Blasting Through the 10 Petaflops Barrier: HACC on the BG/Q

HACC (Hardware/Hybrid Accelerated Cosmology Code) Framework

Salman Habib
HEP and MCS Divisions,
Argonne National Laboratory

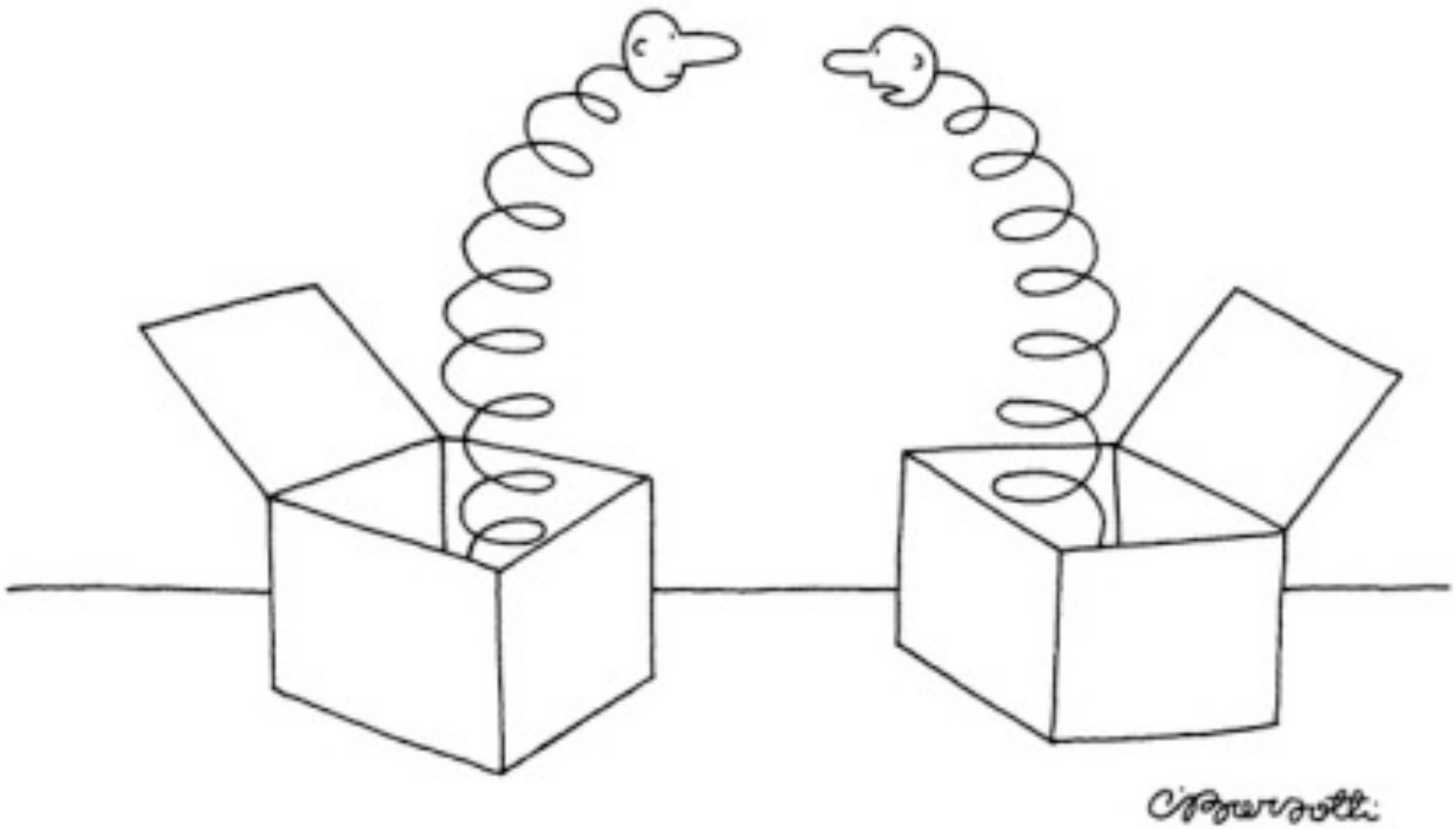
Vitali Morozov
Hal Finkel
Adrian Pope
Katrin Heitmann
Kalyan Kumaran
Tom Peterka
Joe Insley
Venkat Vishwanath
Argonne National Laboratory

David Daniel
Patricia Fasel
Los Alamos National Laboratory

Nicholas Frontiere
Argonne National Laboratory
Los Alamos National Laboratory
University of California, Los Angeles

Zarija Lukic
Lawrence Berkeley National Laboratory

Why Do We Do It?

