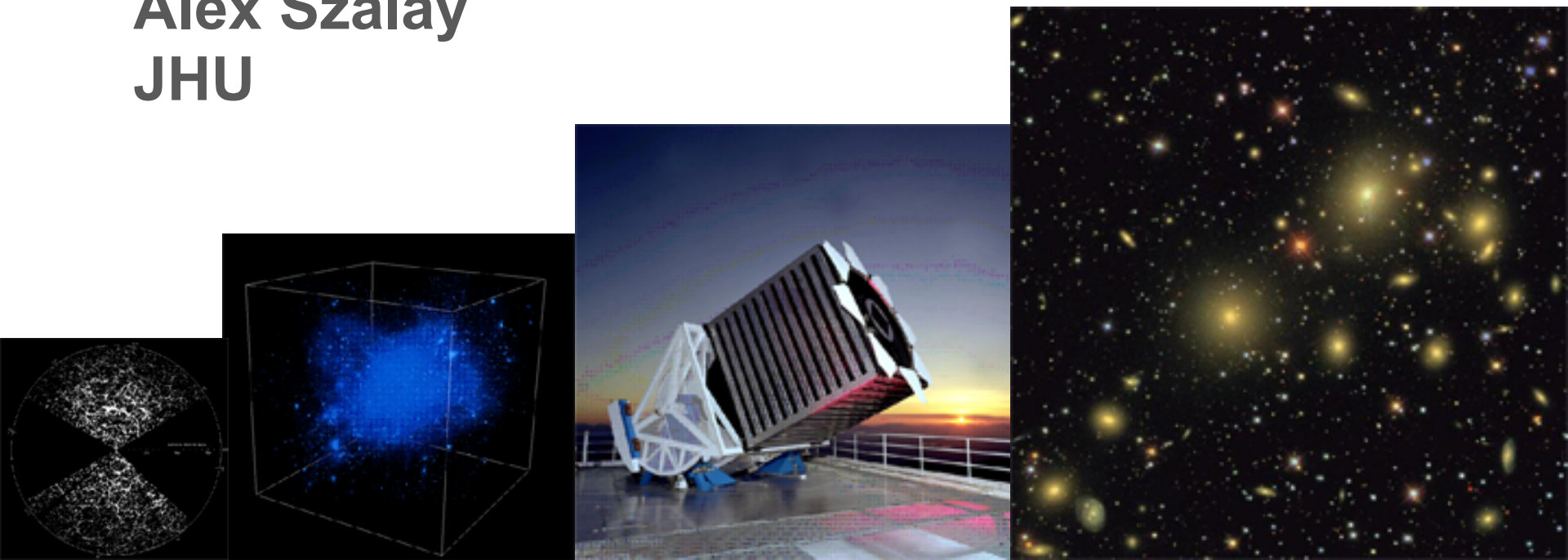
Institute for Data Intensive Engineering and Science

idies

Turning Large Simulations into Numerical Laboratories

Alex Szalay
JHU



Amdahl's Laws

Gene Amdahl (1965): Laws for a balanced system

i. Parallelism: max speedup is S/(S+P)

ii. One bit of IO/sec per instruction/sec (BW)

iii. One byte of memory per one instruction/sec (MEM)

Table 1. Amdahl's laws applied to various system powers.

Operations per second	RAM	Disk I/O bytes/s	Disks for that bandwidth at 100 Mbytes/s/disk	Disk byte capacity (100x RAM)	Disks for that capacity at 1 Tbyte/disk
10^9	Gigabyte	10^8	1	10^{11}	1
10^{12}	Terabyte	10^{11}	1,000	10^{14}	100
10^{15}	Petabyte	10^{14}	1,000,000	10^{17}	100,000
10^{18}	Exabyte	10^{17}	1,000,000,000	10^{20}	100,000,000

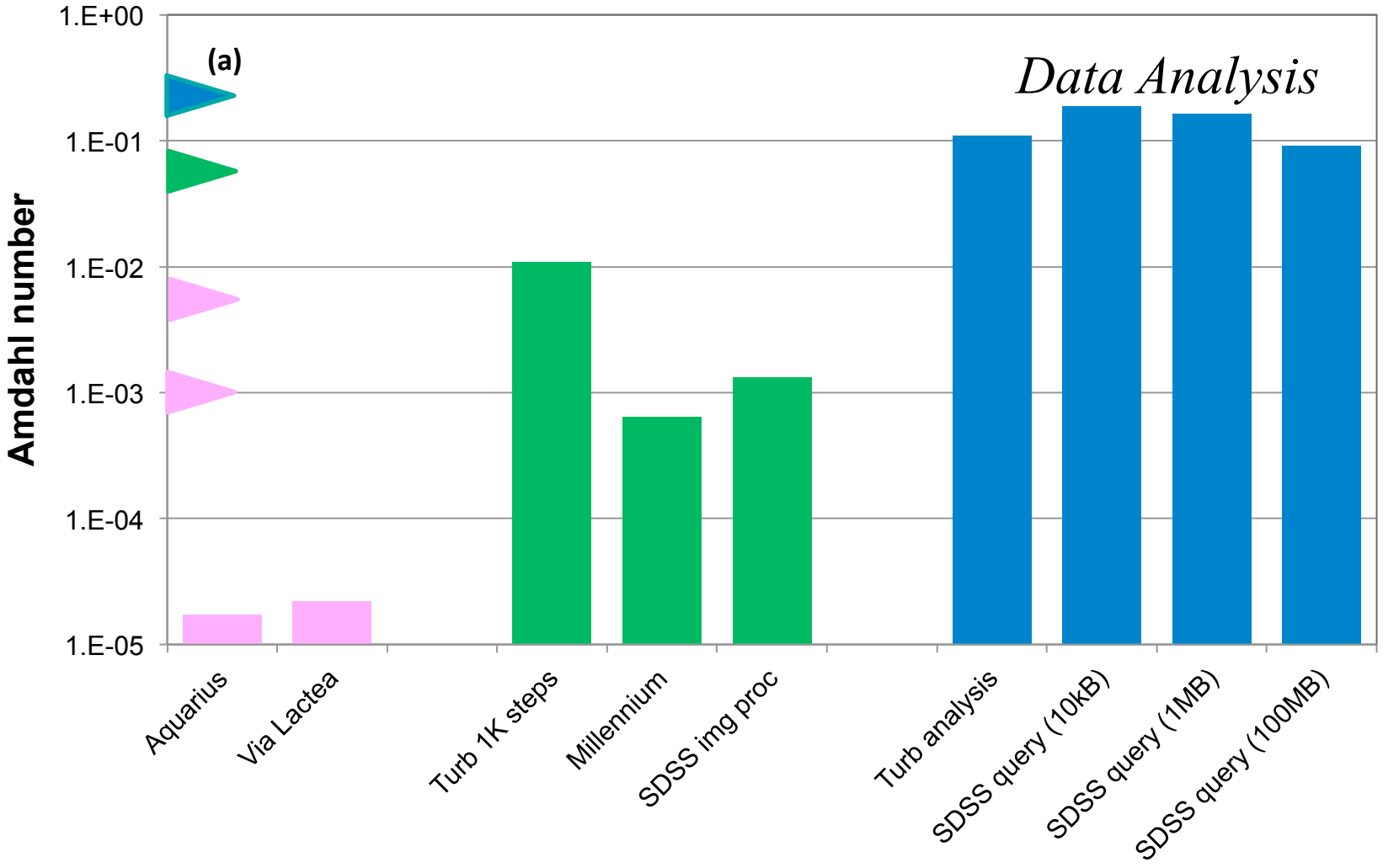
Modern multi-core systems move farther away from Amdahl's Laws (Bell, Gray and Szalay 2006)

