BigData Storage Systems

ICiS 2011: Large Datasets in Astrophysics and Cosmology

Park City, Utah, August 11, 2011

Garth Gibson

Co-Founder and CTO, Panasas Inc.

Professor, School of Computer Science, Carnegie Mellon University

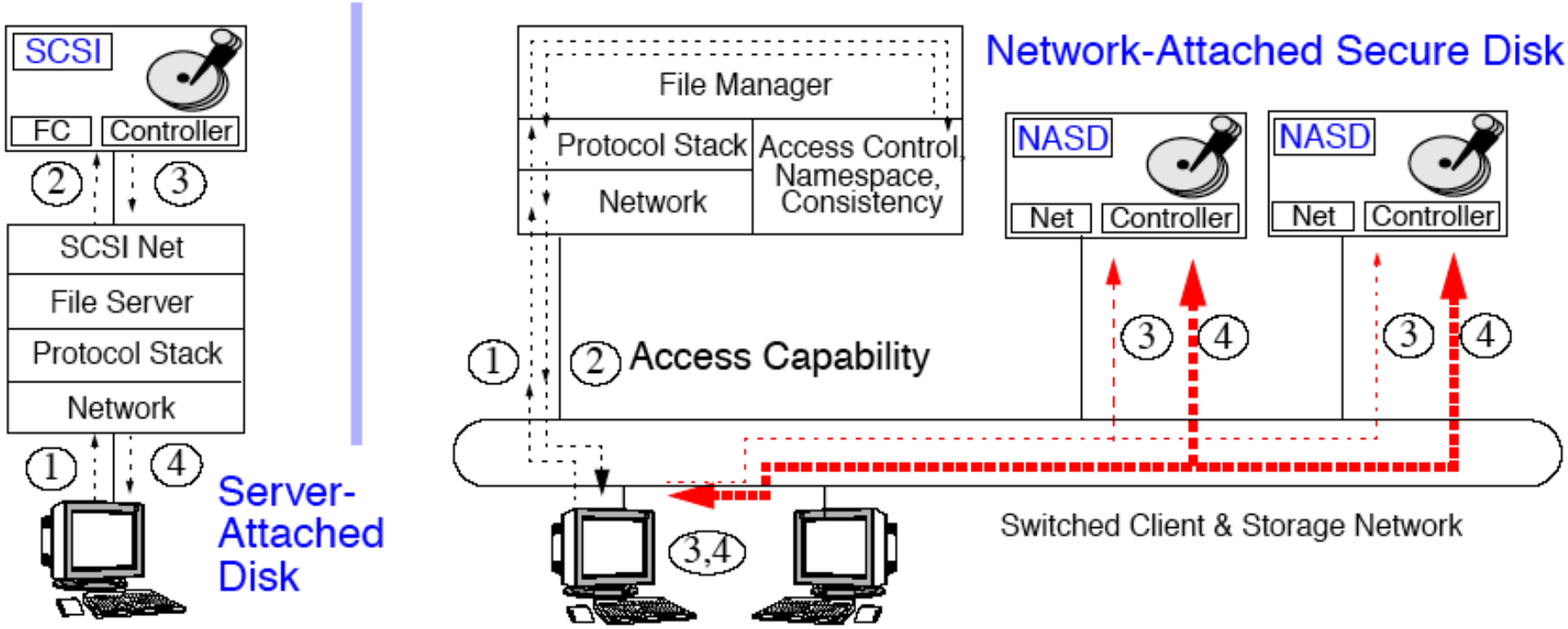
Agenda

- A little history & credentials
- Exascale storage systems
- Technology trends in flash
- Technology trends in magnetic disks
- Transparent middleware for fast N-1 checkpointing
- Astro-inspired work: DB-in-FS
- Google-inspired Data-intensive Scalable Computing
- AstroInformatics: Getting out hands dirty
- AstroInformatics: Reduce the price of Astro-scale data

A Little History & Credentials

Object Storage (DARPA NASD, 95-99)

Before NASD there was store&forward Server-Attached Disks (SAD)
Move access control, consistency out-of-band and cache decisions
Raise storage abstraction: encapsulate layout, offload data access
- Now ANSI T10 SCSI command set standard (OSD 2004, 2009)
- Architecture influenced Lustre, PanFS, GoogleFS, pNFS

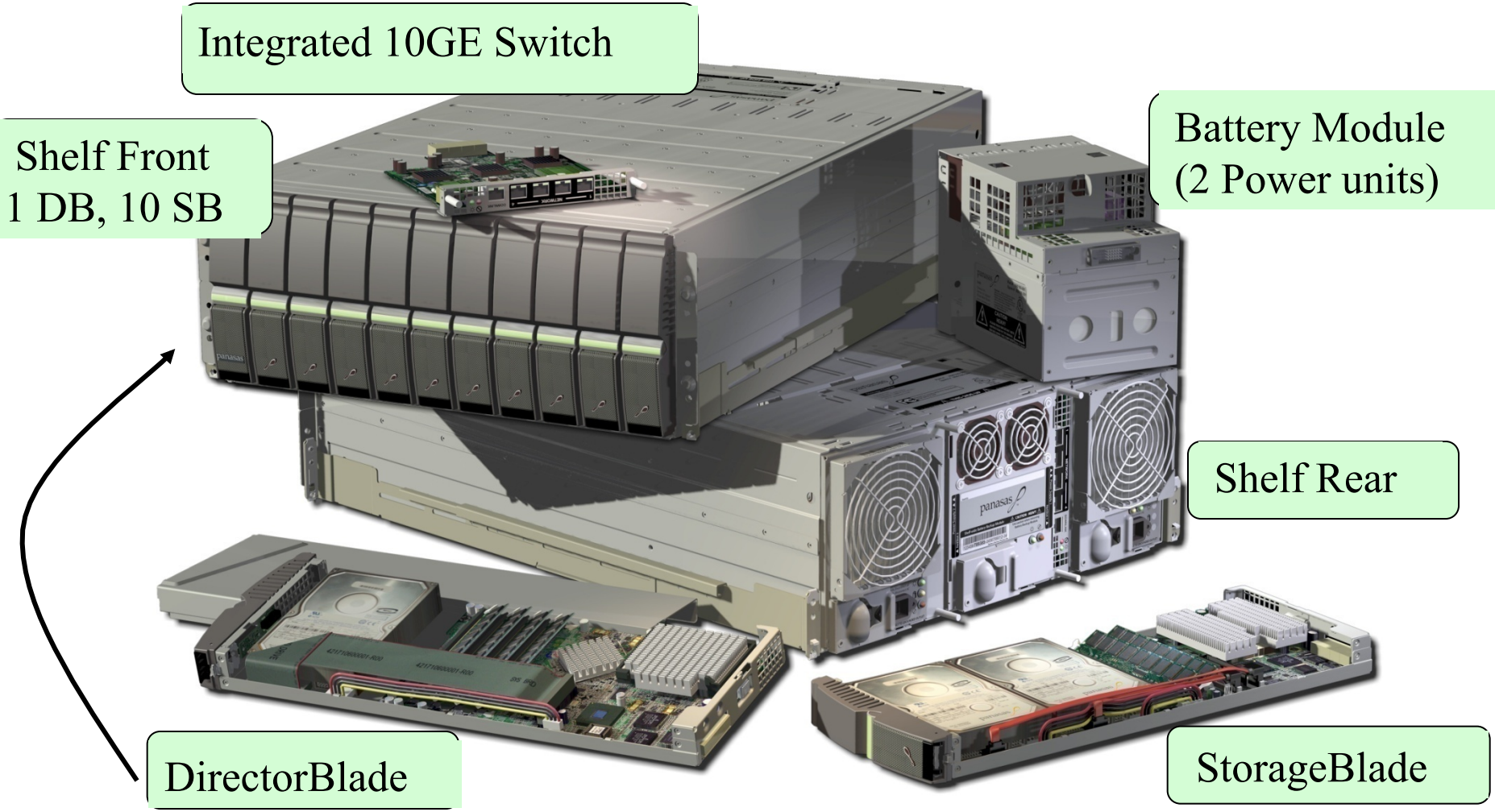


Panasas Spins Out of CMU (1999)

- Storage that accelerates HPC apps
  - 10,000+ clients, 50+ GB/s, 1,000+ storage nodes
  - Primary storage on 1st petascale computer (Los Alamos)
- Founded 1999, shipping solutions since 2003
  - Software innovation, packaged with industry-standard HW
  - Scalable RAID over storage nodes, end-to-end check codes
  - Integrated SSD, extensive HA, snapshot, async mirroring

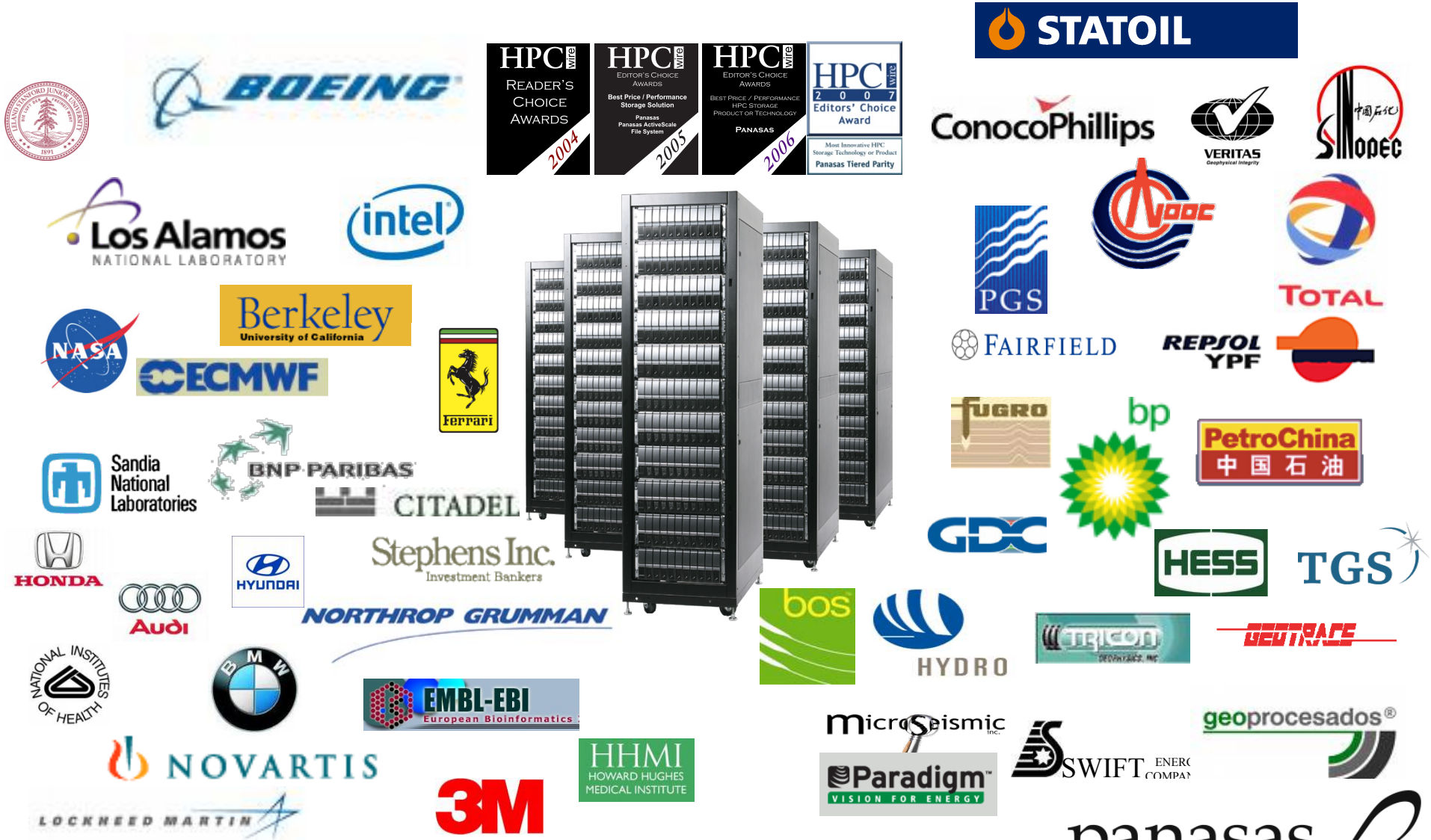
panasas®

Panasas Storage Cluster of Bladeservers



Midplane routes GE, power

Leaders in HPC choose Panasas



PanFS Scales to Petascale for LANL

- Lightning (2004)
  - 2,800 cores across 1,400 nodes
  - 3 GB/sec I/O Bandwidth, 60 TB
- RoadRunner (2009)
  - 12,240 cores (+Cells), 3,060 nodes
    - First Petaflop machine on top500
  - 55+ GB/sec I/O bandwidth, 3 PB
- Cielo (2011)
  - >100,000 cores across 7,000 (growing to 9,000 nodes)
  - >100 GB/sec I/O bandwidth, 10 PB

