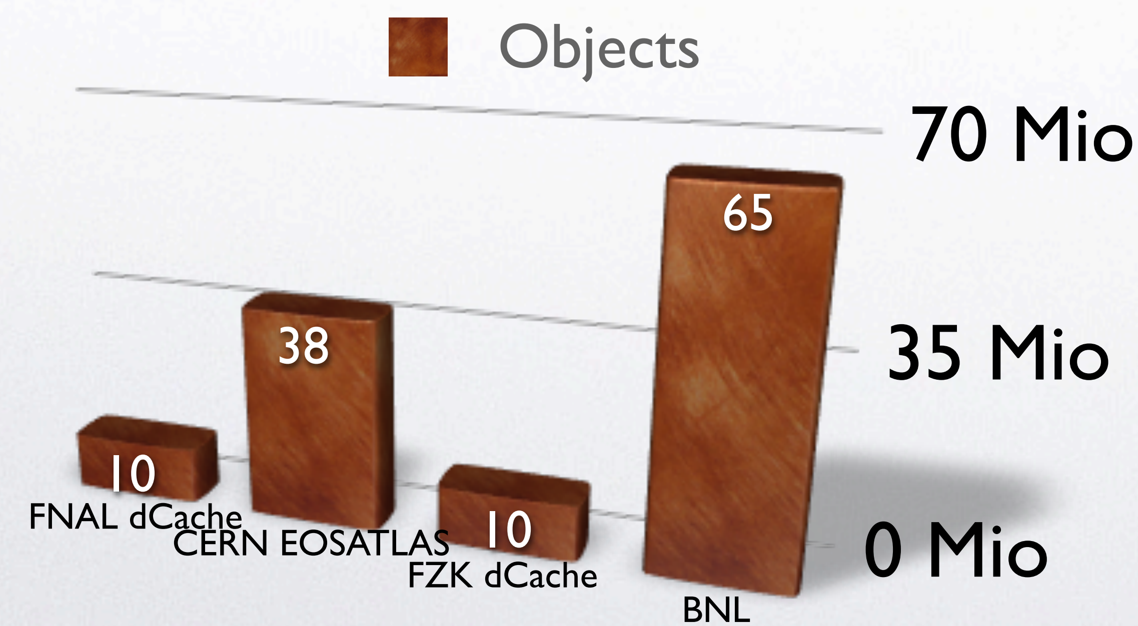


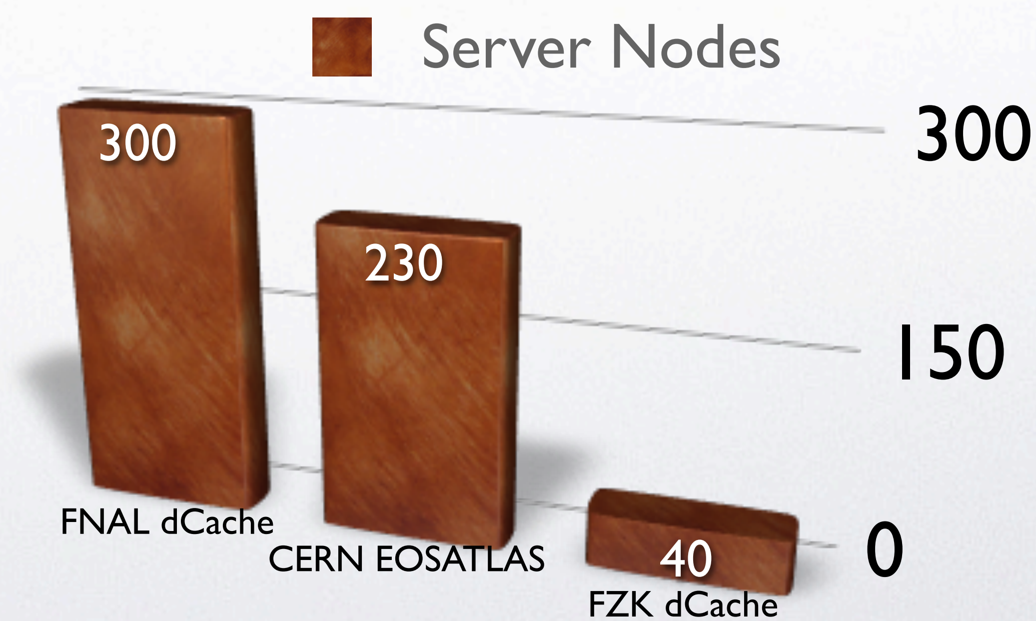
Comparable amount of Disk and Tape Storage

Large LHC Storage Instances ...