Typical Amdahl Numbers

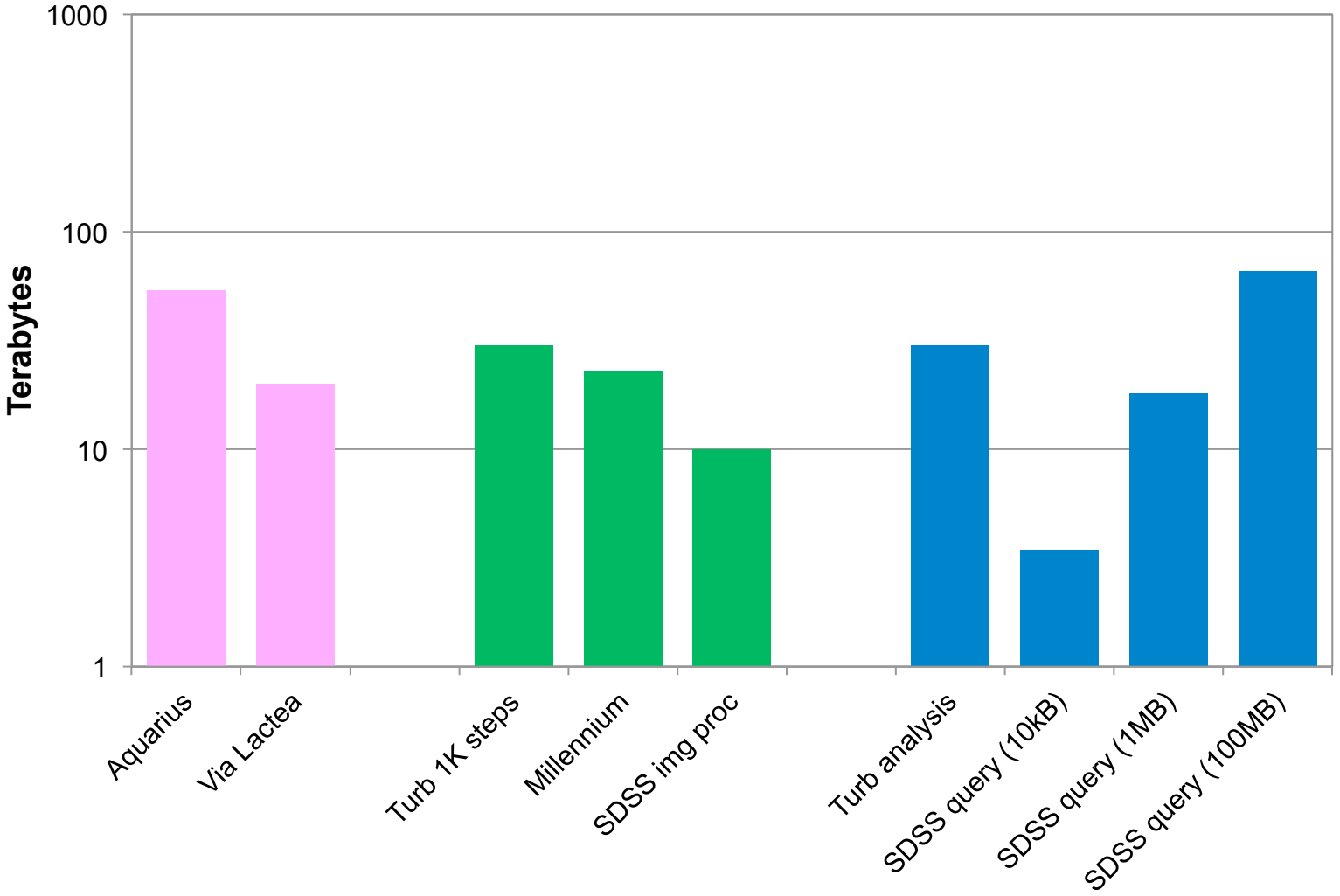
System	CPU count	GIPS [GHz]	RAM [GB]	diskIO [MB/s]	Amdahl	
					RAM	IO
BeoWulf	100	300	200	3000	0.67	0.08
Desktop	2	6	4	150	0.67	0.2
Cloud VM	1	3	4	30	1.33	0.08
SC1	212992	150000	18600	16900	0.12	0.001
SC2	2090	5000	8260	4700	1.65	0.008
GrayWulf	416	1107	1152	70000	1.04	0.506

