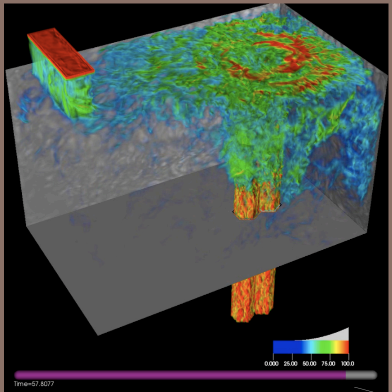
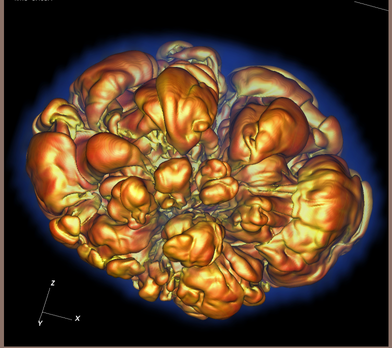
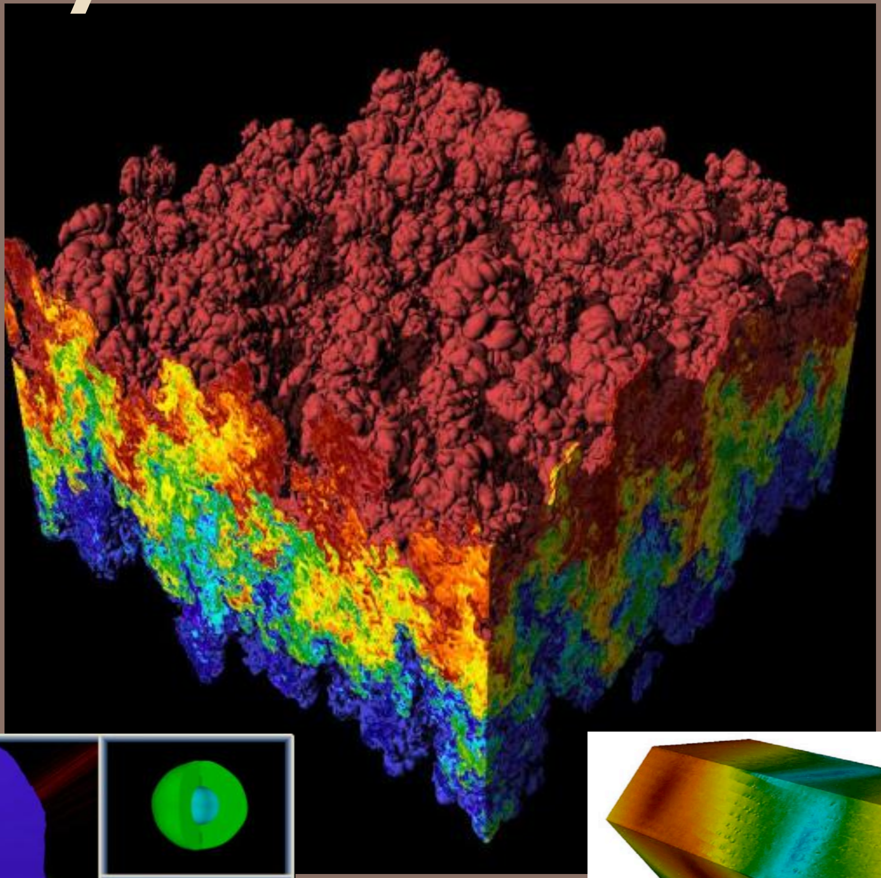
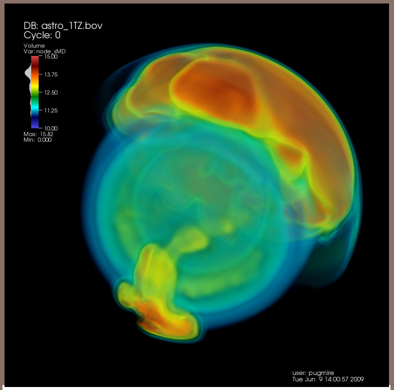
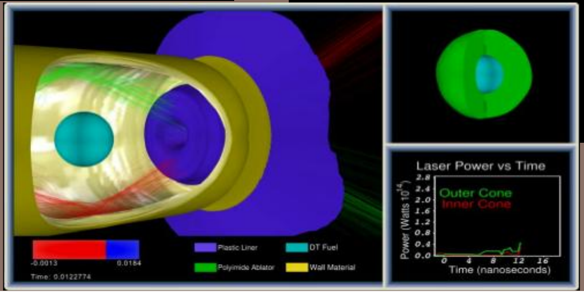
Exascale Analysis & Visualization: Get Ready For a Whole New World

