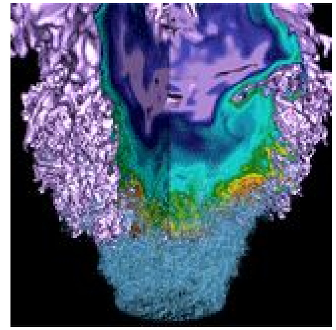
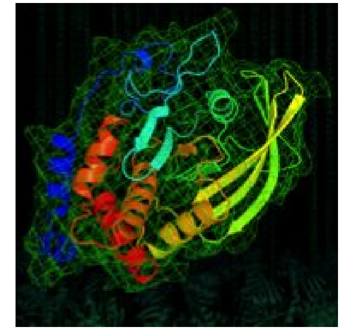
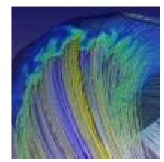
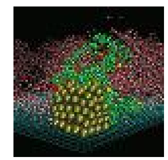
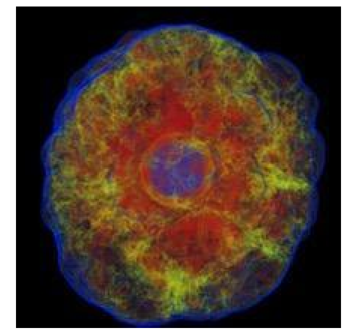
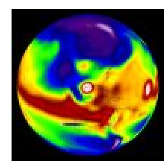
Cori Phase 1 Burst Buffer