Amdahl IO: (sequential IO in bits/sec) / (instructions/sec)

Amdahl Numbers for Data Sets



The Data Sizes Involved



Data in HPC Simulations

- HPC is an instrument in its own right
- Largest simulations approach petabytes
  - *from supernovae to turbulence, biology and brain modeling*
- Need public access to the best and latest through interactive numerical laboratories
- Creates new challenges in
  - *How to write enough output (speed of checkpointing)*
  - *How to move the of data (high speed networking)*
  - *How to look at it (render on top of the data, drive remotely)*
  - *How to interface (smart sensors, immersive analysis)*
  - *How to analyze (value added services, analytics, …)*
  - *Architectures (supercomputers, DB servers, ??)*

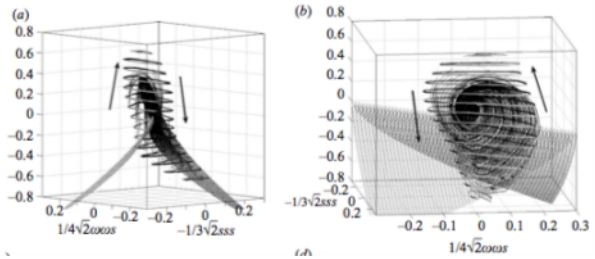
Usage Scenarios

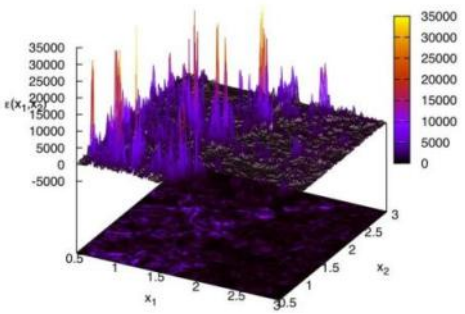
- Huge variations in Data Lifecycle and commitments
  - *On-the fly analysis* *(immediate, do not keep)*
  - *Private reuse* *(short/mid term, local)*
  - *Public reuse* *(mid term)*
  - *Public service portal* *(mid/long term)*
  - *Archival and curation* *(long term)*
- Different from Supercomputer usage patterns
- Wide range of data access patterns, from high speed streams to large random access
- Localized subsets vs global access vs rendering
- Use cases: ***turbulence and cosmology***

Immersive Turbulence

"… the last unsolved problem of classical physics…" Feynman

- **Understand the nature of turbulence**
  - *Consecutive snapshots of a large simulation of turbulence: now 30 Terabytes*
  - *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: 70TB MHD simulation*





- **New paradigm** for analyzing simulations!

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

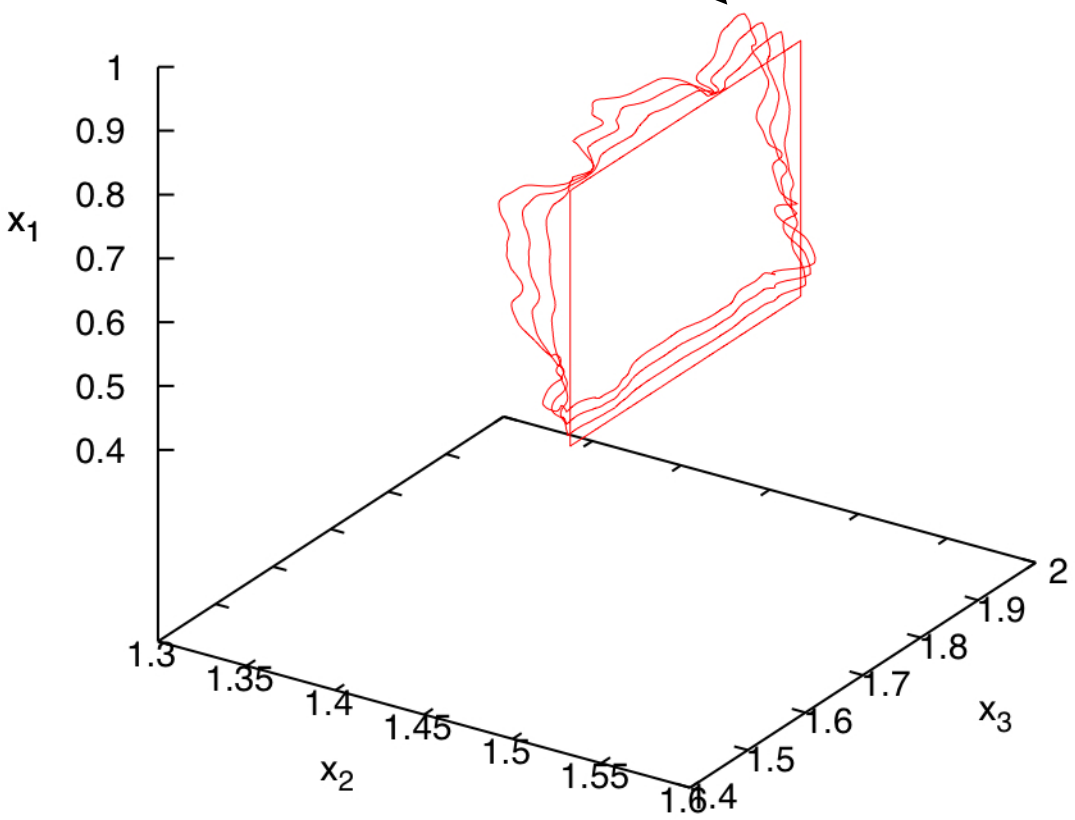
Sample code (fortran 90)

```
!
  do it = 1,15,1
!
    print *,'time = ',time
    time=time-deltat
!
    CALL getvelocity(authkey, dataset1, time, Lagrangian6thOrder , PCHIPInterpolation, 4*n, points, dataout)
 !
   do i=1,4*n
   do k=1,3
      points(k,i)=points(k,i)-dataout(k,i)*deltat
   end do
   end do
!
   if (it.eq.5.or.it.eq.10.or.it.eq.15) then
    do i=1,4*n
      write(10,*) points(1,i),points(2,i),points(3,i)
    end do
      write(10,*) points(1,1),points(2,1),points(3,1)
   endif
   write(10,*) '  '
   end do
!
   endif
```

minus

advect backwards in time !