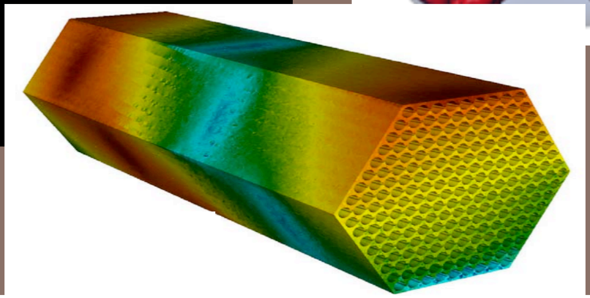












Sept. 16, 2015

Hank Childs, University of Oregon

Before I forget…

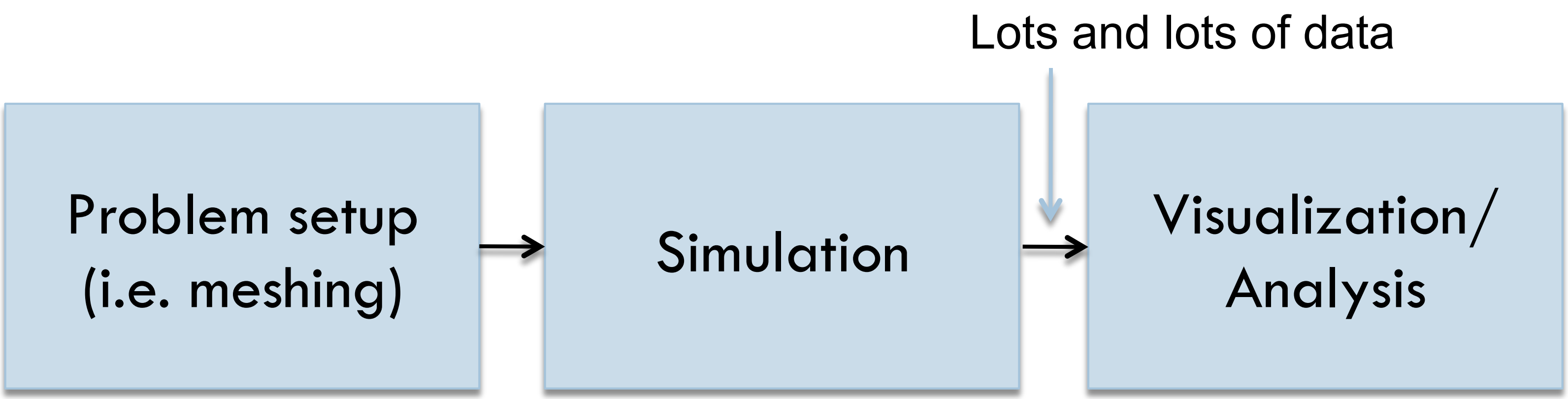
- VisIt: visualization and analysis for very big data
- DOE Workshop for Data Management, Analysis, and Visualization

What I will be talking about

- Theme: looking ahead to the exascale era, and thinking about visualization and analysis
- Talking about research challenges
- (Occasionally) discussing recent research results

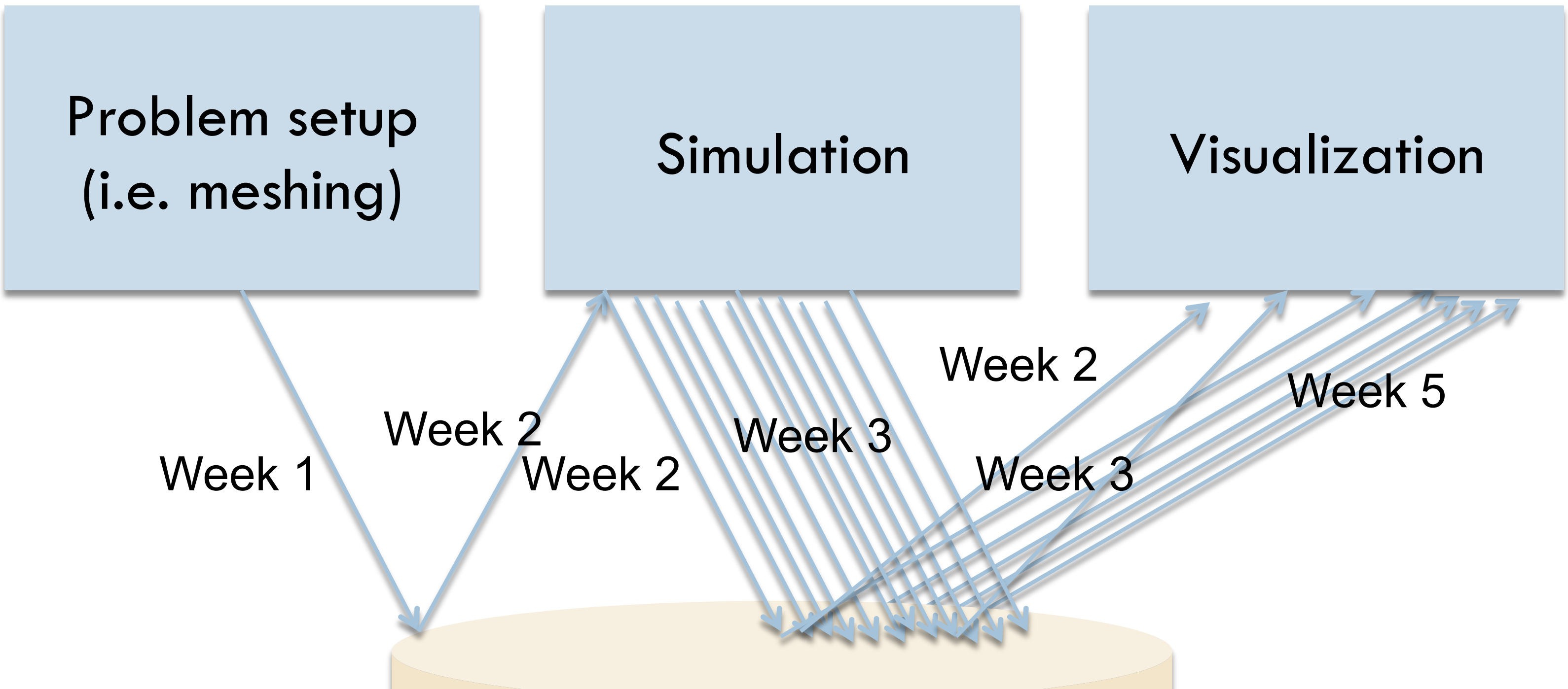
Visualization & analysis is a key aspect of the simulation process.

