

# Numerical Laboratories on Exascale

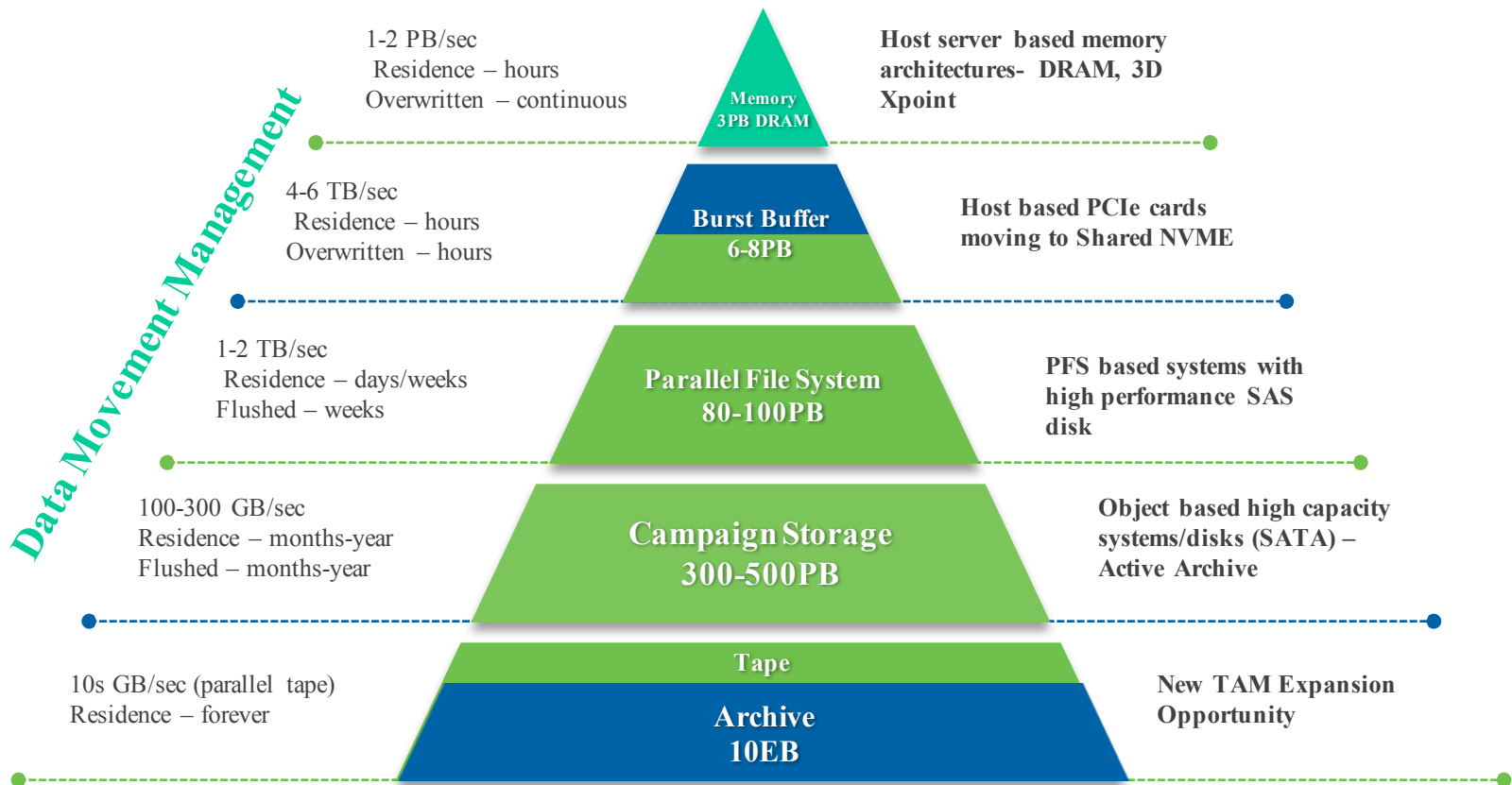
Alex Szalay  
JHU

# Data in HPC Simulations

- HPC is an instrument in its own right
- Largest simulations approach petabytes today
  - *from supernovae to turbulence, biology and brain modeling*
- Need public access to the best and latest through interactive **Numerical Laboratories**
  
- Examples in turbulence, N-body
- Streaming algorithms (annihilation, halo finders)
- Exascale coming

# Towards Exascale

The 'Trinity' System at LANL is leading the way

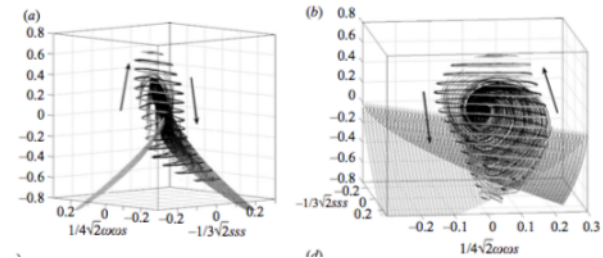


# Immersive Turbulence

“... the last unsolved problem of classical physics...” Feynman

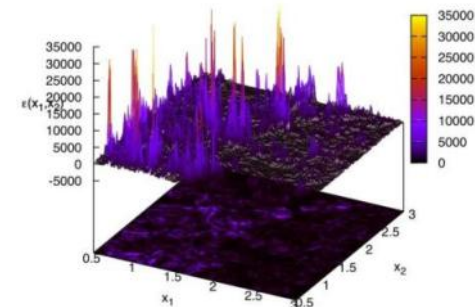
- **Understand the nature of turbulence**

- Consecutive snapshots of a large simulation of turbulence: 30TB
- Treat it as an experiment, **play** with the database!
- **Shoot test particles** (sensors) from your laptop into the simulation, like in the movie *Twister*
- Next step was 50TB MHD simulation
- Now: channel flow 100TB, MHD 256TB