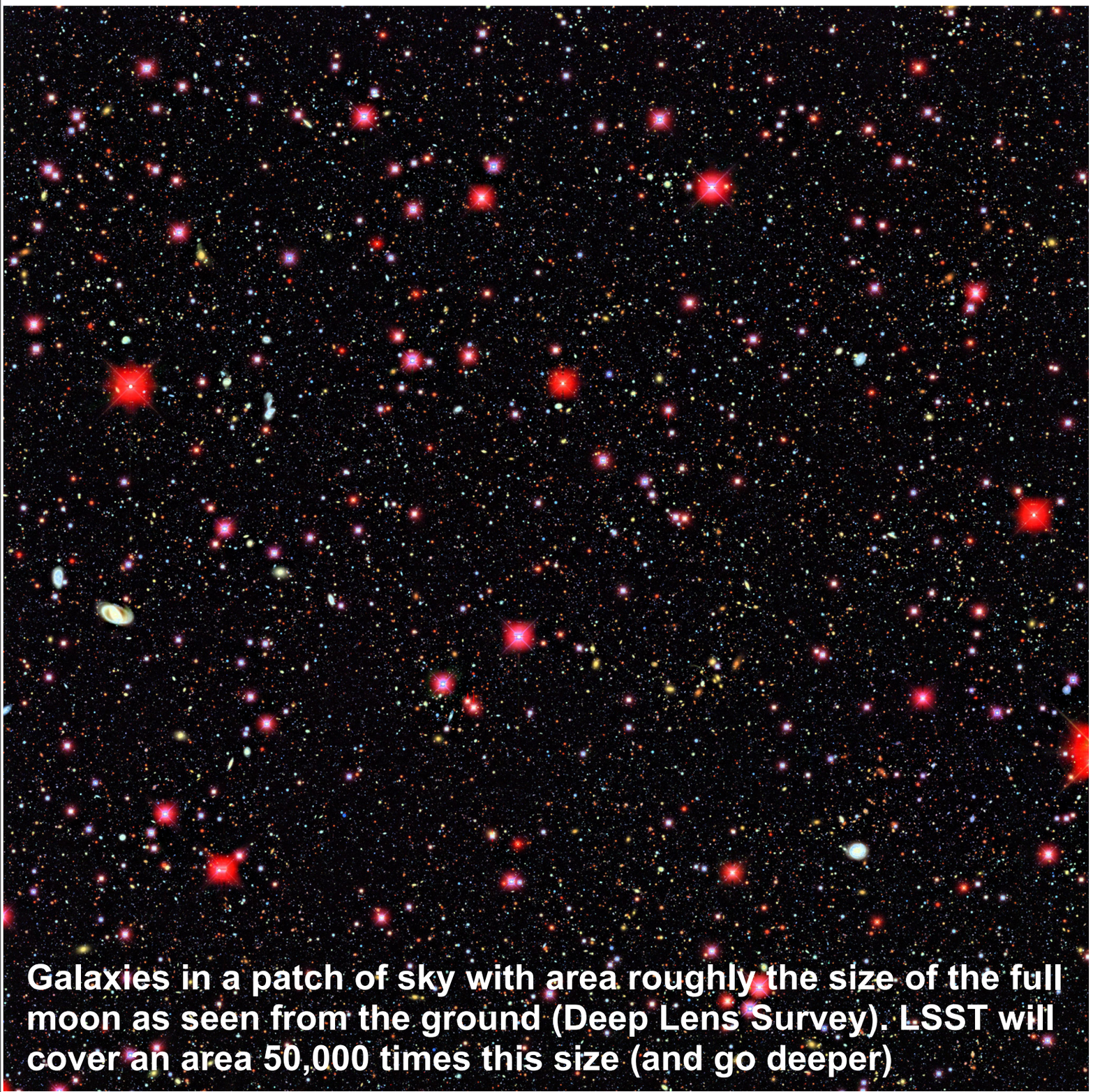


"But before the big boing, what was there?"

Well, not really --

The Dark Universe: Mapping the Sky

Challenge Posed by Cosmic Structure



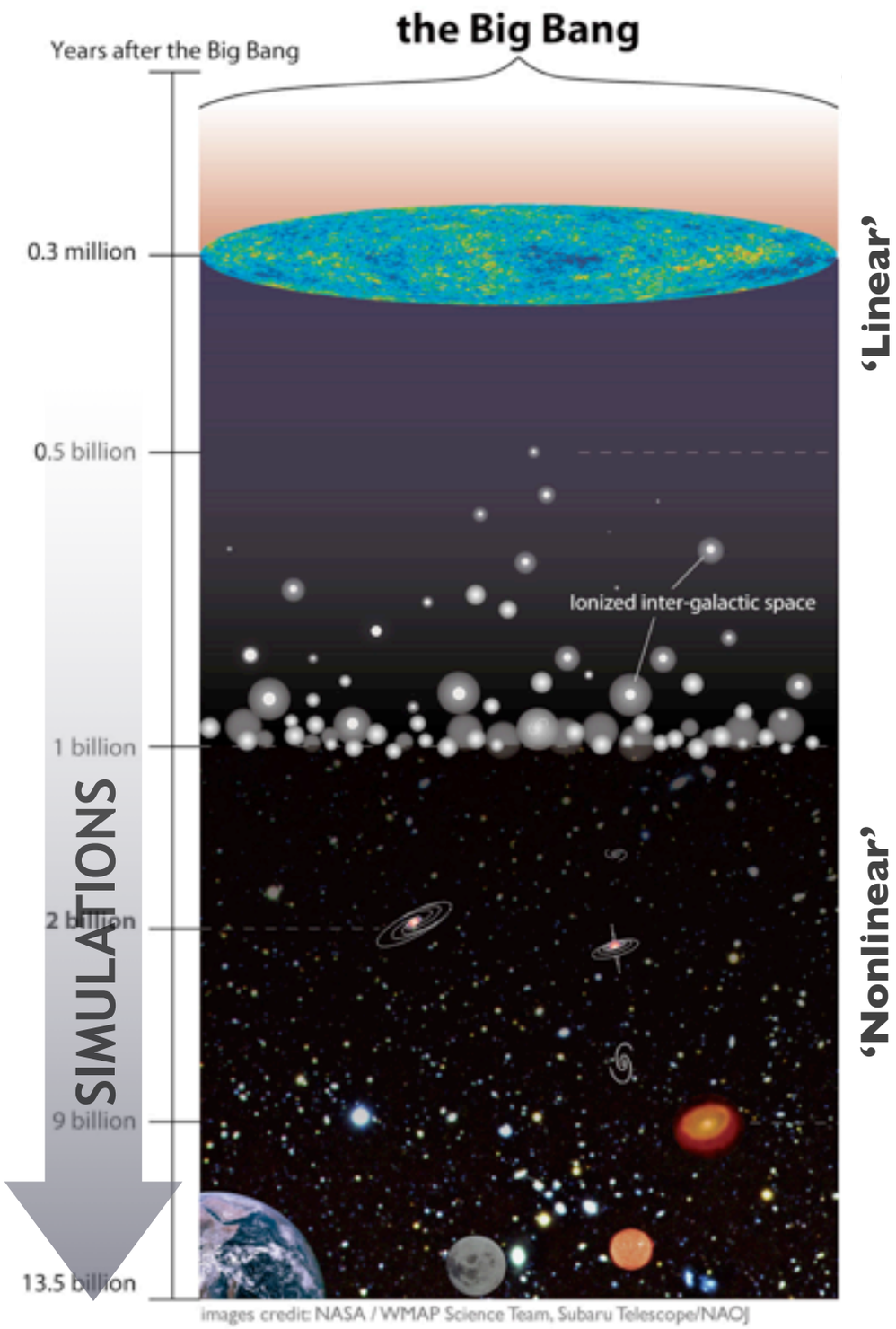
Galaxies in a patch of sky with area roughly the size of the full moon as seen from the ground (Deep Lens Survey). LSST will cover an area 50,000 times this size (and go deeper)