Not possible during DNS



Database design philosophies

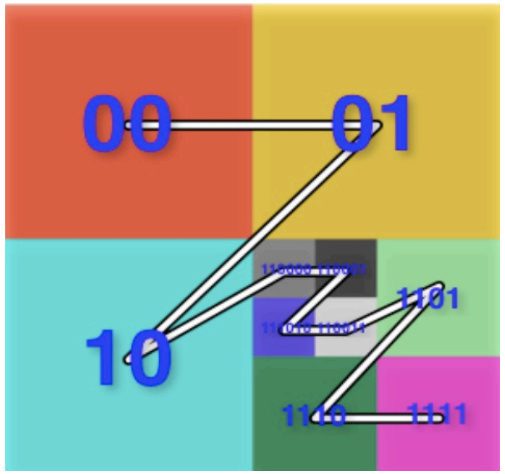
- Move operations as close as possible to the data:

  Most elementary operations in analysis of CFD data (constrained on localilty):

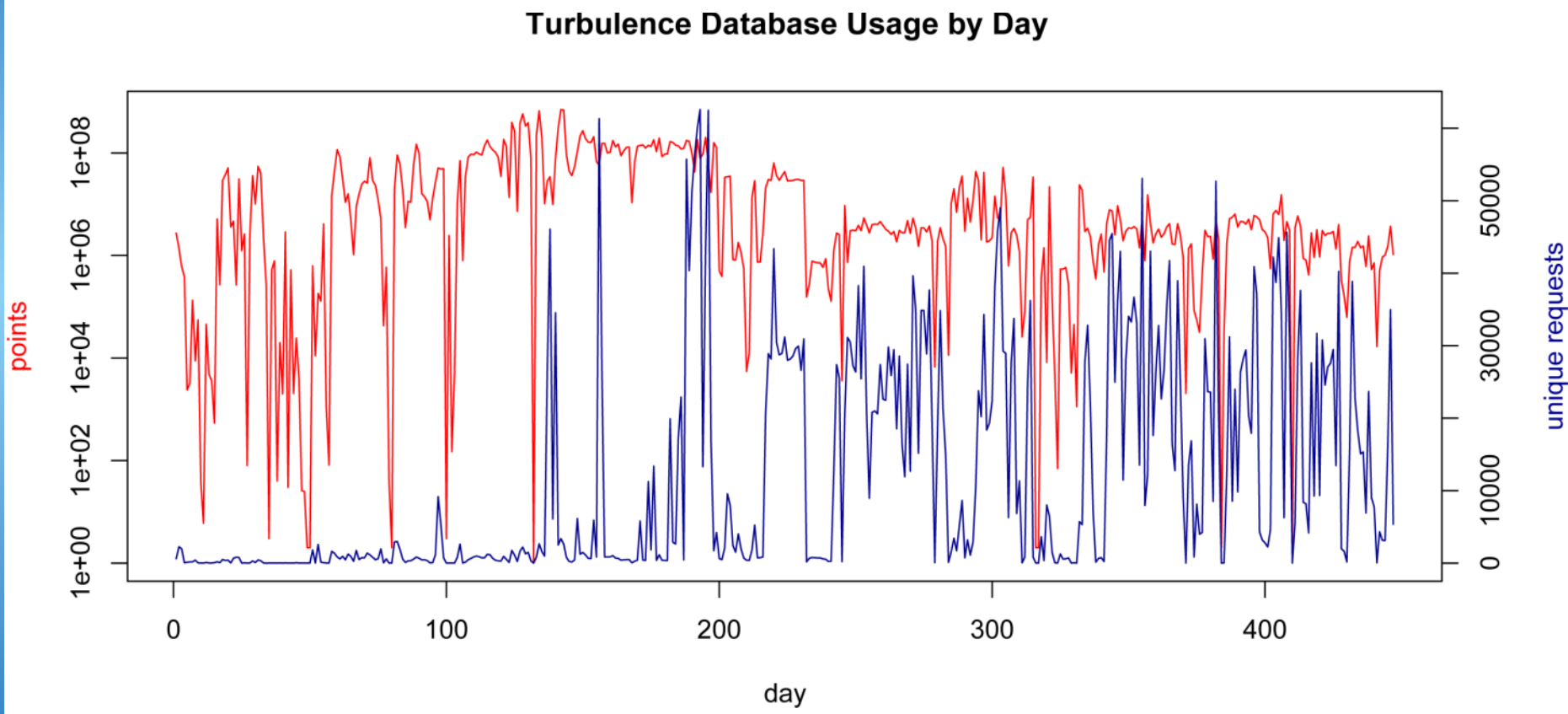
  - Differentiation (high-order finite-differencing)
  - Interpolation (Lagrange polynomial interpolation)

- Storage schema must facilitate rapid searches
  - Most basic search: given *x,y,z,t* position, find field variables (*u,v, w,p*).
  - Define elementary data-cube (optimize size relative to typical queries) and arrange along Z curve and indexing using oct-tree:

$j=$	0	1	2	3	4	5	6	7
$i=0$	0	1	4	5	16	17	20	21
$i=1$	2	3	6	7	18	19	22	23
$i=2$	8	9	12	13	24	25	28	29
$i=3$	10	11	14	15	26	27	30	31
$i=4$	32	33	36	37	48	49	52	53
$i=5$	34	35	38	39	50	51	54	55
$i=6$	40	41	44	45	56	57	60	61
$i=7$	42	43	46	47	58	59	62	63

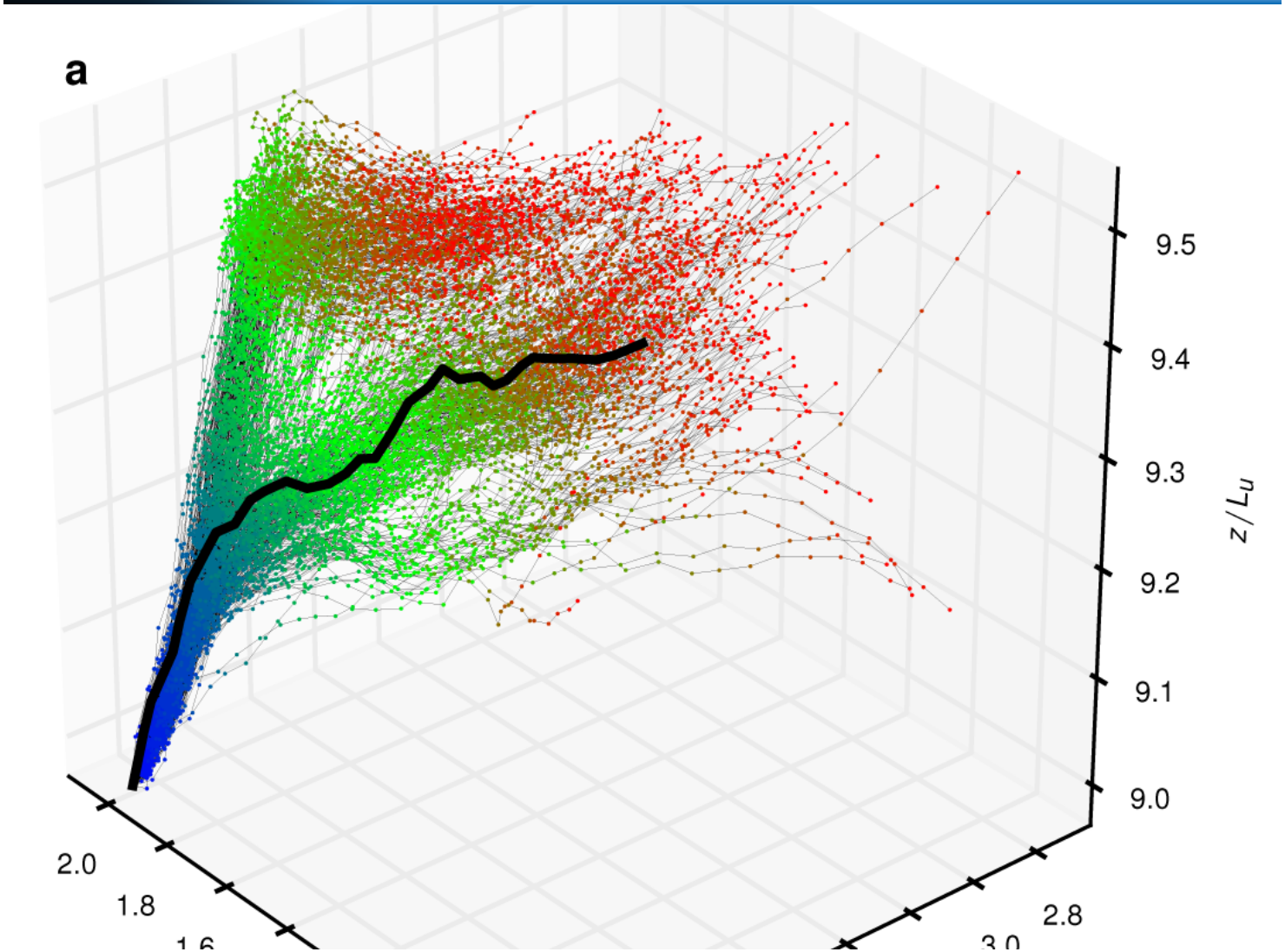


Daily Usage



2011: exceeded 100B points, delivered publicly

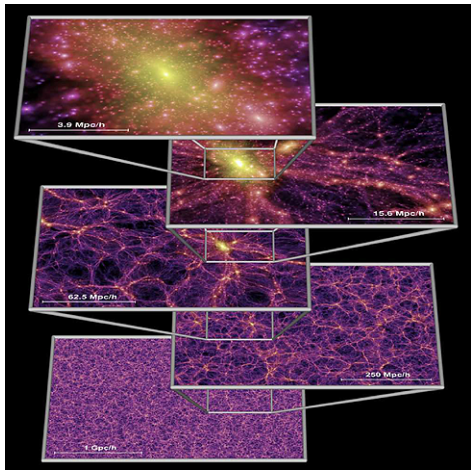
Eyink et al Nature (2013)



Cosmological Simulations

In 2005 cosmological simulations had 10^{10} particles and produced over 30TB of data (Millennium)

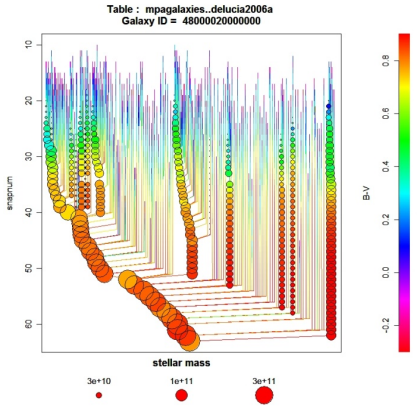
- Build up dark matter halos
- Track merging history of halos
- Use it to assign star formation history
- Combination with spectral synthesis
- Realistic distribution of galaxy types



- Today: simulations with 10^{12} particles and PB of output are under way (MillenniumXXL, Silver River, etc)
- Hard to analyze the data afterwards -> need DB
- What is the best way to compare to real data?