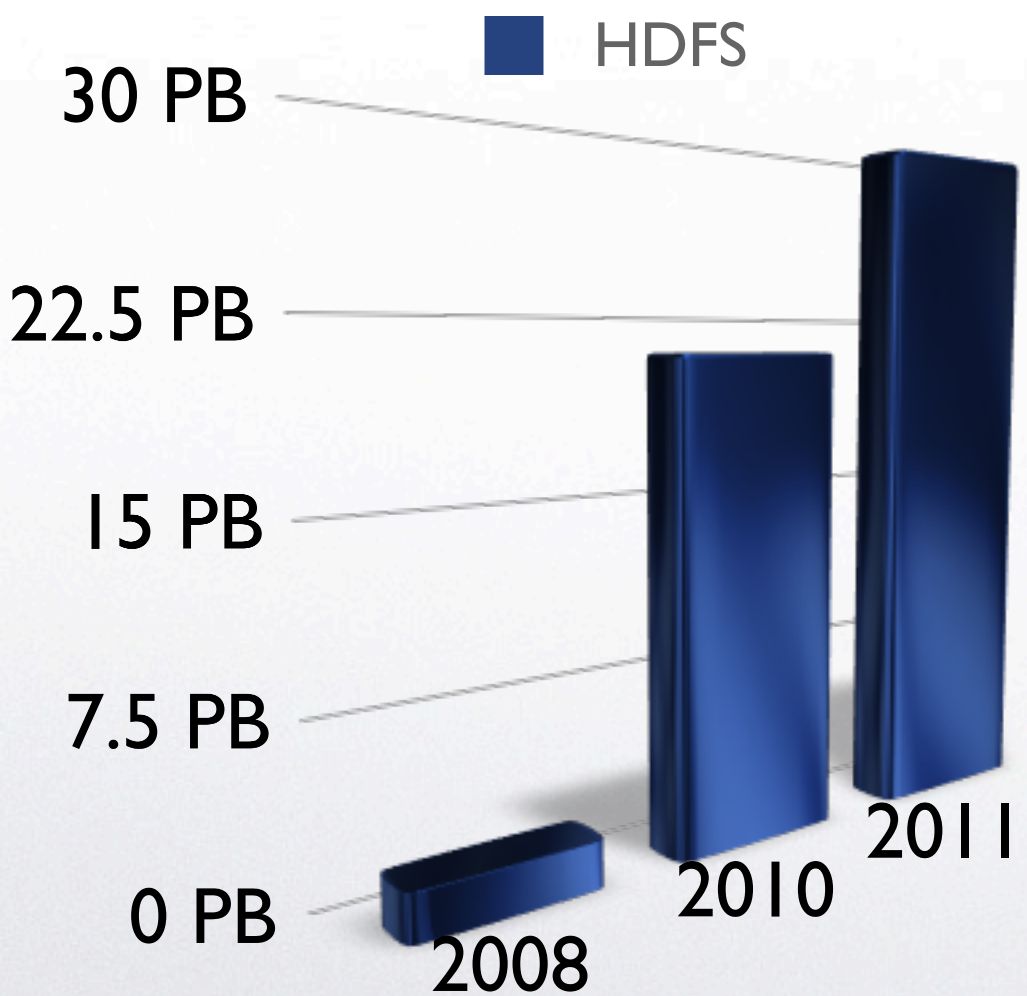






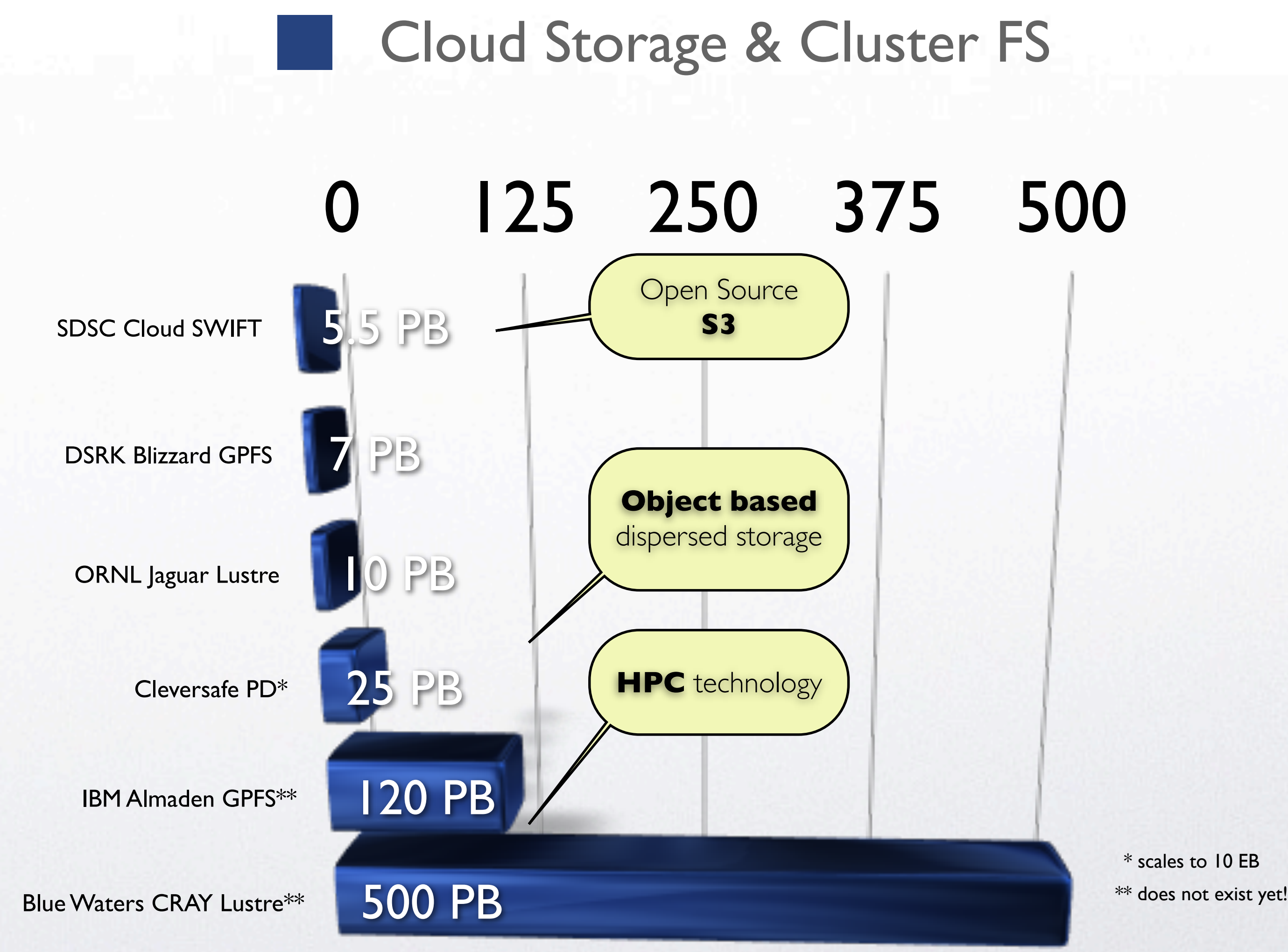
Other Large Storage Systems

Facebook



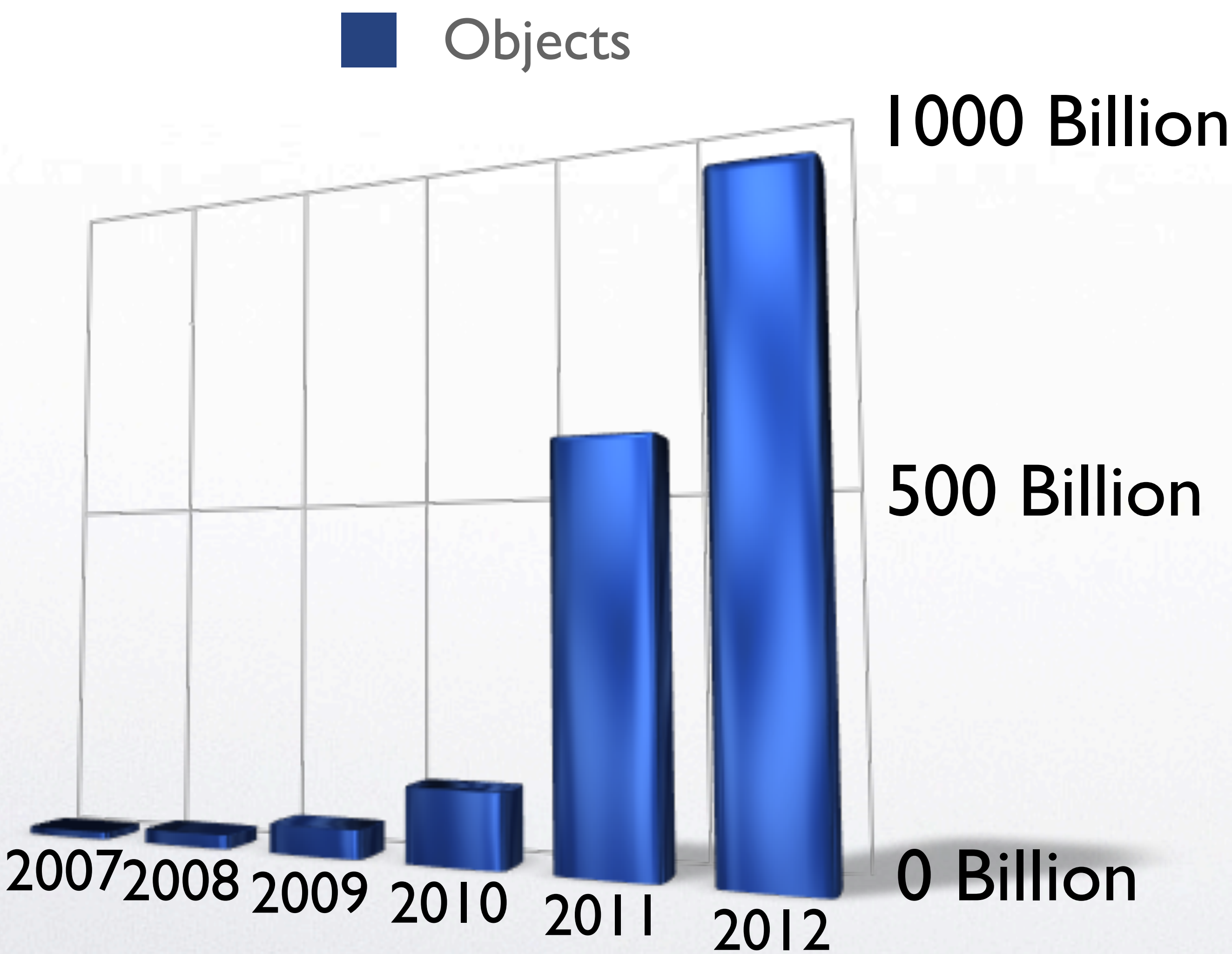
From facebook.com Paul Yang Facebook Engineering

Large Storage Installations



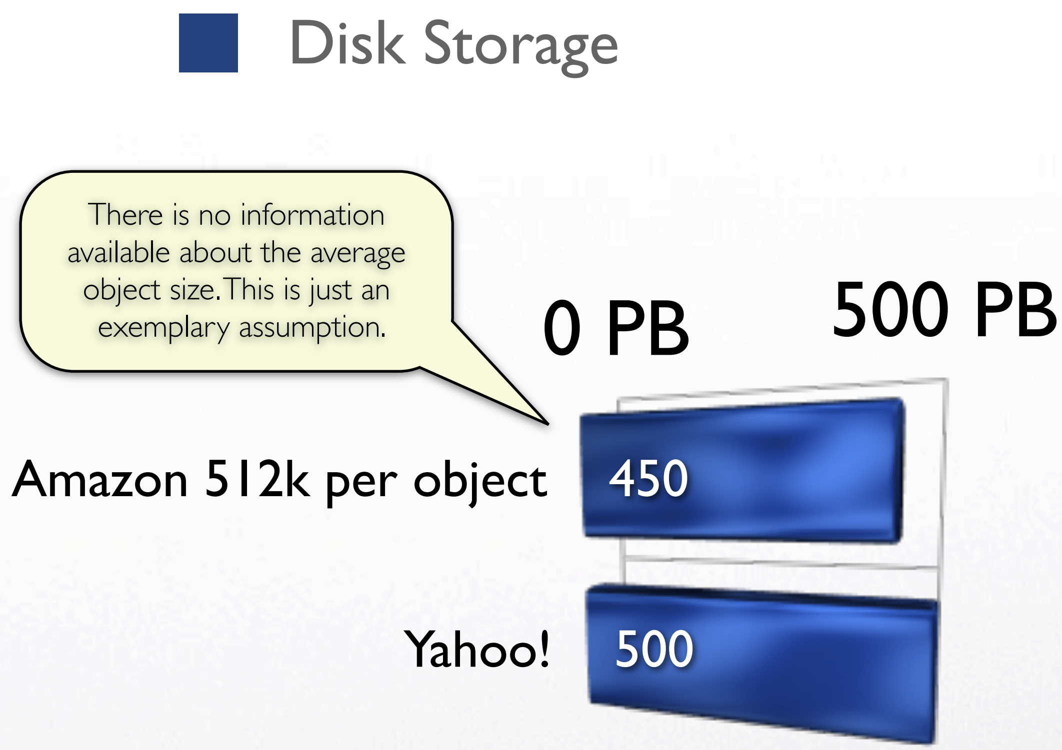
Today: **Cleversafe** 10 EB system would require 4.5M disks and cost several billion $$!

Amazon S3 & Yahoo!