- **Cosmology = Physics + Statistics**
  - Mapping the sky with large-area surveys across multiple bands
  - LSST: ~4 billion galaxies total; ~200,000 galaxies per sq. deg. or ~40K galaxies over a sky patch the size of the moon
  - To 'understand' a dataset this large (~100 PB), we need to model the distribution of matter down to the scales of the individual galaxies, and over the size of the entire survey

Can the entire observable Universe be 'stuffed' inside a supercomputer?

Structure Formation in the Universe: The Basic Paradigm

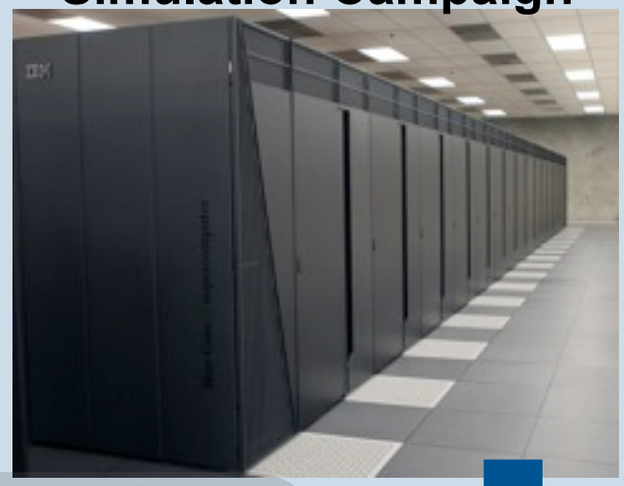
- **Solid understanding of structure formation is a requirement for cosmic discovery**
  - To high accuracy, initial conditions are given by a Gaussian random field
  - Initial perturbations amplified by gravitational instability in a dark matter-dominated Universe
  - Relevant theory is gravity and atomic physics (‘first principles’)
- **Early Universe**
  - Linear perturbation theory very successful (Cosmic Microwave Background)
- **The Universe: ‘Second Half’**
  - Nonlinear domain of structure formation, impossible to treat without large-scale computing



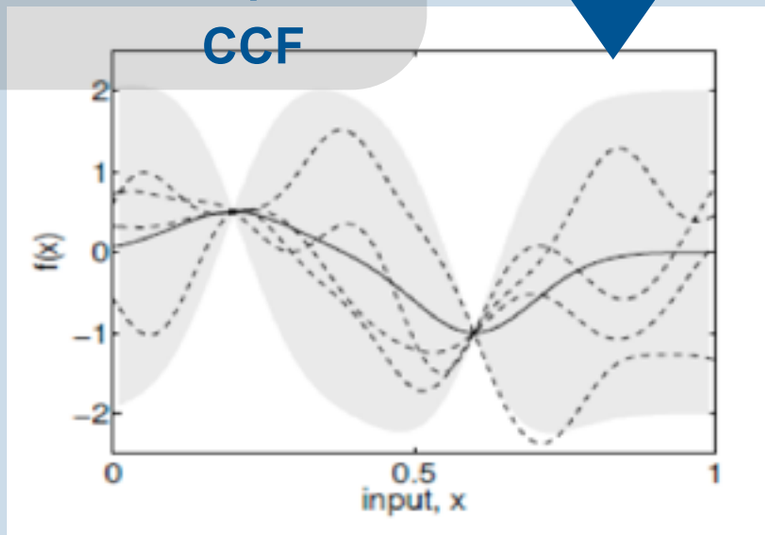
Precision Cosmology: Big Data Meets Supercomputing