Next-Generation Challenges

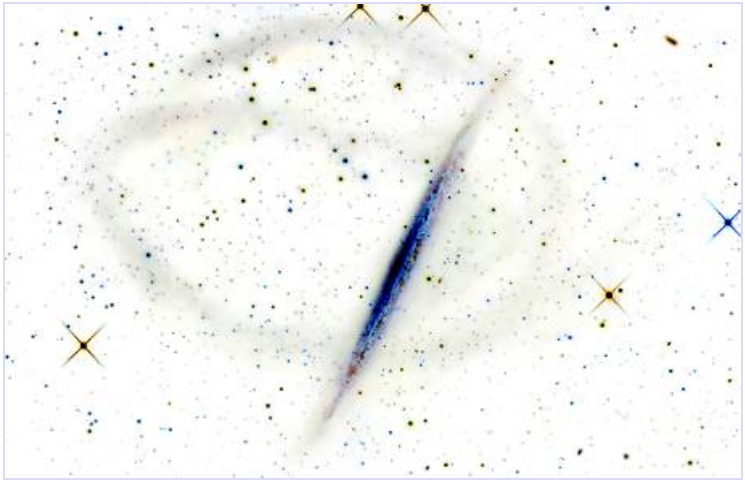
- Millennium DB is the poster child/ success story
  - *600 registered users, 17.3M queries, 287B rows*
    http://gavo.mpa-garching.mpg.de/Millennium/
  - *Dec 2012 Workshop at MPA: 3 days, 50 people*
- Data size and scalability
  - *PB data sizes, trillion particles of dark matter*
  - *Where is the data stored, how does it get there*
- Value added services
  - *Localized (SED, SAM, SF history, posterior re-simulations)*
  - *Rendering (viz, lensing, DM annihilation, light cones)*
  - *Global analytics (FFT, correlations of subsets, covariances)*
- Data representations
  - *Particles vs hydro grid*
  - *Particle tracking in DM data*
  - *Aggregates, uncertainty quantification*



The Milky Way Laboratory

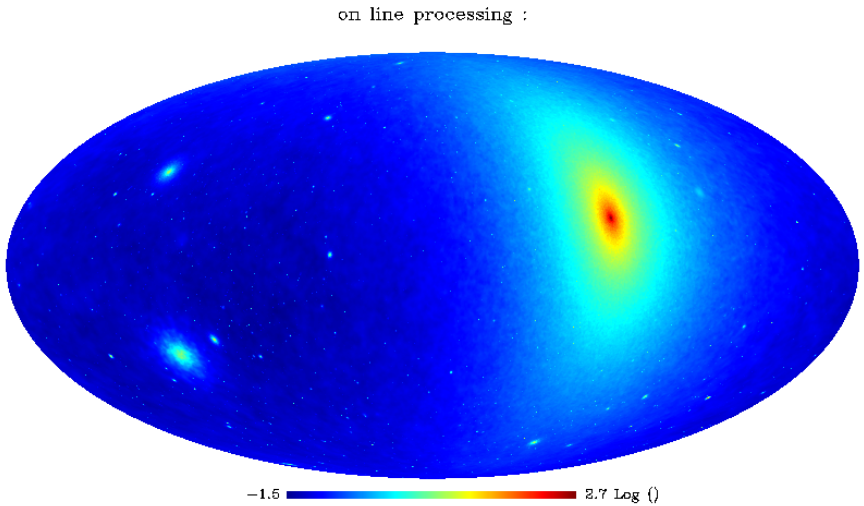
- Use cosmology simulations as an immersive laboratory for general users
- Via Lactea-II (20TB) as prototype, then Silver River (50B particles) as production (15M CPU hours)
- 800+ hi-rez snapshots (2.6PB) => 800TB in DB
- Users can insert test particles (dwarf galaxies) into system and follow trajectories in pre-computed simulation
- Users interact remotely with a PB in ‘real time’