~ 80%/year BW growth, ~ 100%/year Capacity growth

Exascale storage systems: still all about checkpoint/restart

SciDAC Petascale Data Storage Institute

- Eight organizations 2006-2010
  - Carnegie Mellon University, Garth Gibson, PI
  - U. of California, Santa Cruz, Darrell Long
  - U. of Michigan, Ann Arbor, Peter Honeyman
  - Lawrence Berkeley Nat. Lab, John Shalf
  - Oak Ridge National Lab, Phil Roth
  - Pacific Northwest National Lab, Evan Felix
  - Los Alamos National Lab, Gary Grider
  - Sandia National Lab, Lee Ward

www.pdl.cmu.edu

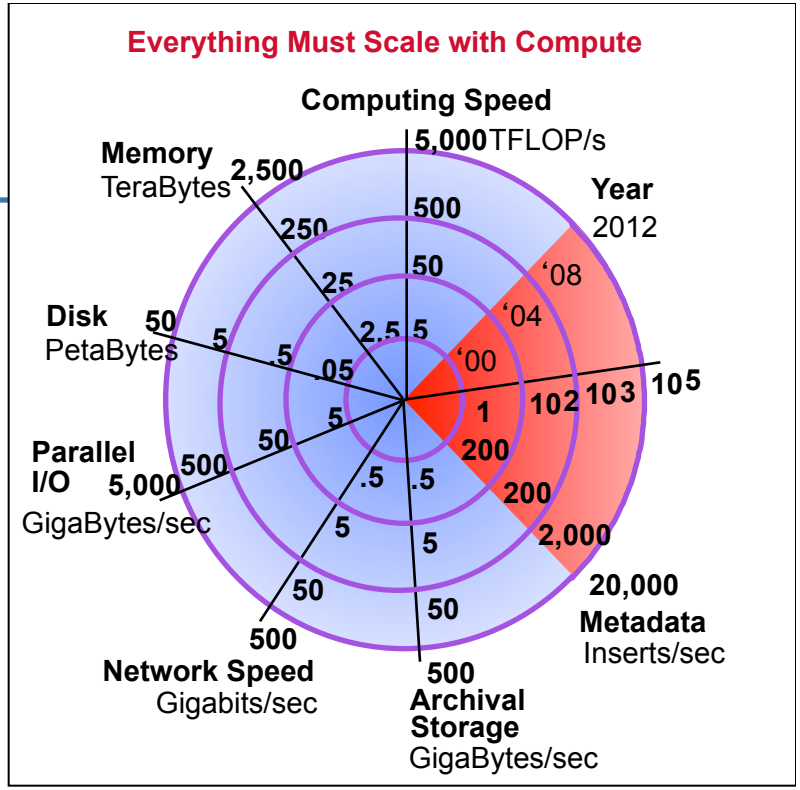
Reaching for Exascale

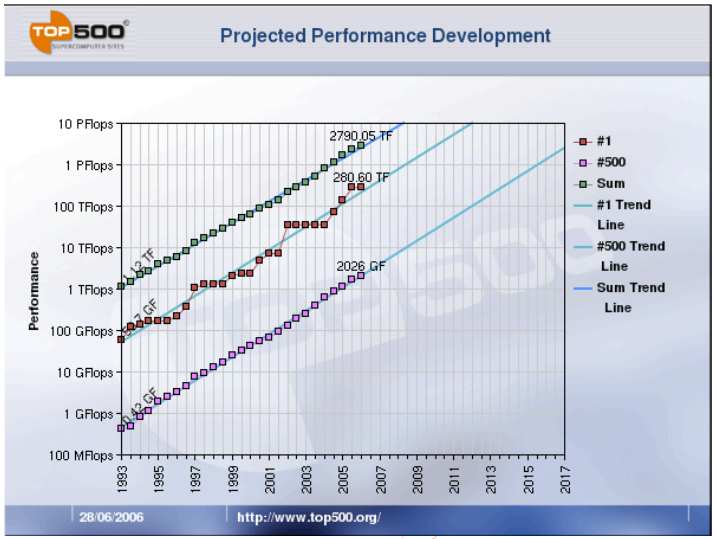
Systems	2009	2015 +1/-0	2018 +1/-0
System peak	**2 Peta**	**100-300 Peta**	**1 Exa**
Power	**6 MW**	**~15 MW**	**~20 MW**
System memory	0.3 PB	5 PB	64 PB (+)
Node performance	125 GF	0.5 TF or 7 TF	2 TF or 10TF
Node memory BW	25 GB/s	0.2TB/s or 0.5TB/s	0.4TB/s or 1TB/s
Node concurrency	12	O(100)	O(1k) or 10k
Total Node Interconnect BW	3.5 GB/s	100-200 GB/s 10:1 vs memory bandwidth 2:1 alternative	200-400GB/s (1:4 or 1:8 from memory BW)
System size (nodes)	18,700	50,000 or 500,000	O(100,000) or O(1M)
Total concurrency	225,000	O(100,000,000) *O(10)-O(50) to hide latency	O(billion) * O(10) to O (100) for latency hiding
Storage	15 PB	150 PB	500-1000 PB (>10x system memory is min)
IO	0.2 TB	10 TB/s	60 TB/s (how long to drain the machine)
MTTI	days	O(1day)	O(1 day)

John Shalf. FSIO, Aug 2010

Storage & Fault Tolerance

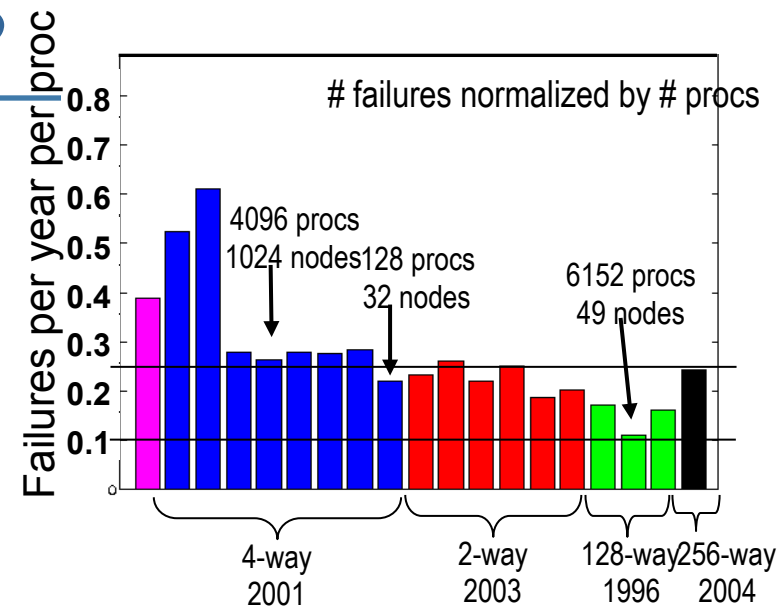
- Checkpoint-restart is key to HPC fault tolerance
- Balanced systems design
  - Linear growth of memory & storage with compute
    - Checkpoint overhead constant
- Top500.org: 100% CAGR → storage BW growth
  - Supporting rapidly growing need for BW drives HPC storage
  - But disk BW has 20% CAGR
  - So # disks grows at 65% CAGR
  - Faster than node count growth → Cost unacceptable to most

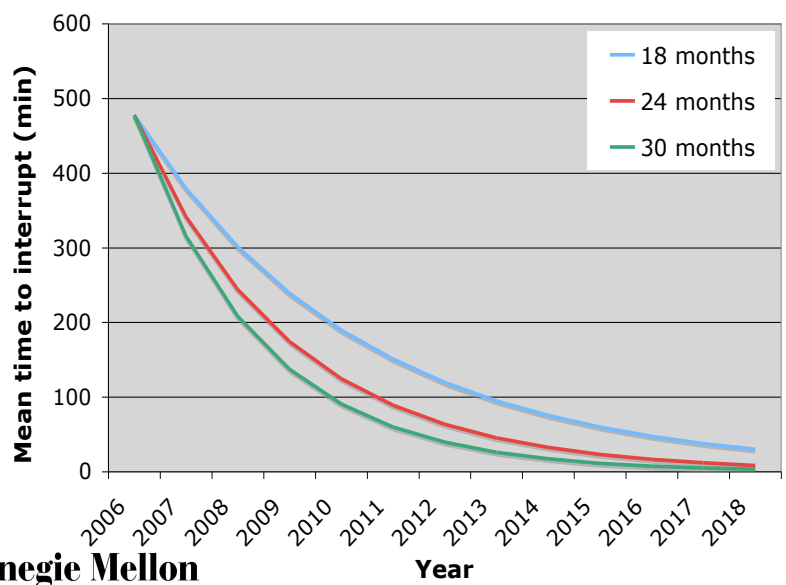


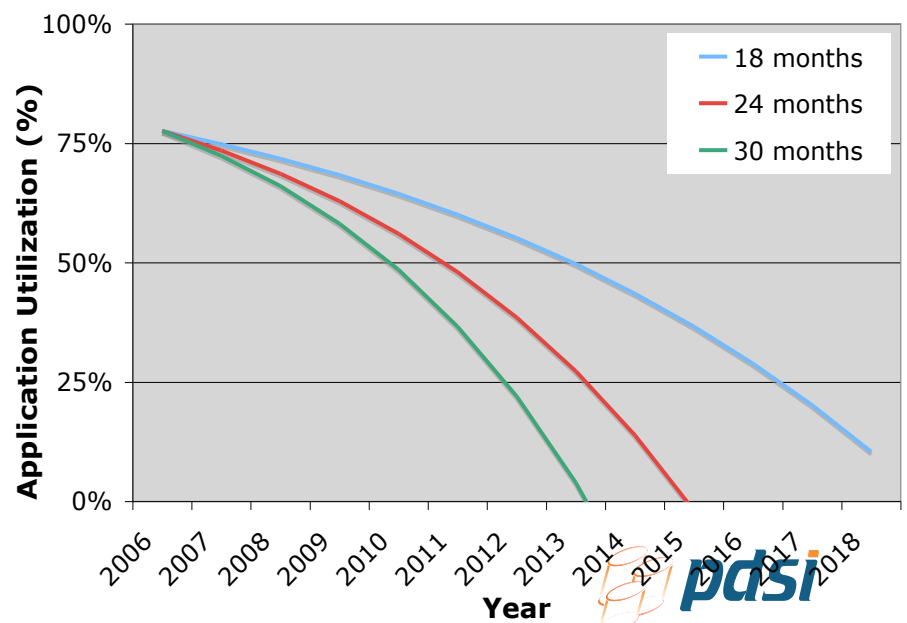


What if MTTI Dropping?

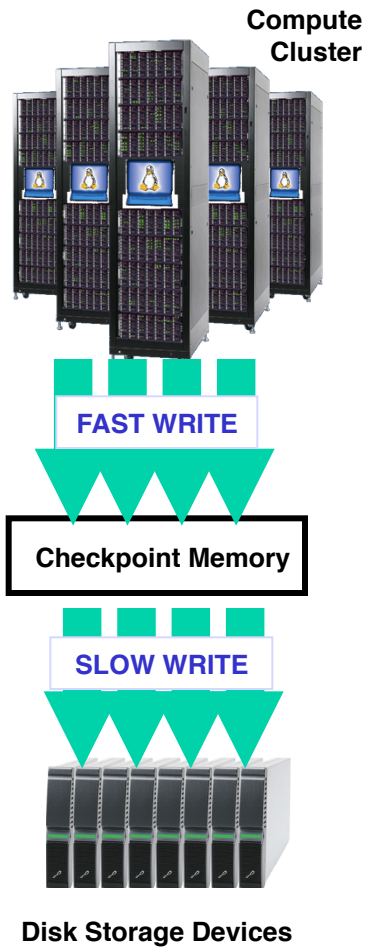
- Parts count will grow, will MTTI drop?
- History says 1/MTTI linear with # parts
- 1 – AppUtilization = t/p + p/(2*MTTI)
  Depends on memory capture time, t,
  Checkpoint interval, p, p^2 = 2*t*MTTI
- So more parts means
  less effective use of big machines



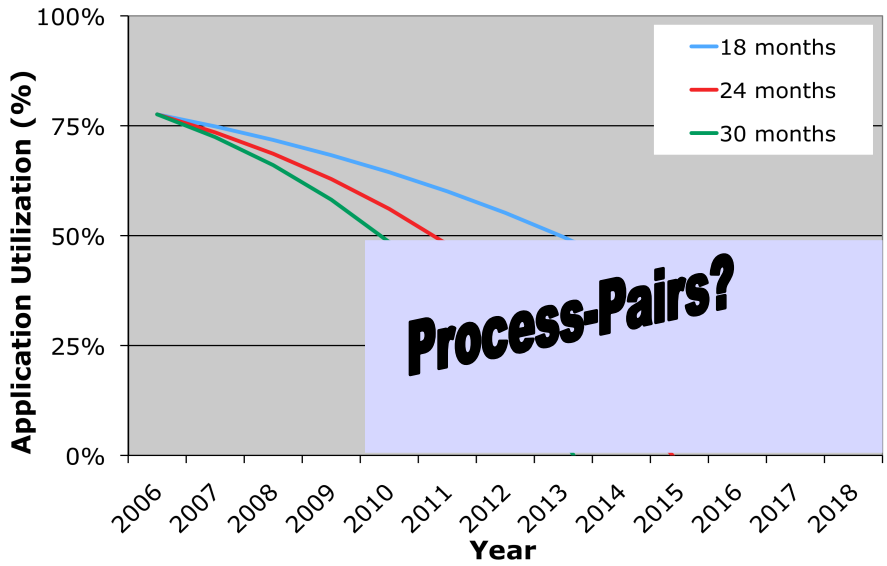




Different approaches needed for 2018



- Dedicated checkpoint device (The Plan)
  - Stage checkpoint thru fast memory (NAND flash)
  - Background write to storage at 10% BW
  - Cost of dedicated flash expected to double storage
  - One time win; component count growth continues
- Classic enterprise process pairs duplication
  - Flat 50% efficiency cost, plus message duplication