From Amazon Web Service Blog

650k Request/s

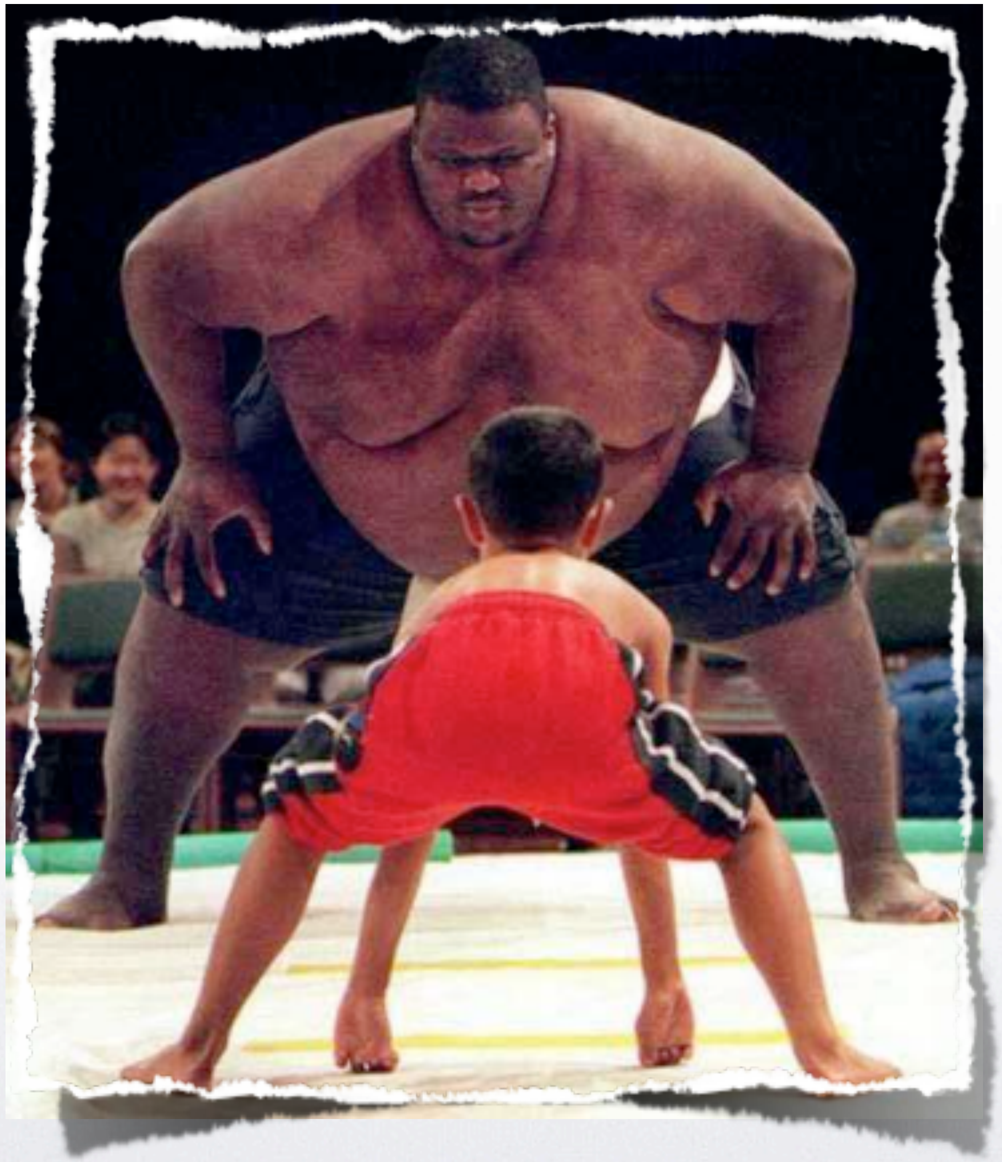


From LTUG 2012

GOOGLE. does not publish these numbers ...

... there are many other Cloud Storage Provider: DropBox, iCloud, Google Drive ...

Is LHC Storage large?



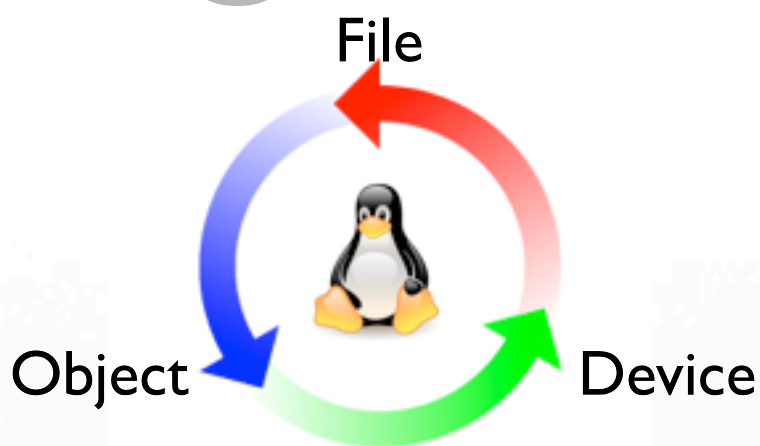
Volume

Objects

LHC Storage is large in volume - not in number of objects!

Technology Trends

Storage Interfaces



Access Protocols

Complex/rich features set — Simple/reduced feature set

POSIX* | POSIX* light | RAP** | S3 | HTTP | REST

standard applications run out of the box with this interface

applications have to be enabled or use a download - process with POSIX - upload approach

* POSIX via AFS, Lustre, GPFS, pNFS or FUSE client
** RAP: remote access protocols like ftp, XRootD, DCap++, rfio

Storage Semantics

1. POSIX(-like) Storage

Based on **filesystems**, **RDBMS** ...

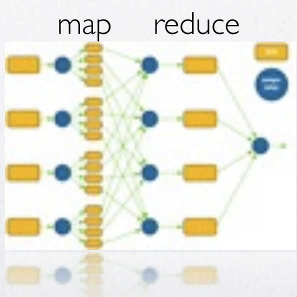
- GPFS, Lustre, AFS, pNFS, OrangeFS, GLUSTER **et.al.**
- CEPH, FUSE **driver for <*xyz*> et.al.**

2. Cloud Storage

Based on **Object Stores**, **DHTs** and **key-value DB's**

- Amazon S3, Swift, Facebook Haystack et al.

3. Map-Reduce Storage



- GoogleFS, HDFS et.al.

Some solutions mix semantics and technology

How many files can a Cluster filesystem have?

"GPFS scans 10 billion files"

Scalability is sufficient for designed use cases. ... still highlights also the problem of hierarchical namespaces.

Richard Freitas, Joseph Slember, Wayne Sawdon, and Lawrence Chiu. GPFS Scans 10 Billion Files in 43 Minutes. 2011.

http://www.almaden.ibm.com/storagesystems/resources/GPFS-Violin-white-paper.pdf

- theoretical **exercise (policy scan)**

  0-size files and 6.5TB of meta data

- **meta data on SSDs (violin memory)**
  - > 1MIOPS @ 4k
  - 4 GB/s

  'would require hundreds of hard disks to reach SSD performance'

Cloud storage is leading a way out!

From POSIX to Cloud API

Web Scale Problems ...

access
aio_cancel
aio_error
aio_read
aio_return
aio_suspend
aio_write
chdir
chmod
chown
close
closedir
creat
dup
dup2
fcntl
fdatasync
fdopen
fstat
fsync
getcwd
link
lio_listio
lseek
mkdir
mkfifo
msync
open
opendir
read
readdir
rename
rewinddir
rmdir
stat
umask
uname
unlink
utime
write

POSIX IO

Simplifications mainly:

- Drop Namespace Hierarchy
- WORM - write once read many
- GetObject [Range] PutObject, Delete Object
- bucket handling

Service

GET Service

Bucket

DELETE Bucket
DELETE Bucket lifecycle
DELETE Bucket policy
DELETE Bucket website
GET Bucket (List Objects)
GET Bucket acl
GET Bucket lifecycle
GET Bucket policy
GET Bucket location
GET Bucket logging
GET Bucket notification
GET Bucket Object versions
GET Bucket requestPayment
GET Bucket versioning
GET Bucket website
HEAD Bucket
List Multipart Uploads
PUT Bucket
PUT Bucket acl
PUT Bucket lifecycle
PUT Bucket policy
PUT Bucket logging
PUT Bucket notification
PUT Bucket requestPayment
PUT Bucket versioning
PUT Bucket website

Object

DELETE Object
Delete Multiple Objects
GET Object
GET Object ACL
GET Object torrent
HEAD Object
POST Object
PUT Object
PUT Object acl
PUT Object - Copy
Initiate Multipart Upload
Upload Part
Upload Part - Copy
Complete Multipart Upload
Abort Multipart Upload
List Parts

S3 API

simplifications-=> simple resilient
scale-out architecture

interface changes =>
compromises

The Cloud Compromise

- increased latency
- eventual consistency
- reduced **but** simpler storage interface
- **goes along with** MapReduce **for efficient data access**
  - move task where the data is, optimize for large IOs, rewrite your application "no POSIX", WORM & append-only
- **proven** scalability **and** manageability **e.g.**
  - 900 Billion Objects in Amazon S3
  - 100 Billion Photos in Facebook 2011
- **can run in** no maintenance **mode**
  - no repair approach - a failed disk or node needs no intervention
  - exchange/migrate all after natural lifecycle of the whole system

... solved the web problem

... is not optimal for non-sequential small reads

... allows large installations to be easy operable and cheap

Cloud: Overall performance and scalability
HPC: Single client performance

Storage Tiering

Tiered storage is the assignment of different categories of data to different types of storage media in order to reduce total storage cost.