- Three main phases to simulation process:



- Visualization & analysis is used in three distinct ways:
  - Scientists <u>confirm</u> their simulation is running correctly.
  - Scientists <u>explore</u> data, leading to new insights.
  - Scientists <u>communicate</u> simulation results to an audience.

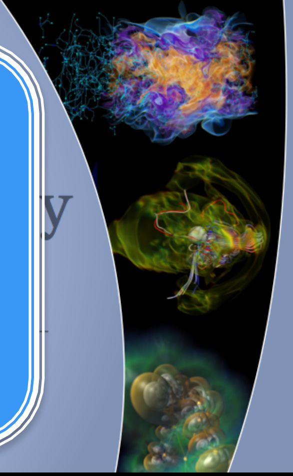
Post-Hoc Analysis (Post-processing)



Note: simulations produce data every cycle, but only store data infrequently (e.g., every Nth cycle)

Starting
exascale
the Sl

I believe some of the things that scared us about exascale will happen well before exascale.



System Parameter	2011	"2018"		Factor Change
System Peak	2 PetaFLOPS	1 ExaFLOP		500
Power	6 MW	≤ 20 MW		3
System Memory	0.3 PB	32 – 64 PB		100 – 200
Total Concurrency	225K	1B × 10	1B × 100	40,000 – 400,000
Node Performance	125 GF	1 TF	10 TF	8 – 80
Node Concurrency	12	1,000	10,000	83 – 830
Network BW	1.5 KB/s	100 GB/s	1000 GB/s	66 – 660
System Size (nodes)	18,700	1,000,000	100,000	50 – 500
I/O Capacity	15 PB	300 – 1000 PB		20 – 67
I/O BW	0.2 TB/s	20 – 60 TB/s		100 – 300

credit: Ken Moreland, Sandia

I/O bandwidth: 0.2TB/s → 20-60TB/s

- Argument #1: “I don’t believe it”

Machine	Year	Time to write memory
ASCI Red	1997	300 sec
ASCI Blue Pacific	1998	400 sec
Jaguar XT4	2007	1400 sec
Roadrunner	2008	1600 sec
Jaguar XT5	2008	1250 sec

ORNL Titan (2012): 1TB/s
ORNL Summit (2017): 1TB/s

c/o Jeremy Meredith & David Pugmire, ORNL

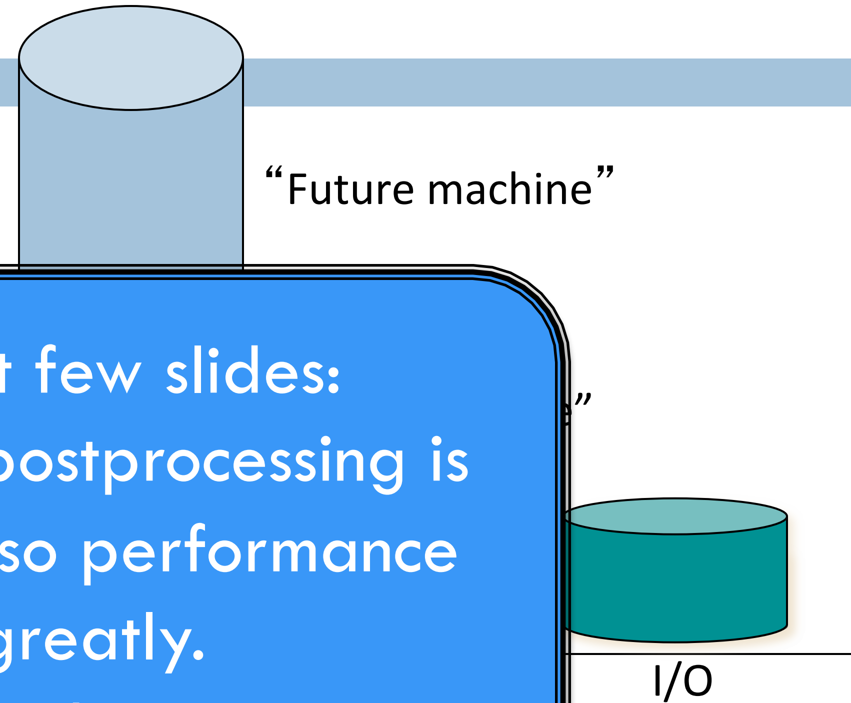
I/O bandwidth: 0.2TB/s → 20-60TB/s

- Argument #2: "it will be even worse than expected because of technology changes"
- Important distinction: WRITE bandwidth vs READ bandwidth
- Old mod
  - Write b
- New mo … e system
  - If bandwidth(SSDs) > bandwidth(PFS) →
    - Write bandwidth > Read bandwidth
  - If bandwidth(SSDs) >> bandwidth(PFS) →
    - Write bandwidth >> Read bandwidth

What are the implications?

I/O and vis/analysis

- Parallel vis & analysis on supercomputer is almost always > sometim
- Amount is typica
- Two big
  1. how much data you have to read
  2. how fast you can read it
- → Relative I/O (ratio of total memory and I/O) is key



Summary of last few slides:
the slowest part of postprocessing is getting slower, and so performance will suffer greatly.
This is bad.

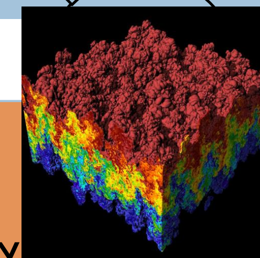
Result: community is moving toward in situ processing

- In situ defined: produce vis & analysis w/o writing to disk
  - Multiple ways to accomplish: loosely coupled vs tightly coupled
- Common perceptions of in situ:
  - Pros:
    - No I/O & plenty of compute
    - Access to more data
  - Cons:
    - Very memory constrained
    - Some operations not possible
      - Once the simulation has advanced, you cannot go back and analyze it
    - User must know what to look at a priori

Do we have our use cases covered? (thinking in broad strokes…)

- Three primary use cases:
  - Exploration
  - Confirmation
  - Communication

?