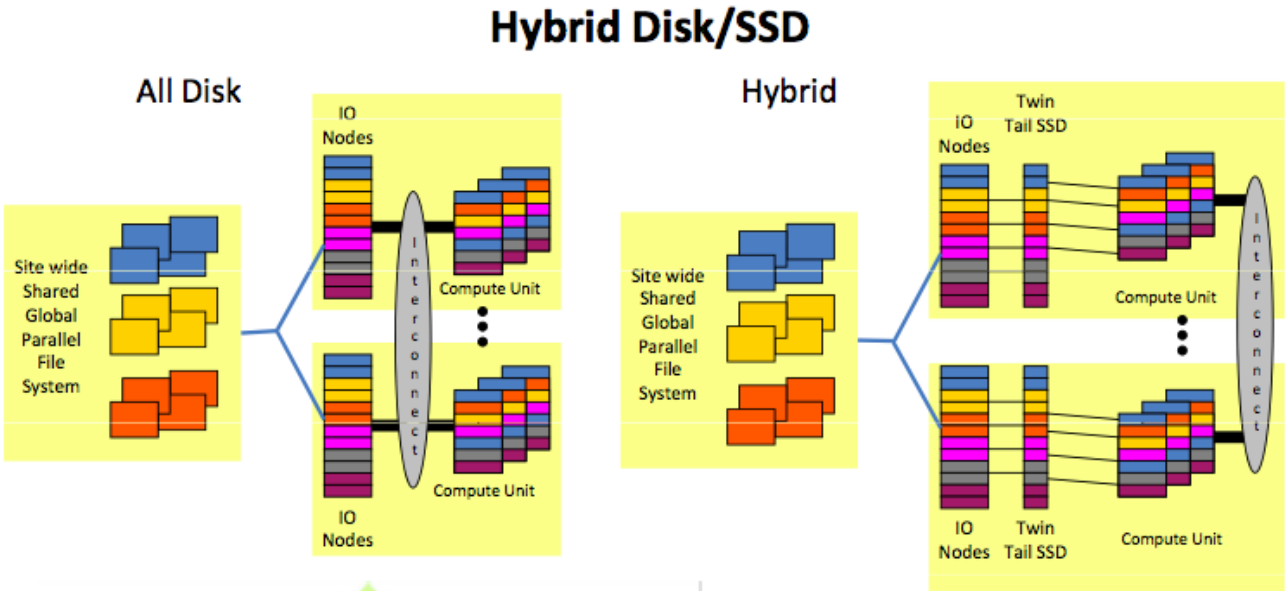


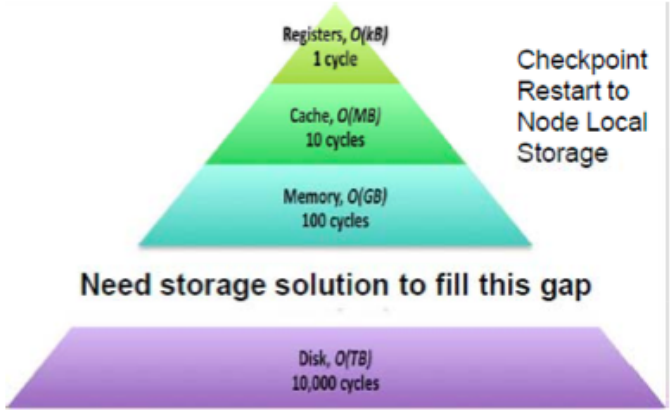
Exascale Storage Bandwidth (Checkpoint/Restart Cluster Fault Tolerance)

G. Grider, FSIO10

- 600PB & 66TB/s
- All disk: $225M
- All SSD: $625M
- Hybrid: $60M

- Double buffer strategy
- Disk BW is 10% of SSD BW
- Disk 100% busy
- 20K disks





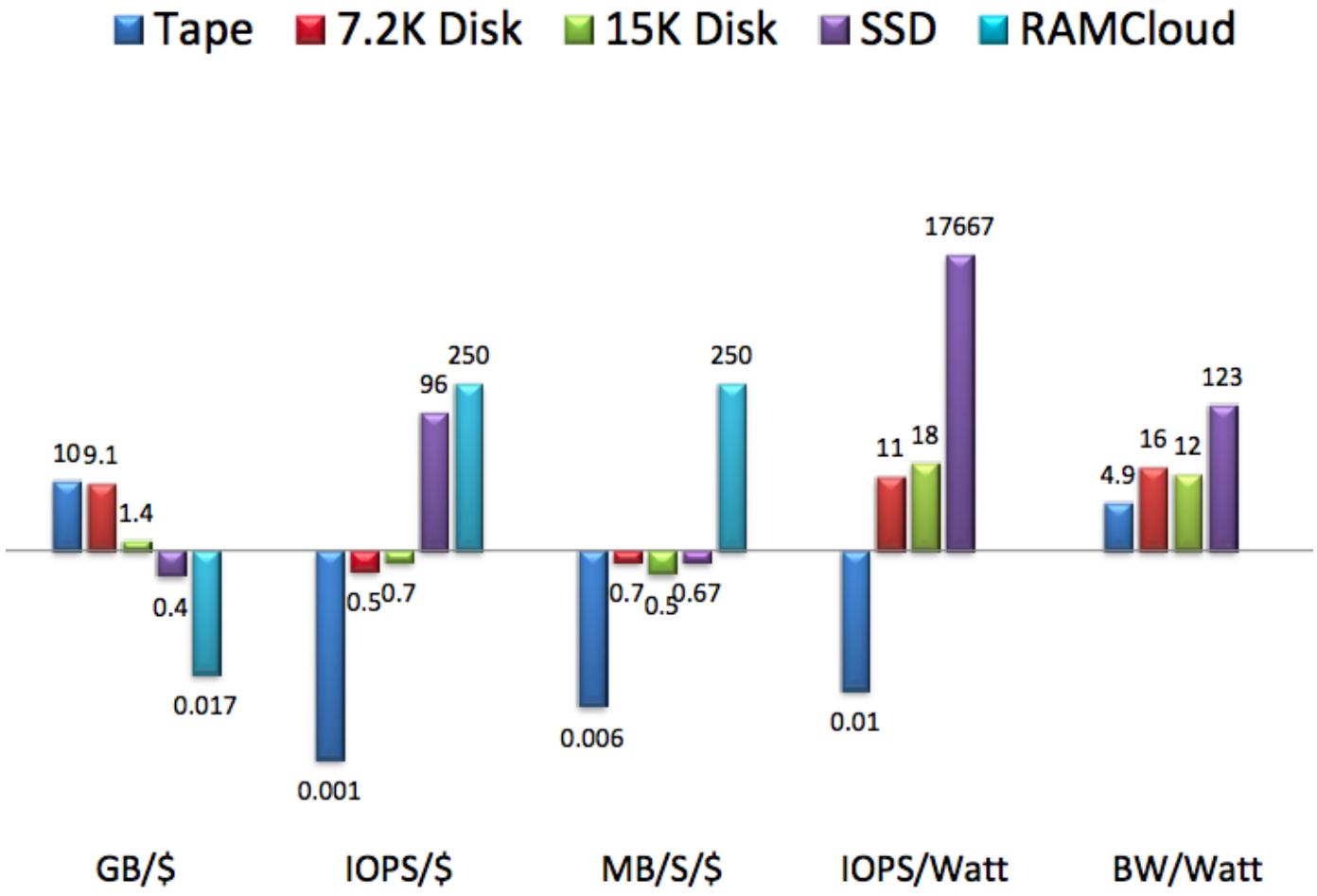
Checkpoint Restart to Node Local Storage

Must move checkpoint device closer to compute memory

- on node – has jitter issues
- at least near node is required
- Leads to Hybrid Storage model

Technology trends in flash: here to stay, but no disk replacement

Some Comparisons



Fab costs too high to sell SSD at disk prices

Solid State Disks

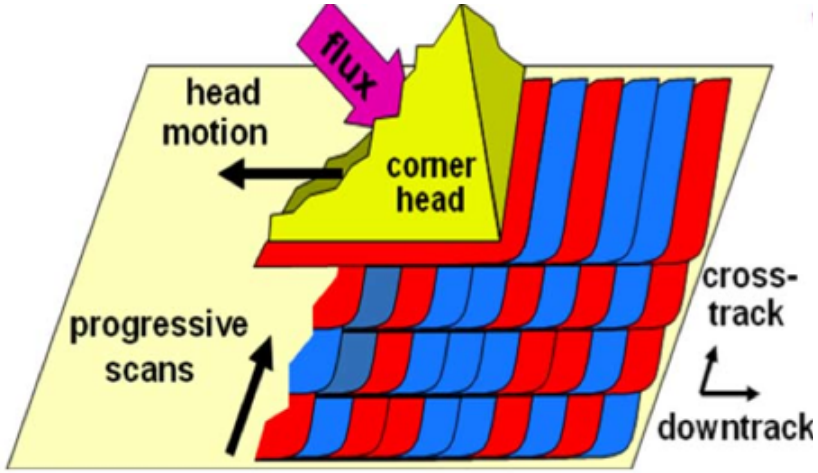
	Steve's #s	My #'s
Wafers per hi capacity drive	1	1
Fab cost	$3.4B+	$5-10B
wafers per 3 year life	1,368,750	9,000,000
Capital cost of fab per SSD unit	$2,500	$550-$1000

The issue is not wafer capacity, it is profit!

Technology trends in magnetic disks: disks are becoming tape

Possible Change in Disk Technology

Shingled-Writing



Garth's simple world view

- HAMR, BPMR:
  big changes in fab/assembly
- Shingled-writing does not need big changes

Shingle-writing means

- Partially overwriting tracks, for closer pitch
- Inability to modify one embedded sector without rewriting cross-track neighbors

Same Problem for Flash

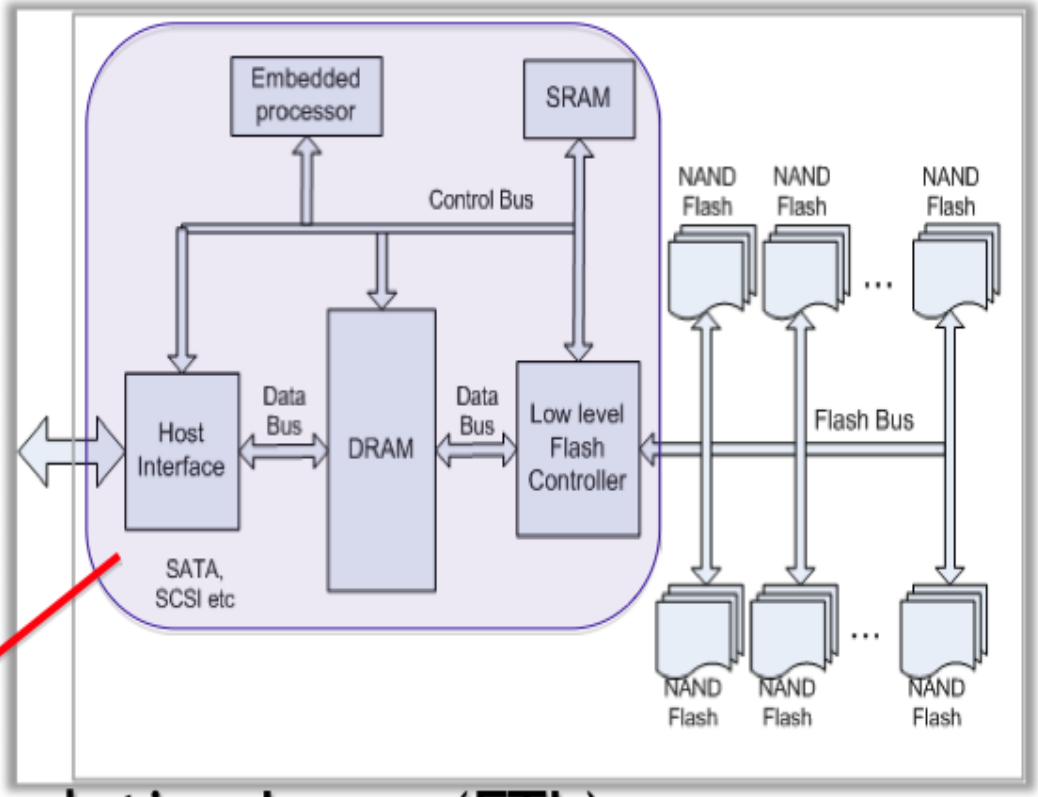
Flash SSD organized as "bands" of "sectors"