Madau, Rockosi, Szalay, Wyse, Silk, Kuhlen, Lemson, Westermann, Blakeley

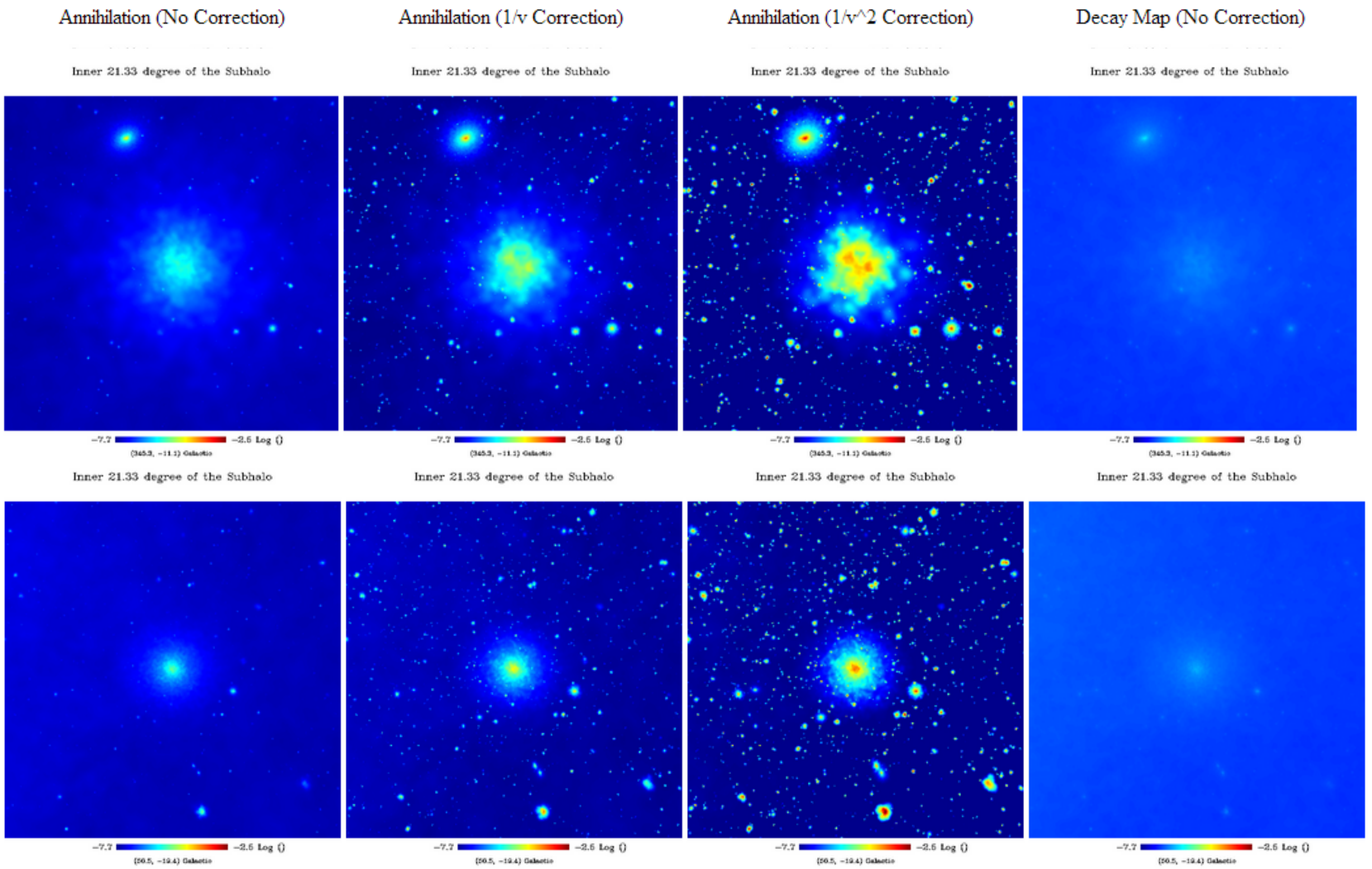


Dark Matter Annihilation

- Data from the Via Lactea II Simulation (400M particles)
- Computing the dark matter annihilation over the whole sky
- Original code by M. Kuhlen runs in 8 hours for a single image
- New GPU based code runs in 24 sec, OpenGL shader language (Lin Yang, 2nd year grad student at JHU)
- Interactive service (design your own cross-section)
- Would apply very well to lensing and image generation