SciDAC-3 Project

Supercomputer Simulation Campaign

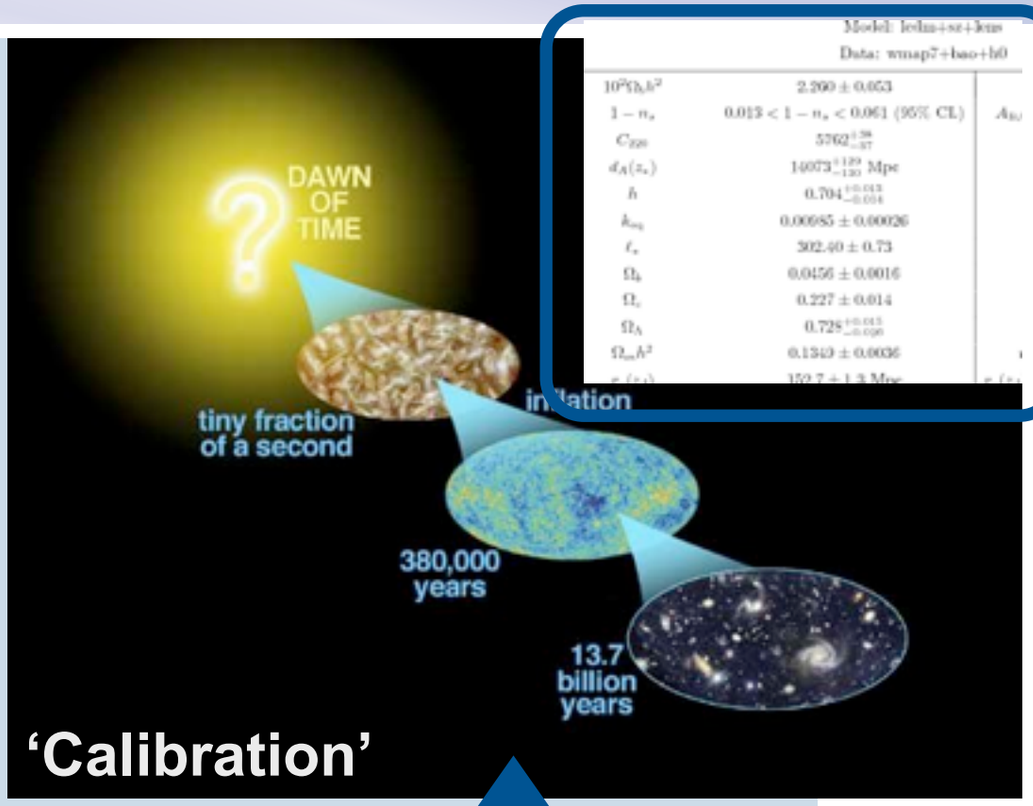


Simulations + CCF



Emulator based on Gaussian Process Interpolation in High-Dimensional Spaces

CCF= Cosmic Calibration Framework

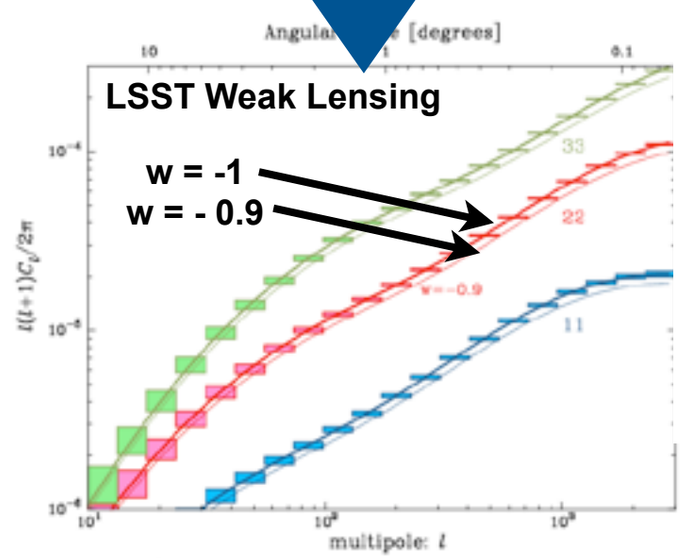


'Calibration'

'Precision Oracle'

