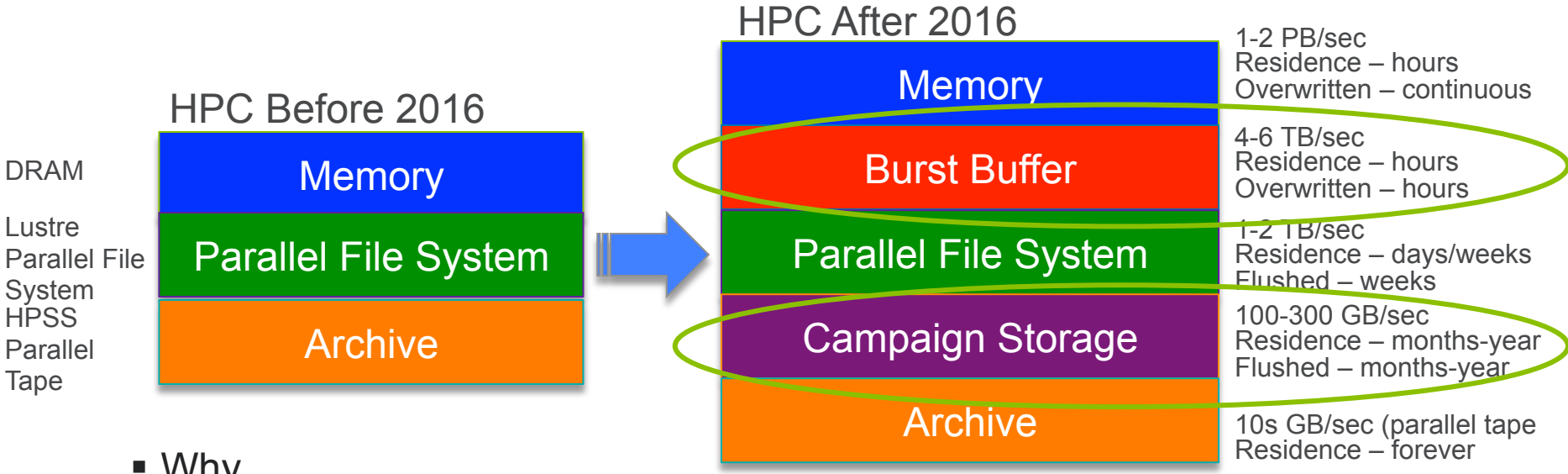
Argonne NATIONAL LABORATORY

FROM FILE SYSTEMS TO SERVICES:
CHANGING THE DATA MANAGEMENT MODEL IN HPC

ROB ROSS, PHILIP CARNS, KEVIN HARMS, JOHN JENKINS, AND SHANE SNYDER	Argonne National Laboratory
GARTH GIBSON, GEORGE AMVROSIADIS, CHUCK CRANOR, AND QING ZHENG	Carnegie Mellon University
JEROME SOUMAGNE AND JOE LEE	The HDF Group
GALEN SHIPMAN AND BRAD SETTLEMYER	Los Alamos National Laboratory