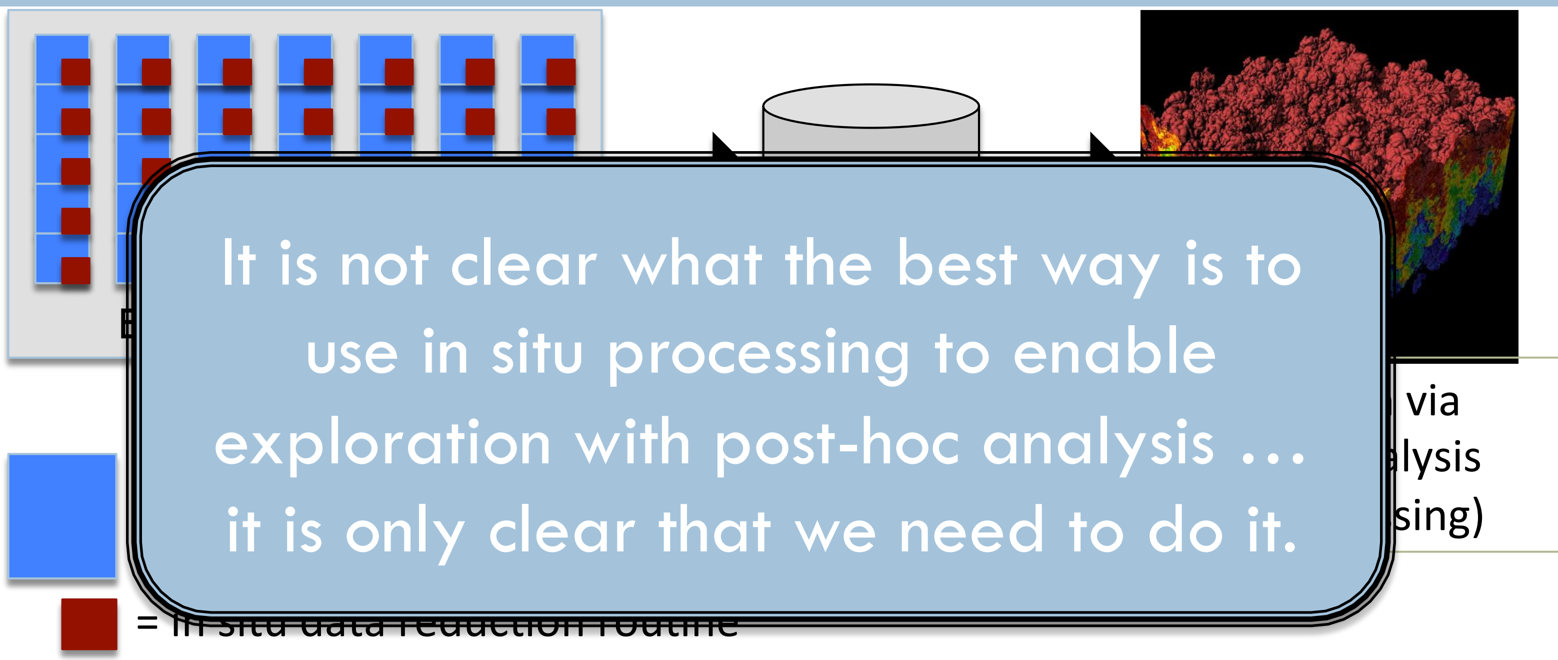
Examples:
Scientific discovery
Debugging

Examples:
Data analysis
Images / movies
Comparison

In situ

Examples:
Data analysis
Images / movies

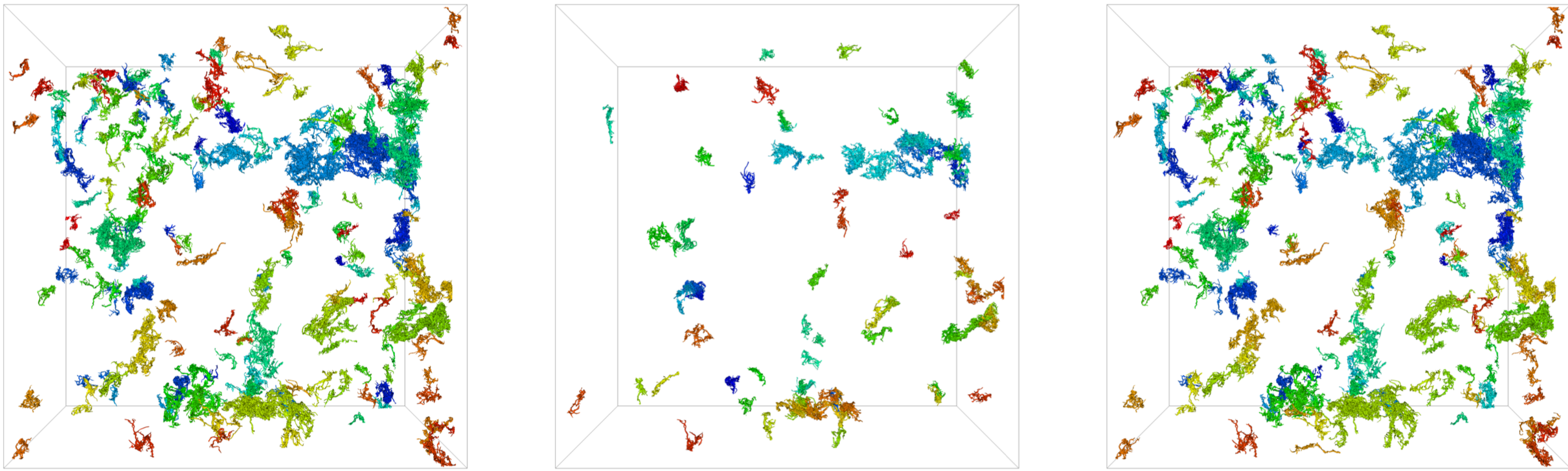
Enabling exploration via in situ processing



It is not clear what the best way is to use in situ processing to enable exploration with post-hoc analysis … it is only clear that we need to do it.