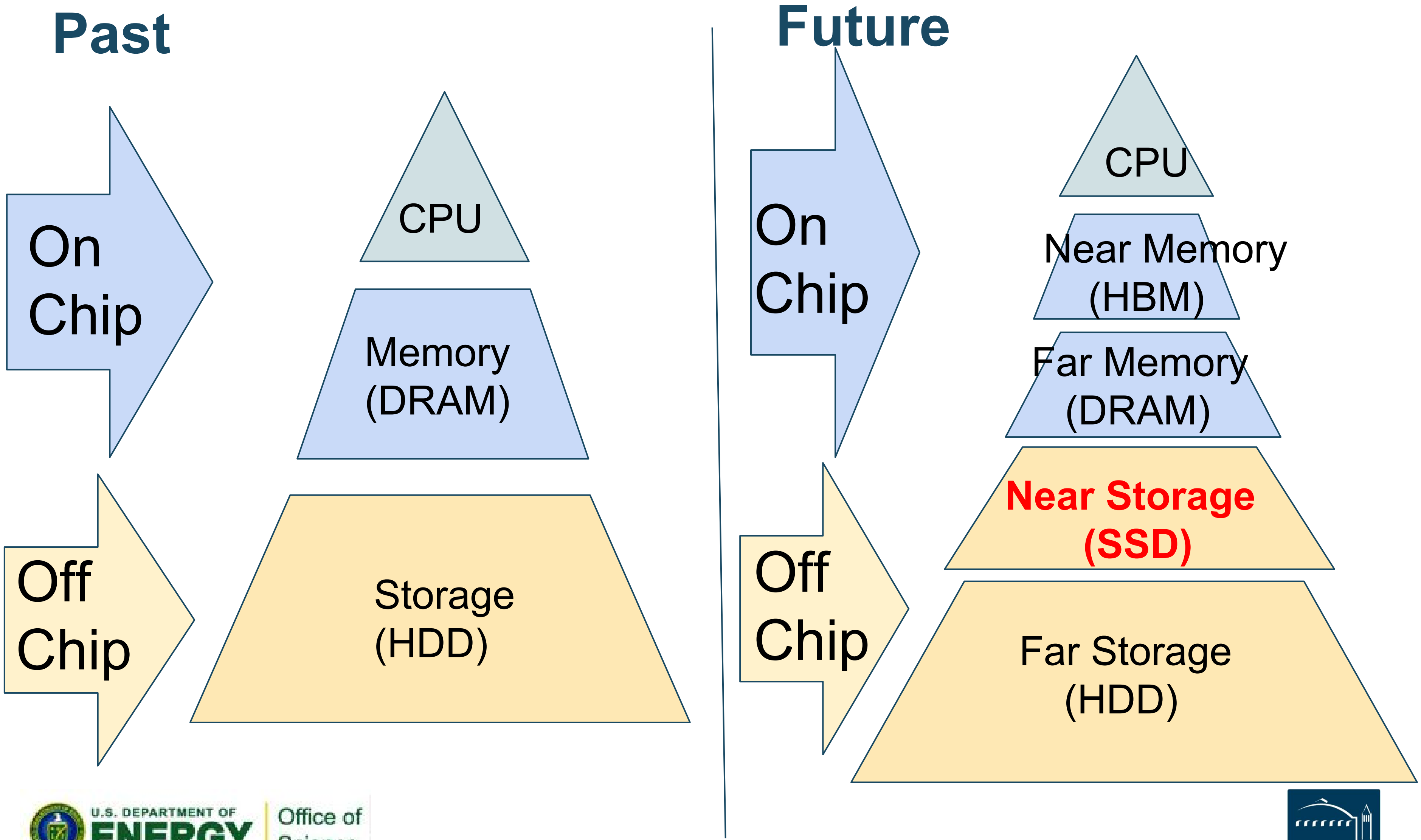

NERSC 40 YEARS at the FOREFRONT

Debbie Bard, Wahid Bhimji, Dave Paul, et. al.

U.S. DEPARTMENT OF ENERGY | Office of Science

BERKELEY LAB

HPC memory hierarchy is changing