CHANGES IMPACTING HPC DATA AND STORAGE

MORE STORAGE/MEMORY LAYERS…

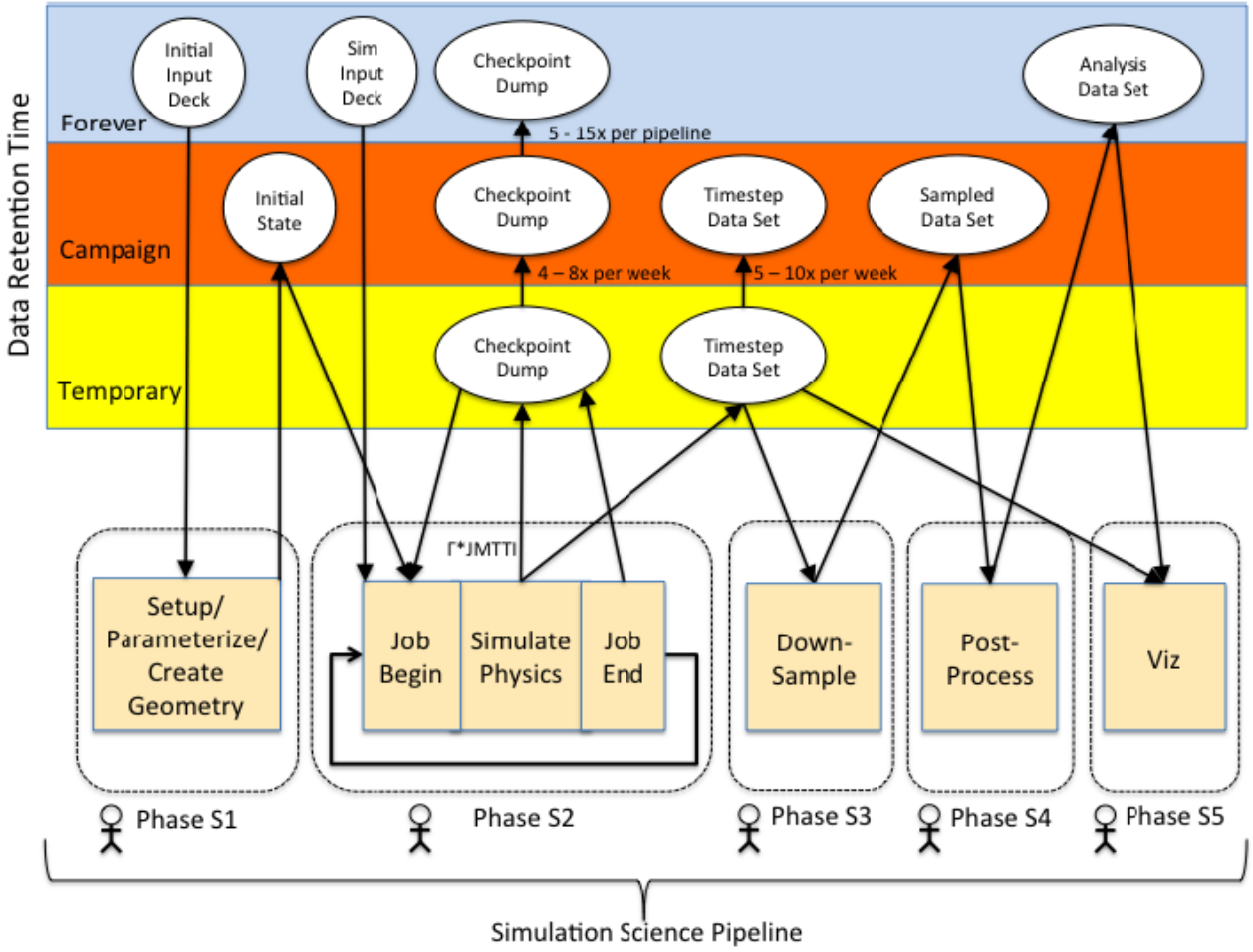


- Why
  - BB: Economics (disk bw/iops too expensive)
  - PFS: Maturity and BB capacity too small
  - Campaign: Economics (tape bw too expensive)
  - Archive: Maturity and we really do need a “forever”

Slide from Gary Grider (LANL).