Must pre-erase band before programming data

Hide erase in FTL

Simple products
rewrite band
on all writes

Smart products
remap LBN
dynamically

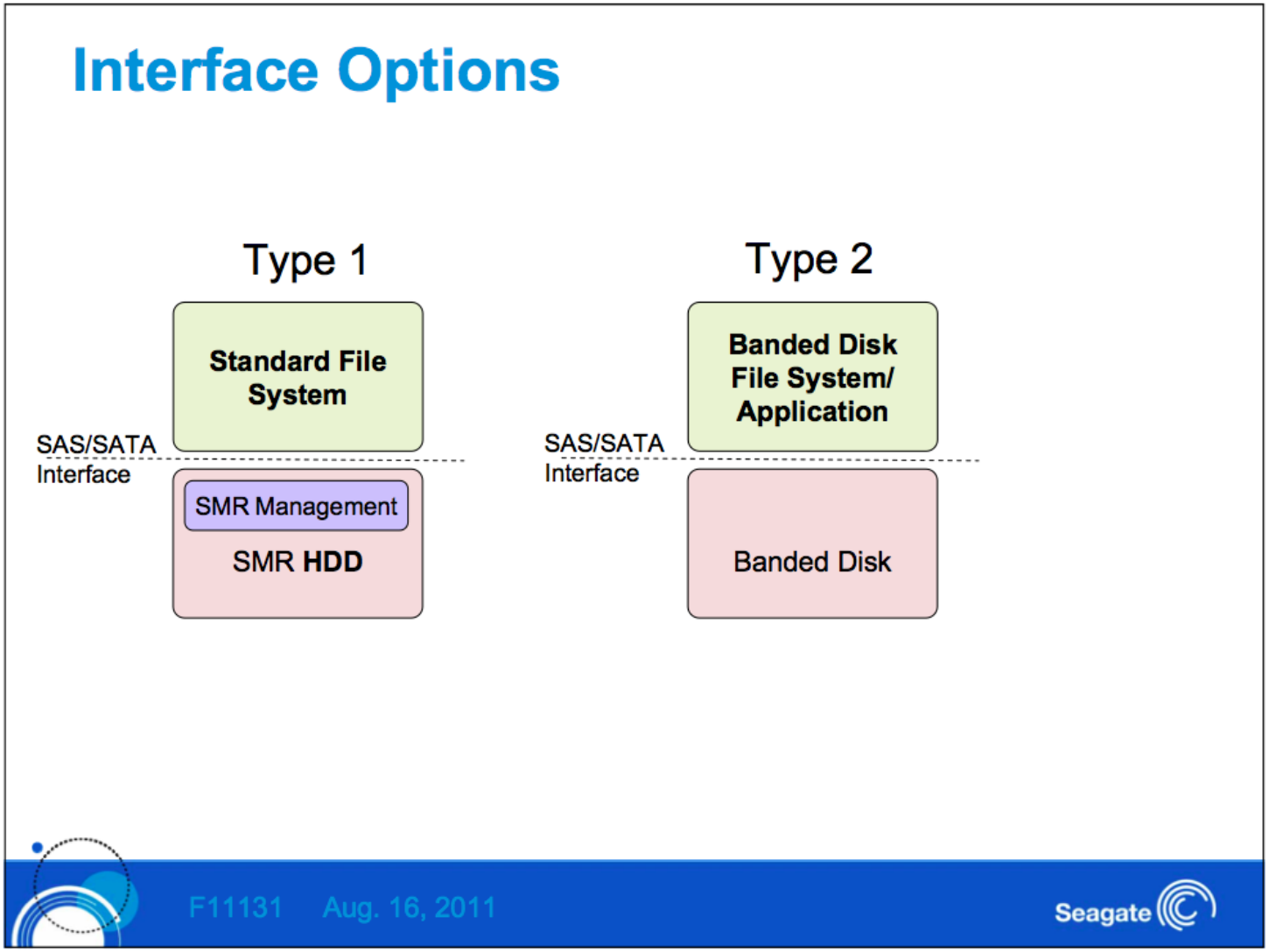


Flash Translation Layer (FTL)

Transparent STL/FTL approach

- Shingled disks implement “translation”
  - Same types of algorithms as Flash
  - Can hire ex-staff of flash industry to jumpstart
  - Data will be correct using existing codes
- Not performance transparent
  - Erase block: 100-1000 X bigger
  - Read-erase-write: 1000-10000 X longer
  - Sure to exceed long tolerable latency thresholds
- Not cost transparent
  - Disk margins < flash margins
  - Yet disk STL needs more resources

A Standards Process is Starting in T13



Shingled Disk Write is really Append

Banded Drives

- Banded Devices
  - Drive divided up into a single Random-write band & multiple sequential write bands
  - All bands are Random-read
  - RD/WR commands address data using a Band # plus LBA offset (RBA) into the band
    - In sequential-write bands the drive always writes to the next sequential block
    - Drives manage band write pointers across power cycles and resets
  - Bands can be 'linked' using Manage Bands command
    - Linked Bands allow RD/WR commands to span multiple bands
    - Links can be changed dynamically by system to manage the user data 'space'
  - Bands are not necessarily aligned to head and media boundaries
  - SCSI Reserve/Release commands supported independently on each band
  - Encryption keys can be aligned with bands enabling independent cryptographic erasure of bands

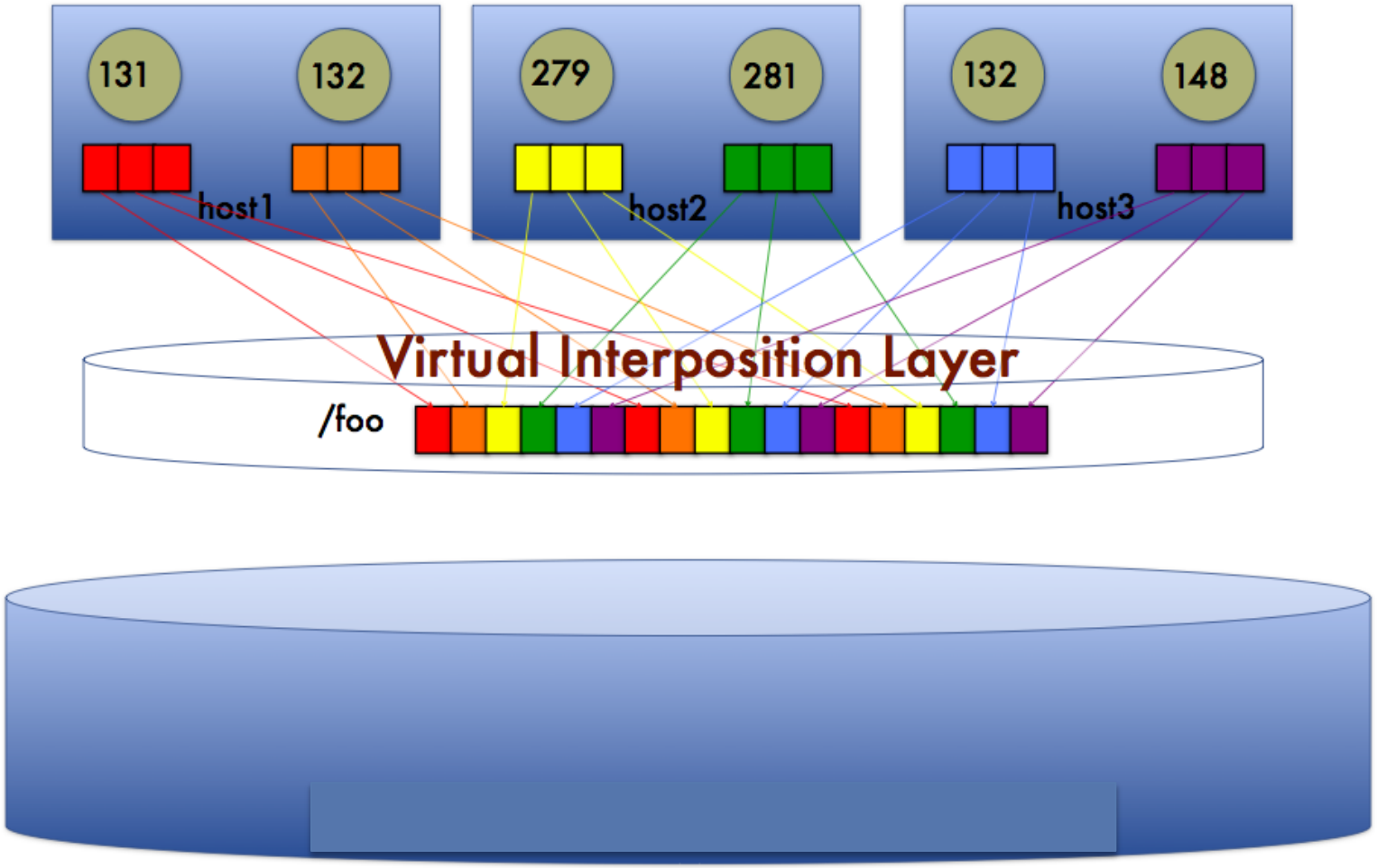
F11131 Aug. 16, 2011 Seagate

Transparent Middleware for Fast N-1 Checkpointing

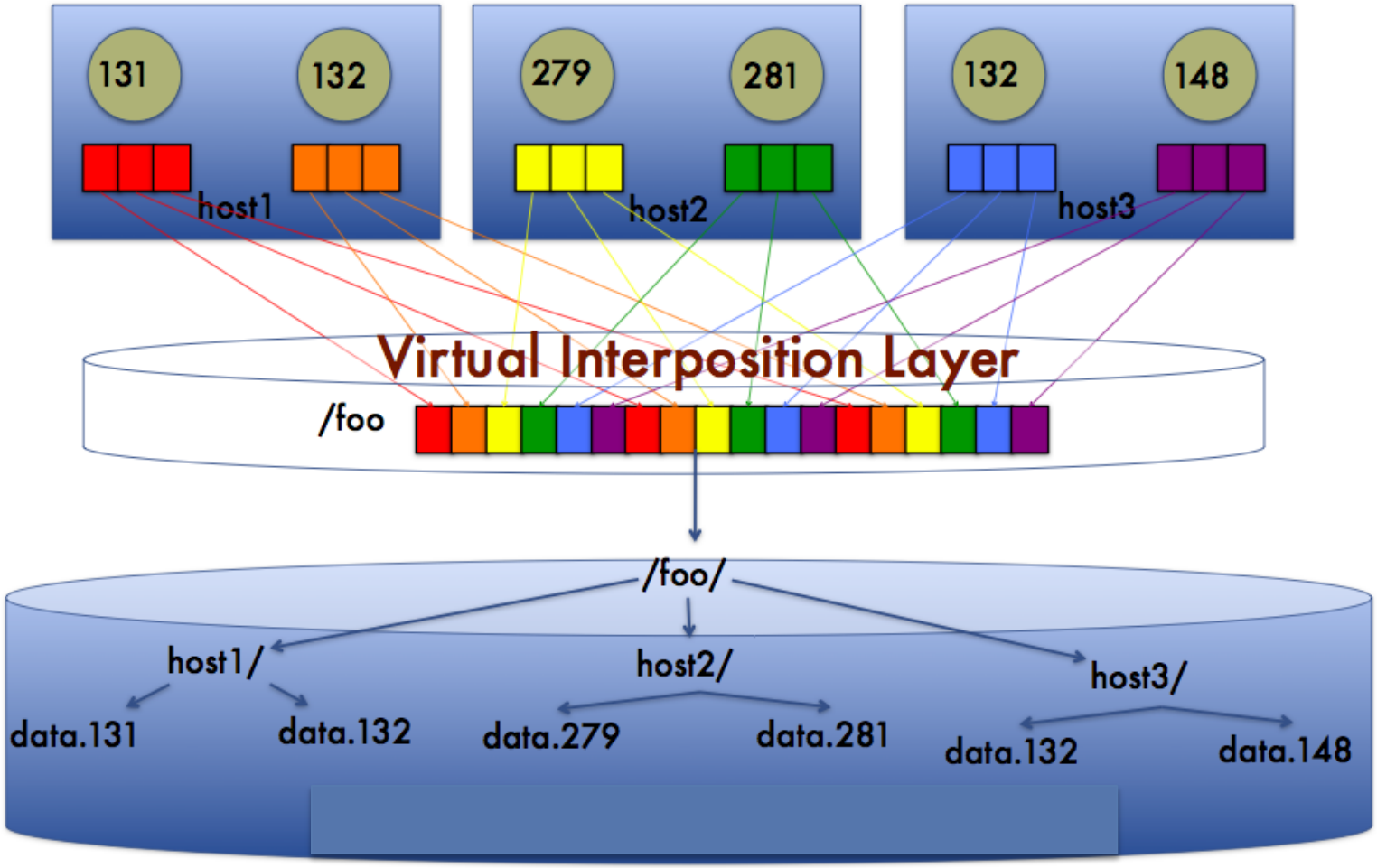
Current Write (N-1) Checkpoint Speed

- File system "capture" slows with concurrency
  - N nodes writing to 1 file, often strided, small writes (e.g. AMR)
- Written data by default organized by app, middleware, file system, but default bad for write, and often read
  - Middleware structures data by allocation in nodes into files
  - GaTech/ORNL ADIOS is new internal data organization
  - File system concurrent write sharing is serialized into logs
  - LANL/CMU PLFS is new internal data organization
  - File system layout avoids any seeks for arriving objects
  - PSC Zest is new internal data organization
- Core technique: optimize for write, not read
  - History: database logs, log-structured filesystem
  - Related: log-structured merge-tree (LSM), BigTable/SSTable