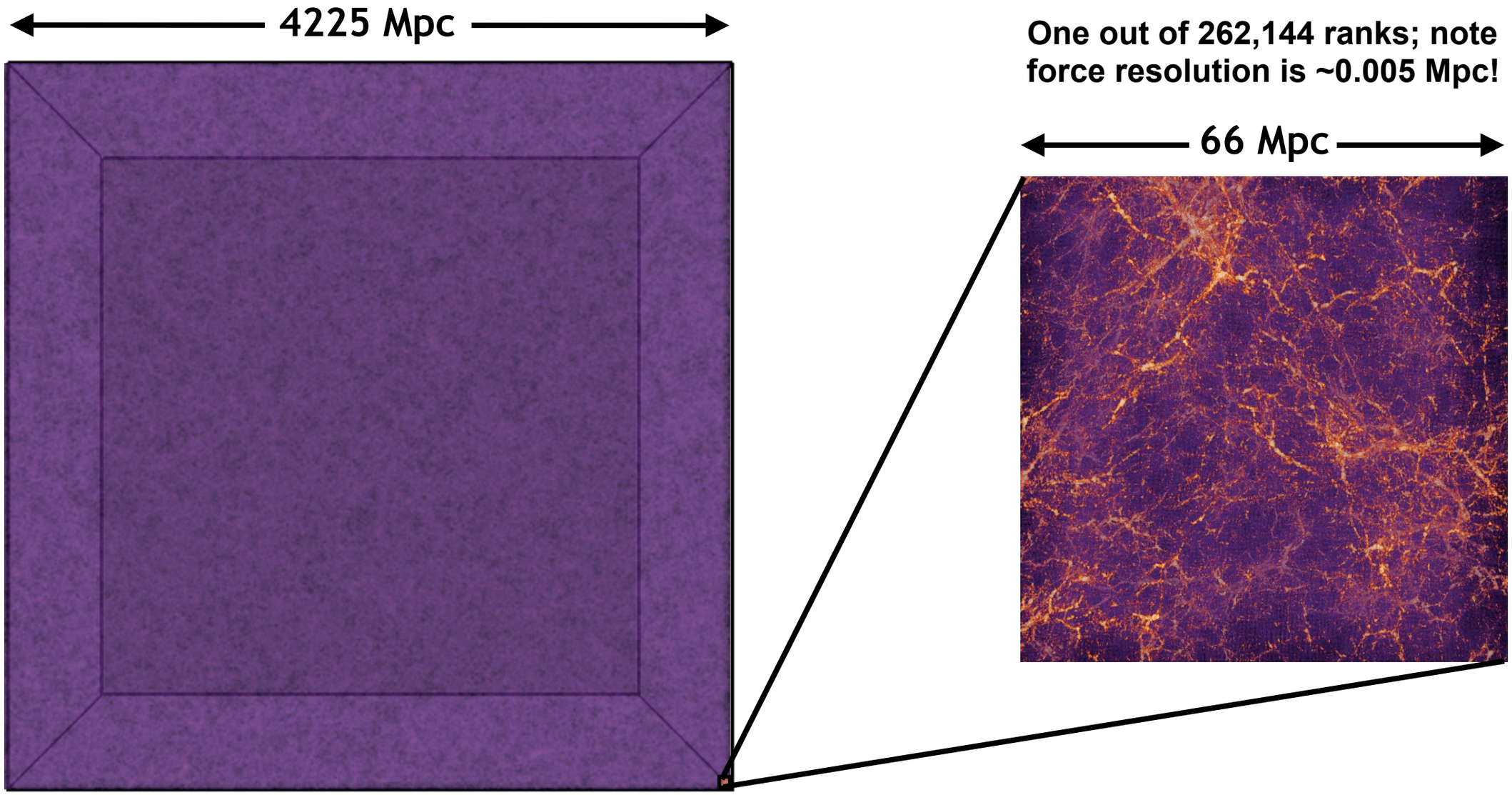
Markov chain Monte Carlo

Mapping the Sky with Survey Instruments



Observations: Statistical error bars will 'disappear' soon!

Capturing Sky Surveys: Trillion Particles in a 'Box'



- **Size:** Volumes = ~100's of cubic Gpc (1 pc = 3.26 light-years)
- To capture individual galaxy mass concentrations over this volume, need trillions of particles (billions of objects with thousands of sampling particles per object) -- simple numerical algorithms useless

1.1 trillion particle HACC science run at z=3 illustrating the dynamic range of a large, high-resolution, cosmological N-body simulation

Large Scale Structure Simulation Requirements

$$\frac{\partial f_i}{\partial t} + \dot{\mathbf{x}}\frac{\partial f_i}{\partial \mathbf{x}} - \nabla\phi\frac{\partial f_i}{\partial \mathbf{p}} = 0, \qquad \mathbf{p} = a^2\dot{\mathbf{x}},$$

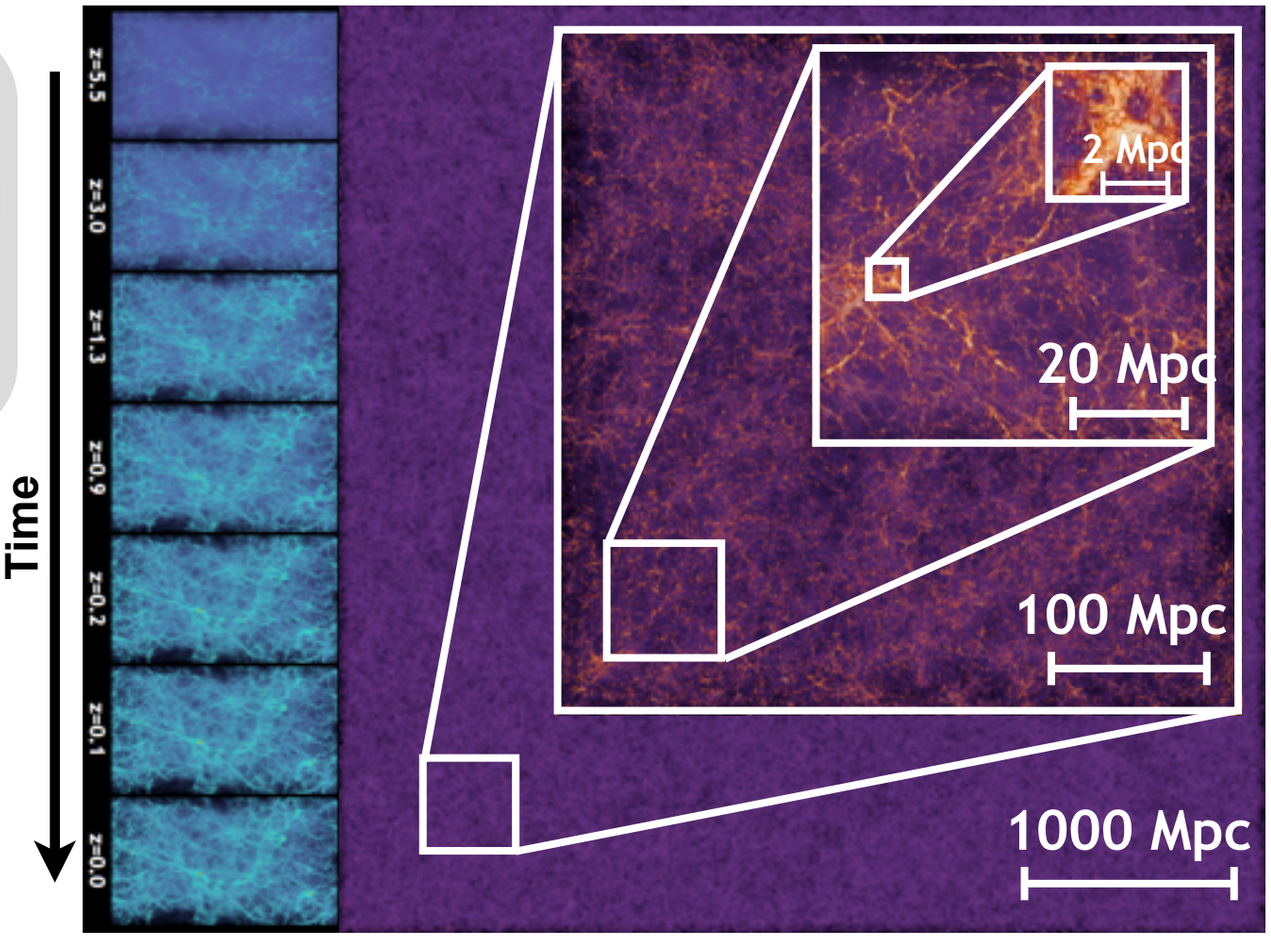
$$\nabla^2\phi = 4\pi G a^2(\rho(\mathbf{x},t) - \langle\rho_{\rm dm}(t)\rangle) = 4\pi G a^2 \Omega_{\rm dm}\delta_{\rm dm}\rho_{\rm cr},$$