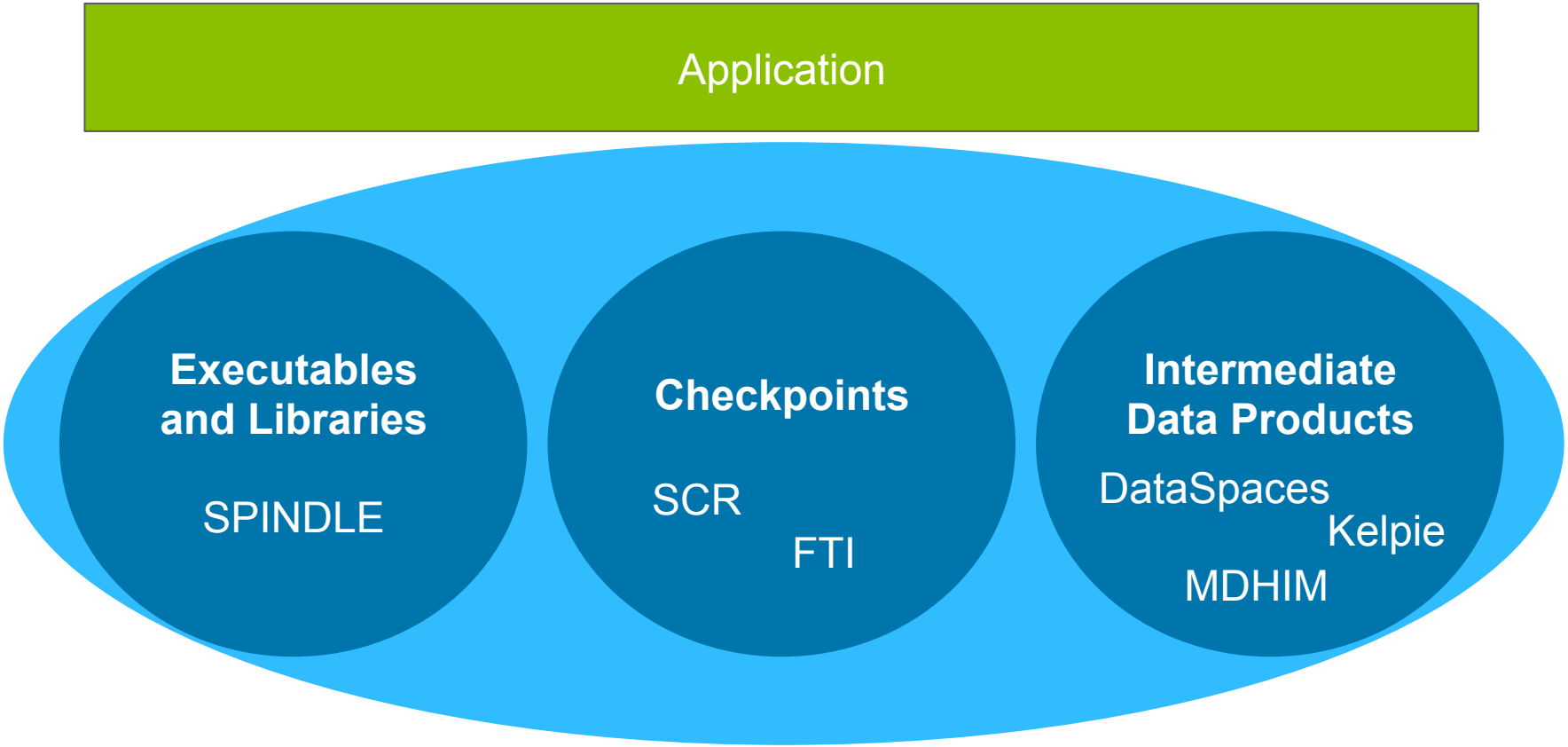
SIMULATION WORKFLOW

APEX Workflows, LANL, NERSC, SNL,
SAND2015-10342 O, LA-UR-15-29113



Argonne NATIONAL LABORATORY

SPECIALIZATION OF DATA SERVICES



	Provisioning	Comm.	Local Storage	Fault Mgmt. and Group Membership	Security
Rusty → **ADLB** *Data store and pub/sub.*	MPI ranks	MPI	RAM	N/A	N/A
DataSpaces ← Manish *Data store and pub/sub.*	Indep. job	Dart	RAM (SSD)	Under devel.	N/A
DataWarp *Burst Buffer mgmt.*	Admin./ sched.	DVS/ lnet	XFS, SSD	Ext. monitor	Kernel, lnet
FTI ← Franck *Checkpoint/restart mgmt.*	MPI ranks	MPI	RAM, SSD	N/A	N/A
Kelpie *Dist. in-mem. key/val store*	MPI ranks	Nessie	RAM (Object)	N/A	Obfusc. IDs
SPINDLE *Exec. and library mgmt.*	Launch MON	TCP	RAMdisk	N/A	Shared secret