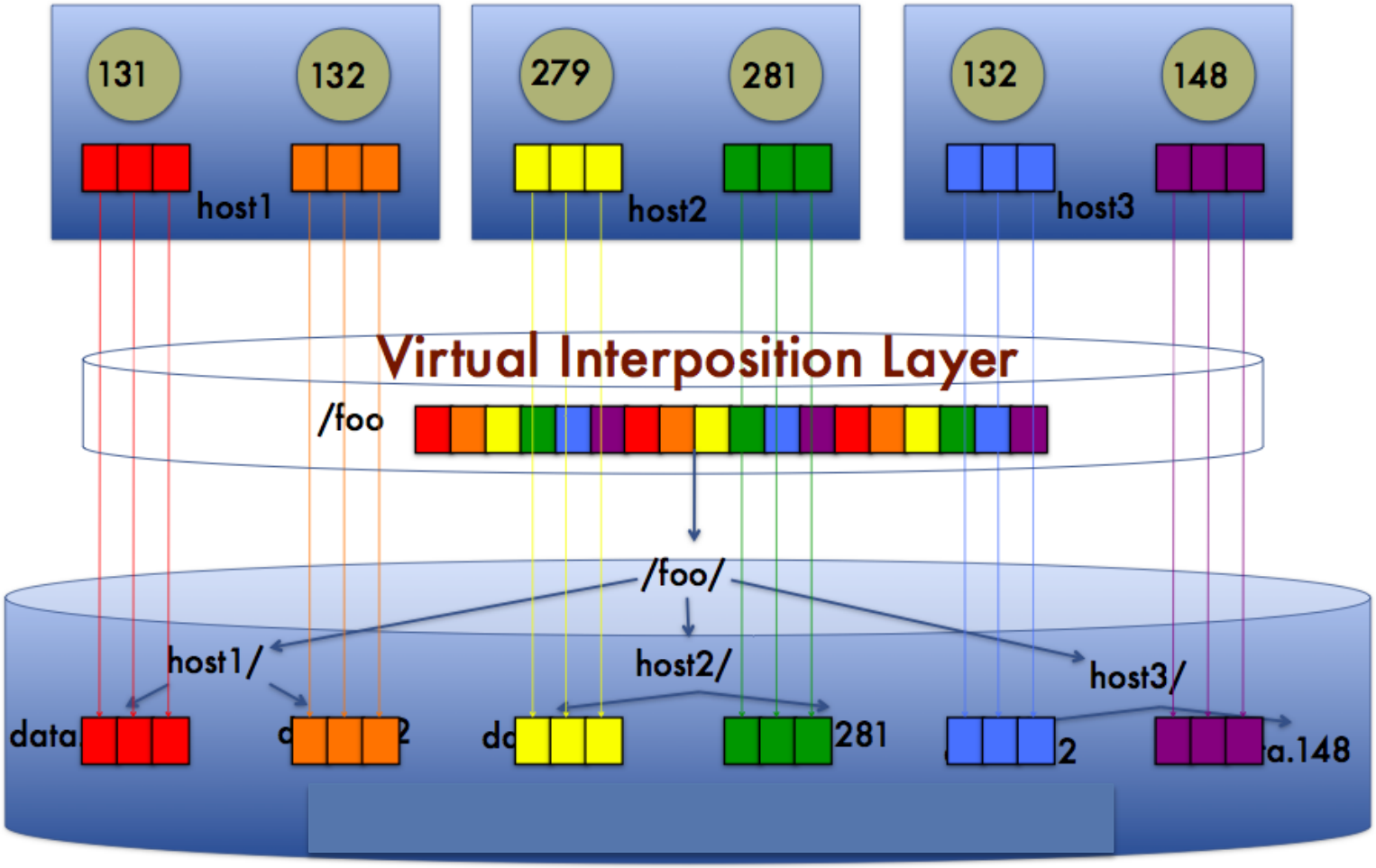
Concurrent Writes Couples Apps



PLFS Decouples Concurrency



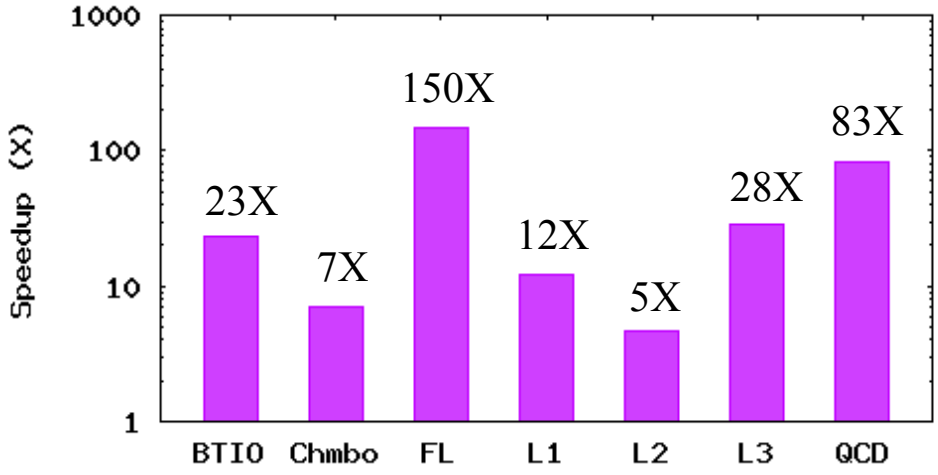
Net: HPC N-1 Write is really N-N

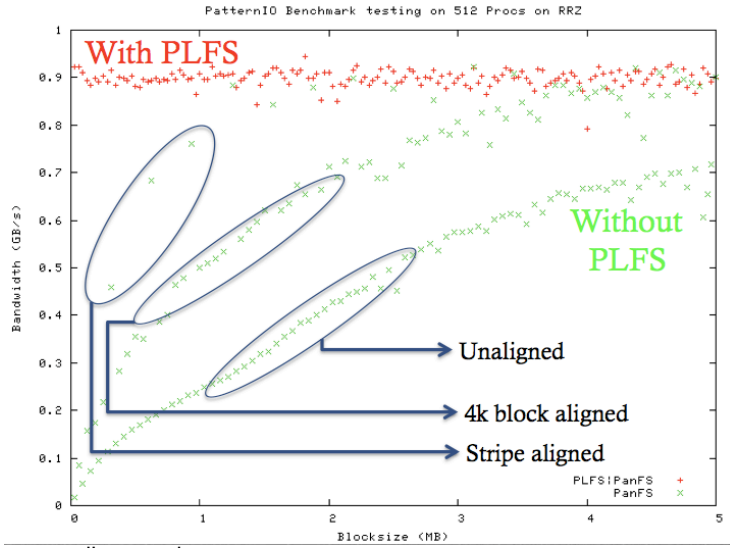


Fast Transparent N-1 Checkpoints

PLFS Checkpoint File System

- Small, strided all-to-1 writing (e.g. AMR codes) means non-scalable fault tolerance
- Employ representation transform to decouple concurrently written streams – parallel log-structured
- Surprisingly good read bandwidth





- Transparent to applications
  - Runs old codes unmodified via FUSE
  - Integrating with MPI-IO (non-FUSE)
- 5-150X wins for DOE apps
- Working at RoadRunner scale
  - Applies to Lustre, PanFS, GPFS, …
  - LANL deployment effort underway
- Insensitive to misalignment errors

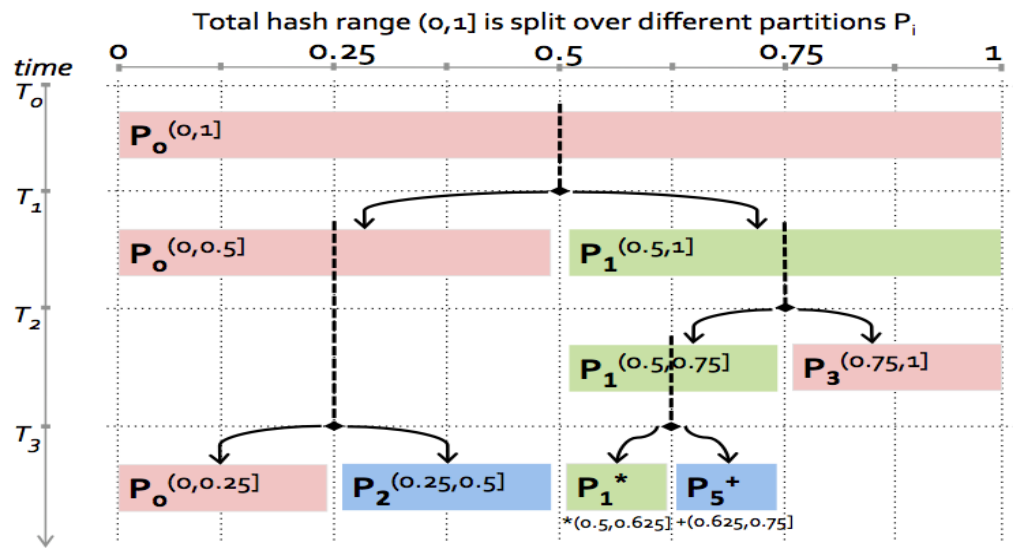
Astro-inspired work: Massive Directories for DB-in-FS

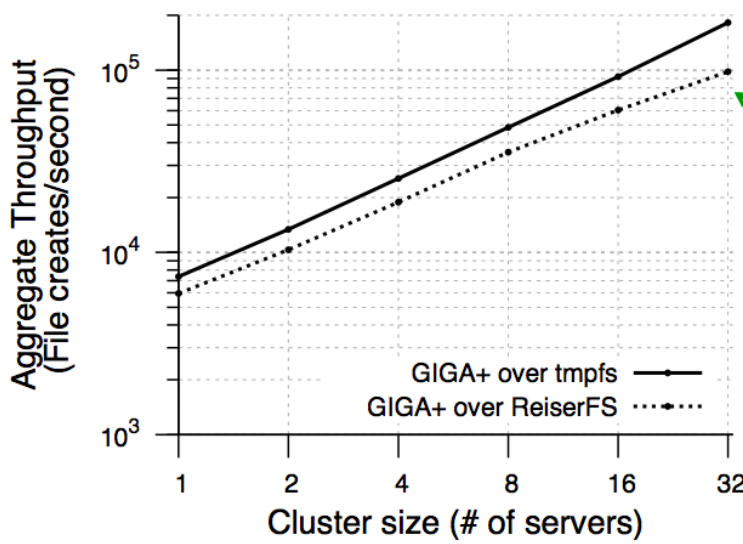
GIGA+: Directories w/ Billions of Files

- Billions of files
  - Posix API
- Extend hashing ideas of 80s/90s
  - Radix rehash
  - Stale client hints
- Decentralize & sync-free growth
  - Buckets split w/o sync with others
  - Rich client hints

FUSE & ONC-RPC

OrangeFS, NNSA



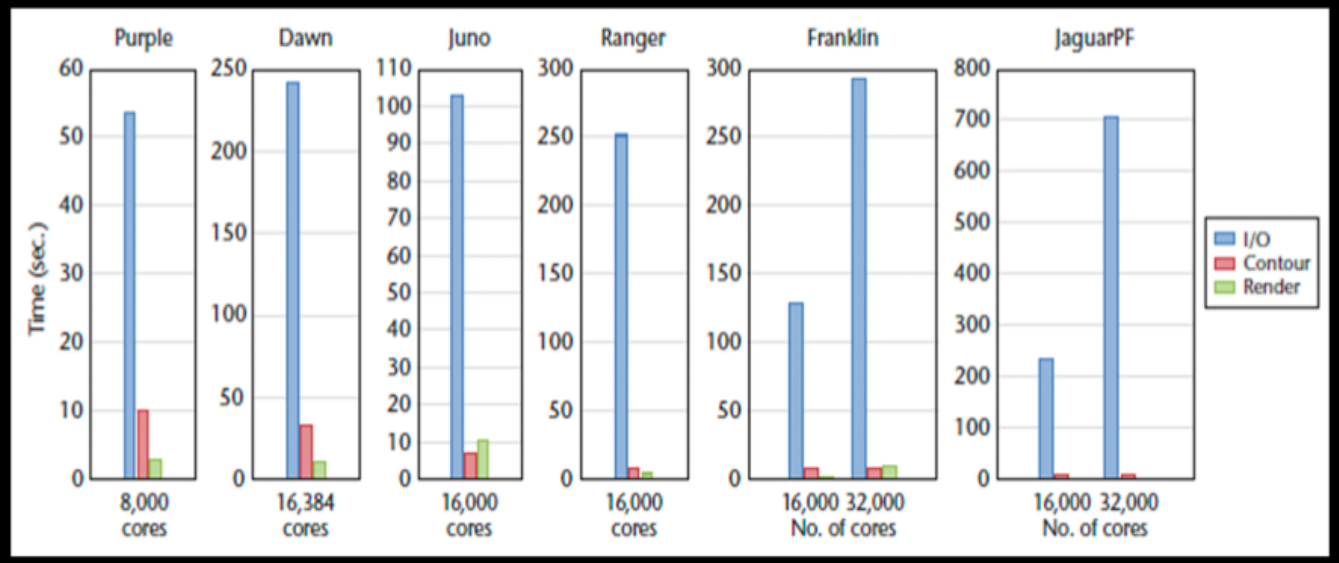


~98K creates/second

Sub-linear scaling due to the implementation decisions of localFS (ReiserFS on disk)

Google-inspired Data-Intensive Scalable Computing

Viz codes don't have enough to do ☺



Core-collapse supernova simulation from the CHIMERA code – @ 3.5 million cells
These results show that, although there is variation across the supercomputers, I/O is the slowest phase by one to two orders of magnitude.