HPC memory hierarchy is changing

Past

On Chip

CPU

Memory (DRAM)

Off Chip

Storage (HDD)

Future

On

CPU

Chip

Far Storage (HDD)

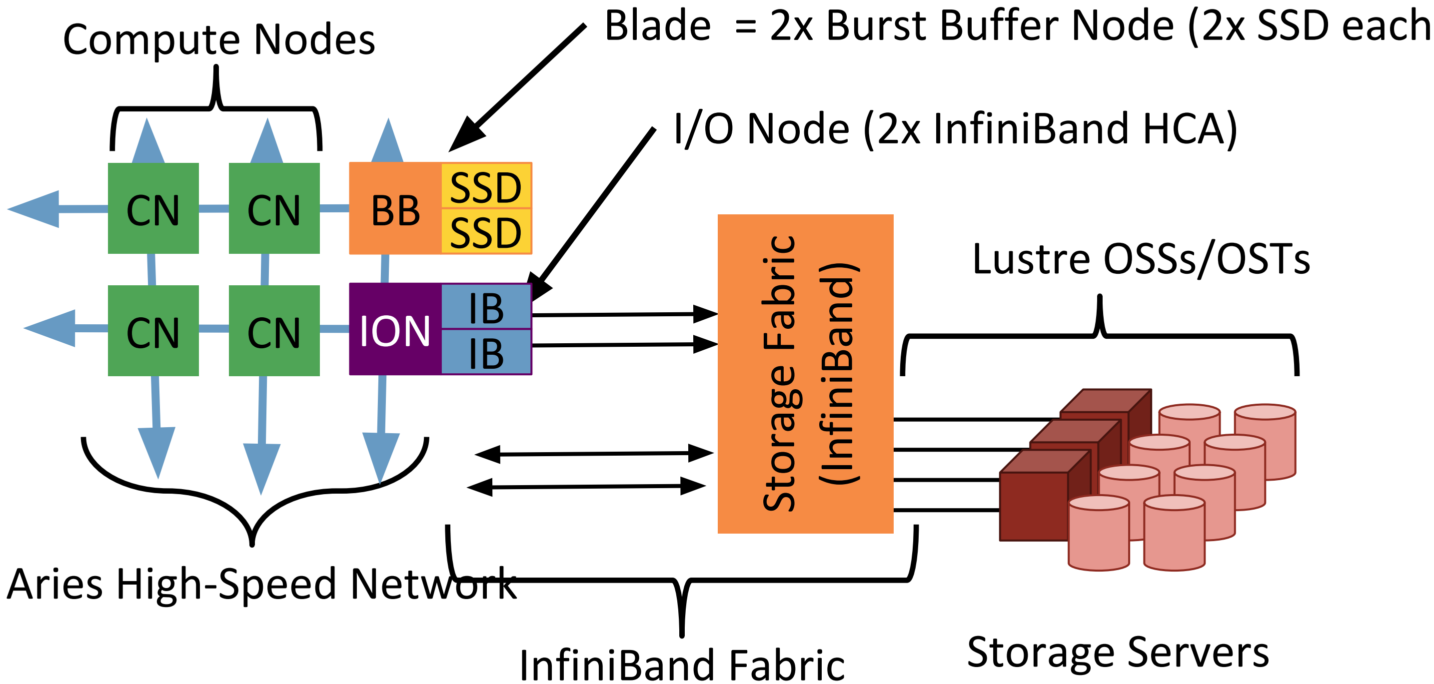
Why a Burst Buffer?