$$\delta_{\rm dm}(\mathbf{x},t) = (\rho_{\rm dm} - \langle\rho_{\rm dm}\rangle)/\langle\rho_{\rm dm}\rangle),$$

$$\rho_{\rm dm}(\mathbf{x},t) = a^{-3}\sum_i m_i \int d^3\mathbf{p} f_i(\mathbf{x},\dot{\mathbf{x}},t).$$

Cosmological Vlasov-Poisson Equation

- **Resolution:**
  - Force resolution has to be ~kpc, a dynamic range of a million to one, also controls time-stepping
  - Local overdensity variation is ~million to one
- **Physics:**
  - Gravity dominates at scales greater than ~Mpc
  - At small scales: galaxy distribution modeling
- **Computing 'Boundary Conditions':**
  - Total memory in the PB+ class
  - Performance in the 10 PFlops+ class
  - Wall-clock of ~days/week, in situ analysis



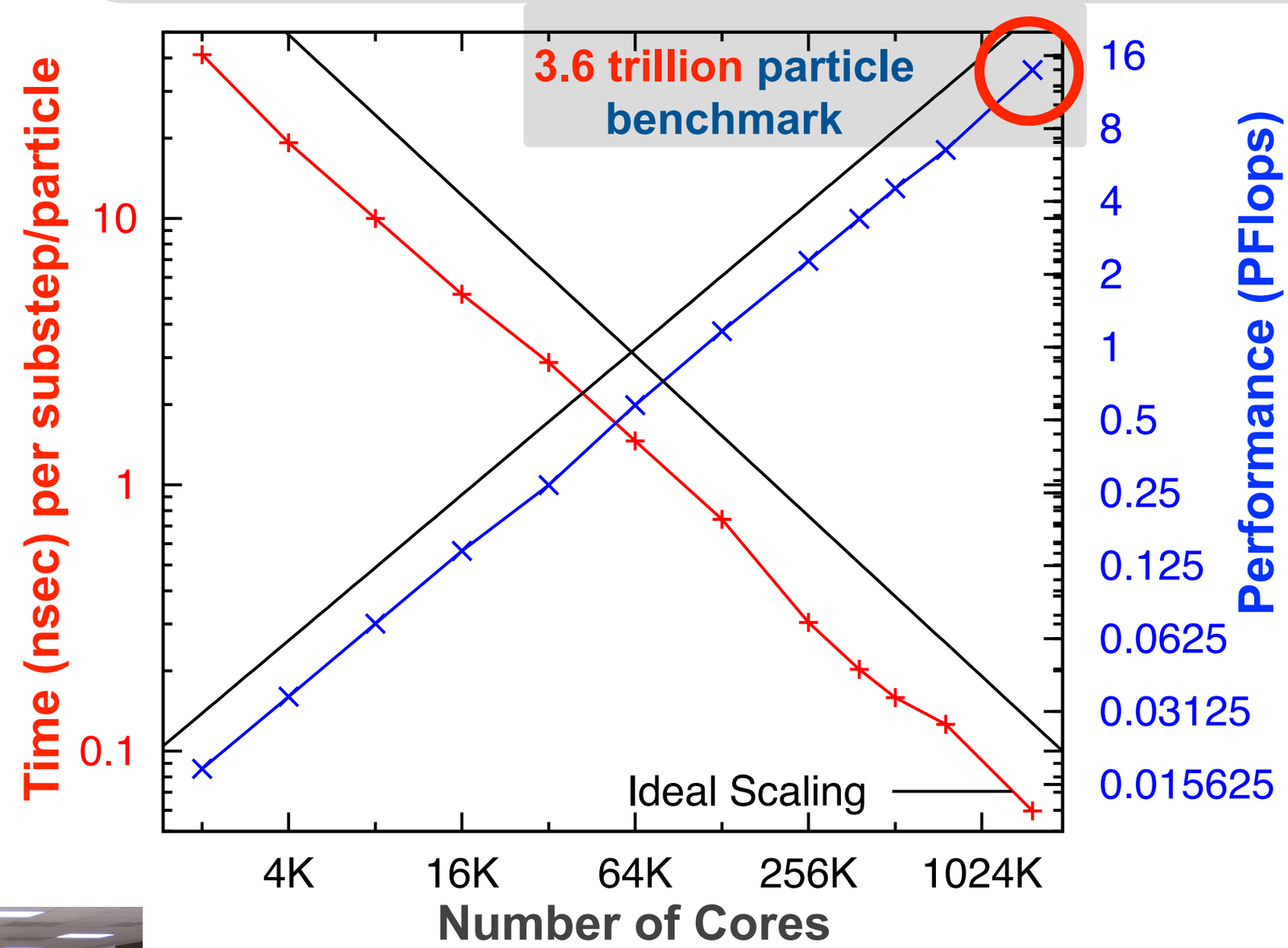
Gravitational Jeans Instablity

Can the Universe be run as a short computational 'experiment'?