– *Courtesy of Childs, et. al.: Extreme Scaling of Production Visualization Software on Diverse Architectures*

UNCLASSIFIED – LA-UR: 11-04588 Los Alamos UCF 9

VisIO: Idea…

- What if we collocated compute resources powerful enough to run visualization algorithms with storage?
  - ….. Distributed File Systems!
- Setup a visualization cluster with integrated storage on each node.
  - ….. Data Intensive Supercomputing (DISC)!
- Now we can have each process independently load a portion of the data set for the given time step from the local storage bus (SAS, SATA, etc).
  - No sharing with other nodes or other clusters!

UNCLASSIFIED – LA-UR: 11-04588

Los Alamos NATIONAL LABORATORY

UCF

12

The DISC Stack

- MapReduce/Hadoop
  - Indep. checkpoint/restart
- Programming layer
  - DB “compiler”, not common
- Database layer
  - Incremental update
  - Log structured merge trees
  - Lots of variants (DoD too)
  - Google has a 76+ PB DB !
- Storage Layer
  - All about cheap, but replaceable (PVFS, PanFS, S3)
- Hardware
  - All about cheap, Google vs Amazon

Job Control Layer (MapReduce)

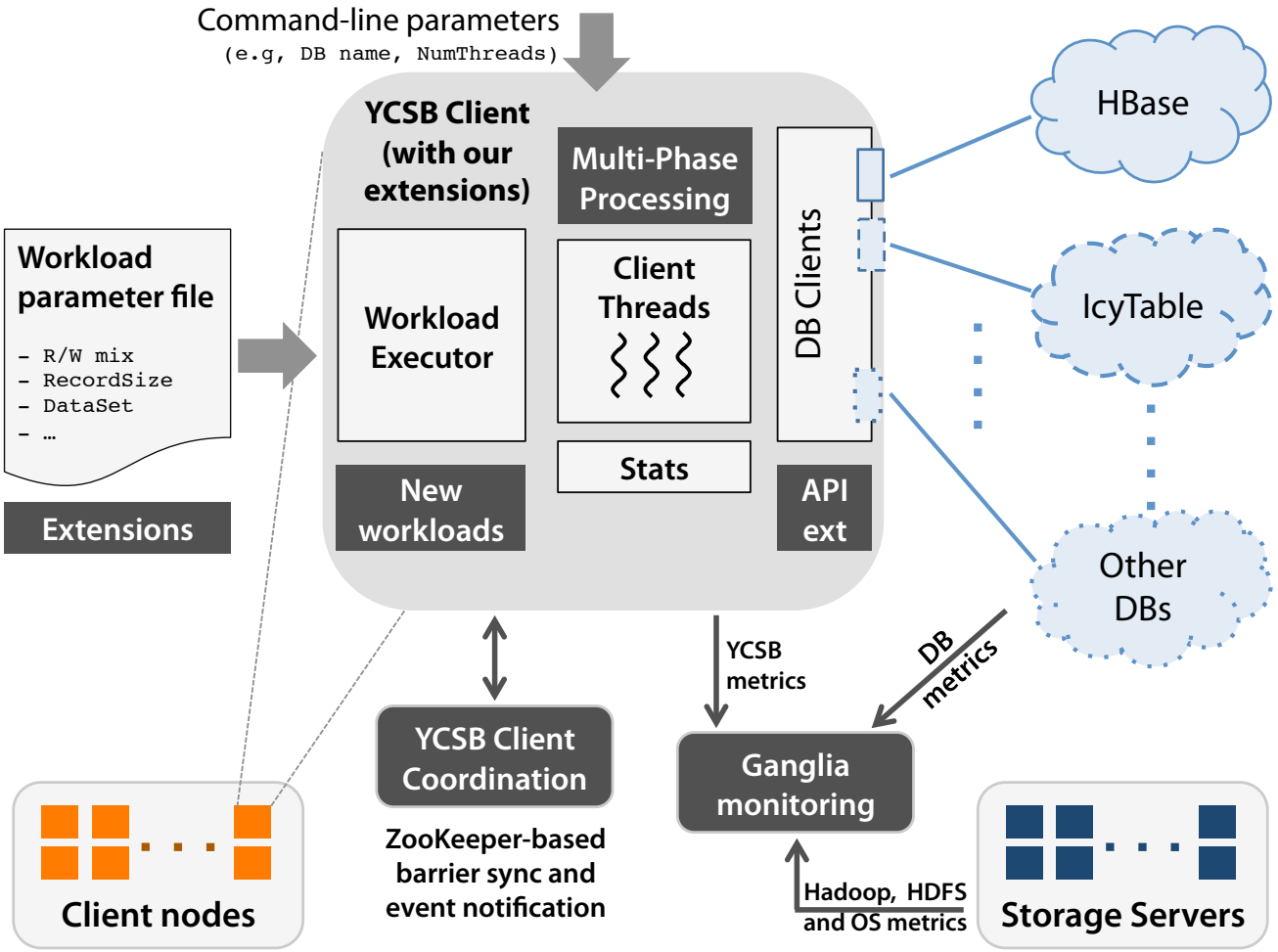
Programming Layer (HIVE?)

Database Layer (BigTable)

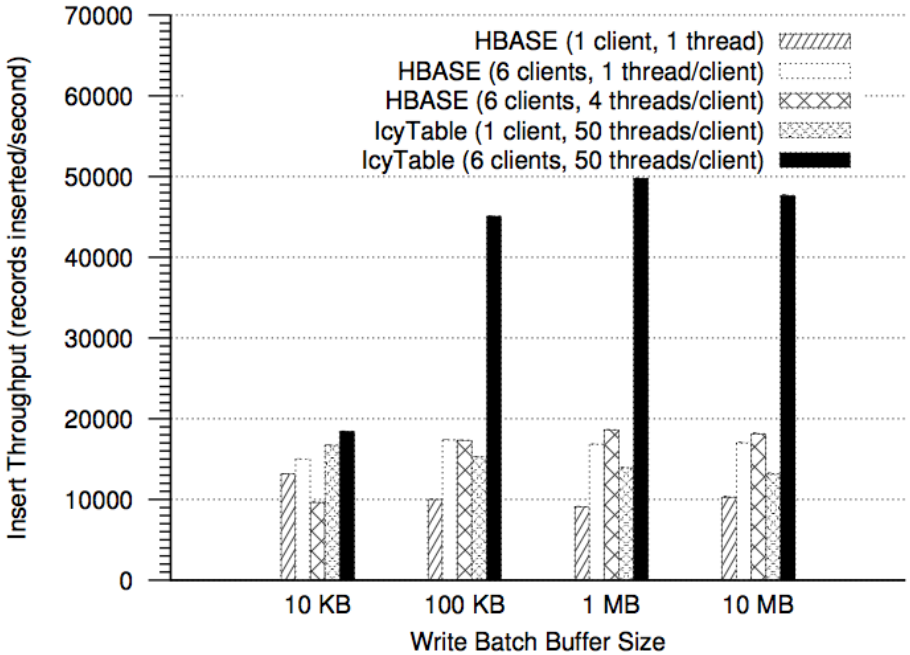
Storage Layer (HDFS)

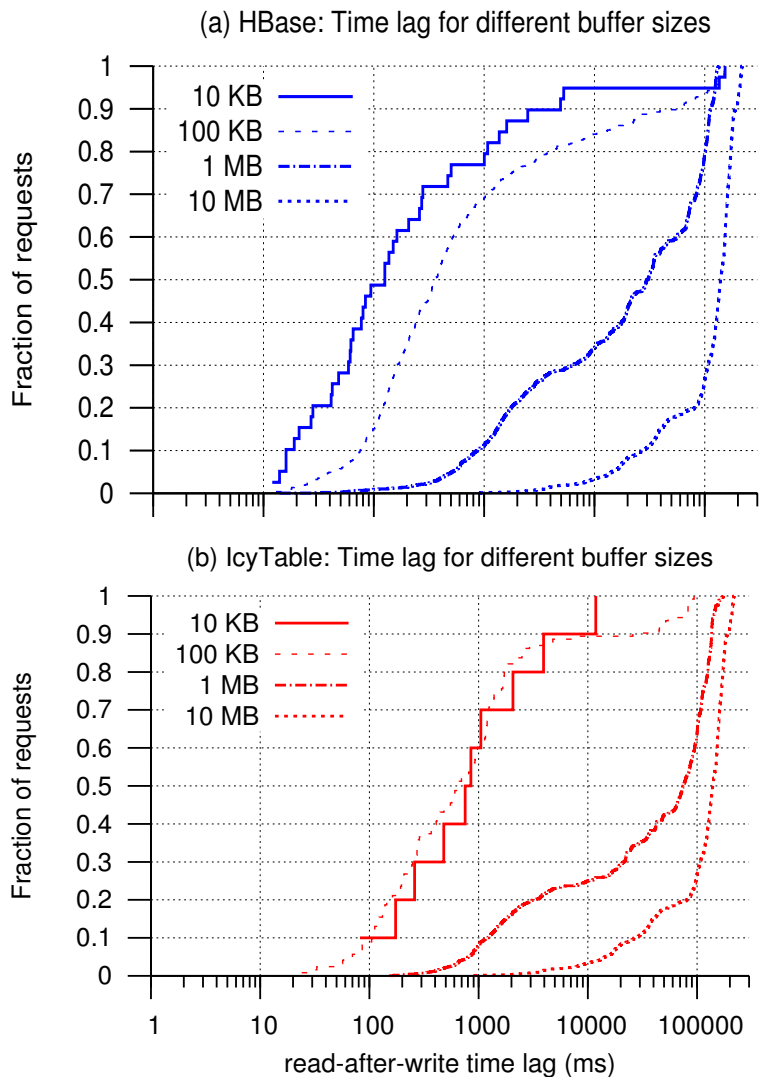
Hardware (storage rich cluster)

YCSB++: Benchmarking BigTable etc



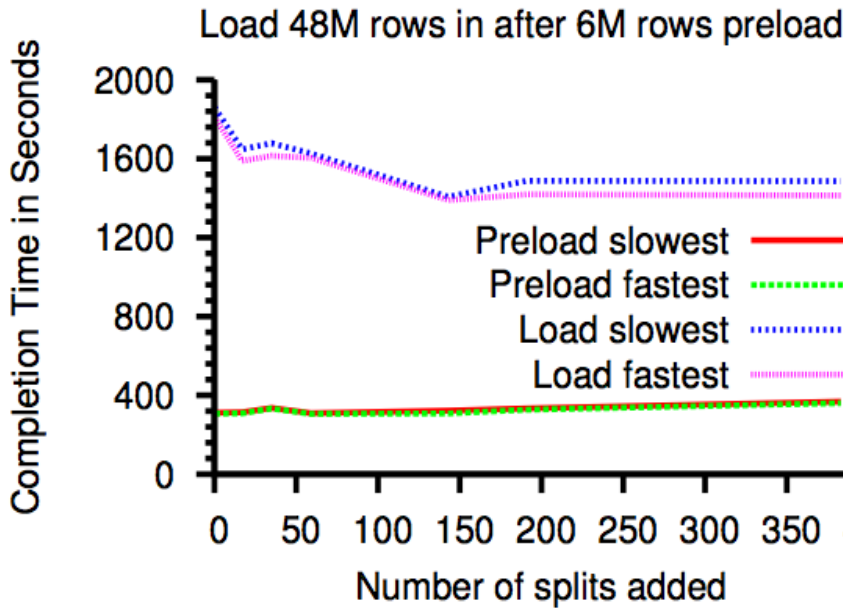
Batch writers & eventual consistency

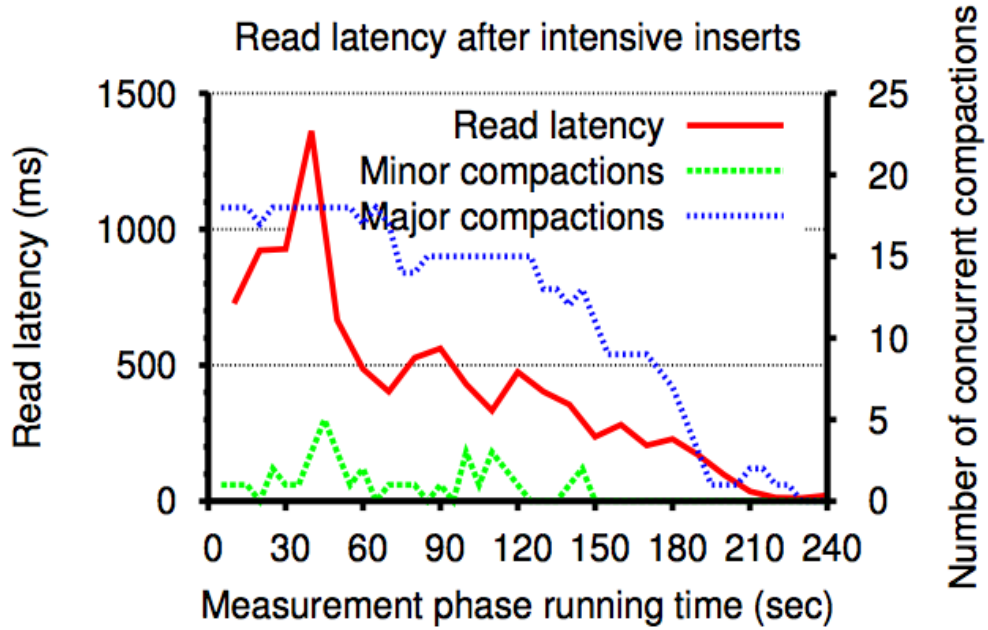




- Deferred write wins, but visible latency can be 100 secs

Pre- (and post-) Tablet Splitting





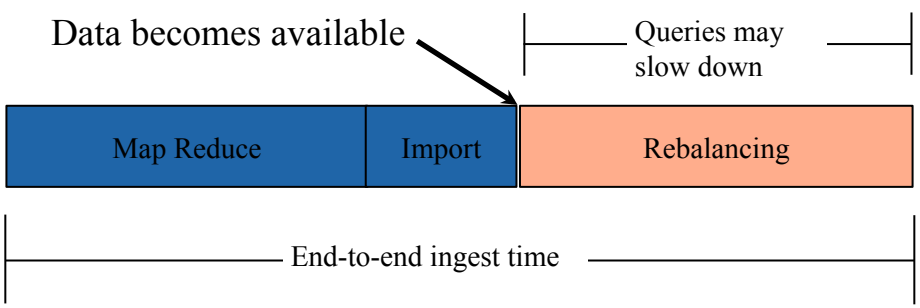
- 6 servers
  - Per server: Preload 1M rows; Load 8M rows; Measure @100 ops/s
- 20% faster load if pre-split
  - post-load rebalancing hurts *for minutes*