- **Handle spikes in IO bandwidth requirements without increasing size of PFS**
  - Reduce job wallclock time
  - Compute resources idle during IO bursts
- **Disk-based PFS bandwidth is expensive**
  - Capacity is relatively cheap
- **SSD bandwidth is relatively cheap**
- **-> Separate bandwidth and spinning disk**
  - Provide high BW without wasting PFS capacity
  - Leverage Cray Aries network speed

Burst Buffer implementation

- **High BW SSDs in service nodes, directly attached to Aries network**
- **Software creates pool of available memory**
  - DataWarp service daemons
  - DataWarp file system (using DVS, LVM, XFS)
  - Integrated with SLURM
- **Allocation portions of this pool to users per-job, or in a persistent reservation**
- **Users see a POSIX-compatible FS**
- **Can stage data in and out from BB to PFS**
  - Before/after compute job starts - *saves compute time*
  - Asynchronously during compute job

Burst Buffer Architecture



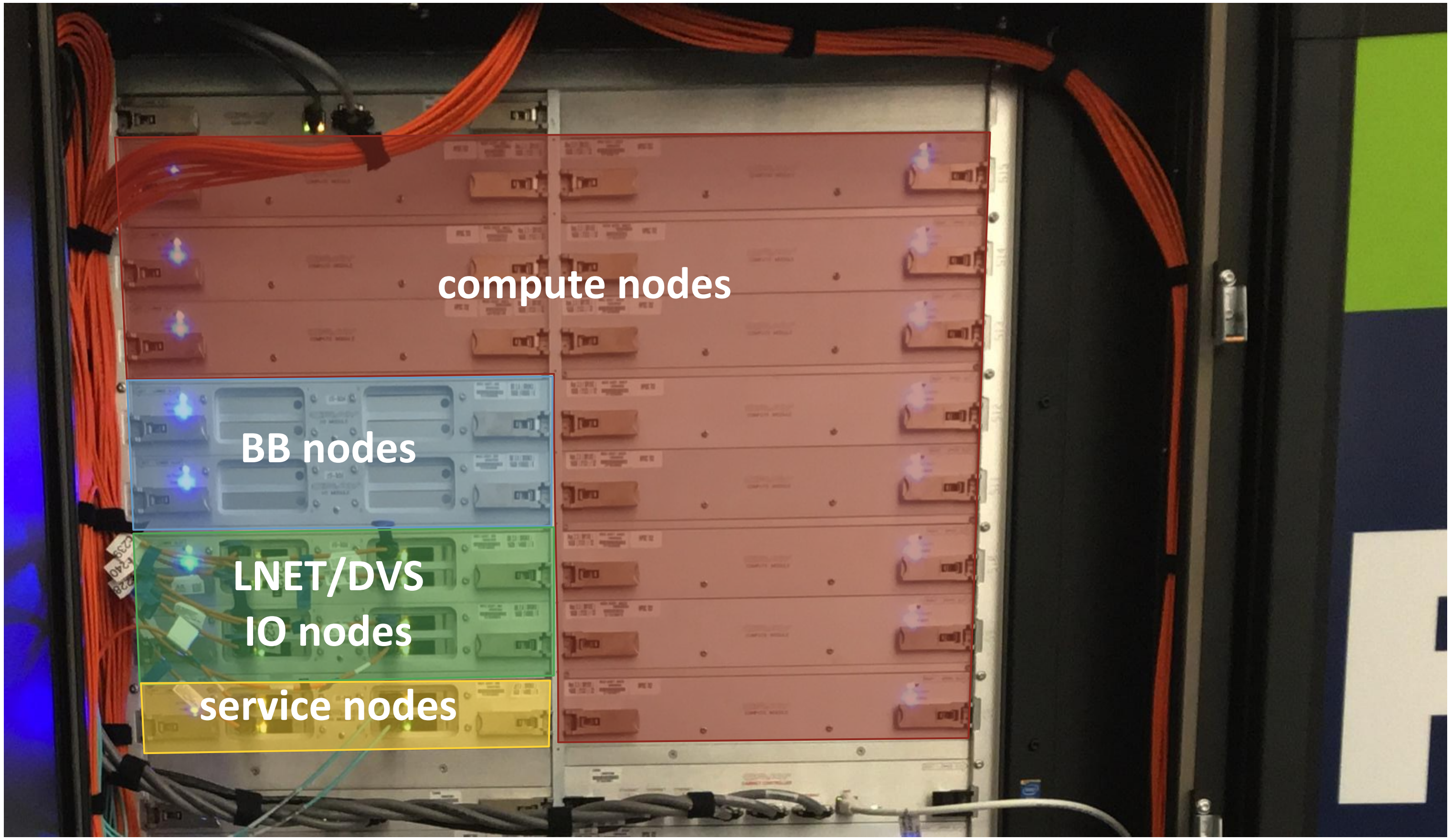
- **Cori Stage 1 configuration: 920TB on 144 BB nodes (288 x 3.2 GB SSDs)**
- **>1.5 PB total coming with Cori Phase 2**
- **Lustre: 30PB PFS**