- Examples
  - *HSM* Systems
    - Lustre^{HSM}, GPFS^{HSM}, dCache, CASTOR
  - *HHD/SSH* = HD + SSD Cache
  - *VTL* - Virtual Tape Libraries
  - *CAS* - Content-addressed Storage

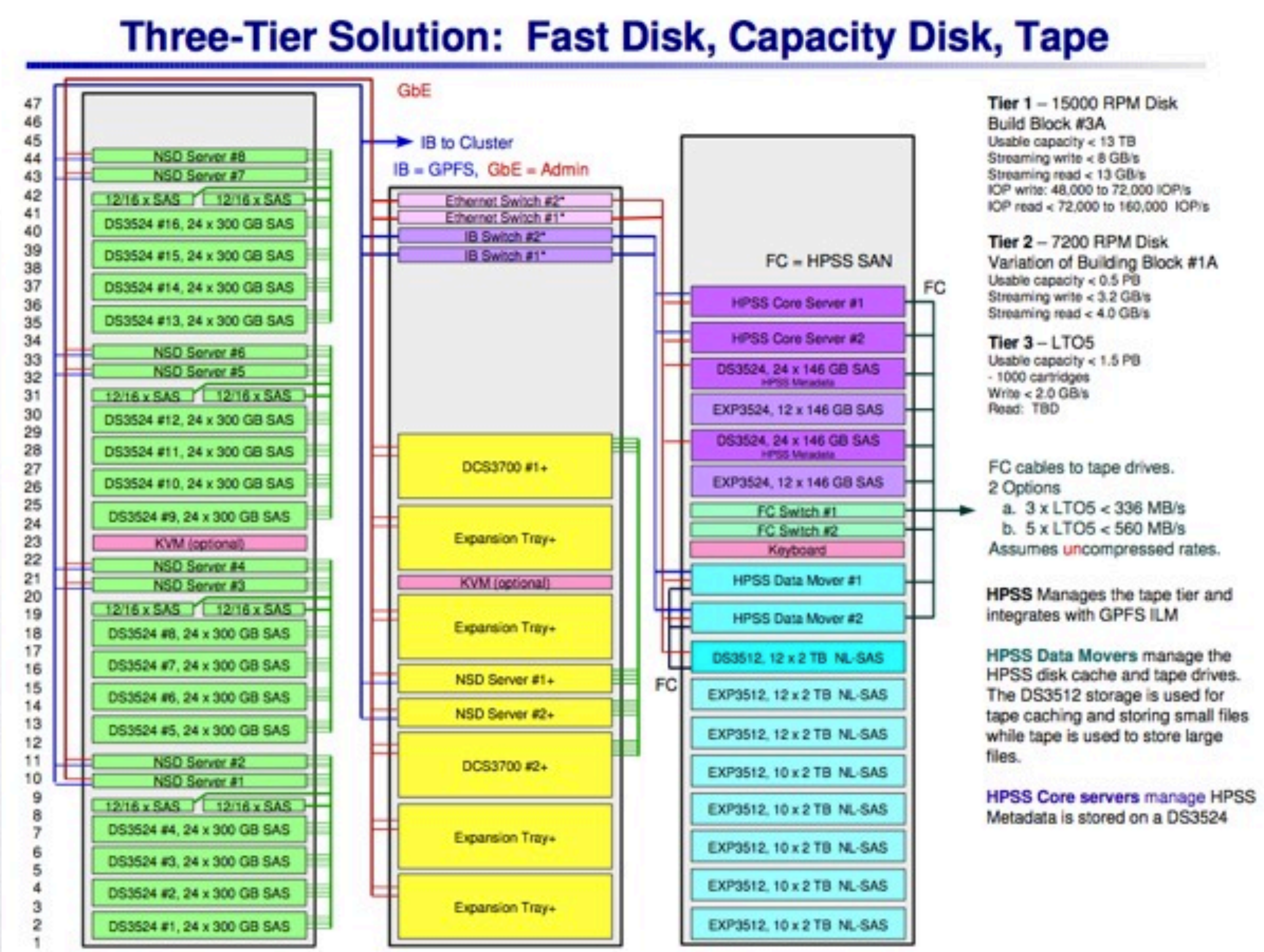
LHC storage demonstrated that HSM is not a perfect solution **random user file recalls**

Large impact for the **OS**, **DB's** and **meta data** stores - not for volume

hides the **tape IO** characteristics.

Cloud storage with **POSIX** API.

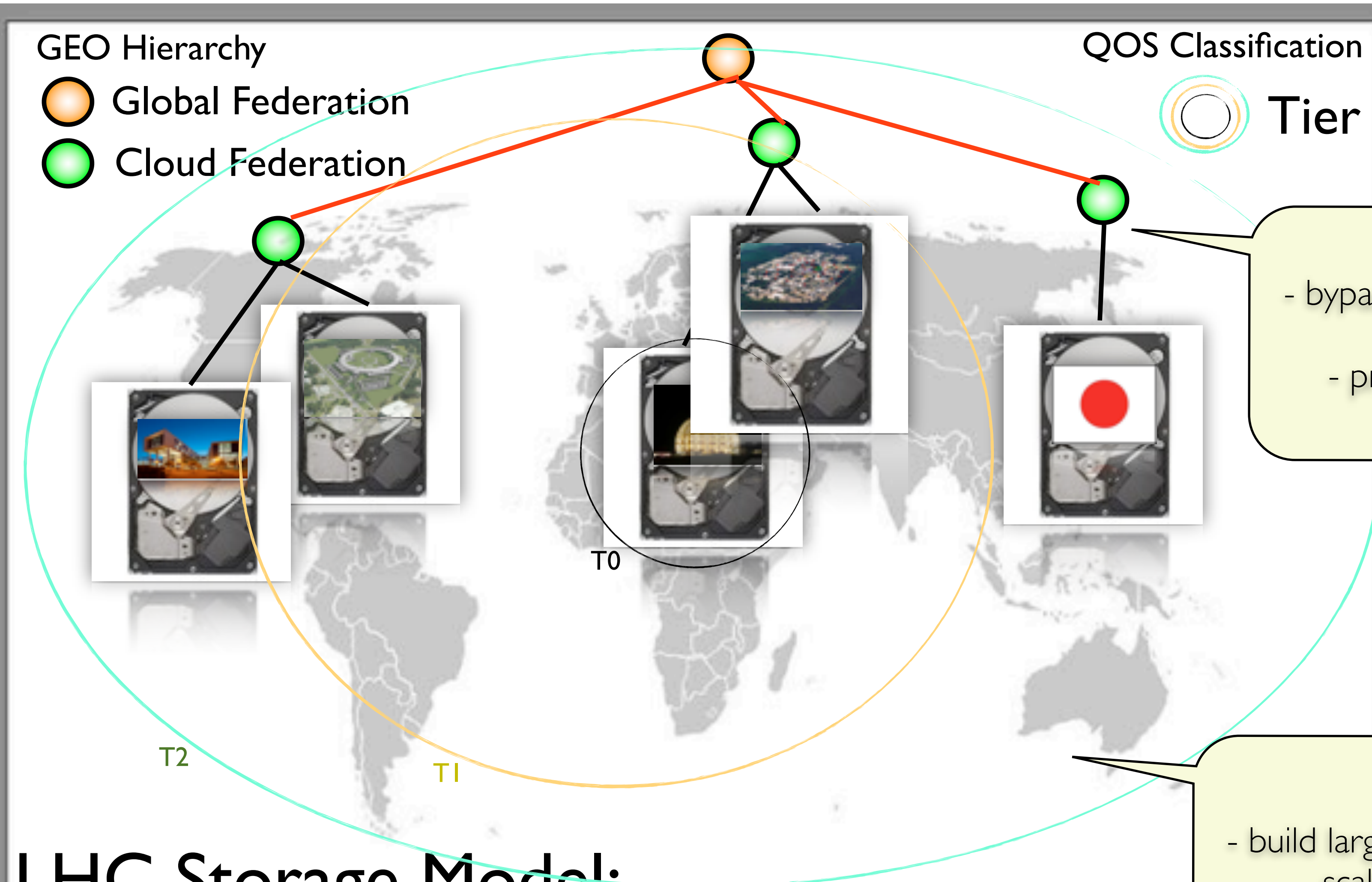
Multi Tier Storage Solutions



- allow to shape/optimize performance
  - e.g. convert stream perf. to IOPS
  - save money having the largest capacity on the cheapest media
- does not work for all work loads
  - the dimension and performance parameters of the tiers must meet usage pattern.
  - otherwise:
    - no guaranteed gain
    - no savings

Promising approach:
combination of cloud storage as a capacity store **+ front-end** with fine-grained and performant IO interface e.g. HSM enabled filesystems, dCache, XRootD FRM

Scalability by Aggregation & Federation



LHC Storage Model:

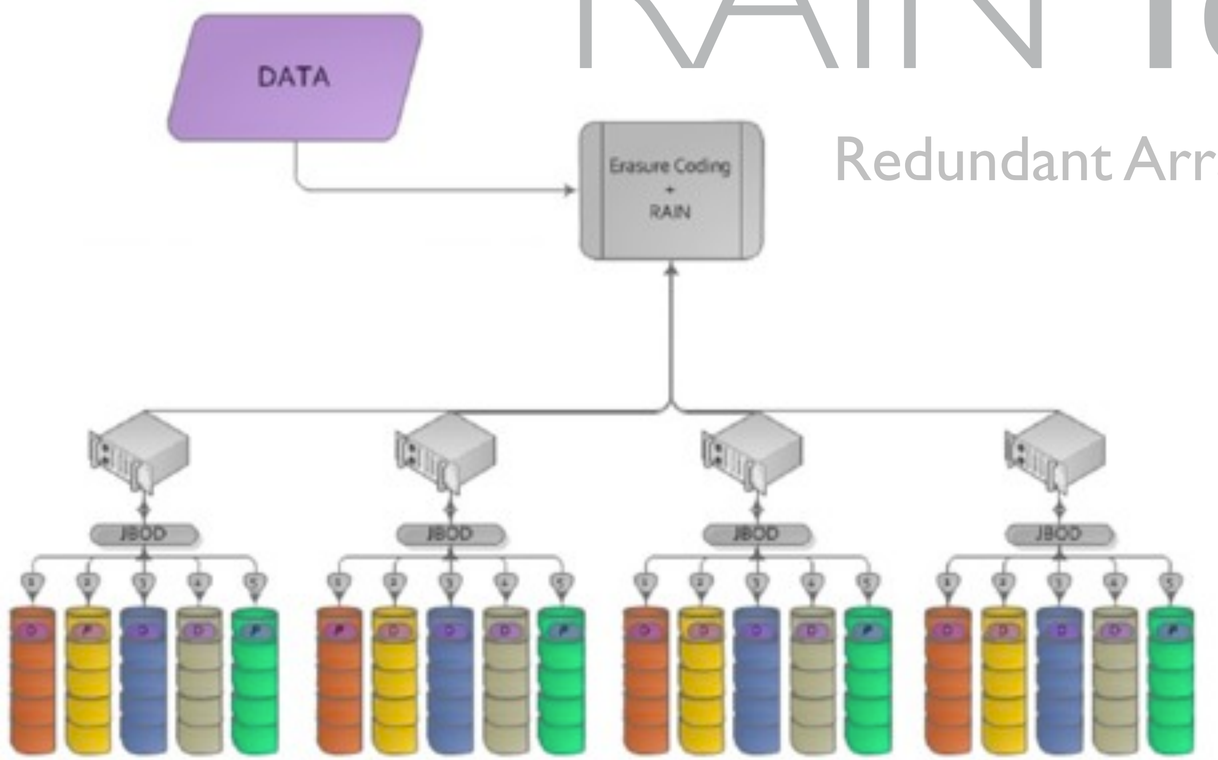
Aggregation of many storage sites into one virtualized storage system.
Requires centralized tracking for placement and location.