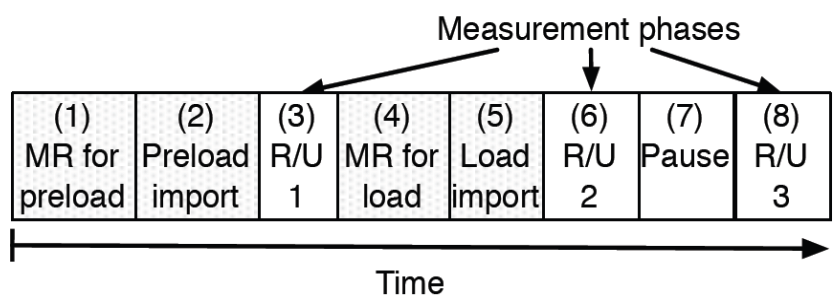
Phase	Workload
Pre-load	Pre-load 6M rows in range $[0, 12B]$
Pre-split	Pre-split tablet $[0, 72M]$ evenly
Load	Load 48M rows in range $[0, 72M]$
Measurement 1	Half update and half read for 4 minutes with 600 ops/s target
Sleep	Sleep for 5 minutes
Measurement 2	Same as measurement one

Table 2 – Different phases in pre-split experiment

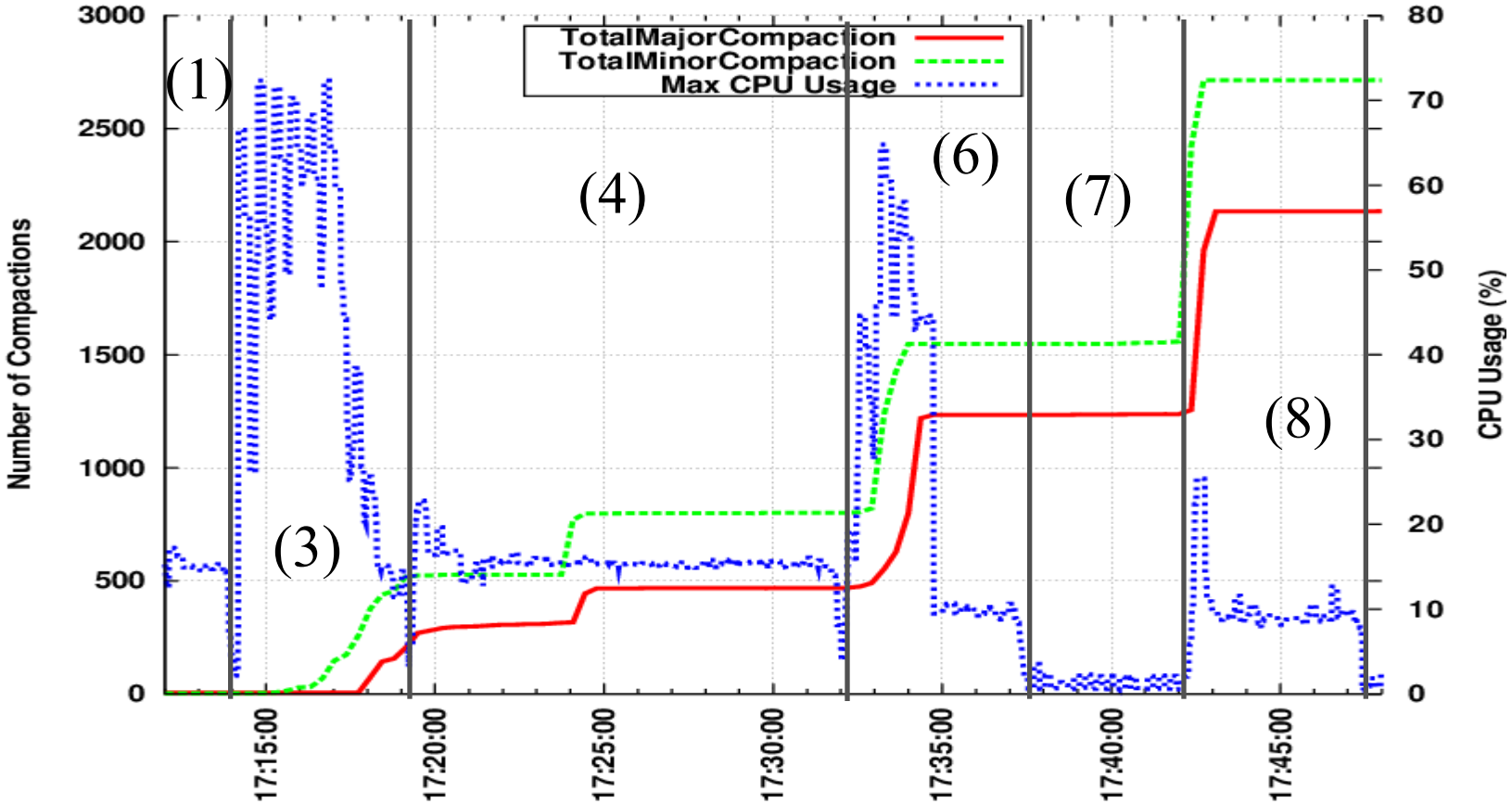
Improving Ingest Speed: Bulk Load

- Faster ingest is ***format*** with MapReduce, ingest/import with bulk load, ***rebalance*** during measurement phase
- Test: preload, monitor/measure, format bulk, bulk load, monitor/measure, sleep 5 minutes, monitor/measure
  - Per server: Preload 1M rows; Load 8M rows; Measure @ 100 ops/s
- Import turns out to be nearly instant, but rebalancing is not
  - Load 48M rows one at a time: 1400-1600 secs, 23-26 mins
  - Bulk load, including formatting time: 5-12 mins (***2-5X faster***)

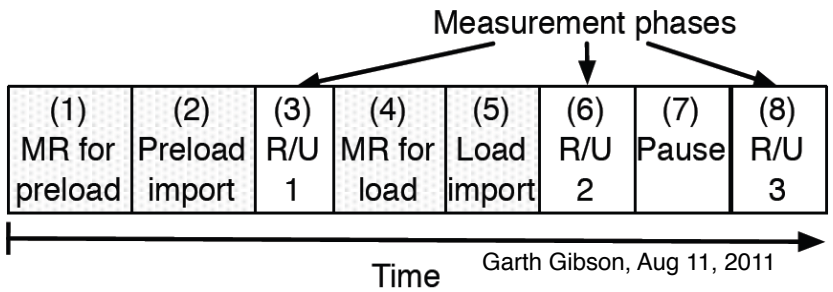




Rebalancing Timeline (54 Servers/36 MapFiles)



- Phase 1 rebalancing starts late
- Too much rebalancing work



My latest thoughts BigData management

- Google Insight
  - Data encapsulated in continually updating database is relatively slow to access (synchronized)
  - On disk formats are easy to parse, immutable
  - Flush-and-link on disk files: database snapshot
  - Query snapshot with direct processing of dataset
- YCSB++ inspired extension
  - Do same thing on ingest – bulk load arriving data
- "Database" for incidental access
  - Interactive "small" queries
  - Cleaning of data

AstroInformatics: Getting our hands dirty

http://www.pdl.cmu.edu/AstroDISC/index.shtml

SQLShare – View Query | Parallel Data Lab Project: Astro-DISC

HOME
NEWS
EVENTS
CURRENT PROJECTS
RECENT PROJECTS
PUBLICATIONS
SEMINARS & EDUCATION
PEOPLE
AFFILIATES & SPONSORS
ABOUT
MEMBERS
INTERNAL

Google™ Site Search
Search

Astro-DISC pages:
Overview
Research Goals
Sub-Projects
- DISC-Distances
- DISC-Finder
- DISC-Holes
- DISC-Index
- DISC-Quasars
- DISC-SkyMap
People
Publications & Posters
Code Distribution
- DISC-Distances
- DISC-Finder
- DISC-Quasars
- DISC-SkyMap
Sponsors & Partners
Fun Stuff
- The Universe in a Nutshell

Astro-DISC: Astronomy and Cosmology Applications of Data-Intensive Supercomputing

Contact: Eugene Fink, Julio López, Garth Gibson

We are developing new algorithms, data structures, and software tools for the analysis of massive astronomical and cosmological datasets. The goal is to enable the efficient processing of massive astrophysics datasets, with sizes in the tens of terabytes, by scaling up standard algorithms and designing new ones. We are developing programming abstractions and tools to address these challenges and to extend the developed techniques to other fields of science.

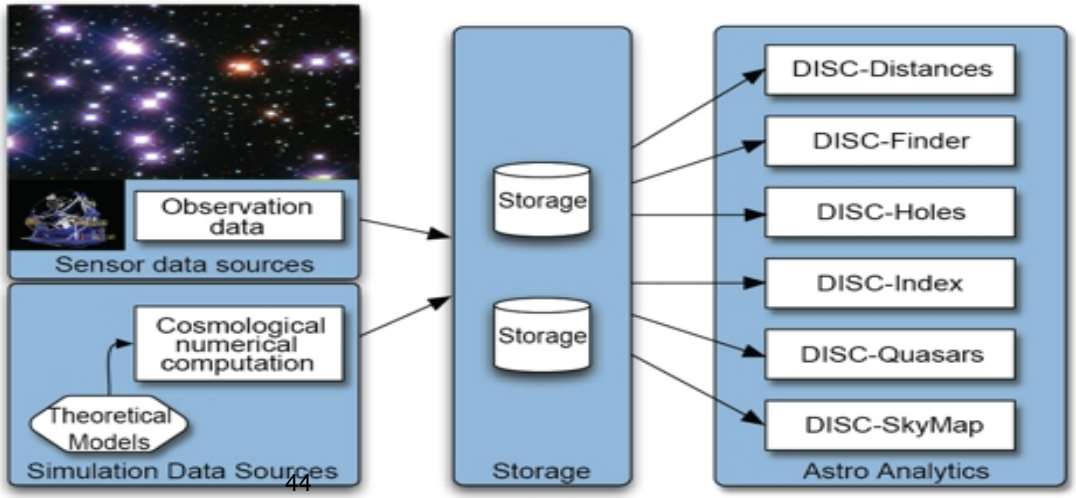
RESEARCH AREAS

- Distributed algorithms and data structures
- Indexing and analysis of massive structured data

CHALLENGES

- Developing distributed versions of standard astronomy applications
- Creating efficient algorithms for new astronomical challenges, in collaboration with astronomers and cosmologists
- Building a general-purpose toolkit for the analysis of massive astronomical observations and cosmological simulations

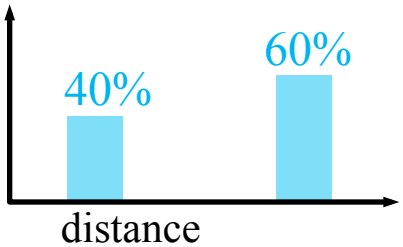
ASTROPHYSICS ANALYTICS PROCESS



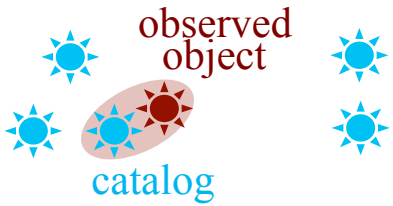
W/ Di Matteo, Croft (CMU), Wood-Vassey (U.Pitt)

- Friends of Friends:
  Identification of galaxy clusters

- Correlation functions:
  Analyzing distribution of distances between galaxies



- Spatial matching:
  Identification of observed objects in the catalog



- Quasar detection:
  Identifying quasars based on the five passband fluxes

W/ Di Matteo, Croft (CMU), Wood-Vassey (U.Pitt)

- Friends of Friends:

Astro-DISC: Code Distribution

If you would like to use the software developed by Astro-DISC research, please follow these links to download it. Please note that this is research software, which is not "plug and play", and it may take some work to figure out how to run it.

If you need help, please e-mail **eugenefink@cmu.edu**. We would love to see people trying out our code and we are always happy to help.

- DISC-Distances: Analyzing the distribution of distances between galaxies
- DISC-Finder: Identifying clusters of astronomical objects
- DISC-Quasars: Identifying distant quasars in sky surveys (coming soon)
- DISC-SkyMap: Indexing astronomical objects (coming soon)

Please review our software license.