Meeting the Challenge: HACC on the BG/Q

- New Cosmological N-Body Framework
  - Designed for extreme performance AND portability, including heterogeneous systems
  - Supports multiple programming models
  - Memory efficient
  - In situ analysis framework
  - Production science code

13.94 PFlops, 69.2% peak, 90% parallel efficiency on 1,572,864 cores/MPI ranks, 6.3M-way concurrency



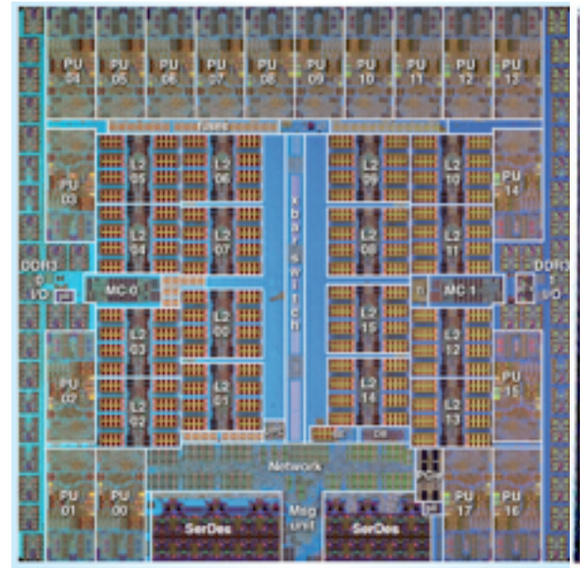
HACC weak scaling on the IBM BG/Q (MPI/OpenMP)

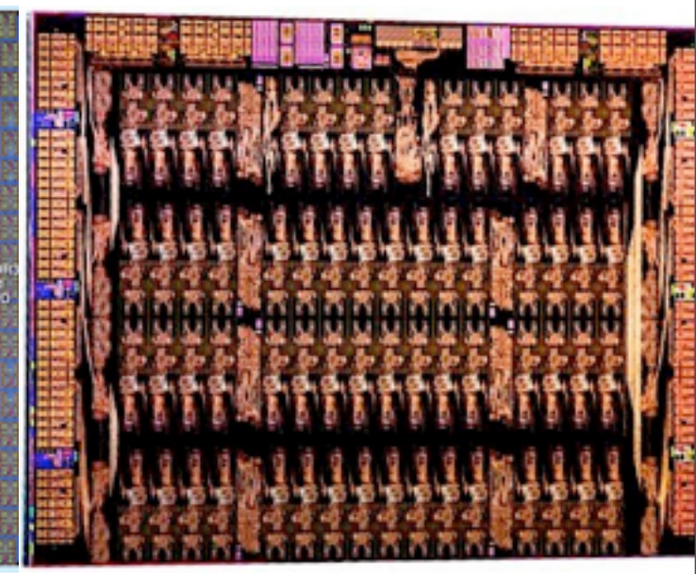
Co-Design vs. Code Design

- **HPC Myths**
  - The magic compiler
  - The magic programming model/ language (DSL)
  - Special-purpose hardware
  - Co-Design?
- **Dealing with (Current) HPC Reality**
  - Follow the architecture
  - Know the boundary conditions
  - There is no such thing as a 'code port'
  - Think out of the box
  - Get the best team
  - Work together

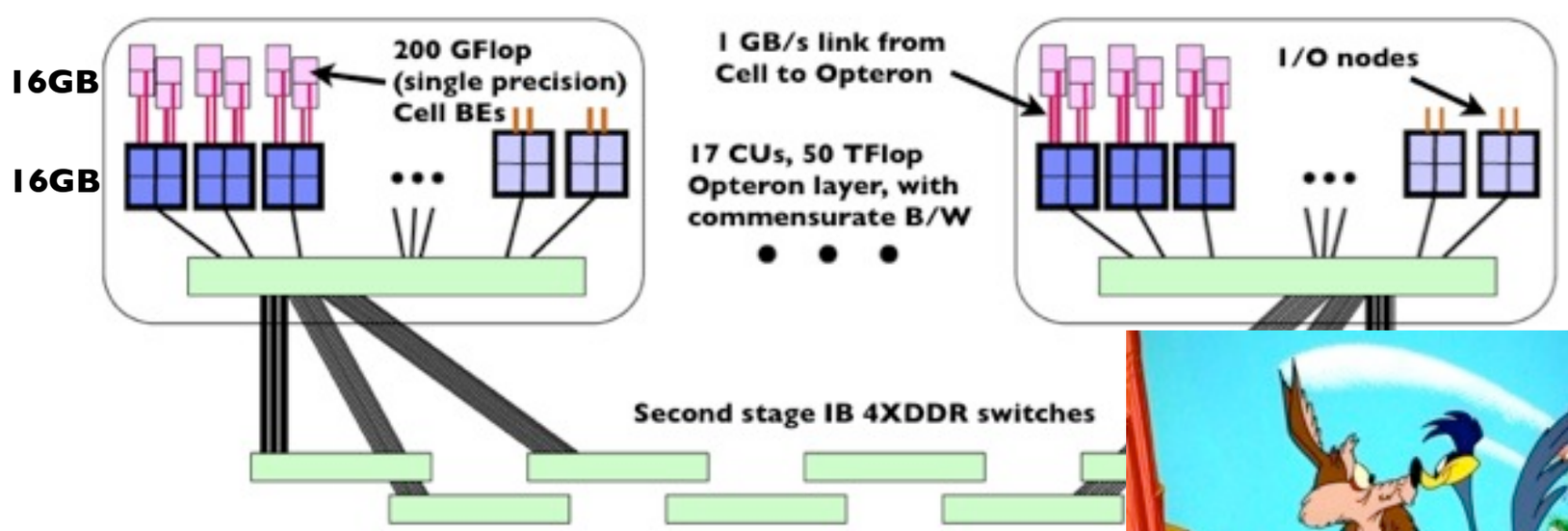
BQC:
- 16 cores
- 205 GFlops, 16 GB
- 32 MB L2, crossbar at 400 GB/s (memory connection is 40 GB/s)
- 5-D torus at 40 GB/s