- Requirement: must transform the data in a way that both reduces and enables meaningful exploration.

Select Current Activity #1: Wavelet Compression



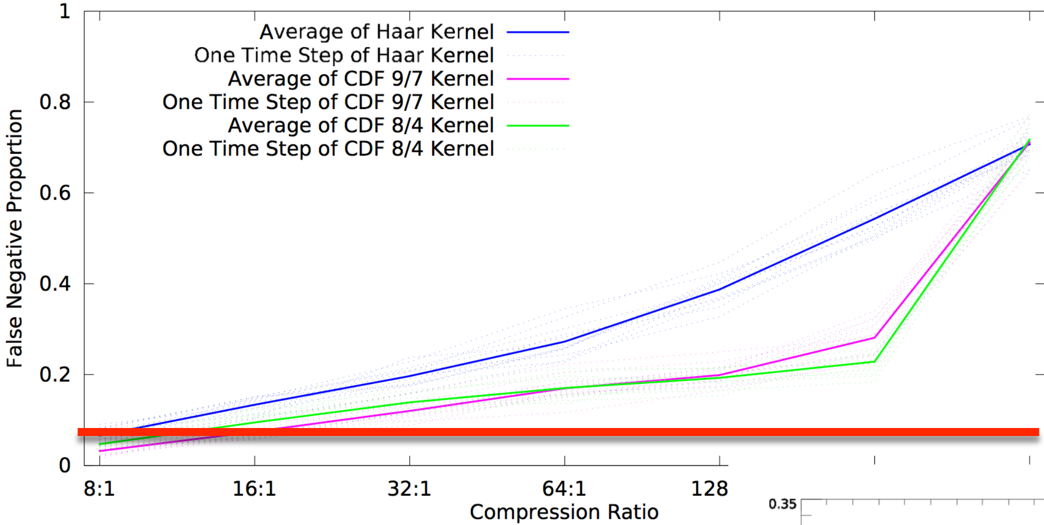
(a) Image from analysis of a $4,096^3$ turbulent flow data set.

(b) 8:1 compressed data using a multi-resolution technique and the Haar kernel.

(c) 128:1 compressed data using prioritized coefficients and the CDF 9/7 kernel.

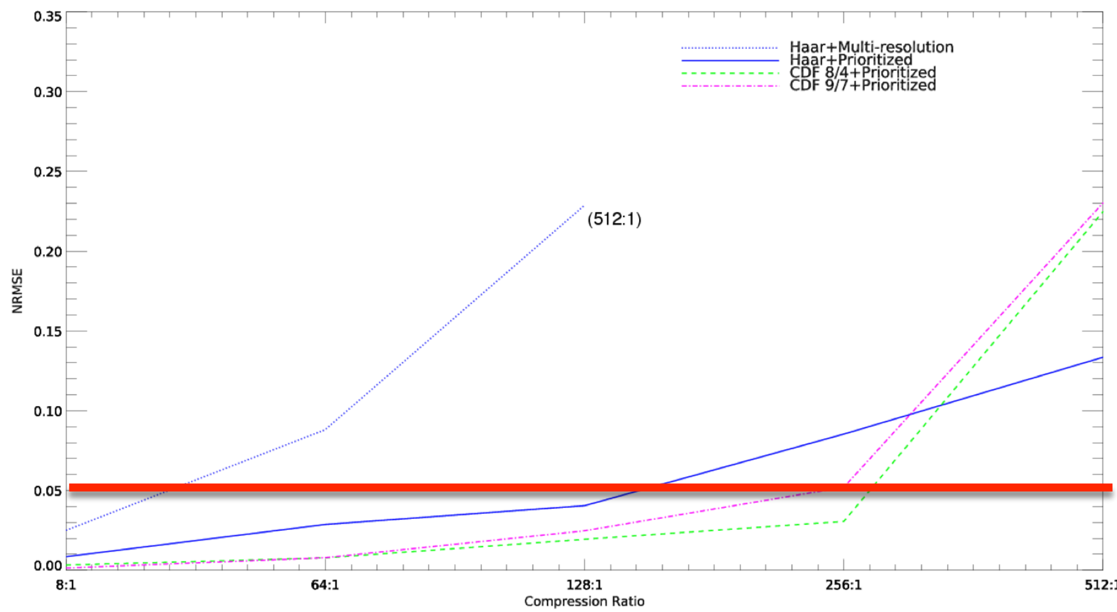
S. Li, K. Gruchalla, K. Potter, J. Clyne, and H. Childs. Evaluating the Efficacy of Wavelet Configurations on Turbulent-Flow Data. In Proceedings of IEEE Symposium on Large Data Analysis and Visualization, Oct. 2015. To appear.