Federation
- bypass central aggregation bottleneck for *ro* access
- profit from knowledge of GEO hierarchy

Aggregation
- build large storage system out of limited-scalable storage sub-systems

RAIN Technology

Redundant Array of Inexpensive Nodes



RAID: uses redundancy algorithm on device level

RAIN: uses redundancy algorithm on file/object level distributed over nodes

RAIN combined with block checksumming is a perfect match for very large storage systems

- **space** overhead **configurable** via algorithm e.g. Reed Solomon 10+3 = 30% overhead
- allows to **scale rebuild performance** independently of stripe widths
- allows to **repair** device failures and **silent corruptions**
- allows to **scale IO performance** per object
- **reduces** the **data loss** scenario compared to RAID
- **no** need for hardware RAID **controller** or **multi-path** storage

Available e.g. by NetApp, PanFS, Cleversafe ... GPFS implemented as native RAID (~RAIN) on AIX.

RAIN replaces RAID in large storage systems

Trends & Standards

- GLUSTER & CEPH **are providing an S3 and object store interface**

  Commercial importance of S3!

- LUSTRE & HDFS **work on distributed/ federated namespaces**

  Separating namespaces is still the easiest 'technology' to scale storage.

- pNFS - client in RedHat 6.2
  - **new** NFS v4.2 **standard** coming

    pNFS has not been widely accepted (yet) by the storage industry

    - defines SSC (server side copy), application data blocks, space reservation, sparse files and IO advise, targeting virtualized data centers

Will pNFS **ever displace native clients in** Lustre & GPFS?

Storage Virtualization

Storage as a Service

Virtualization

- **Virtualization of Storage**
  - not new concept
    - Cluster Filesystems use storage virtualization on block, disk, file and filesystem level and virtualization of tape via HSM etc.
    - important for fully virtualized data centers hosting VM and database images
- **Storage in virtualized Environments**
  - run storage system as a virtual appliance in a virtual machine e.g. the GLUSTER Storage Appliance
  - simplifies deployment and configuration 'Storage as a Service'
  - allows on-the-fly deployment of a distributed storage system in cloud environments
  - allows performance confinement within a virtual machine
  - GLUSTER reports only a 5% degradation inside a virtual machine

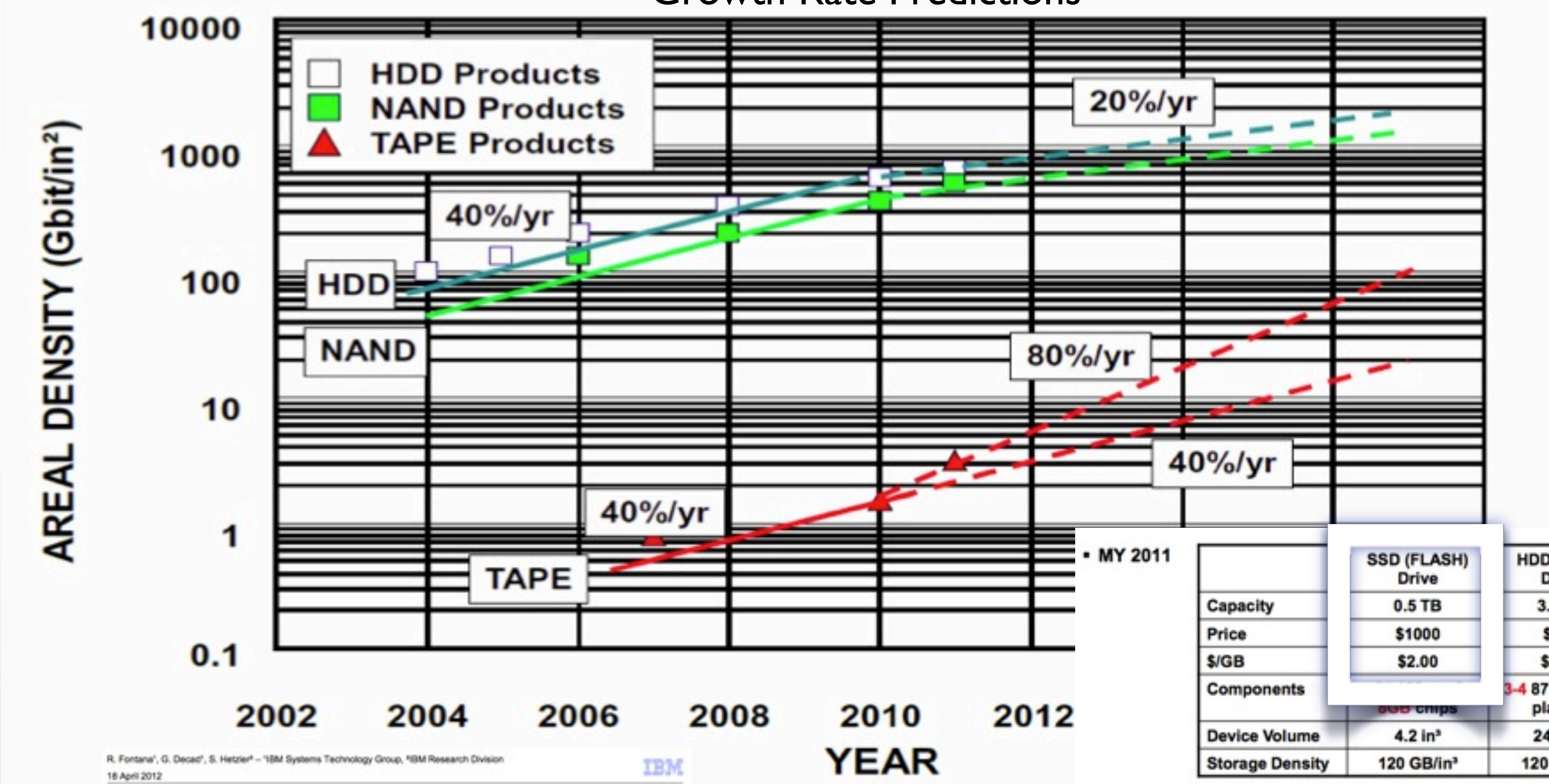
Limitation: Storage is stateful and can not quickly be moved or exchanged once it is filled.

1 PB @ 1GB/s = 11,5 days

Market Trends

Storage Density & Volume Metrics

Growth Rate Predictions



- MY 2011

	SSD (FLASH) Drive	HDD (DISK) Drive	LTO5 (TAPE) Cartridge
Capacity	0.5 TB	3.0 TB	1.5 TB
Price	$1000	$150	$50
$/GB	$2.00	$0.05	$0.03
Components	[illegible] chips	3-4 87 mm disk platters	1 tape cartridge (12.5 mm x 820 m)
Device Volume	4.2 in³	24.2 in³	14.8 in³
Storage Density	120 GB/in³	120 GB/in³	101 GB/in³