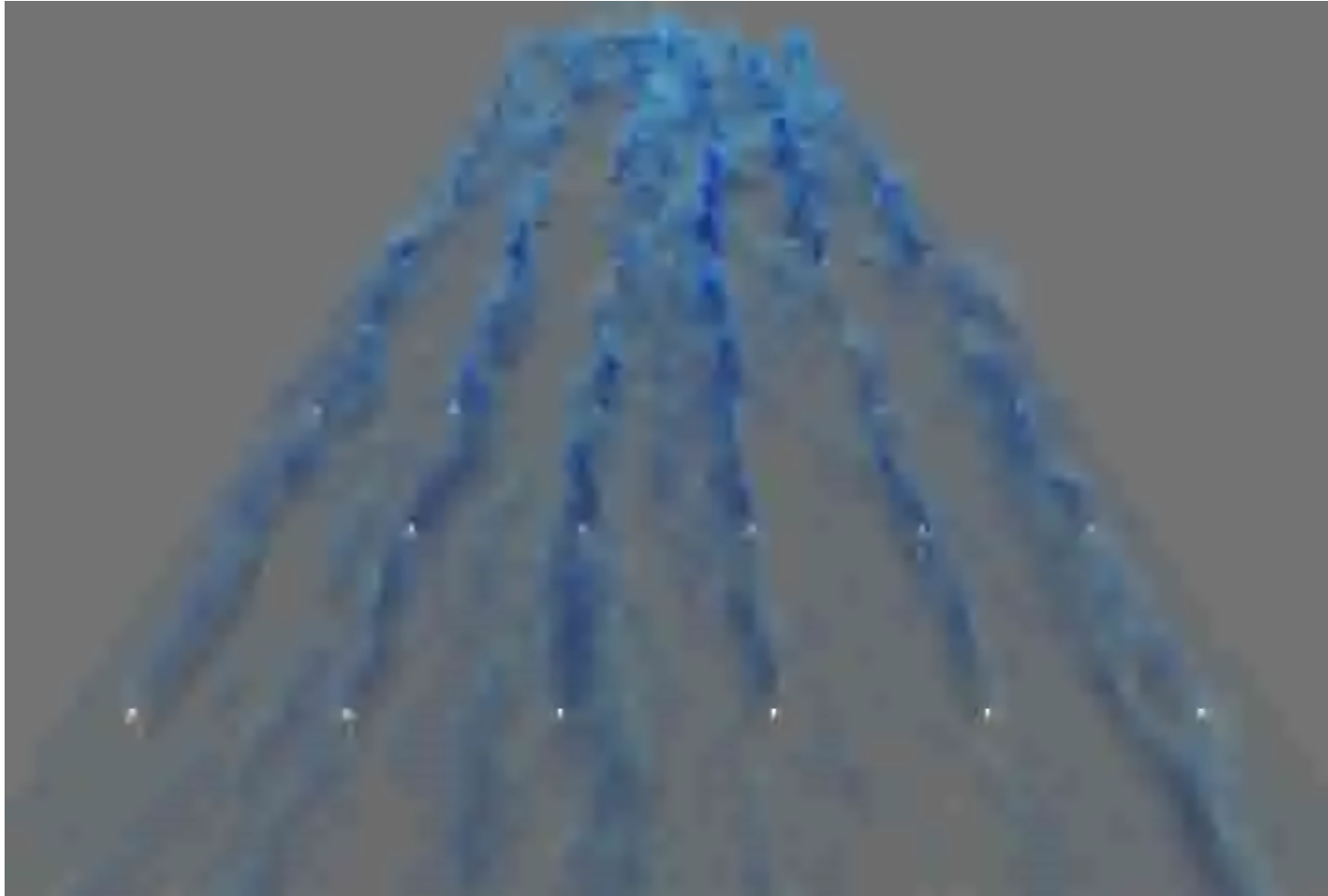
- **New paradigm** for analyzing simulations

**20 trillion points queried to date!**



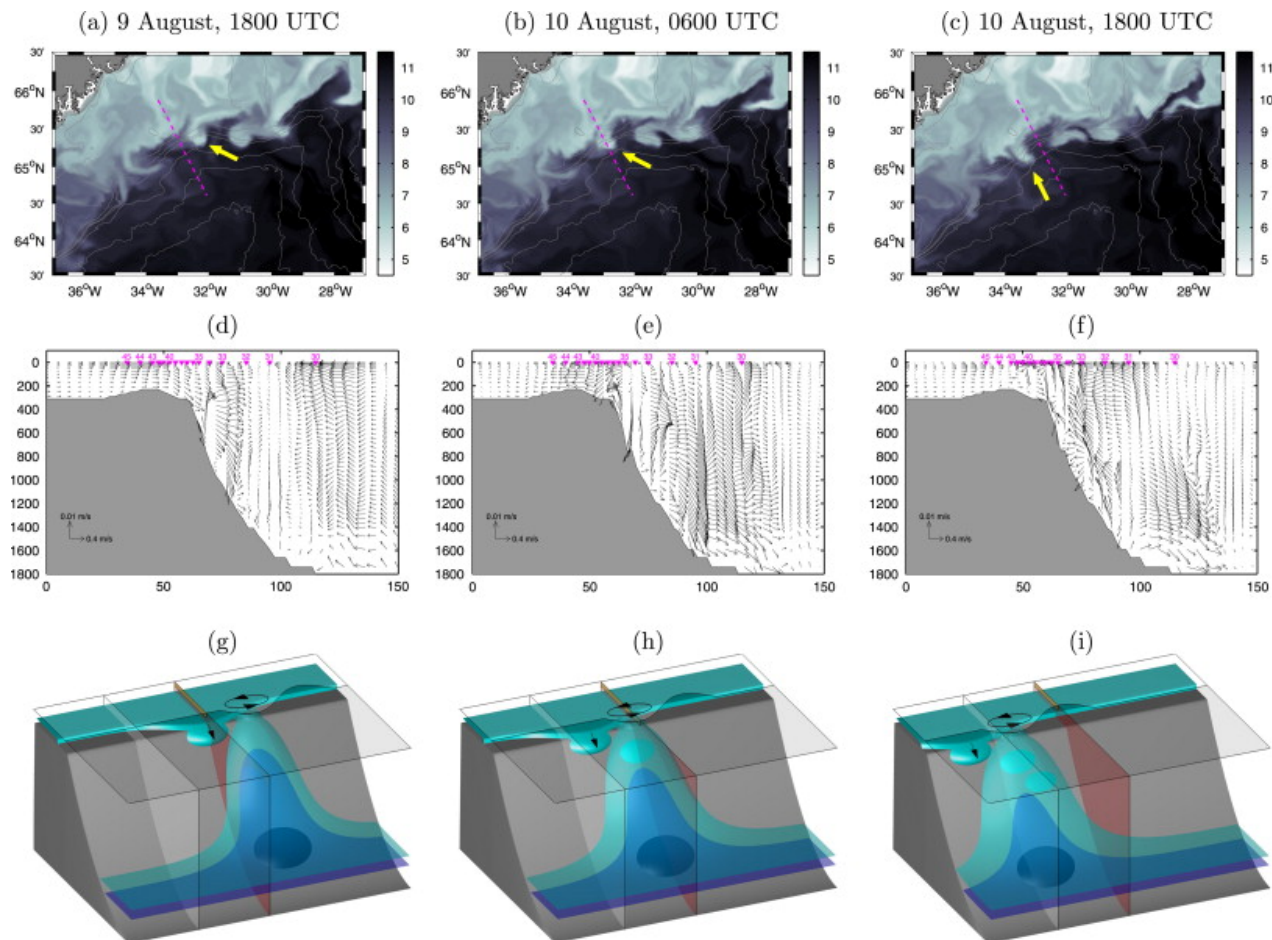
with C. Meneveau (Mech. E), G. Eyink (Applied Math), R. Burns (CS)