COMPOSING DATA SERVICES

OUR GOAL

Enable composition of data services for DOE science and systems

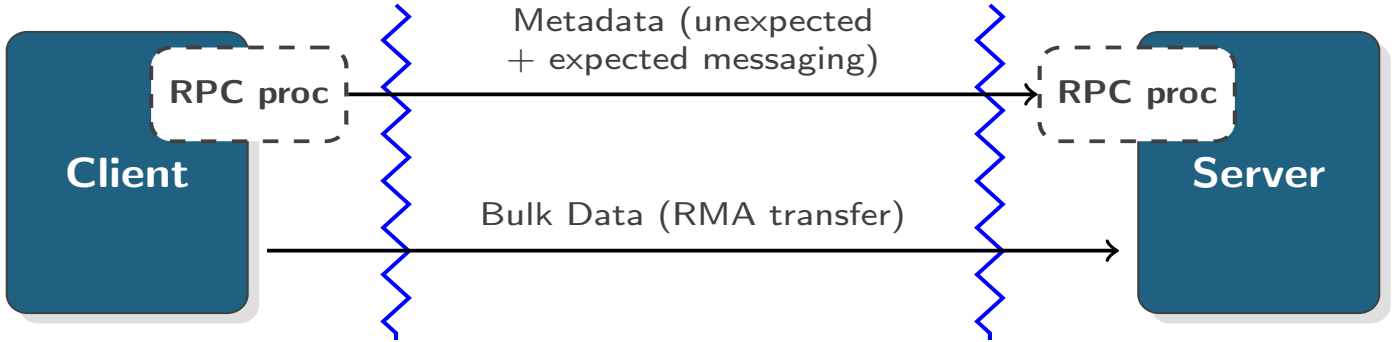
- Application-driven
  - Identify and match to science needs
  - Traditional data roles (e.g., checkpoint, data migration)
  - New roles (e.g., equation of state/opacity databases)
- Develop/adapt building blocks
  - **Communication**
  - **Concurrency**
  - Local Storage
  - Resilience
  - Authentication/Authorization

COMMUNICATION: MERCURY

https://mercury-hpc.github.io/

Mercury is an RPC system for use in the development of high performance system services. Development is driven by the HDF Group with Argonne participation.

- Portable across systems and network technologies
- Efficient bulk data movement to complement control messages
- Builds on lessons learned from IOFSL, Nessie, lnet, and others



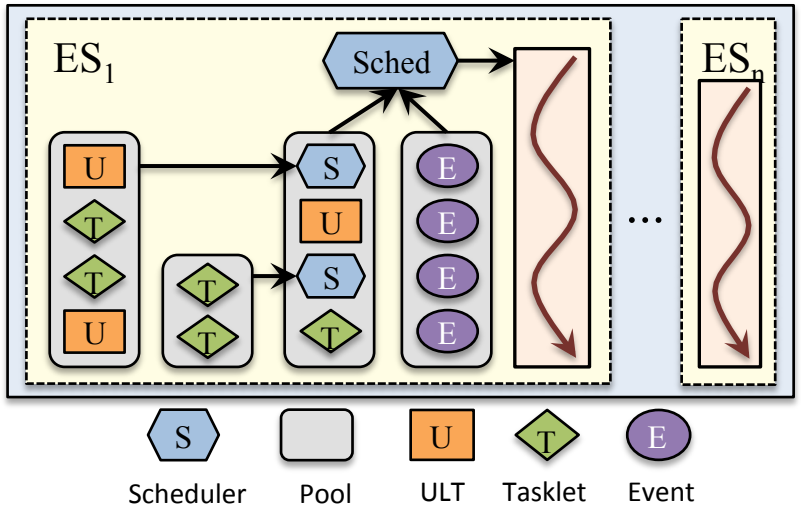
CONCURRENCY: ARGOBOTS

https://collab.cels.anl.gov/display/argobots/

Argobots is a lightweight threading/tasking framework.

- Features relevant to I/O services:
  - Flexible mapping of work to hardware resources
  - Ability to delegate service work with fine granularity across those resources
  - Modular scheduling
- We developed asynchronous bindings to:
  - Mercury
  - LevelDB
  - POSIX I/O
- Working with Argobots team to identify needed functionality (e.g., idling)