Select Current Activity #1: Wavelet Compression



Analysis #1: error is too big beyond 8:1 compression

Analysis #2: error is acceptable up to 256:1 compression

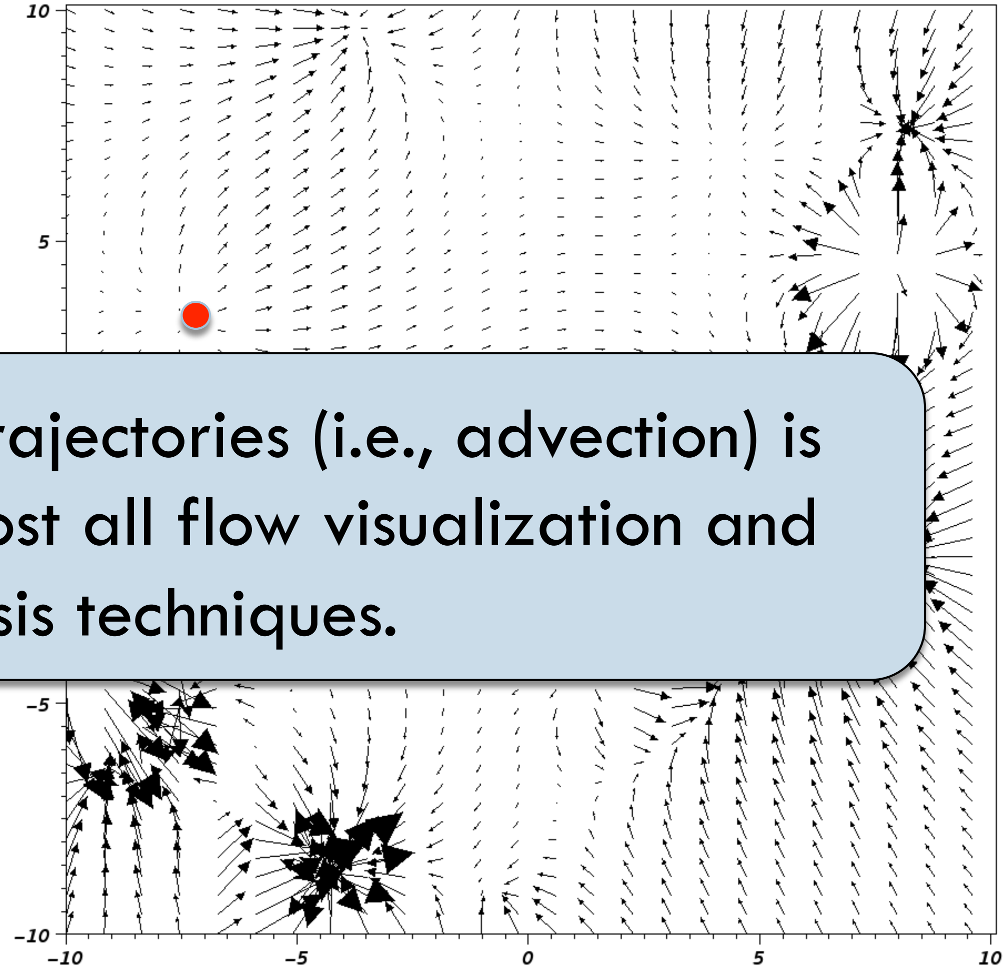


Select Current Activity #2: Particle Advection

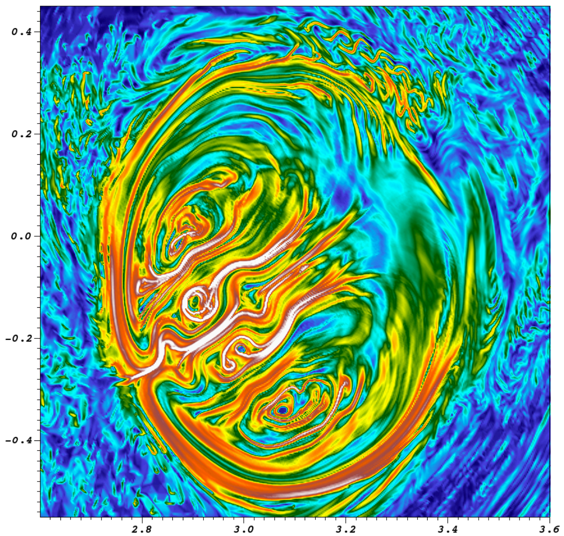
- Displace massless particle based on velocity field
- S(t) = position of curve at time t

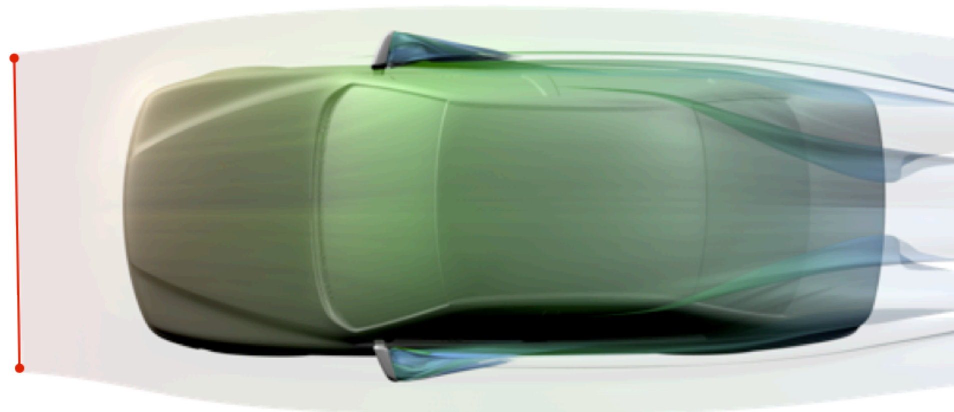
Analyzing particle trajectories (i.e., advection) is foundational to almost all flow visualization and analysis techniques.