Changing the Cross Section

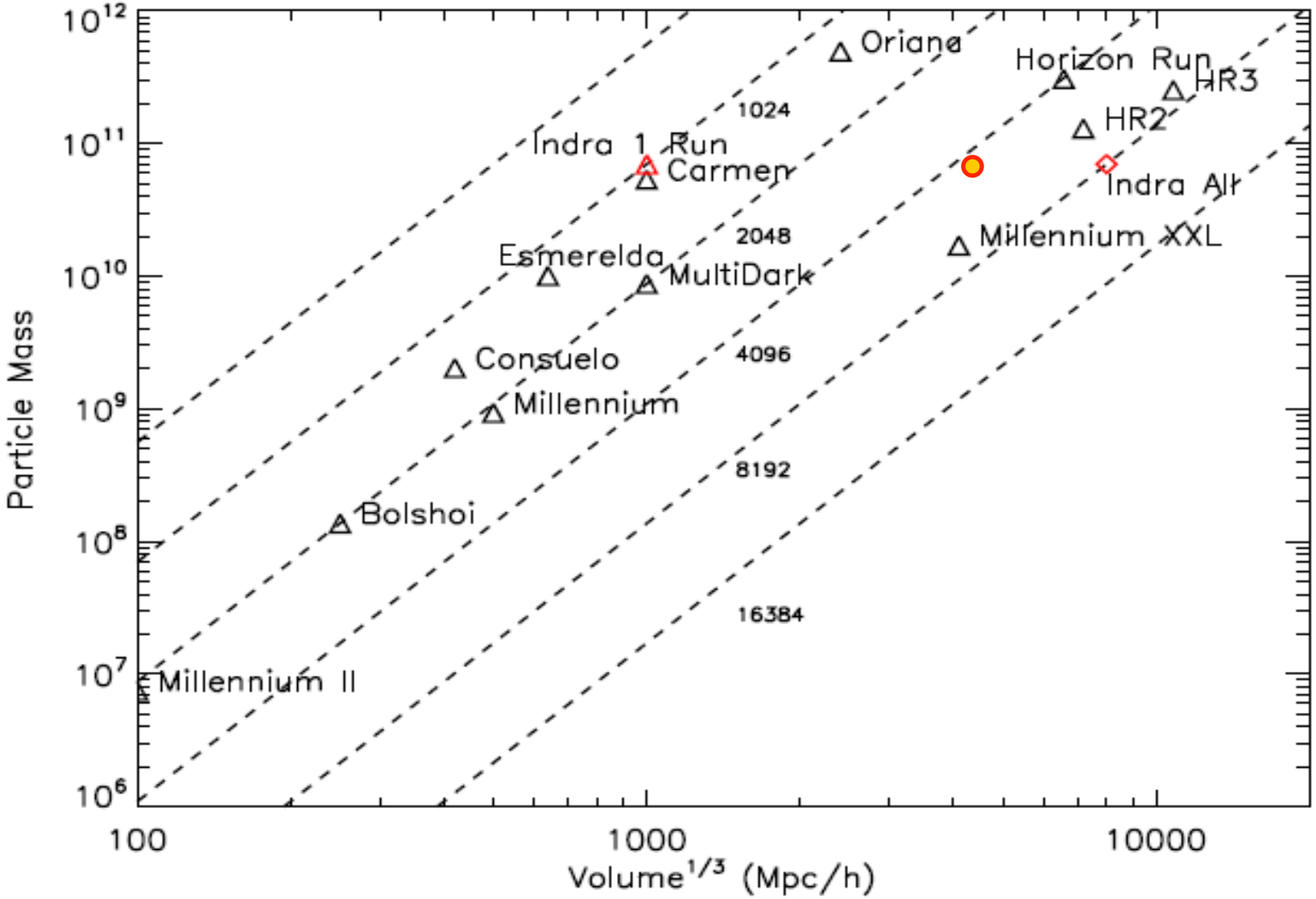


The Indra Simulations

- Quantifying cosmic variance/uncertainty
- Suite of dark matter *N*-body simulations
  - *Gadget2 code*
  - *512 different 1 Gpc/h box, 1024^3 particles per simulation*
  - *Data loaded into SQL database, will be available to the public*
  - *Random initial conditions, WMAP7 cosmology*
- Particle data:
  - *Ids, positions, velocities for 64 snapshots of each simulation*
- Halo catalogs:
  - *Standard Friends-Of-Friends (and others), linked to particles*
- Fourier modes:
  - *Course density grid for 512 time steps of each run*

Bridget Falck (ICG Portsmouth), Tamás Budavári (JHU), Shaun Cole (Durham), Daniel Crankshaw (JHU), László Dobos (Eötvös), Adrian Jenkins (Durham), Gerard Lemson (MPA), Mark Neyrinck (JHU), Alex Szalay (JHU), and Jie Wang (Durham/Beijing)

Current N-body Simulations



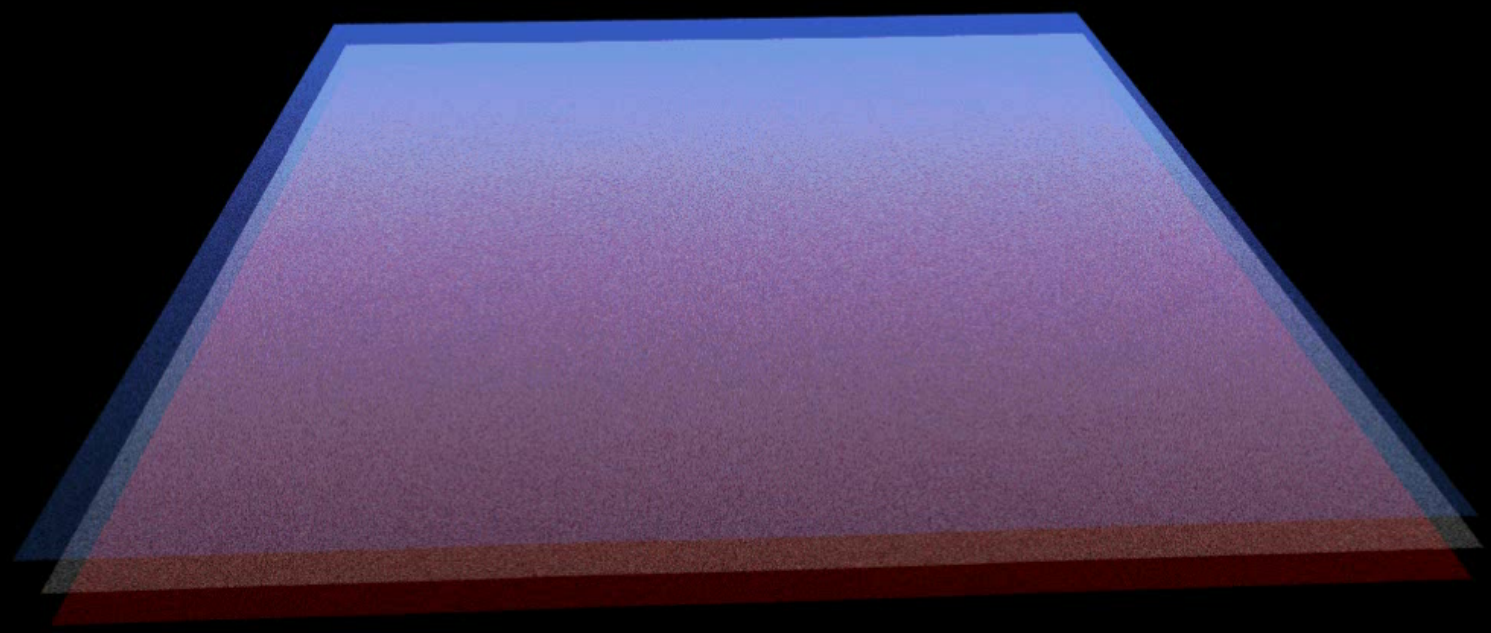
Architectural Challenges

- How to build a system for the posterior analysis?
- Where should data be stored
  - *Not directly at the supercomputer (too expensive storage)*
  - *Computations and visualizations must be on top of the data*
  - *Need high bandwidth to data source*
- Scheduling of complex I/O access patterns
  - *Databases are a good model, but are they scalable?*
  - *Google (Dremel, Tenzing, Spanner: exascale SQL)*
  - *Augmented with value-added analytic services (SciDB, etc)*
- Data organization
  - *Cosmology simulations are not hard to partition (scale-out)*
  - *Use fast, cheap storage for data streaming (sequential)*
  - *Consider a tier of large memory systems (random access)*

Summary

- Amazing progress in 7 years
- Millennium is prime showcase of how to use simulations
- Community is now using the DB as an instrument
- New challenges emerging:
  - *Petabytes of data, trillions of particles*
  - *Increasingly sophisticated value added services*
  - *Need a coherent strategy to go to the next level*
- It is not just about storage, but how to integrate access and computation
- Fill the gap between DB server and supercomputer
- Justification: increase the re-use of the output of SC

Streaming Visualization of Turbulence



Kai Buerger, Technische Universitat Munich, 24 million particles

Visualization of the Vorticity