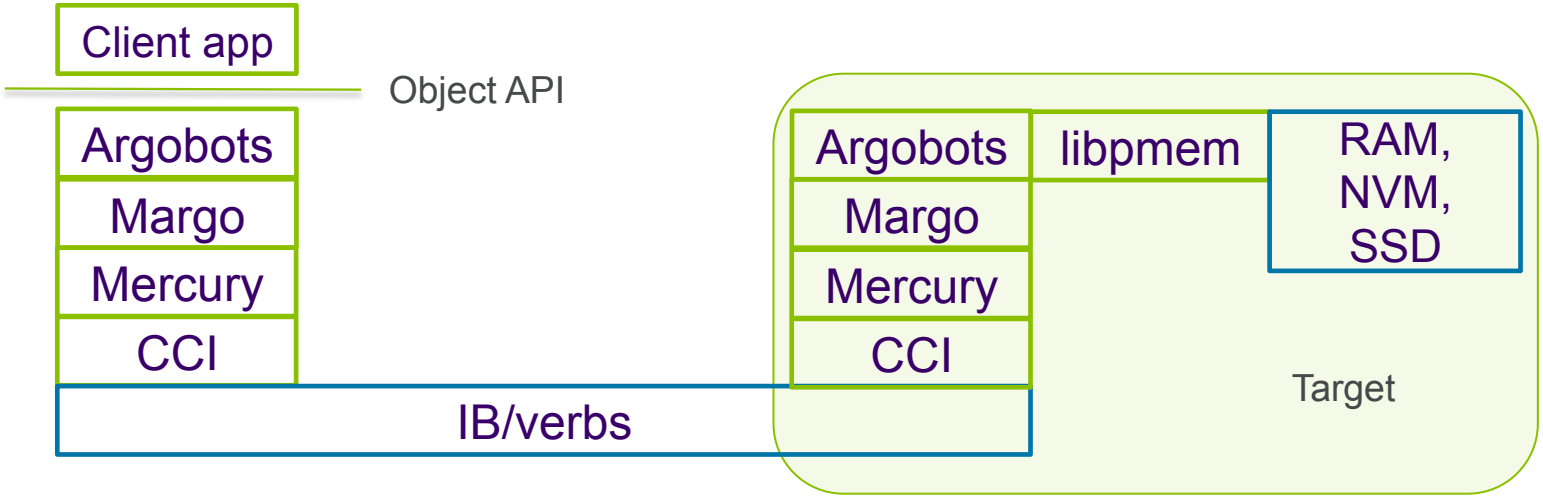
Argobots Execution Model



THREE EXAMPLE SERVICES

1. REMOTELY ACCESSIBLE OBJECTS

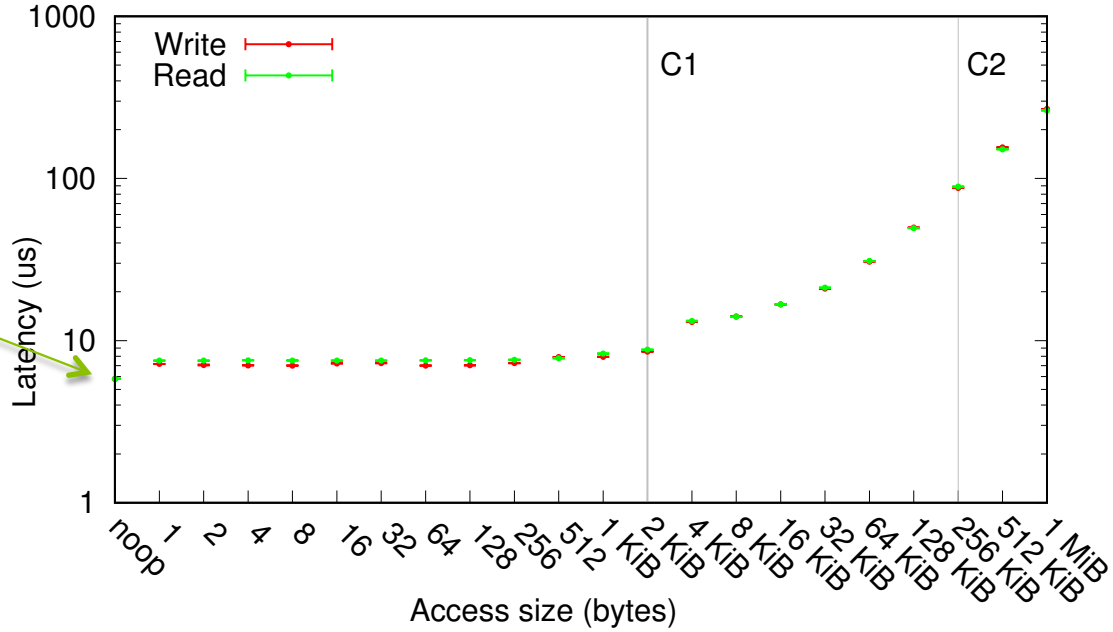
- API for remotely creating, reading, writing, destroying fixed-size objects/extents
- libpmem (http://pmem.io/nvml/libpmemobj/) for management of data on device