Burst Buffer Architecture Reality

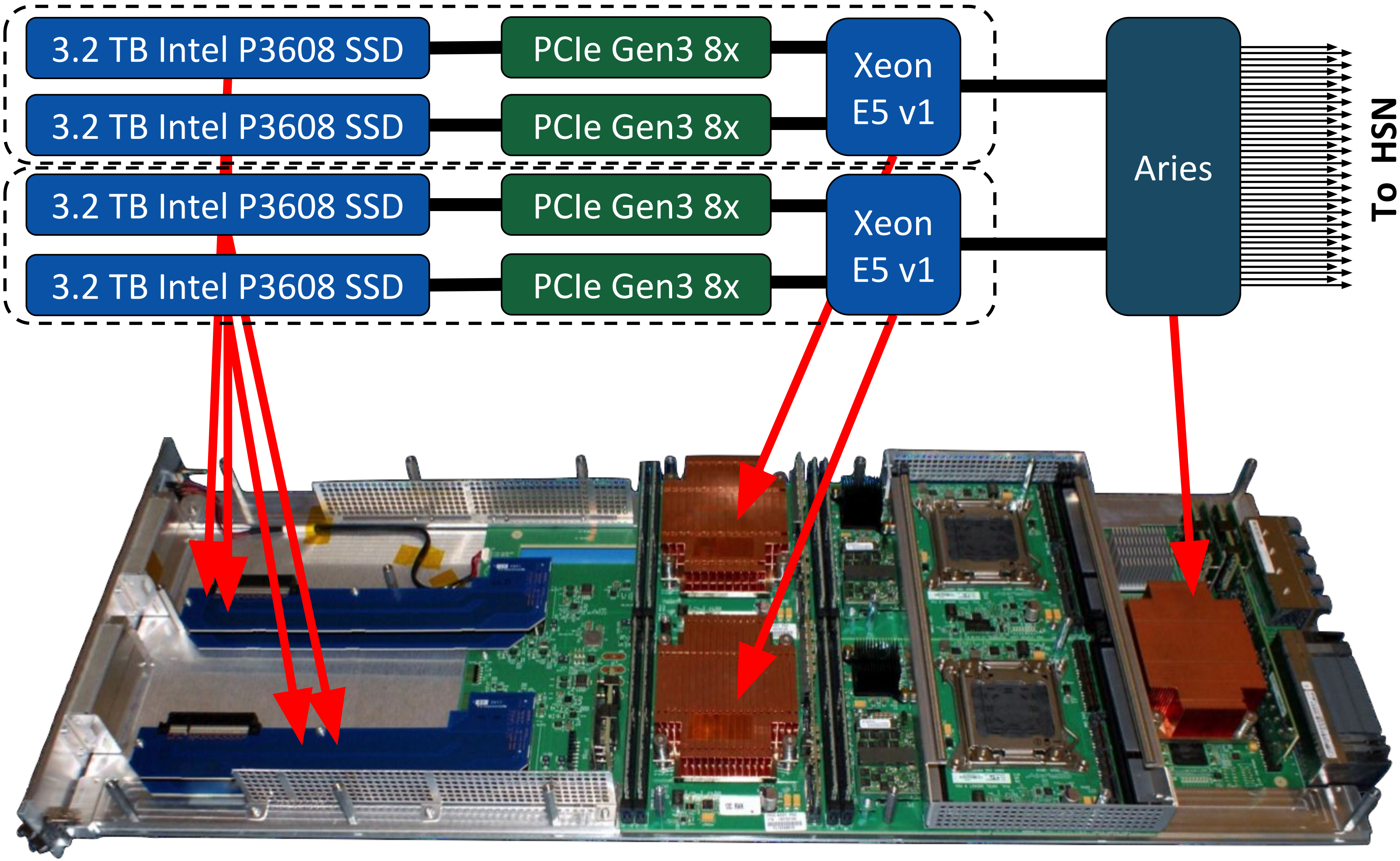
Photo from Glenn Lockwood

BB nodes scattered throughout HSN fabric

2 BB blades/chassis (12 nodes/cabinet) in Phase I



Burst Buffer Blade = 2xNodes



Burst Buffer Feature Timeline

NERSC

Stage 3

In-transit processing and filtering

Stage 2

Transparent caching mode

we are here

Stage 1

Striping, "bad" I/O pattern remediation, per-job and persistent allocations

API will provide means to
- set cache size
- set read-ahead window size
- force consistency point (flush)

Stage 0

Static mapping of compute to BB node, manual data migration

3Q14 → 4Q14 → 1Q15 → 2Q15 → 3Q15 → 4Q15 → 1Q16 → 2Q16 → 3Q16

Burst Buffer Early User Program

- Aug 10th: solicited proposals for BB Early Users program.
  - Award of exclusive early use of BB on Cori P1, plus help of NERSC experts to optimise application for BB.
- Selection criteria include:
  - Scientific merit;Computational challenges; Cover range of BB data features; Cover range of DoE Science Offices.
- Great interest from the community, 29 proposals received.
- Decided to support more applications than we'd originally anticipated
  - some applications already had LDRD funding at LBNL, and existing support from NERSC staff.
- ~20 applications not supported by NERSC staff, but do have early access to Cori P1 and the BB.

Burst Buffer Early Users

Burst Buffer User Case	Active Early Users
IO Bandwidth: Writes (checkpointing)	• Nyx/BoxLib astro simulations • VPIC IO
IO Bandwidth: Reads	• Electron Cryo-microscopy image analysis
"Bad" IO pattern, eg. high IOPs	• Spark analytics framework • ALS TomoPy
Workflow coupling and visualization	• Climate simulation, analysis and visualization
in transit / in-situ analysis	• ChomboCrunch / VisIt carbon sequestration simulation
Staging intermediate data	• Phoenix3D radiation transport simulation • ALICE data analysis (HEP)

~40 active Burst Buffer Users

Challenges ...

- **Initial instabilities resolved in early patches**
  - Early Users are the best testers of a new system!
  - New issues cropping up as use patterns are extended
- **Minor usability issues being addressed by Cray**
  - Syntax sensitivities, informative error codes, improved documentation...
- **Work is on-going, in collaboration with Cray, to understand user application IO patterns and performance**
  - optimising configuration of hardware/software for widely varying use cases is an interesting challenge
  - e.g. balance of performance in writing large vs small files

Performance testing ongoing

- **Burst Buffer is exceeding (nearly all) benchmark performance targets**
  – Work on-going to improve MPIO shared file write
  – IOPs particularly impressive
  – Out-performs Lustre (Lustre also exceeds requirements)