# Simulation of Windfarms

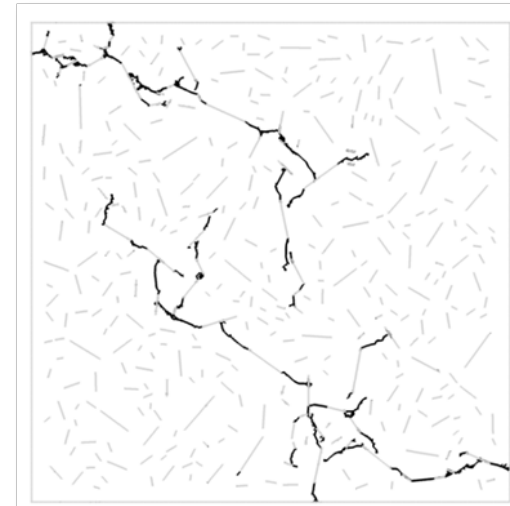
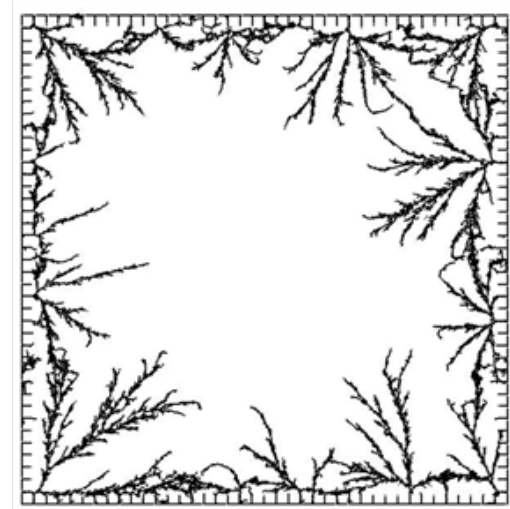
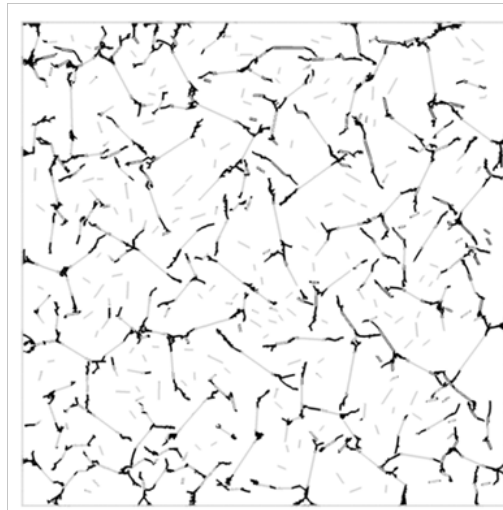
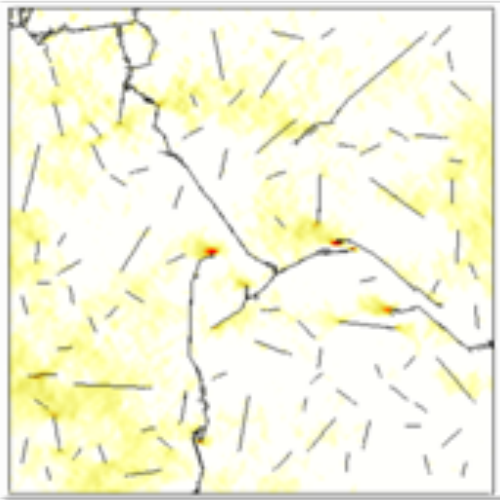


# Oceanography

Hydrostatic and non-hydrostatic simulations of dense waters cascading off a shelf: The East Greenland case (Magaldi, Haine 2015)



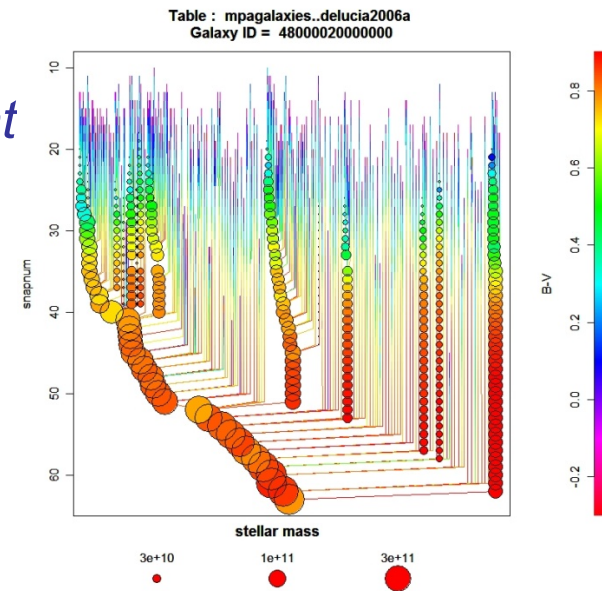
# Materials Science



Daphalapurkar, Brady, Ramesh, Molinari. JMPS (2011)

# Cosmology Simulations

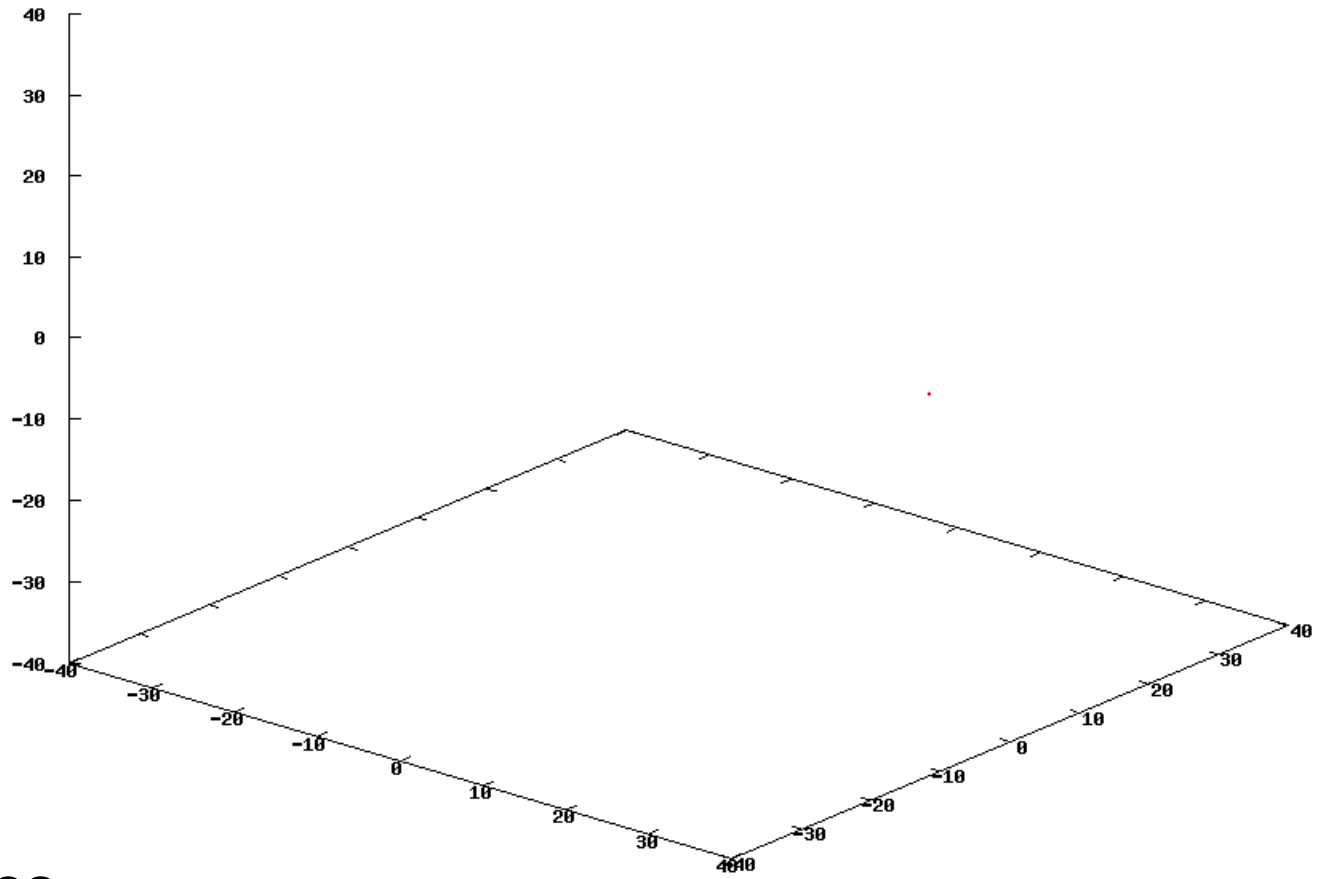
- Simulations are becoming an instrument on their own
- Millennium DB is the poster child/ success story
  - *Built by Gerard Lemson*
  - *600 registered users, 17.3M queries, 287B rows*
  - <http://gavo.mpa-garching.mpg.de/Millennium/>
  - *Dec 2012 Workshop at MPA: 3 days, 50 people*
- Data size and scalability
  - *PB data sizes, trillion particles of dark mat*
- Value added services
  - *Localized*
  - *Rendering*
  - *Global analytics*