- Quasar detection:

Identifying quasars based on the five passband fluxes

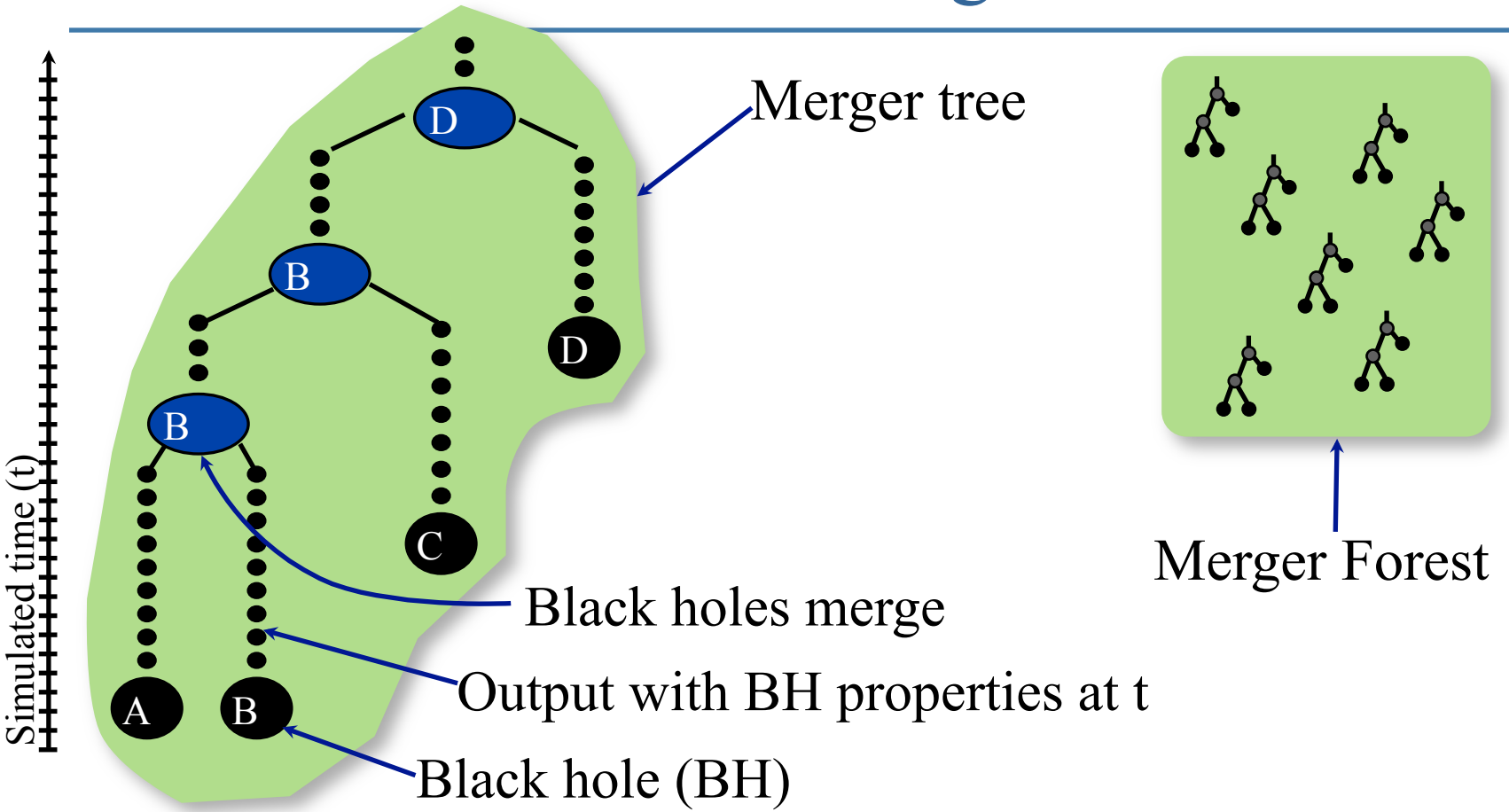
Recipes for Baking Black Forest Databases

Building and Querying Black Hole Merger Trees from Cosmological Simulations

Julio López, Colin Degraf, Tiziana DiMatteo,
Bin Fu, Eugene Fink, Garth Gibson
Carnegie Mellon

NSF award CCF-1019104, and The Gordon and Betty Moore Foundation.
CMU PDL OpenCloud Cluster, and NSF allocation of Advanced Computing Resources at Kraken (NICS)

Black Hole Merger Trees



- Merger forest is in the order of hundreds of millions of trees
- # BH properties >> # mergers (by 2+ orders of magnitude)

AstroInformatics: Blackhole DB in SQLShare

https://sqlshare.escience.washington.edu/sqlshare#s=query/billhowe/blackhole

SQLSHARE

blackhole x

Logged in: garthgibsoncmu@gmail.com

Your datasets
All datasets
Shared datasets
Recent activity... 0
Recently viewed »

Upload dataset
New query

POPULAR TAGS

protein	21
protein	15
ecoli	11
oceanography	10
optode	10
suna	10
tsg	10
fluorimeter	10
h1	8
albumin	6
eh1	4
cph1	4
coffee	3
mims	3
blood	3
salary	2

blackhole Viewable by everyone

Last modified: Aug 5, 2011 2:03 AM billhowe@washington.edu

DISC-Holes: Data Models for Black-hole Datasets from Cosmological Simulations
Contact: Julio López, Eugene Fink, Garth Gibson

http://www.pdl.cmu.edu/AstroDISC/DISC-Holes.shtml

astronomy

```
SELECT * FROM [billhowe].[table_blackholeF0D68]
UNION ALL SELECT * FROM [billhowe].[table_blackhole7D23A]
UNION ALL SELECT * FROM [billhowe].[table_blackholeF94AB]
UNION ALL SELECT * FROM [billhowe].[table_blackholeE5106]
UNION ALL SELECT * FROM [billhowe].[table_blackhole482AA]
UNION ALL SELECT * FROM [billhowe].[table_blackholeA4C7A]
UNION ALL SELECT * FROM [billhowe].[table_blackhole3AF8D]
UNION ALL SELECT * FROM [billhowe].[table_blackholeD8AA3]
UNION ALL SELECT * FROM [billhowe].[table_blackholeA643D]
UNION ALL SELECT * FROM [billhowe].[table_blackholeD7922]
```

Derive dataset | Create snapshot | Download

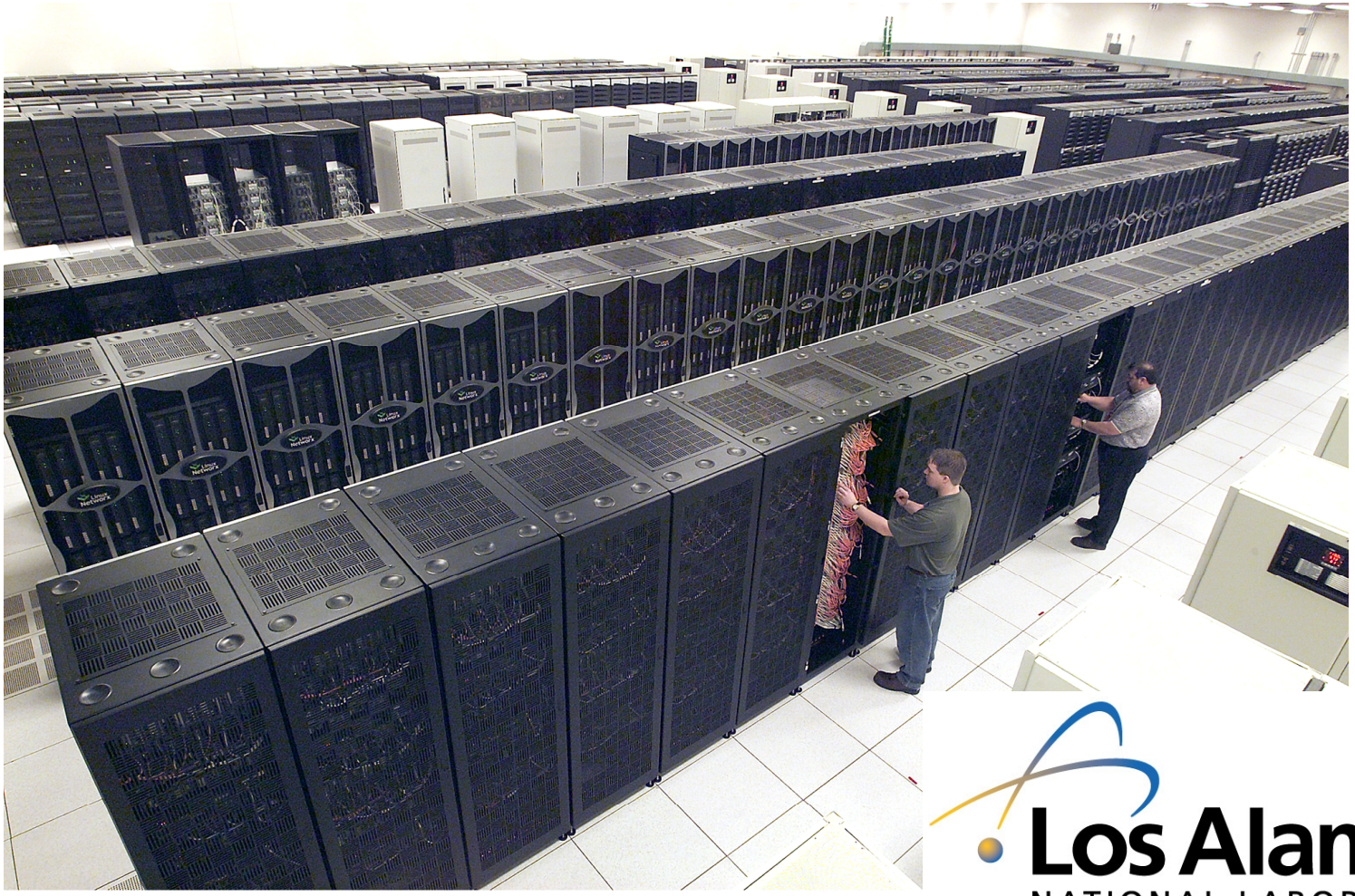
DATASET PREVIEW Rows 1 - 100 of 866916 | Columns 13 of 13

<< first < prev 1 2 3 4 5 next > last >>

bhid	time_q	mass	mdot	rho	csound	vrel	pos_x	pos_y	pos_z	vgas_x	vgas_
126	0.149092	5E-05	0.000122	0.00468788	38.2642	47.0772	580.6047198	260.2525321	2335.2431058	10.86	-14.77
126	0.149462	5.00826E-05	0.000128787	0.0046578	38.2434	45.6068	581.5205795	258.9971326	2336.0664937	10.4068	-14.99
126	0.149833	5.04327E-05	0.00012495	0.00448316	40.8896	43.4131	582.4068202	257.6959212	2336.9211122	10.069	-15.00
126	0.150205	5.07737E-05	0.000130975	0.00410368	40.5617	40.4161	583.2577857	256.3466823	2337.8131857	10.0086	-14.63
126	0.150577	5.11324E-05	0.000137242	0.00389083	40.0676	38.6103	584.0787433	254.9457105	2338.7472577	10.0007	-14.11
126	0.150951	5.15096E-	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	-

AstroInformatics: Reducing the price of Astro-scale data

LANL was going to trash this!



Los Alamos
NATIONAL LABORATORY
EST. 1943

NSF to the Rescue

- Large scale clusters for systems researchers
  - For dedicated use, long periods of time (days, weeks)
  - Allow replacement of any and all software
- NSF funds PRObE (2011-2014)
  - Parallel Reconfigurable Observational Environment
  - New Mexico Consortium (NMC)
    - Andree Jacobson, Katharine Chartrand
    - Joint unit of U. of NM, NM Inst. of Mining &Tech., NM State U.
    - Located in Los Alamos, NM
  - Los Alamos National Laboratory (Gary Grider)
  - Carnegie Mellon University (Garth Gibson)
  - University of Utah (Robert Ricci)

PRObE Inspired AstroInformatics Facility

- Recycle a supercomputer with new disks
  - 1000 x 2-4 x 2-4TB = 4-16 PB
    - @ $200/disk, < $1M in new parts
  - Compute sufficient for Viz if “a few percent” is valid
- Run Hadoop/HDFS & MPI/PLFS/HDFS
  - Slower for compute, but “no cost cycles”
- Still need datacenter, power & cooling
  - NSF awarded PRObE < $1M/yr to operate
- Extend SQLshare to AstroInformatics_Share

Research Sponsors

Office of Science

NSF

DEPARTMENT OF DEFENSE · UNITED STATES OF AMERICA

IRHPIT INSTITUTE FOR RELIABLE HIGH PERFORMANCE INFORMATION TECHNOLOGY

ASCR

GORDON AND BETTY MOORE FOUNDATION

OAK RIDGE National Laboratory

Los Alamos NATIONAL LABORATORY EST. 1943

panasas®

YAHOO!

Google™

Companies of Parallel Data Consortium: APC, EMC, Facebook, Google, Hewlett-Packard, Hitachi, IBM, Intel, LSI, Microsoft, NEC, NetApp, Oracle, Seagate, Symantec, VMware

Carnegie Mellon
Parallel Data Laboratory

pdsi

garth@cs.cmu.edu, garth@panasas.com