  - v(t, p): velocity at time t and position p
  - S'(t): derivative of the integral curve at time t

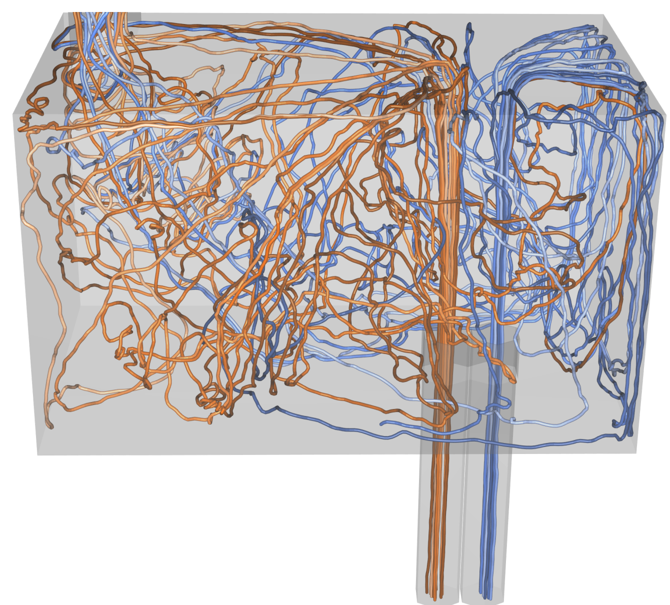


Advection is the basis for many flow visualization techniques





Courtesy Garth, Kaiserslautern



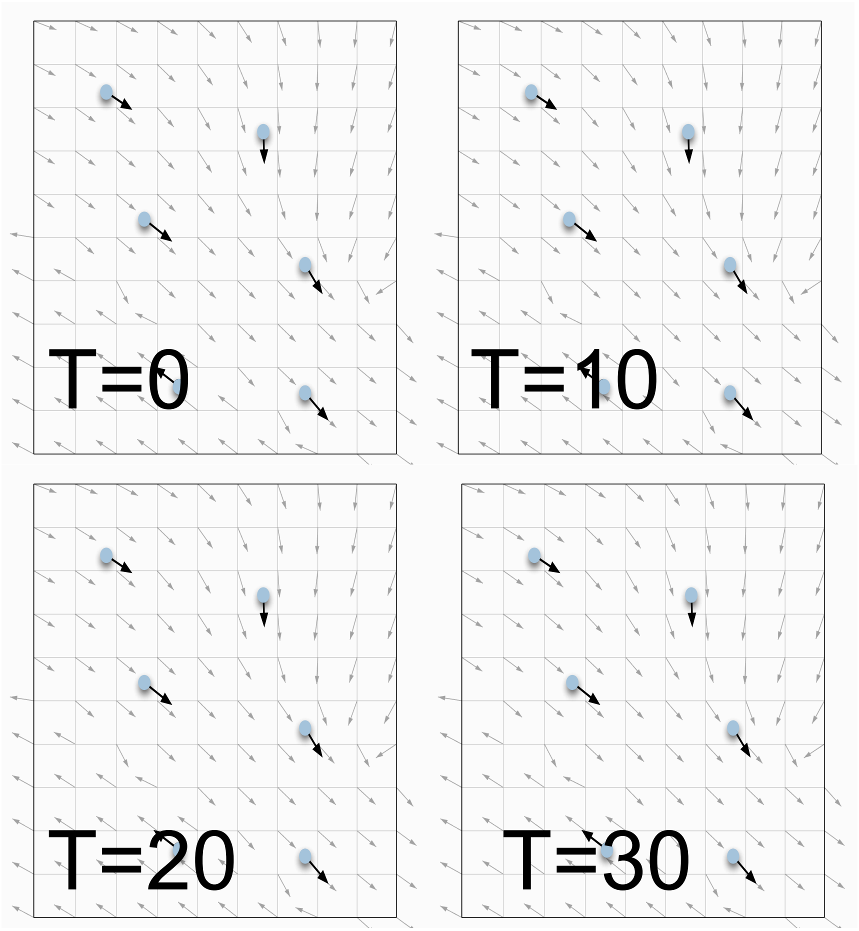
- Takeaways:
  - Used for diverse analyses
  - Diverse computational loads

Select Current Activity #2: Particle Advection

- Post hoc exploratory particle advection of time-varying data
- This means:
  - Save time-varying data from simulation
  - After simulation, explore data using particle advection techniques
    - IMPORTANT: locations of desired particles are not known a priori

A. Agranovsky, D. Camp, C. Garth, E. W. Bethel, K. I. Joy, and H. Childs. Improved Post Hoc Flow Analysis Via Lagrangian Representations. In Proceedings of the IEEE Symposium on Large Data Visualization and Analysis (LDAV), pages 67–75, Paris, France, Nov. 2014. Best paper award.

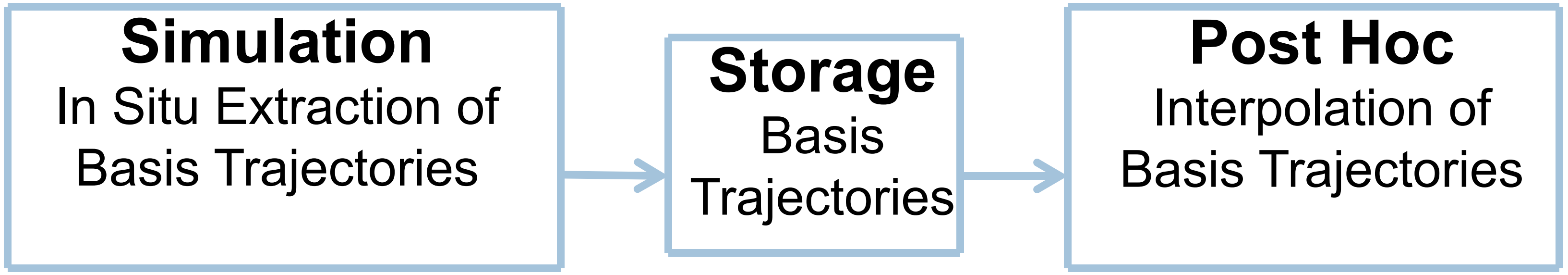
Post Hoc Exploratory Particle Advection: Traditional Method