2014 Forecast

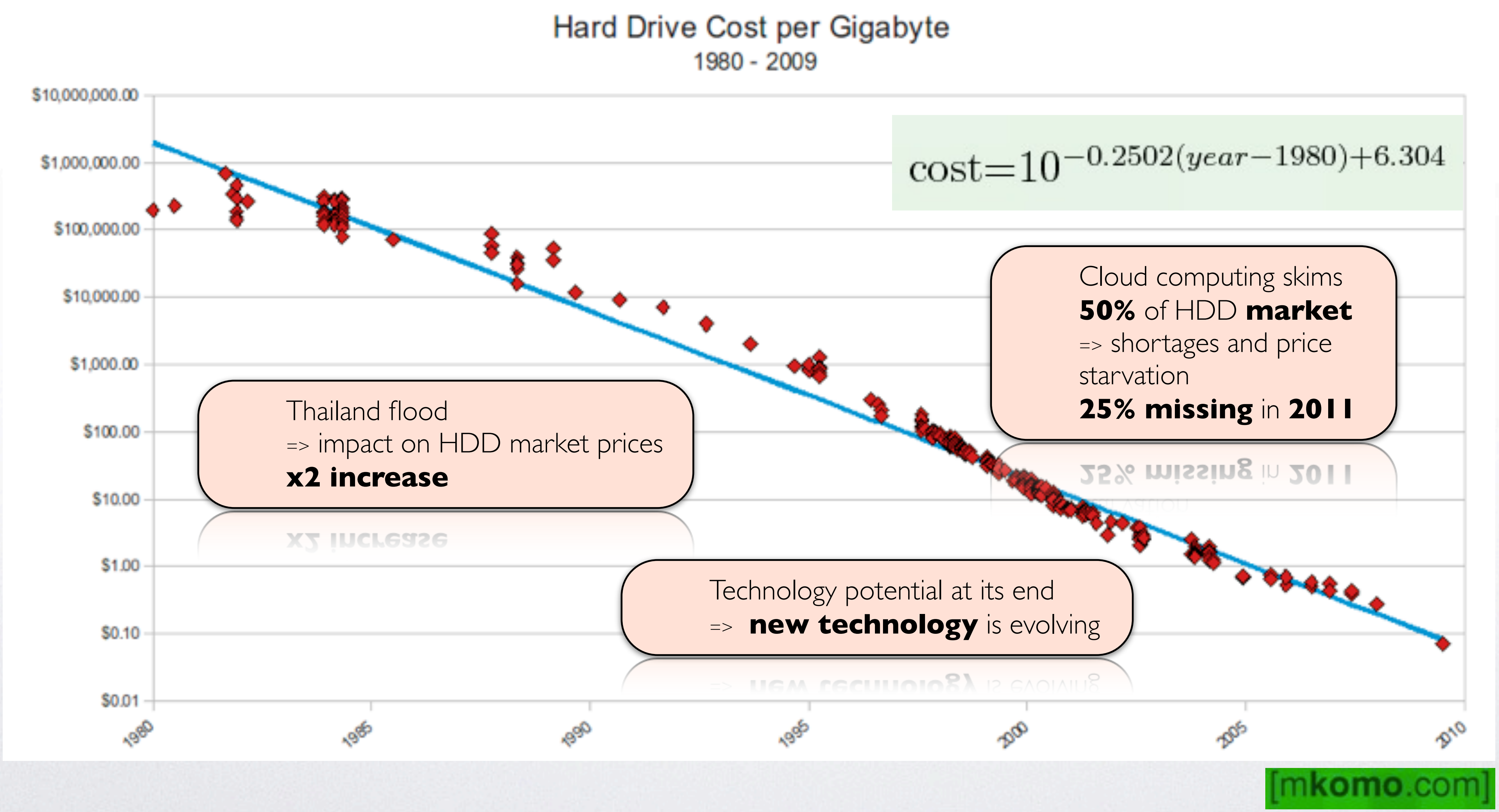
R. Fontana¹, G. Decad¹, S. Hetzler² – ¹IBM Systems Technology Group, ²IBM Research Division
18 April 2012
IBM

Scenarios for 2014

Areal Density Growth (Specifics)	Historical 40%/yr--TAPE 40%/yr--HDD 40%/yr--NAND	Conservative 40%/yr--TAPE 20%/yr--HDD 20%/yr--NAND	Tape Aggressive 80%/yr--TAPE 20%/yr--HDD 20%/yr--NAND
TAPE			
-- Areal Density	4.8 Gbit/in²	4.8 Gbit/in²	12.0 Gbit/in²
-- Minimum Feature	[illegible]	[illegible]	[illegible]
-- Cartridge Capacity	6.0 TB	6.0 TB	15. 0 TB
-- Volumetric Density	[illegible]	[illegible]	[illegible]
HDD			
-- Areal Density	2500 Gbit/in²	1300 Gbit/in²	1300 Gbit/in²
-- Minimum Feature	0.010 um	0.018 um	0.018 um
-- HDD Capacity[1]	12.0 TB	6.0 TB	6.0 TB
-- Volumetric Density	480 GB/in³	240 GB/in³	240 GB/in³
NAND Flash			
-- Areal Density	1300 Gbit/in²	700 Gbit/in²	700 Gbit/in²
-- Minimum Feature	0.012 um	0.016 um	0.016 um
-- Chip Capacity	32 GB	24 GB	24 GB
-- SSD Capacity[2]	2 TB	1.2 TB	1.2 TB
-- Volumetric Density	480 GB/in³	300 GB/in³	300 GB/in³

Tape growth rate **2-4x** HDD & NAND

Consumer HDD Prices



Cloud computing skims **50%** of HDD **market** => shortages and price starvation **25% missing** in **2011**

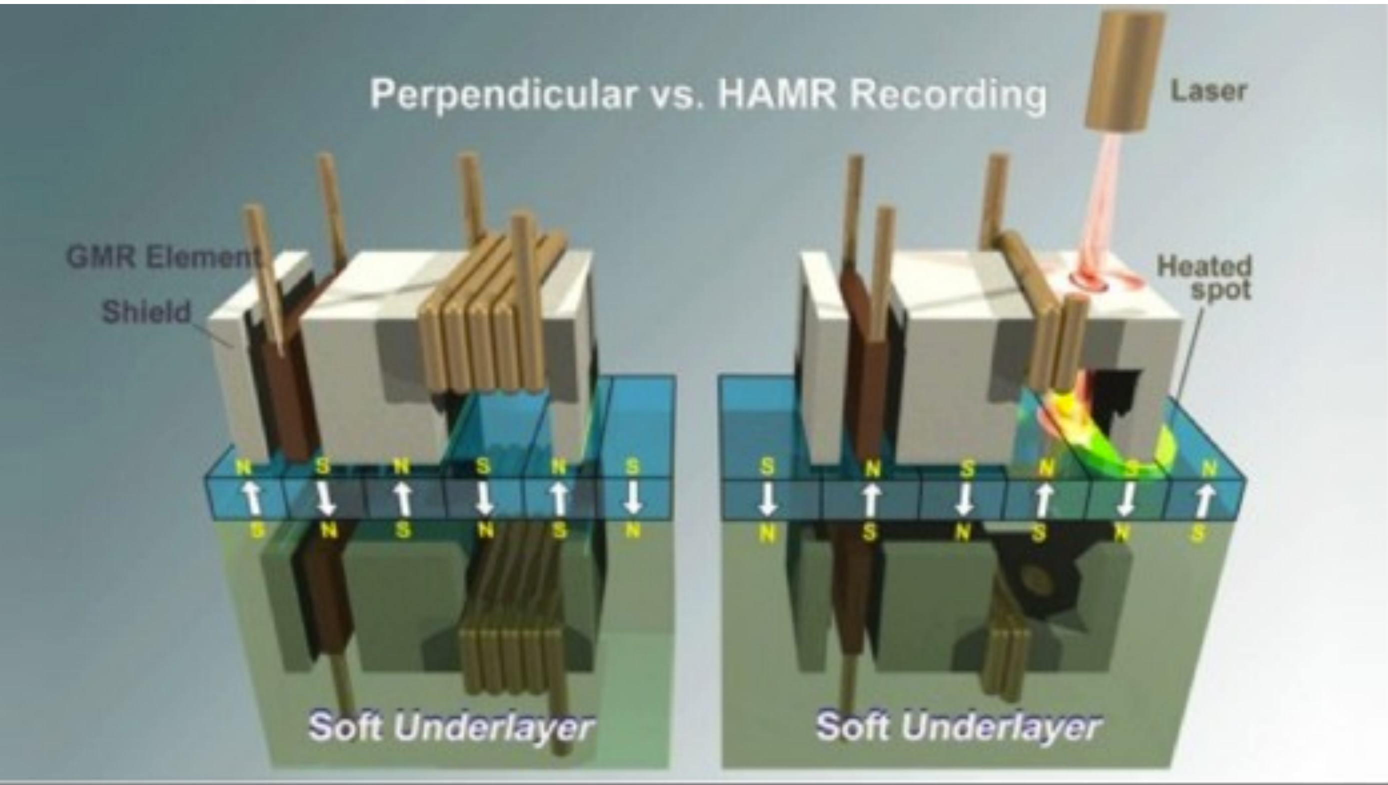
Thailand flood => impact on HDD market prices **x2 increase**