P. Carns et al. “Enabling NVM for Data-Intensive Scientific Services.” INFLOW 2016, November 2016.

1. REMOTELY ACCESSIBLE OBJECTS: HOW MUCH LATENCY IN THE STACK?

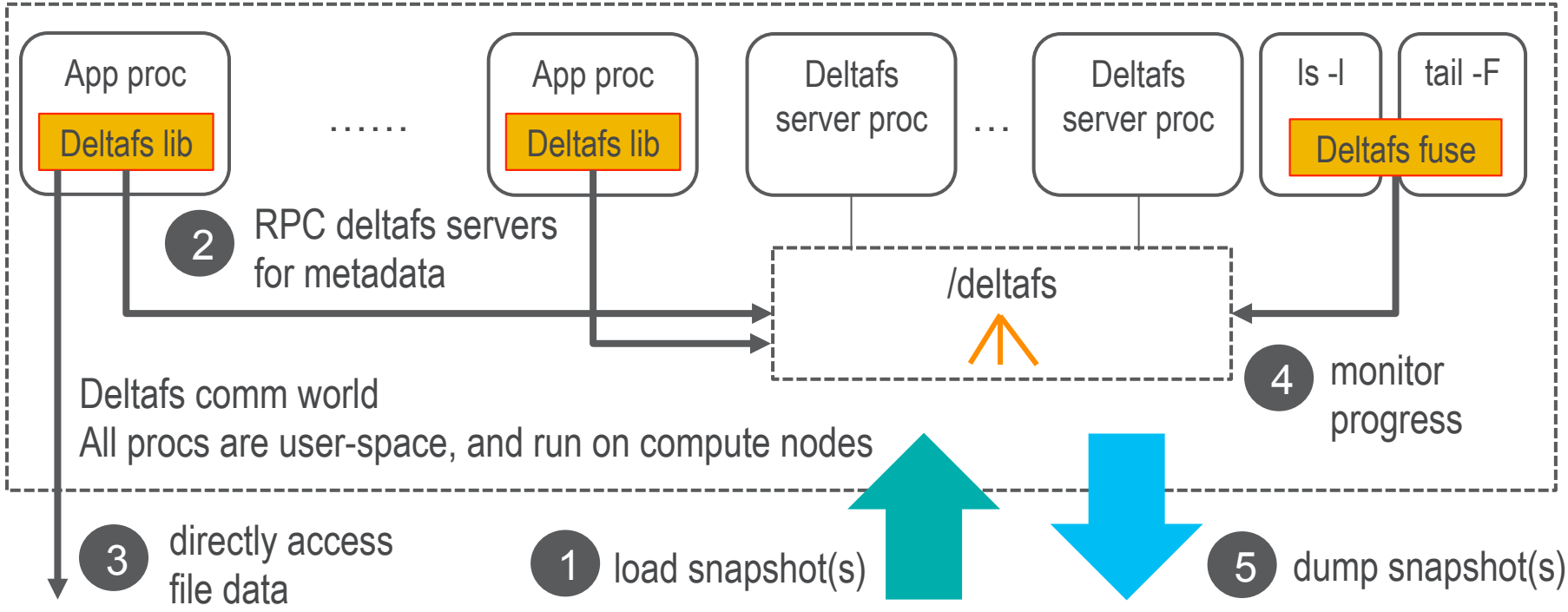
5.8 usec NOOP



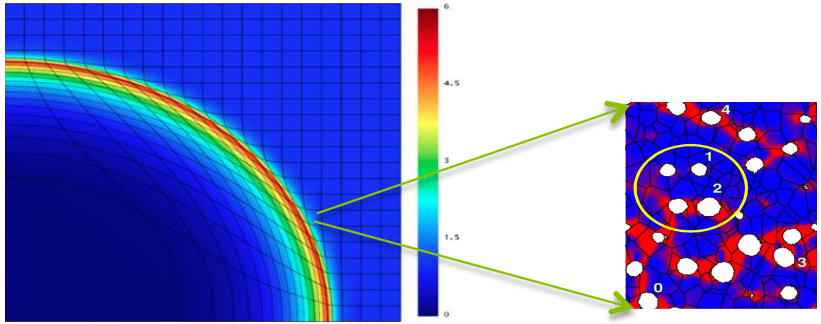
FDR IB, RAM disk, 2.6 usec round-trip (MPI) latency measured separately

2. TRANSIENT FILE SYSTEM VIEWS: DELTAFS

Supporting legacy POSIX I/O in a scalable way.