	140 Burst Buffer Nodes : 1120 Compute Nodes ; 4 processes/node					
	IOR Posix FPP		IOR MPIO Shared File		IOPS	
	Read	Write	Read	Write	Read	Write
Required (GB/s) or IOPS	820	820	656	656	7200000	7200000
Best Measured	905	873	803	351	12591978	12527427
Lustre (peak SOW)	708	751	573	223	-	-

*Bandwidth tests: *8 GB block-size 1MB transfers IOPS tests: 1M blocks 4k transfer*

Example application (preliminary)

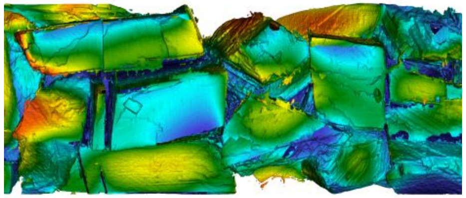
ChomboCrunch

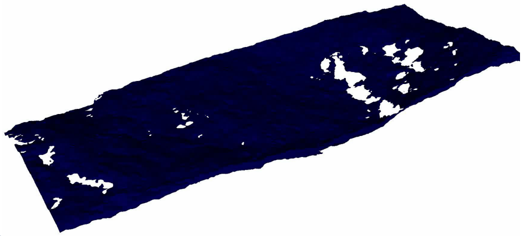
Andrey Ovsyannikov, LBL

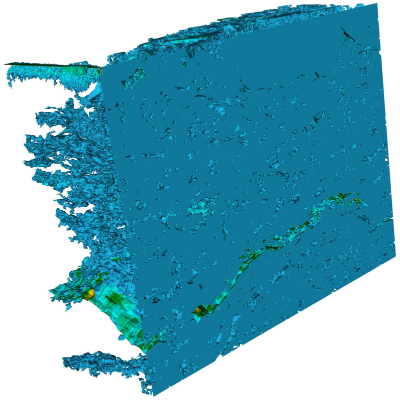
Simulates pore-scale reactive transport processes associated with carbon sequestration

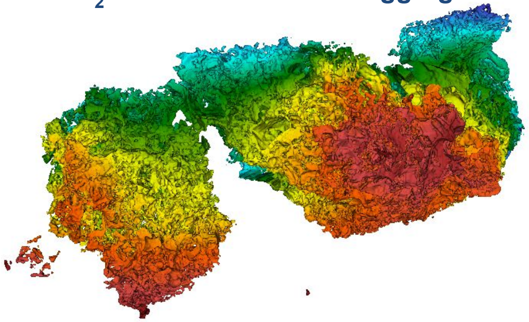
- Applied to other subsurface science areas:
  - Hydrofracturing (aka “fracking”)
  - Disposing of used fuel
- Extended to engineering applications
  - Lithium ion battery electrodes
  - Paper manufacturing (hpc4mfg)

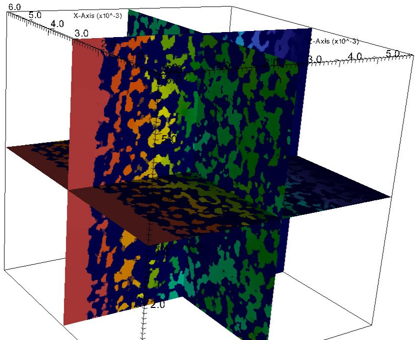
- *Common feature: ability to perform direct numerical simulation from image data of arbitrary heterogeneous, porous materials*