Technology potential at its end => **new technology** is evolving

May 2012 0.05 $/GB

New HDD Technology

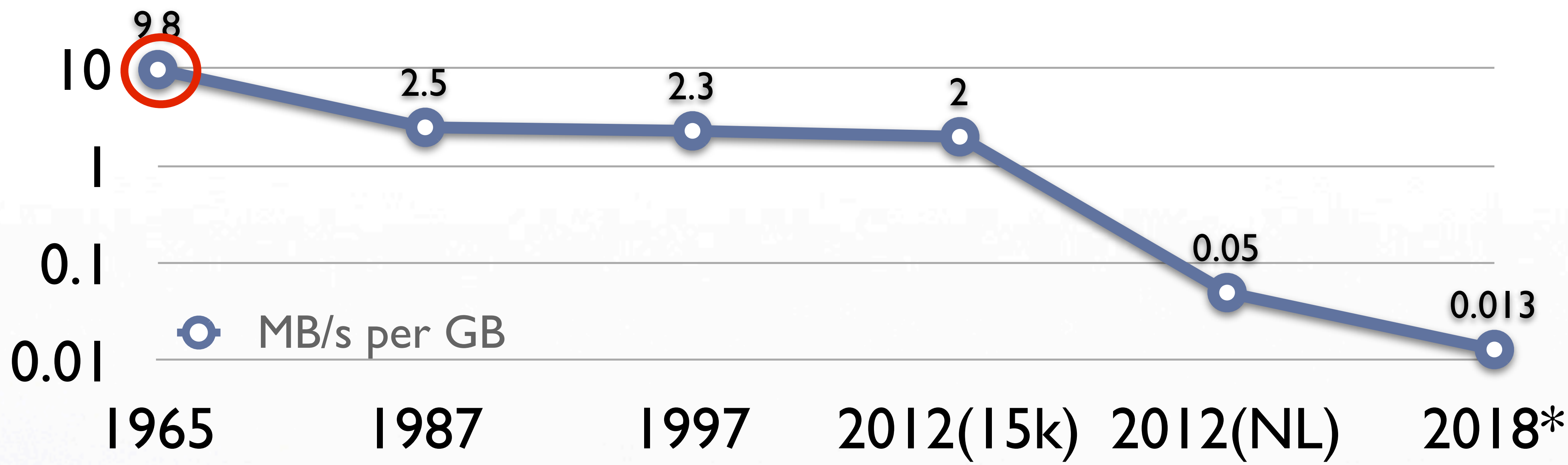
Heat Assisted Magnetic Recording



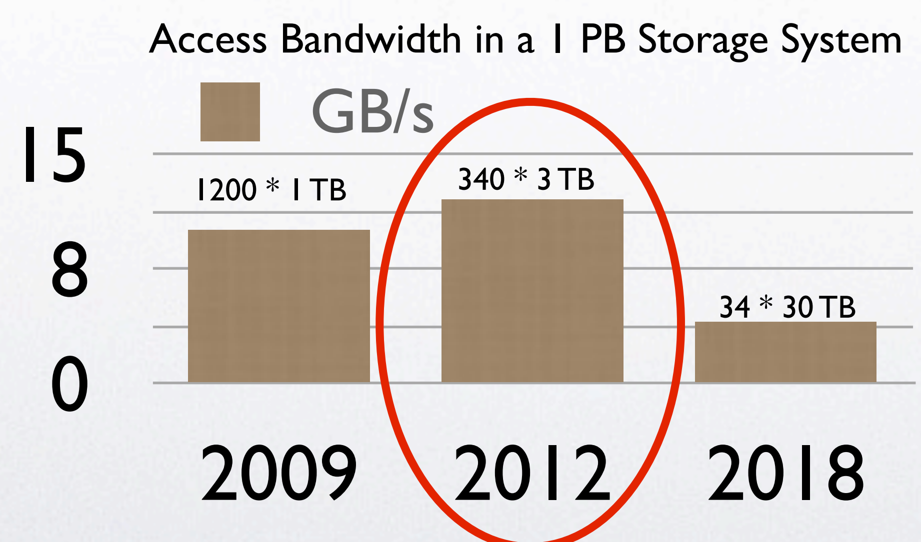
Seagate has achieved a milestone 1 terabit per square inch storage density using heat-assisted magnetic recording (HAMR) technology

- *HAMR* limit is 5-10 Tbit/in^2
~30-60 TB 3.5" HDD
- large production level 2016-2017
- expensive technology
- areal growth rate lower ~20-25%

HDD Performance Capacity Ratio



Assumption:
2018 30 TB HDD @ 380 MB/s

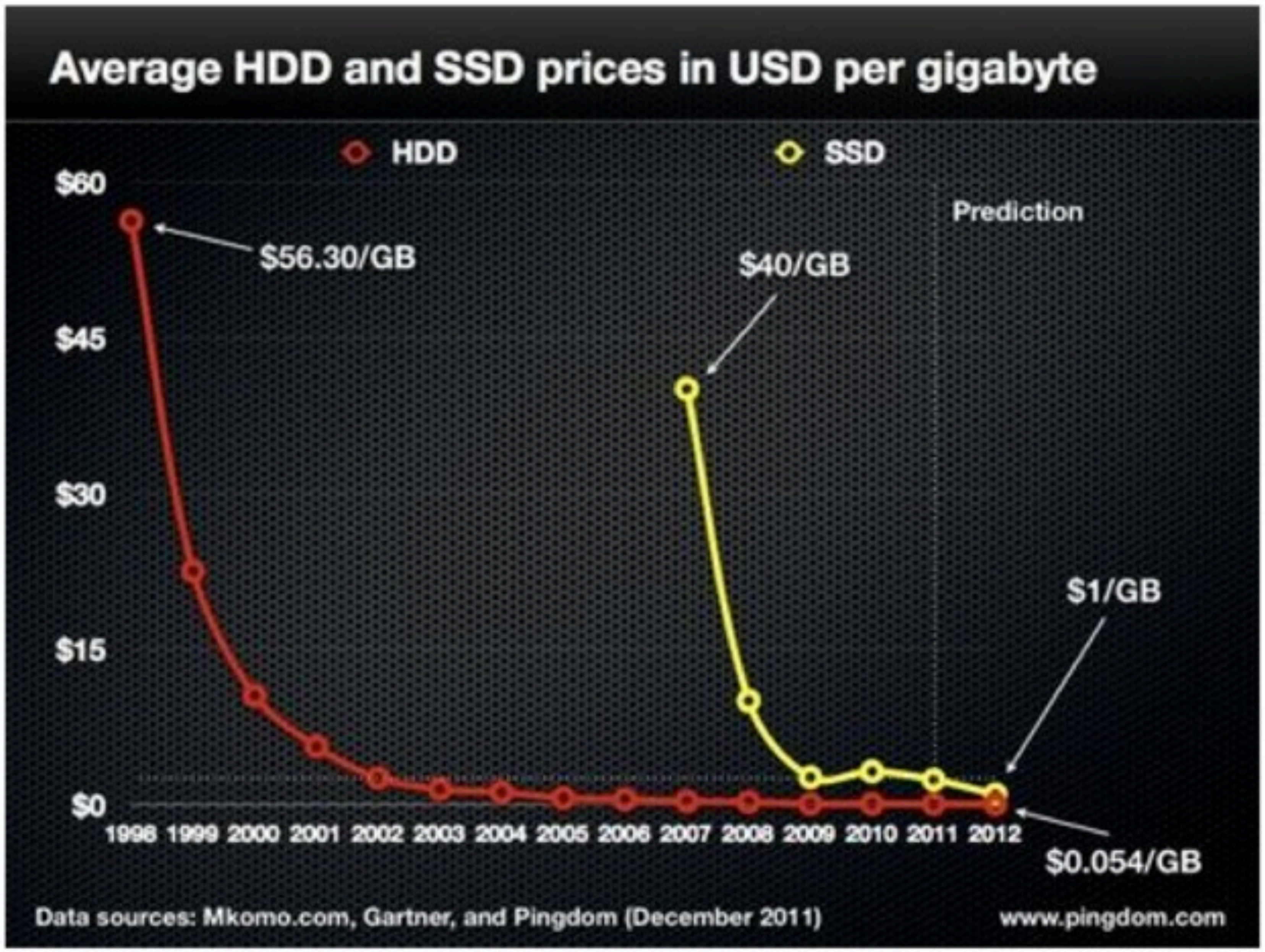


* Gary Grider, Exa-Scale FSIO, 07/2010, LANL.

Source: Raymond L. Paden, Ph.D. - IBM Deep Computing - 28th IEEE Conference on Massive Data Storage

Increasing **capacity reduces bandwidth** to data! Critical crossover point?

SSD


Average HDD and SSD prices in USD per gigabyte
HDD
SSD
Prediction
$60
$45
$30
$15
$0
$56.30/GB
$40/GB
$1/GB
$0.054/GB
1998 1999 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012
Data sources: Mkomo.com, Gartner, and Pingdom (December 2011)
www.pingdom.com

SSD

Examples

Type	Capacity	Streaming Rate r/w	IOPS r/w
Controller (Rack of RamSan-820)	1 PB	168 GB/s	18.9M
Block Device	1 TB	500/380 MB/s	15-35k
PCIe	1.2 TB (ioDrive2) 12/16 TB (Z-Drive)	1.5/1.3 GB/s 7.2 GB/s	500k/140k 2.5M

5.1.2012 "Fusion-io Breaks One Billion IOPS Barrier"

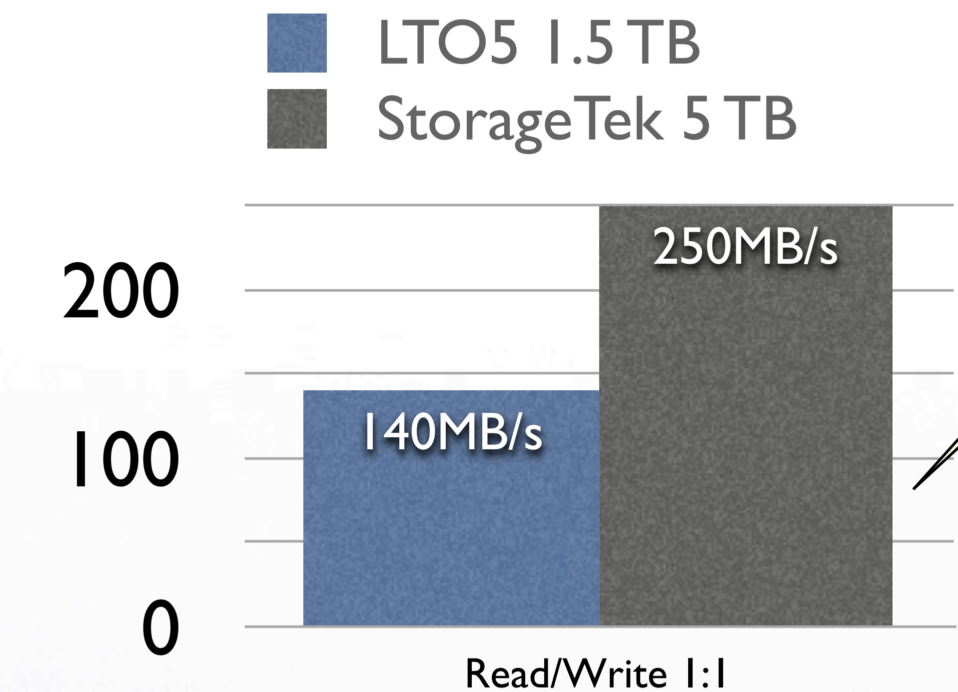
SSDs can **deliver IOPS & bandwidth** *en masse*