# Bring Your Own Dwarf (Galaxy)

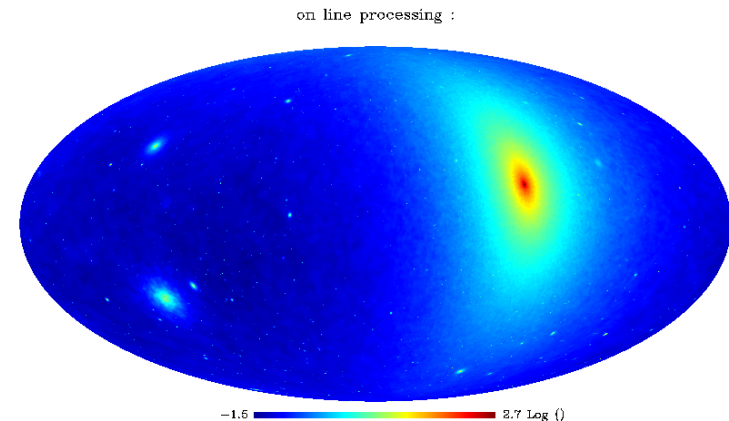
Wayne Ngan  
Brandon Bozek  
Ray Carlberg  
Rosie Wyse  
Alex Szalay  
Piero Madau



Via Lactea-II  
forces from halos

# Dark Matter Annihilation

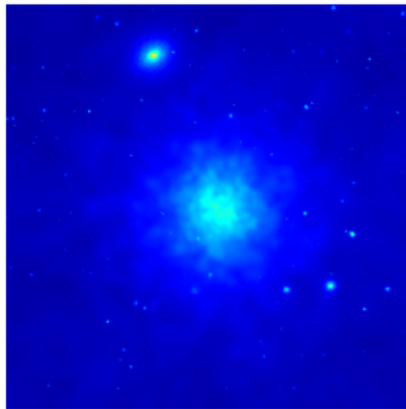
- Data from the Via Lactea II Simulation (400M particles)
- Computing the dark matter annihilation
  - *simulate the Fermi satellite looking for Dark Matter*
- Original code by M. Kuhlen runs in **8 hours** for a single image
- New GPU based code runs in **24 sec**, Point Sprites, Open GL shader language. [Lin Yang (Forrest), grad student at JHU]
- Interactive service (design your own cross-section)
- Approach would apply very well to gravitational lensing and image generation (virtual telescope)



# Changing the Cross Section

Annihilation (No Correction)

Inner 21.33 degree of the Subhalo

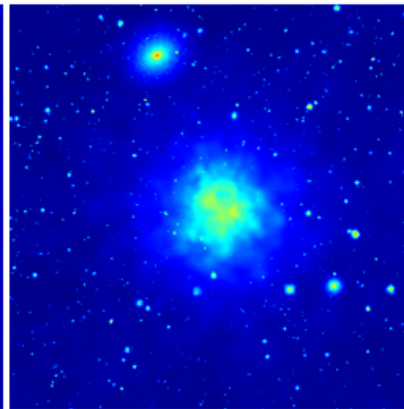


-7.7  -2.5 Log ( )  
(345.3, -11.1) Galactic

Inner 21.33 degree of the Subhalo

Annihilation (1/v Correction)

Inner 21.33 degree of the Subhalo

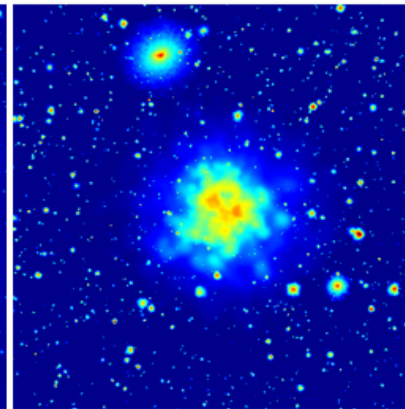


-7.7  -2.5 Log ( )  
(345.3, -11.1) Galactic

Inner 21.33 degree of the Subhalo

Annihilation (1/v^2 Correction)

Inner 21.33 degree of the Subhalo

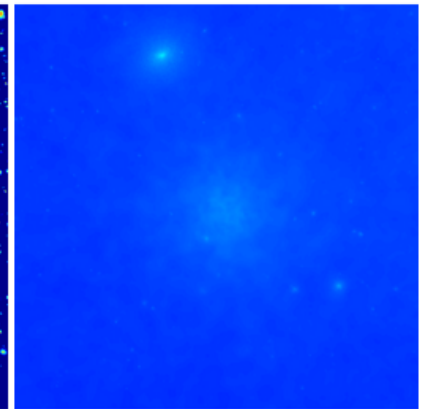


-7.7  -2.5 Log ( )  
(345.3, -11.1) Galactic

Inner 21.33 degree of the Subhalo

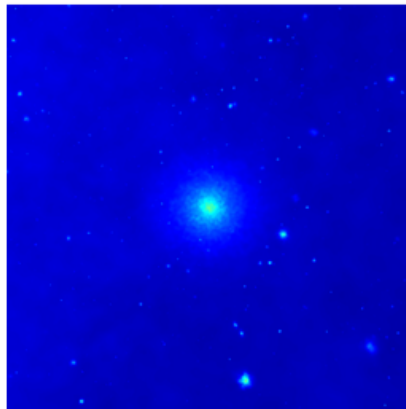
Decay Map (No Correction)


Inner 21.33 degree of the Subhalo

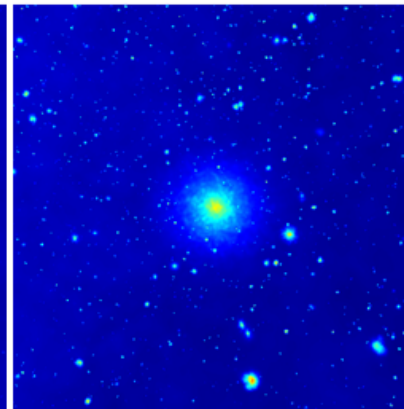



-7.7  -2.5 Log ( )  
(345.3, -11.1) Galactic

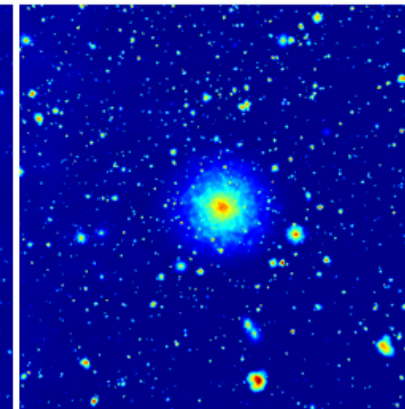
Inner 21.33 degree of the Subhalo




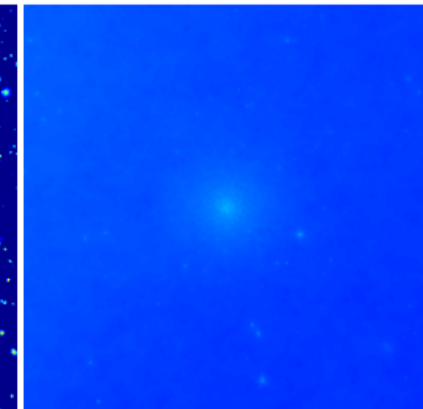
-7.7  -2.5 Log ( )  
(50.5, -19.4) Galactic