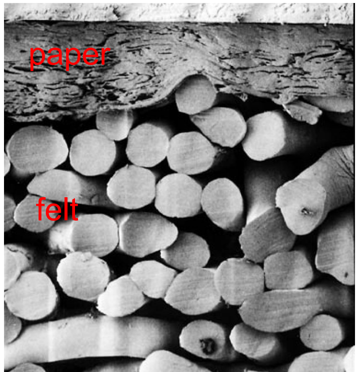

pH on crushed calcite in capillary tube


Transport in fractured dolomite


Flooding in fractured Marcellus shale


O_2 diffusion in Kansas aggregate soil

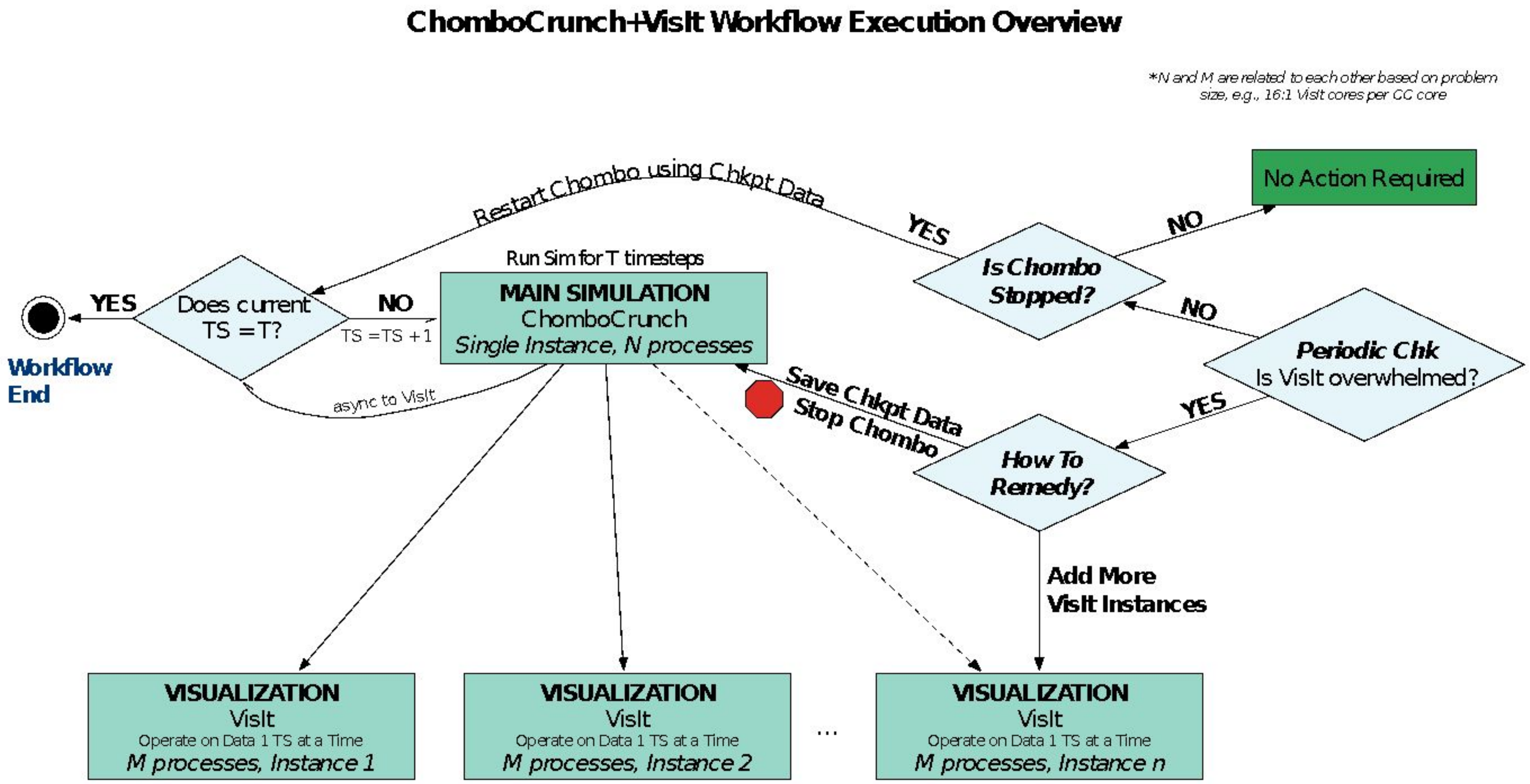

Electric potential in Li-ion electrode



Paper re-wetting

Example application (preliminary)

ChomboCrunch + VisIt workflow



Example application (preliminary)

- **Running more MPI processes gives higher-resolution simulations -> produces larger output plot files**
- Benchmark: reactive flow in a cylinder channel randomly packed by a set of spheres
  - Mesh resolution varies from 256^3 (512 MPI ranks) to 2048*512*512 (16384 MPI ranks)