Interesting for **LSS** to **handle meta data** work loads

Tape

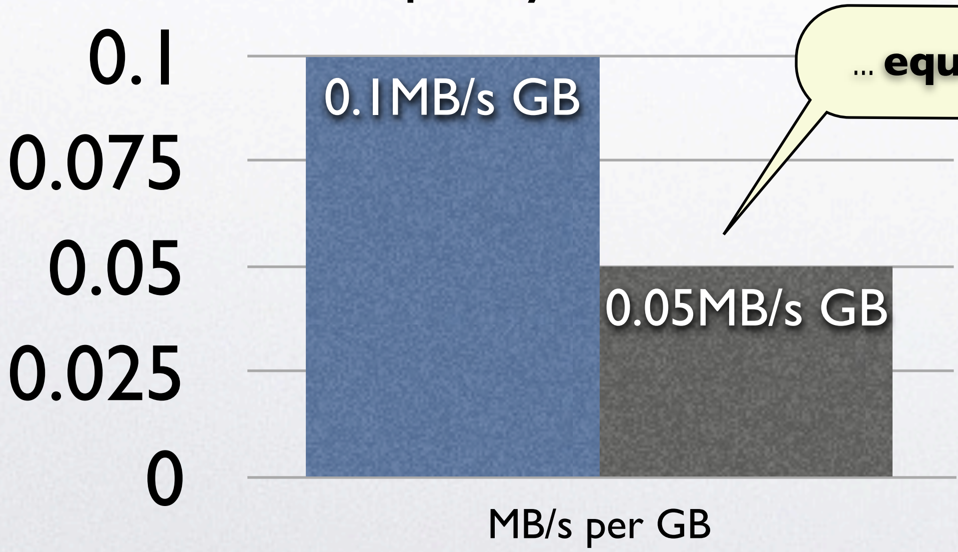
... **alive** ... market is growing!

Streaming Performance

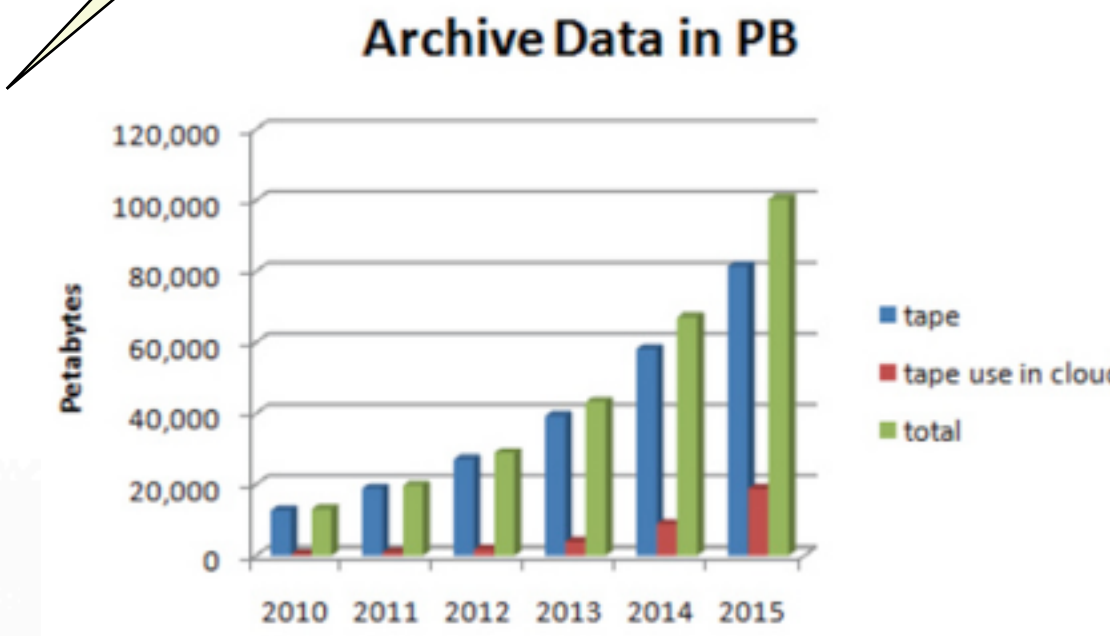


... **faster** than disks ...

Performance:Capacity Ratio



... **equal/better** than NL disks ...



Source: Enterprise Strategy Group

	Generation 1	Generation 2	Generation 3	Generation 4	Generation 5	eration 6	Generation 7	Generation 8
Compressed Capacity	200 GB	400 GB	800 GB	1.6 TB	3 TB	B	16 TB	32 TB
Native Capacity	100 GB	200 GB	400 GB	800 GB	1.5 TB	TB	6.4 TB	12.8 TB
Compressed Data Rate	up to 40 MB/s	up to 80 MB/s	up to 160 MB/s	up to 240 M	up to 280 MB/s	o 525 MB/s	up to 788 MB/s	up to 1180 MB/s
Native Data Rate	up to 20 MB/s	up to 40 MB/s	up to 80 MB/s	up to 120 M	up to 140 MB/s	o 210 MB/s	up to 315 MB/s	up to 472 MB/s

Note: Compressed capacities for generations 1-5 assume 2:1 compression. Compressed capacities for generations [illegible] (achieved with larger compression history buffer).
Source: The LTO Program. The LTO Ultrium roadmap is subject to change without notice and represents goals and objectives only.
Linear Tape-Open, LTO, the LTO logo, Ultrium, and the Ultrium logo are registered trademarks of HP, IBM and Quantum in the US and other countries.

Tape most **reliable and cheapest** volume storage for archiving and backup **in future projections**

Storage Connectivity

Infiniband

	SDR	DDR	QDR	FDR-10	FDR	EDR
Year	1999	2004	2008		2011	[illegible]
4X	**8 Gbit/s**	**16 Gbit/s**	**32 Gbit/s**	**41.2 Gbit/s**	**54.54 Gbit/s**	**100 Gbit/s**
12X	24 Gbit/s	48 Gbit/s	96 Gbit/s	123.6 Gbit/s	163.64 Gbit/s	[illegible]
1X	2 Gbit/s	4 Gbit/s	8 Gbit/s	10.3 Gbit/s	13.64 Gbit/s	25 Gbit/s

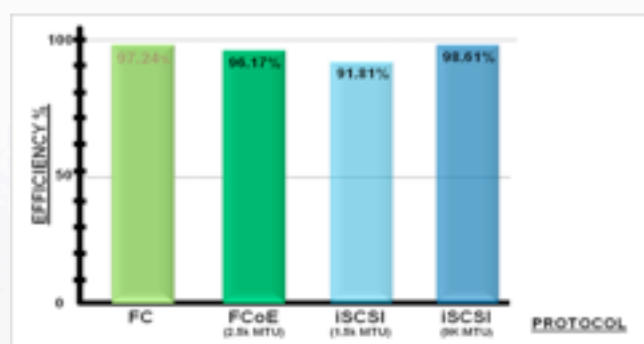
SAS

Year	**Gbit/s**
2004	3
2009	6
2013 (?)	**12**

Fibre Channel

ME	**Gbit/s full duplex**	**Availability**
1GFC	1.6	1997
2GFC	3.2	2001
4GFC	6.4	2004
8GFC	12.8	2005
10GFC Serial	20.4	2008
16GFC	25.6	2011
20GFC	**40.8**	20??

iSCSI/iSCSIoE



Ethernet

1 Gbit	1999
10 Gbit	2002
40/100 Gbit	2009/?

PCIe

2003 1.0a	2 Gbit/s
2005 1.1	2 Gbit/s
2007 2.0	4 Gbit/s
2010 3.0	8 Gbit/s
? 4.0	16 Gbit/s

Standard: x1,x4,x8,16,x32 lanes
Common: x1, x4, x8 cards

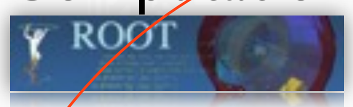
Connectivity **doubles** ...

"Storage Biosphere"

What can **we** learn from Internet Storage?

Environment

Computation



Storage

- look at large storage in its context
  - storage is serving applications, not driving them
    - (LHC) storage is used in a distributed environment implementing a lot of functionality on a higher level

Web Revolution:
Redesign storage **around** their **applications**

Don't stick to standards and/or products!

Set or choose your **standard** according to **requirements** and use it consistently everywhere!

What we should think about:
1 Unification of Storage, Computing and Network
2 Appliances for storage and computing
3 Rethink tier/replication model
4 Move redundancy into the application layer instead of storage systems?
5 Base distributed storage on object stores?

Future

Importance of storage systems will increase in the future

- internet + emersion of mobile devices drives **unseen growth of storage** needs => huge market implications + technology push
  - chance to **profit** from and **contribute** to large community projects
  - commercial solutions follow market demands => options not only for HPC
  - extreme large scale systems based on **elastic object store** in combination with **elastic databases** providing meta data views
    - LHC storage approach is compatible - useful to adopt big data technology
    - over time LHC storage might leave 'comfort zone' where things still scale easily with used technology
  - **Exabyte** storage for big data mining will become a **new norm** within years

Thank you for your attention!