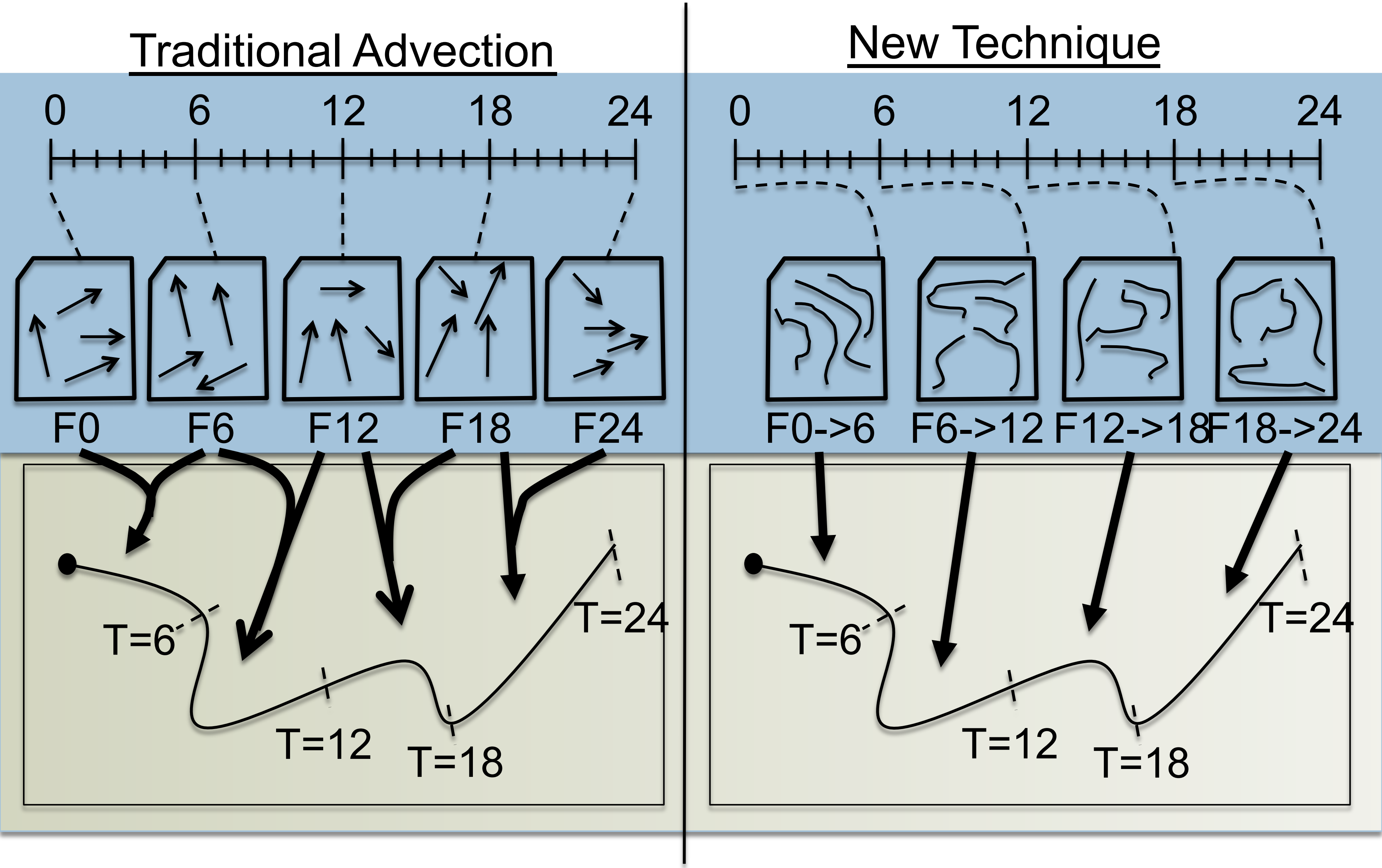


- Save time slices of vector fields
- Integrate (RK4) using interpolation, both spatially and temporally

Our Algorithm – Two Phase

- Phase 1: in situ extraction of basis trajectories
- Phase 2: post-hoc analysis, where new trajectories are interpolated from basis





Traditional Advection
0
6
12
18
24
F0
F6
F12
F18
F24
T=6
T=12
T=18
T=24
New Technique
0
6
12
18
24
F0->6
F6->12
F12->18
F18->24
T=6
T=12
T=18
T=24

Our New Method Provides Significant Improvement Over The Traditional Method

- Accuracy
  - increases of up to 10X, with the same storage as traditional techniques
- Storage
  - decreases of up to 64X, with the same accuracy as traditional techniques
- Speed
  - interaction up to 80X faster
- Keys to success:
  - we can access more data in situ than we could post hoc
  - accepted method (post hoc) producing poor answers

Summary

- “Exascale” ramifications for visualization & analysis:
  - Slow I/O will change our modus operandi
  - So we will need to run in situ
  - And exploration is important
    - So we need to think about operators that transform and reduce data in situ
    - And how they tradeoff between accuracy and data reduction
- Many uncovered topics:
  - SW for many-core architectures, power consumption, specific forms of in situ, fault tolerance, outputs from multi-physics codes, ensemble analysis, etc.