-7.7  -2.5 Log ( )  
(50.5, -19.4) Galactic



-7.7  -2.5 Log ( )  
(50.5, -19.4) Galactic



-7.7  -2.5 Log ( )  
(50.5, -19.4) Galactic

Yang, Silk, Szalay, Wyse, Bozek, Madau (2014)

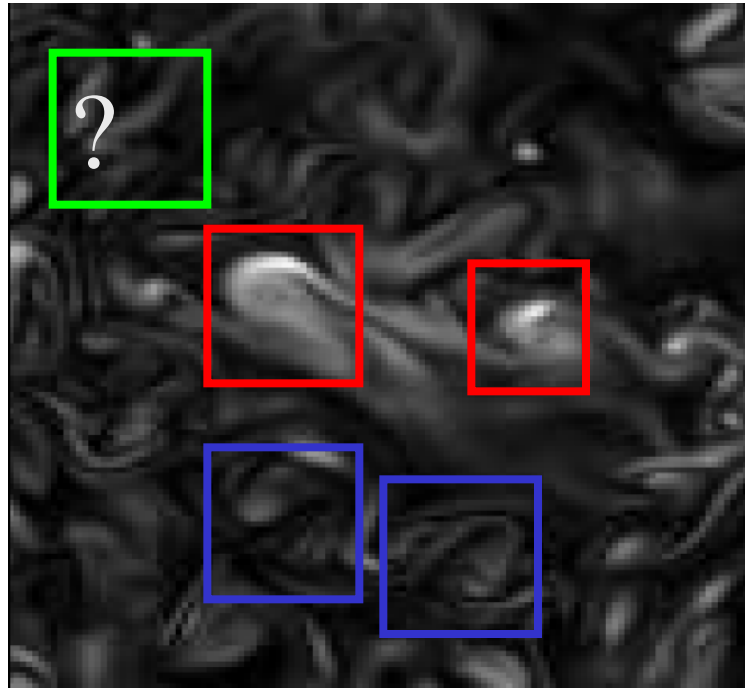
# Exascale Numerical Laboratories

---

- Posterior interactive analysis of sims becoming popular
- Comparing simulation and observational data crucial!
- Similarities between Turbulence/CFD, N-body, ocean circulation and materials science
- Differences as well in the underlying data structures
  - *Particle clouds / Regular mesh / Irregular mesh*
- Innovative access patterns appearing
  - *Immersive virtual sensors/Lagrangian tracking*
  - *User-space parallel operators, mini workflows on GPUs*
  - *Posterior feature tagging and localized resimulations*
  - *Machine learning on HPC data, streaming algorithms*
  - *Joins with user derived subsets, even across snapshots*
  - *Data driven simulations/feedback loop/active control of sims*

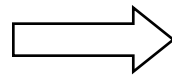
# Applications of ML to Turbulence

Renyi  
divergence



**Vorticity**

**Similarity between regions**



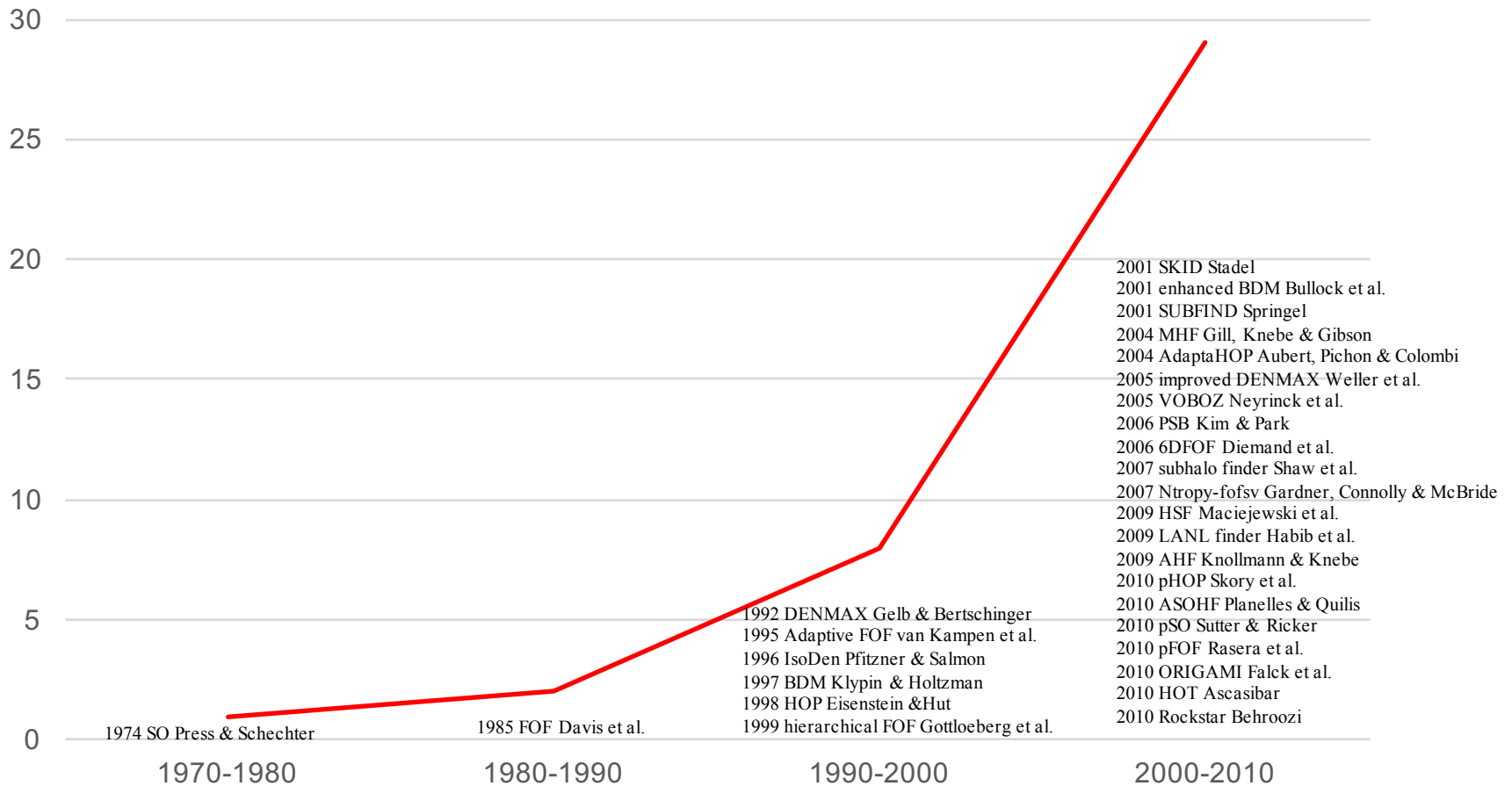
□ clustering,

□ classification,

□ anomaly detection

with J. Schneider, B. Póczos, CMU

# Halo finding algorithms



— Cumulative number of halo finders as a function of time

The Halo-Finder Comparison Project  
[Knebe et al, 2011]