# MPI ranks	File size	Requested BB capacity	**BB BW**	Lustre BW
512	7.4GB	1 BB node	**1.8GB/s**	0.44GB/s
2048	29.5GB	4 BB nodes	**4.7GB/s**	1.5GB/s
16384	236 GB	16 BB nodes	**34.2GB/s**	8.4GB/s

Burst Buffer enables high-resolution analysis of simulations

User Experience so far (tentative)

- **Writing large files (with large block I/O) is fast (checkpointing use case)**
- **Reading/Writing small files (or small I/O transfers) is problematic in some cases**
  - Generally in many cases our BB performance is worse than our Lustre filesystem (which is high-performance).
  - Client-side caching helps Lustre performance
- **Some jobs with many files fail at large scale on compute nodes**
- **Still some system instabilities**

Working to profile applications I/O and tune burst buffer stripe sizes and other configurables...

Expected performance improvements

1. DVS client-side caching

- Lustre has client-side caching, currently DVS does not
- Will help small R/W transfers on BB
- Can currently use "iobuf" library for user-side caching

2. Smaller granularity

- "Grain" is minimum amount of space allocated on each BB node, currently 213GB
  - E.g. request 500GB BB allocation - get 3x213GB "grains".

3. MPI-IO shared file performance

- Change underlying file striping on BB

We're working with Cray to improve BB performance

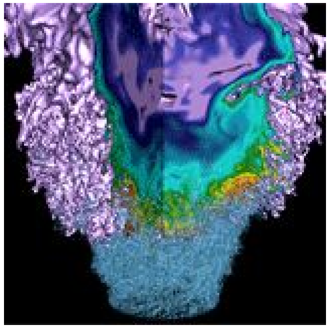
Timeline for Access

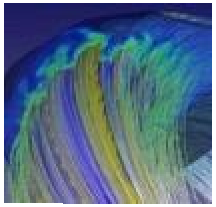
- **Currently only Early Users have access**
- **Plan to allow all users access shortly after we've accepted the BB**
  - Dependent on system being stable - i.e. no kernel panics that bring down compute nodes, major bugs fixed.

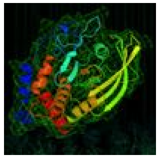
Summary

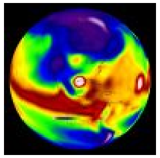
- **BB/DataWarp functional and meets most SOW performance targets**
  - out-performs Lustre in benchmarks
- **Variety of issues affecting user progress**
  - but these are being resolved and users making some progress
- **Some performance pay-off for real use-cases but not all.**
  - We are learning/tweaking configuration
  - Working with Cray to implement performance improvements
- **Profiling I/O patterns to optimise and starting to build whole workflows using the Burst Buffer.**

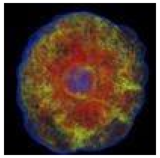
Thankyou

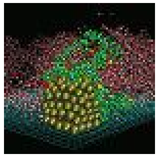












NERSC 40 YEARS at the FOREFRONT

U.S. DEPARTMENT OF ENERGY | Office of Science

BERKELEY LAB

Use Cases by BB feature

Application	I/O bandwidth: reads	I/O bandwidth: writes (checkpointing)	High IOPs	Workflow coupling	In-situ / in-transit analysis and visualization	Staging intermediate files/ pre-loading data
Nyx/Boxlib		X		X	X	
Phoenix 3D		X		X		X
Chomo/Crunch + Visit		X		X	X	
Sigma/UniFam/Sipros	X	X	X			X
XGC1	X	X				X
PSANA				X	X	X
ALICE	X					
Tractor			X	X		X
VPIC/IO					X	X
YODA			X			X
ALS SPOT/TomoPy	X			X	X	X
kitware				X	X	

Use Cases by BB feature

Application	I/O bandwidth: reads	I/O bandwidth: writes (checkpointing)	High IOPs	Workflow coupling	In-situ / in-transit analysis and visualization	Staging intermediate files/ pre-loading data
Electron cryo-microscopy						X
htslib						X
Falcon	X	X				
Ray/HipMer	X	X	X			X
CESM	X	X				
ACME/UV-CDAT					X	X
GVR		X				
XRootD				X		X
OpenSpeedShop	X	X				
DL-POLY		X				
CP2K		X				
ATLAS	X		X			X