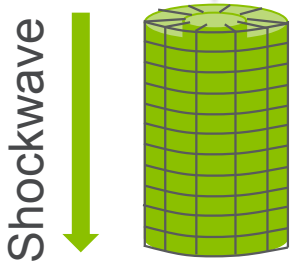


3. CONTINUUM MODEL COUPLED WITH VISCOPLASTICITY MODEL



Lulesh continuum model:
- Lagrangian hydro dynamics
- Unstructured mesh

Viscoplasticity model [1]:
- FFT based PDE solver
- Structured sub-mesh

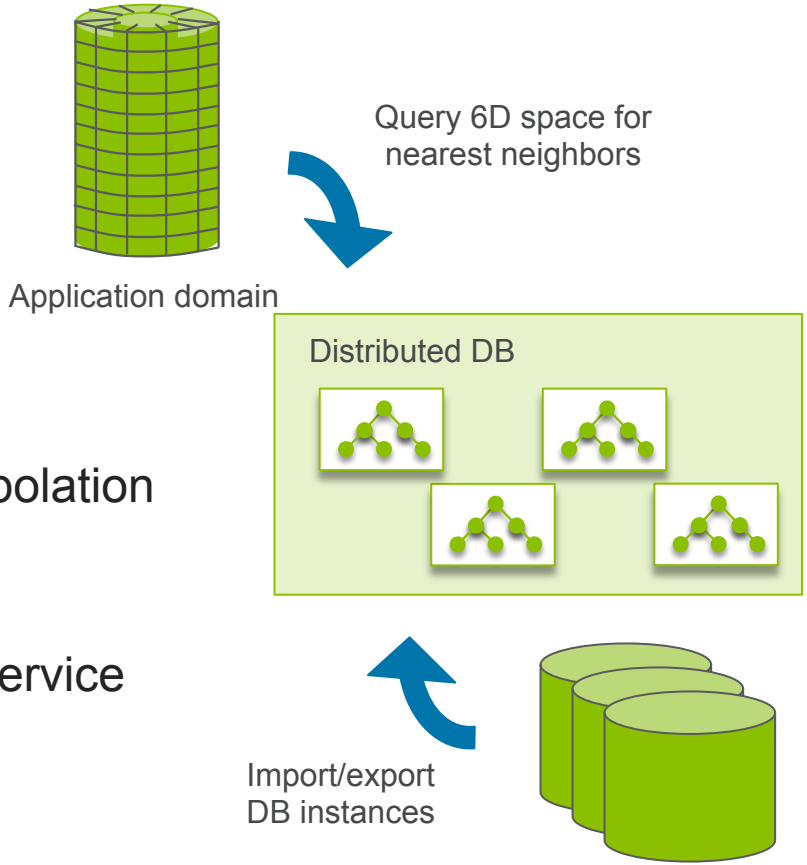


- Future applications are exploring the use of multi-scale modeling
- As an example: Loosely coupling continuum scale models with more realistic constitutive/response properties
  - e.g., Lulesh from ExMatEx
- Fine scale model results can be cached and new values interpolated from similar prior model calculations

R. Lebensohn et al, Modeling void growth in polycrystalline materials, Acta Materialia, http://dx.doi.org/10.1016/j.actamat.2013.08.004.

3. FINE SCALE MODEL DATABASE

- Goals
  - Minimize fine scale model executions
  - Minimize query/response time
  - Load balance DB distribution
- Approach
  - Start with a key/value store
  - Distributed approx. nearest-neighbor query
  - Data distributed to co-locate values for interpolation
  - Import/export to persistent store
- Status
  - Mercury-based, centralized in-memory DB service
  - Investigating distributed, incremental nearest-neighbor indexing



FINAL THOUGHTS

- Stage is set for distributed services in HPC
  - Richer resource management
  - Increasing emphasis on workflows
  - Convergence of data intensive and computational science

- If we're going to "get rid of POSIX", we need alternative(s)

- Real opportunity to make life easier for applications
  - And have fun doing it!

THIS WORK IS SUPPORTED BY THE DIRECTOR, OFFICE OF ADVANCED SCIENTIFIC COMPUTING RESEARCH, OFFICE OF SCIENCE, OF THE U.S. DEPARTMENT OF ENERGY UNDER CONTRACT NO. DE-AC02-06CH11357.

Argonne NATIONAL LABORATORY