# Memory issue

All current halo finders require to load all the data into memory



Each snapshot from the simulation with  $10^{12}$  particles will require 12 terabytes of memory



To build a scalable posterior solution we need to develop an algorithm with **sublinear** memory usage

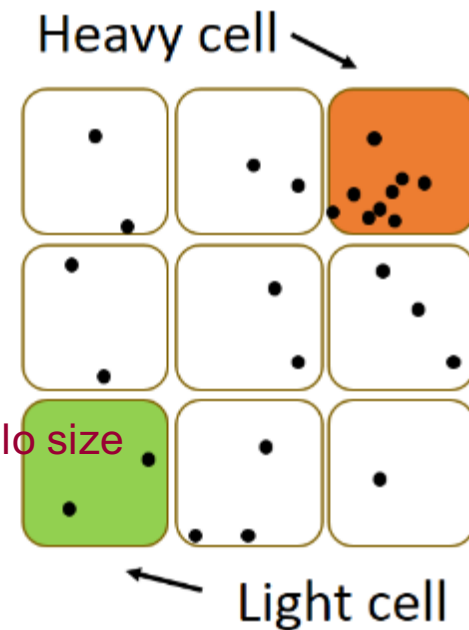
# Streaming Solution

Our goal:

- Reduce halo-finder problem to one of the existing problems of streaming algorithms
- Apply ready-to-use algorithms

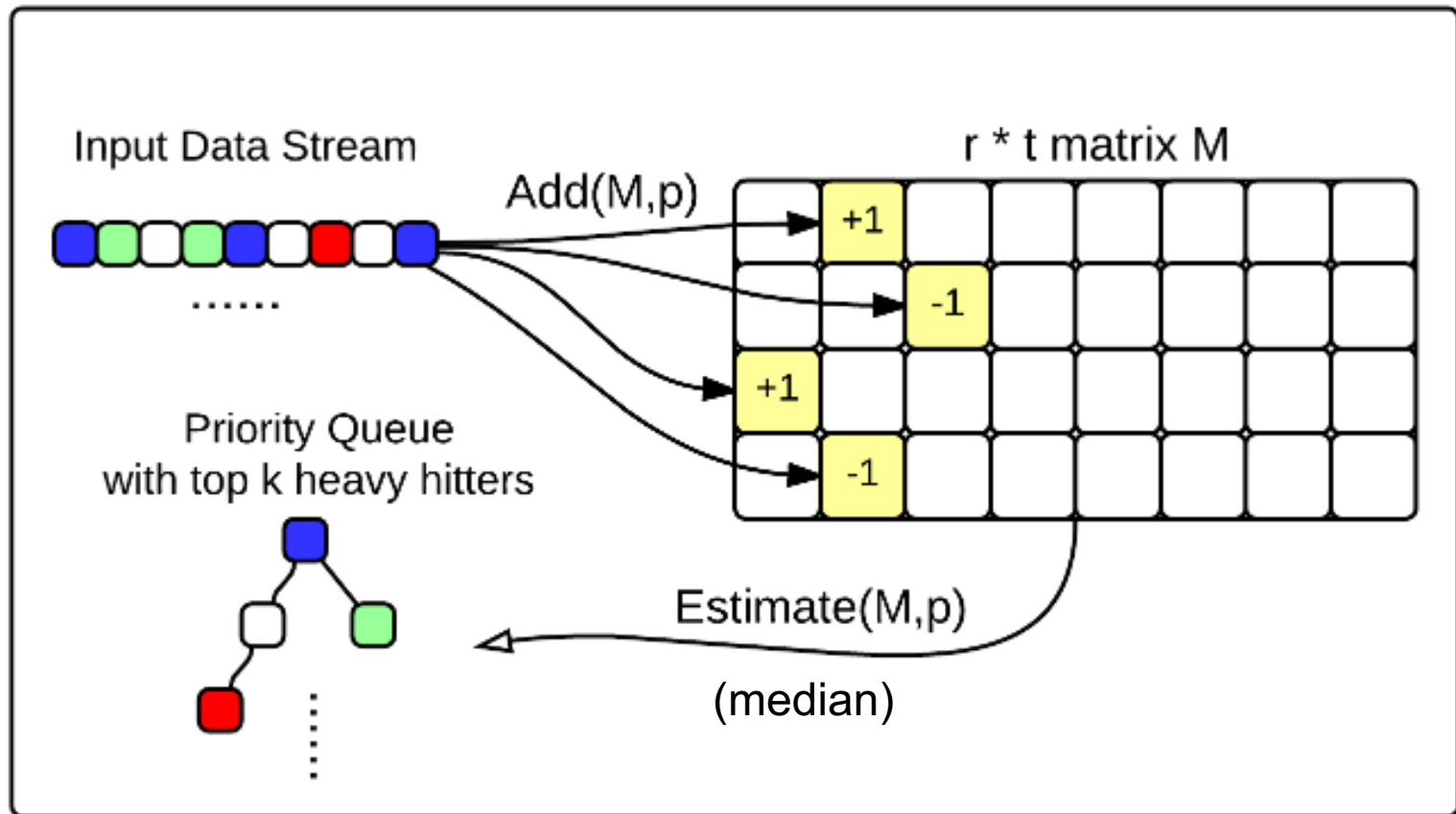
haloes  $\approx$  heavy hitters?

- To make a reduction to heavy hitters we need to discretize the space.
- Naïve solution is to use 3D mesh:
  - Each particle now replaced by cell id
  - Heavy cells represent mass concentration
  - Grid size is chosen according to typical halo size





# Count Sketch



# Emerging Challenges

- Data size and scalability
  - *PB, trillion particles, dark matter*
  - *Where is the data located, how does it get there*
- Value added on-demand services
  - *Localized (SED, SAM, star formation history, resimulations)*
  - *Rendering (viz, lensing, DM annihilation, light cones)*
  - *Global analytics (FFT, correlations of subsets, covariances)*
  - *Spatial queries*
- Data representations
  - *Particles vs hydro vs boundaries*
  - *Particle tracking in DM data*
  - *Aggregates, summary of uncertainty quantification (UQ)*
  - *Covariances, ensemble averages*

# FileDB

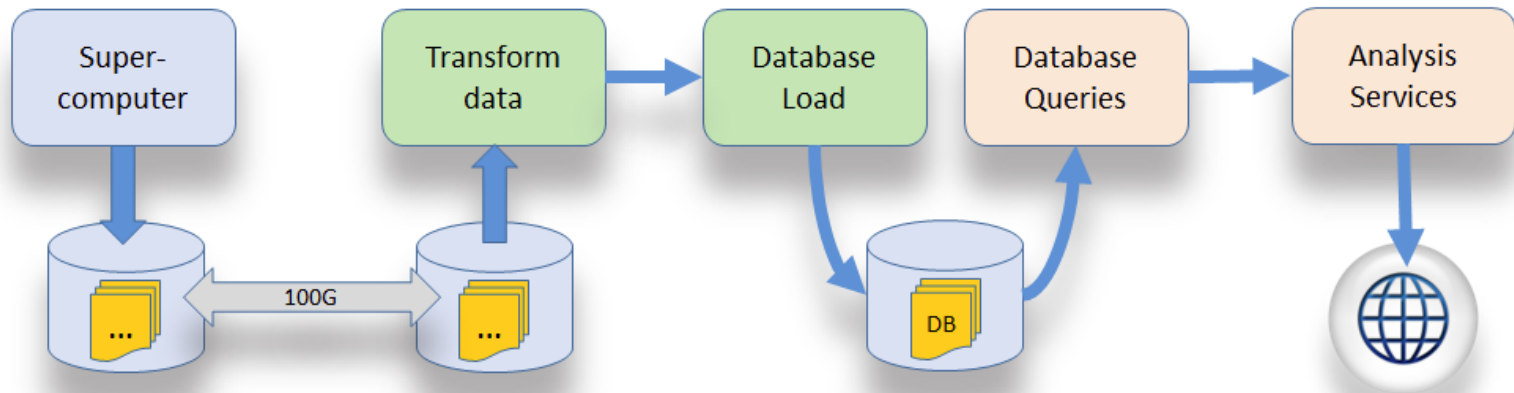
- Localized access pattern greatly aided by indexing
  - *Space-filling curves (Peano-Hilbert, Morton) map 3D to 1D*
  - *Access patterns along these maximally sequential (HDD)*
  - *Extraction of subvolume maps onto B-tree range queries*



But...

- Loading more than 50TB of data into a DB is painful, transactions not needed, write once read many
- Simulation snapshots are heavily partitioned
- Why not just use the native output files with the DB?
  - *Attach files, store only the index in the DB (G. Lemson, JHU)*
- Building a 4PB test system with 16 servers, 40G

# Current Scenario

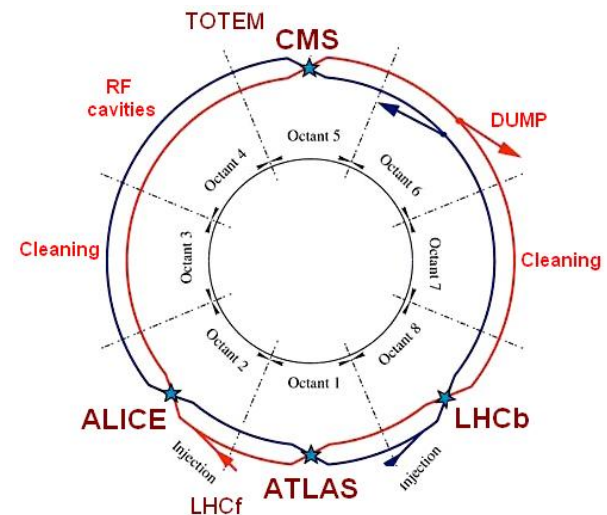


# How Do We Prioritize?

- Data Explosion: science is becoming data driven
- It is becoming “too easy” to collect even more data
- Robotic telescopes, next generation sequencers, *complex simulations*
- **How long can this go on?**
  
- “Do I have enough data or would I like to have more?”
- No scientist ever wanted less data....
- But: Big Data is synonymous with Noisy/Dirty Data
- How can we decide how to collect data that is *more relevant* ?

# LHC Lesson

- LHC has a single data source, \$\$\$\$\$
- Multiple experiments tap into the beamlines
- They each use **in-situ** hardware triggers to filter data
  - *Only 1 in 10M events are stored*
  - *Not that the rest is garbage, just sparsely sampled*
- Resulting “small subset” analyzed many times **off-line**
  - *This is still 10-100 PBs*
- Keeps a whole community busy for a decade or more



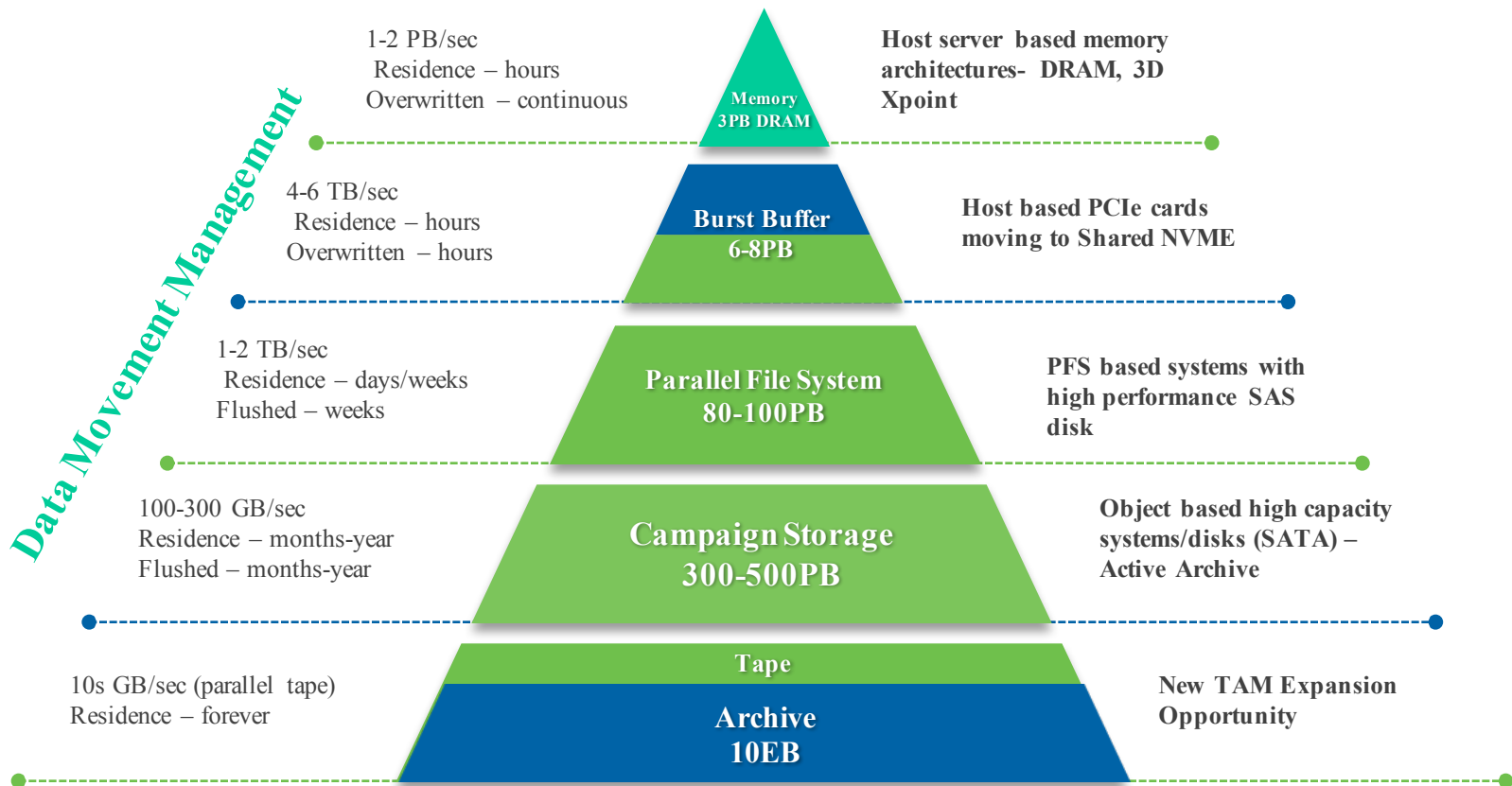
# Exascale Simulation Analogy

---

- Exascale computer running a community simulation
- Many groups plugging their own “triggers” (in-situ), the equivalents of “beamlines”
  - *Keep very small subsets of the data*
  - *Plus random samples from the field*
  - *Immersive sensors following world lines or light cones*
  - *Buffer of timesteps: save precursor of events*
- Sparse output analyzed offline by broader community
- Cover more parameter space and extract more realizations (UQ) using the saved resources

# Using the Memory Hierarchy

The 'Trinity' System at LANL is leading the way

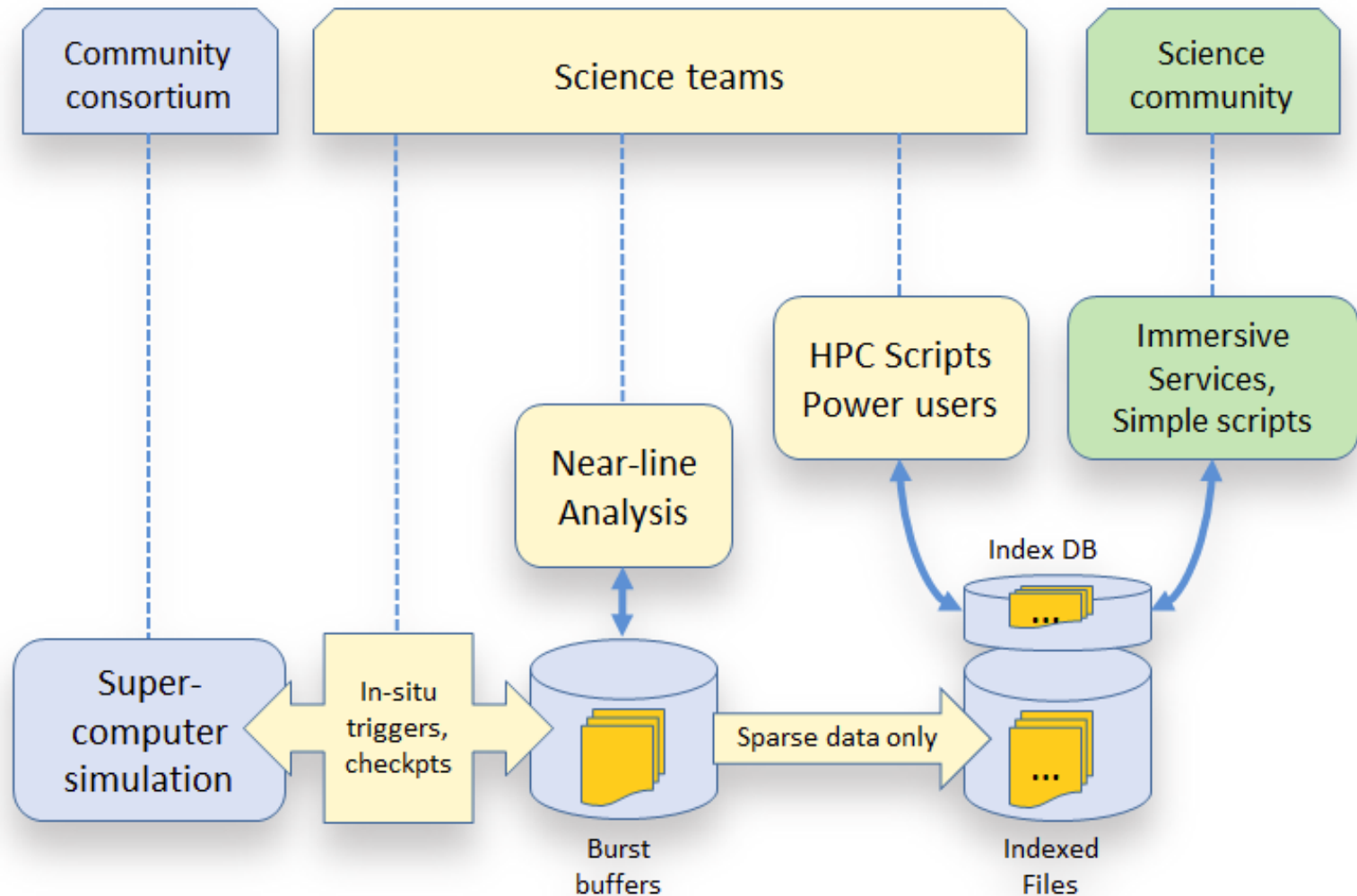




# Architectural Implications

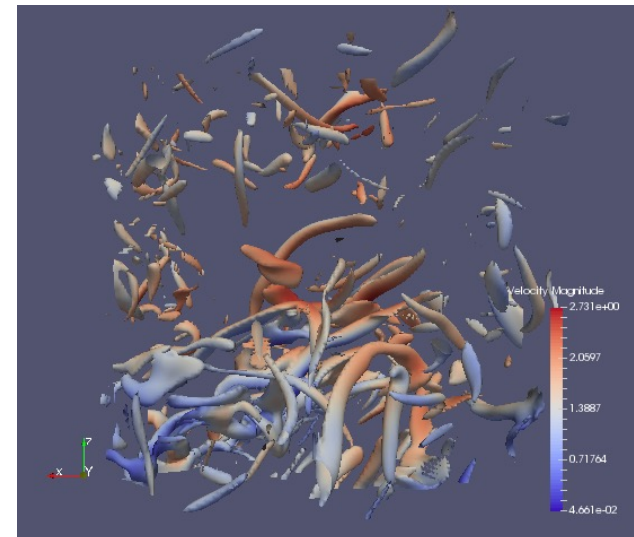
- In-situ: global analytics and “beamline” triggers, two stage, light-weight, and scheduler
- Simple API for community buy-in
- Very high selectivity to keep output on PB scales
- Burst buffers for near-line analyses
- Need to replace DB storage with smart object store with additional features (seek into objects)
- Build a fast DB index on top (SQL or key-value?) for localized access patterns
- Parallel high level scripting tools (iPython.parallel?)
- Simple immersive services and visualizations

# Future Scenario



# Testing Burst Buffer Triggers

- Use data in the Trinity Burst Buffers
- Allocate about 2% of CPU to compute triggers in-situ
- Store results in secondary storage for viz
- Extract high-vorticity regions from turbulence sim
- Data compression not very high, but good proof of concept
- Model applies to light-cones in N-body, cracks in Material Science



Hamilton, Burns, Ahrens, Szalay et al (2016)

# Summary

- Science is increasingly driven by data (big and small)
- Computations getting even closer to the data
- Simulations are becoming first-tier instruments
- Changing sociology – archives analyzed by individuals
- Need Numerical Laboratories for the simulations
  - *Provide impedance matching between the HPC experts and the many domain scientists*
- Exascale: razor-sharp balance of in-situ triggers and off-line follow-up
- Clever in-situ use of burst buffers promising
- **On Exascale everything is a Big Data problem**