Xeon Phi:
- 60 cores
- 1 TFlops, 8 GB
- 32 MB L2, ring at 300 GB/s (connects to cores and memory)
- 8 GB/s to host CPU





Average performance speed-up on ~10 applications codes on Titan is ~2 (ranging from 1.few to 7), but of Titan's 27 PFlops, only 2.5 PFlops are in the CPU! What is wrong with this picture?

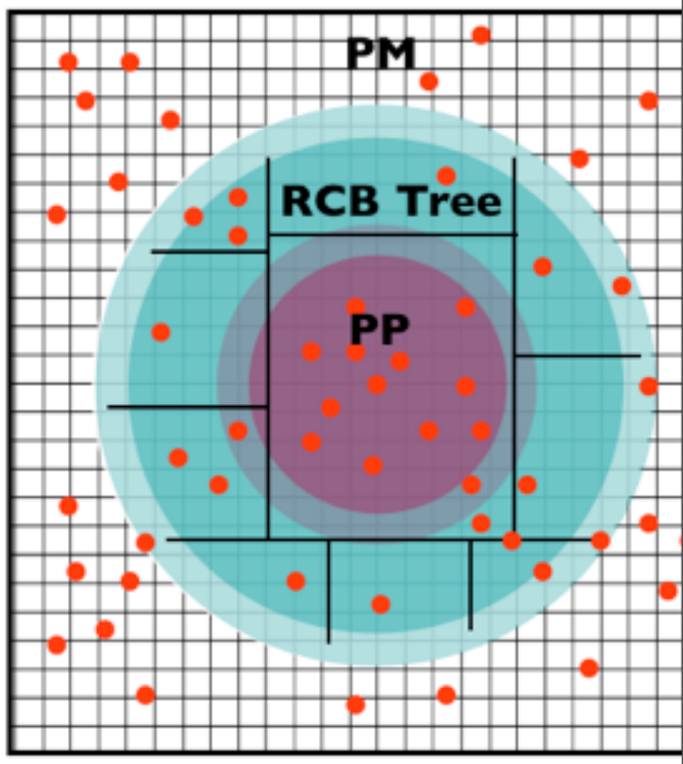


Roadrunner

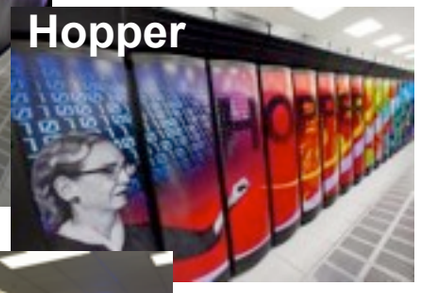
Opening the HACC 'Black Box': Design Principles

Andrew White Dec 7, 2007 + What if you had a petaflop/s

- **Optimize Next-Generation Code 'Ecology':** Numerical methods, algorithms, mixed precision, data locality, scalability, I/O, in situ analysis -- life-cycle significantly longer than architecture timescales
- **Framework design:** Support a 'universal' top layer + 'plug-in' optimized node-level components; minimize data structure complexity and data motion -- support multiple programming models
- **Performance:** Optimization stresses scalability, low memory overhead, and platform flexibility; assume 'on your own' for software support, but hook into tools as available (e.g., ESSL FFT)
- **Optimal Splitting of Gravitational Forces:** Spectral Particle-Mesh melded with direct and RCB tree force solvers, short hand-over scale (dynamic range splitting ~ 10,000 X 100)
- **Compute to Communication balance:** Particle Overloading
- **Time-Stepping:** Symplectic, sub-cycled (uses Hamiltonian Maps)
- **Force Kernel:** Highly optimized force kernel takes up large fraction of compute time, no look-ups due to short hand-over scale
- **Production Readiness:** runs on all supercomputer architectures

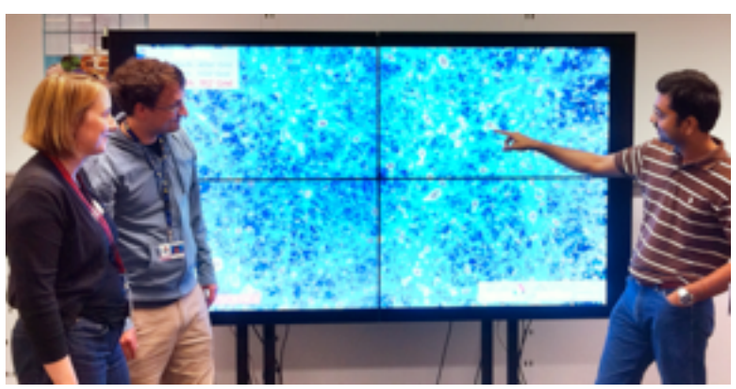


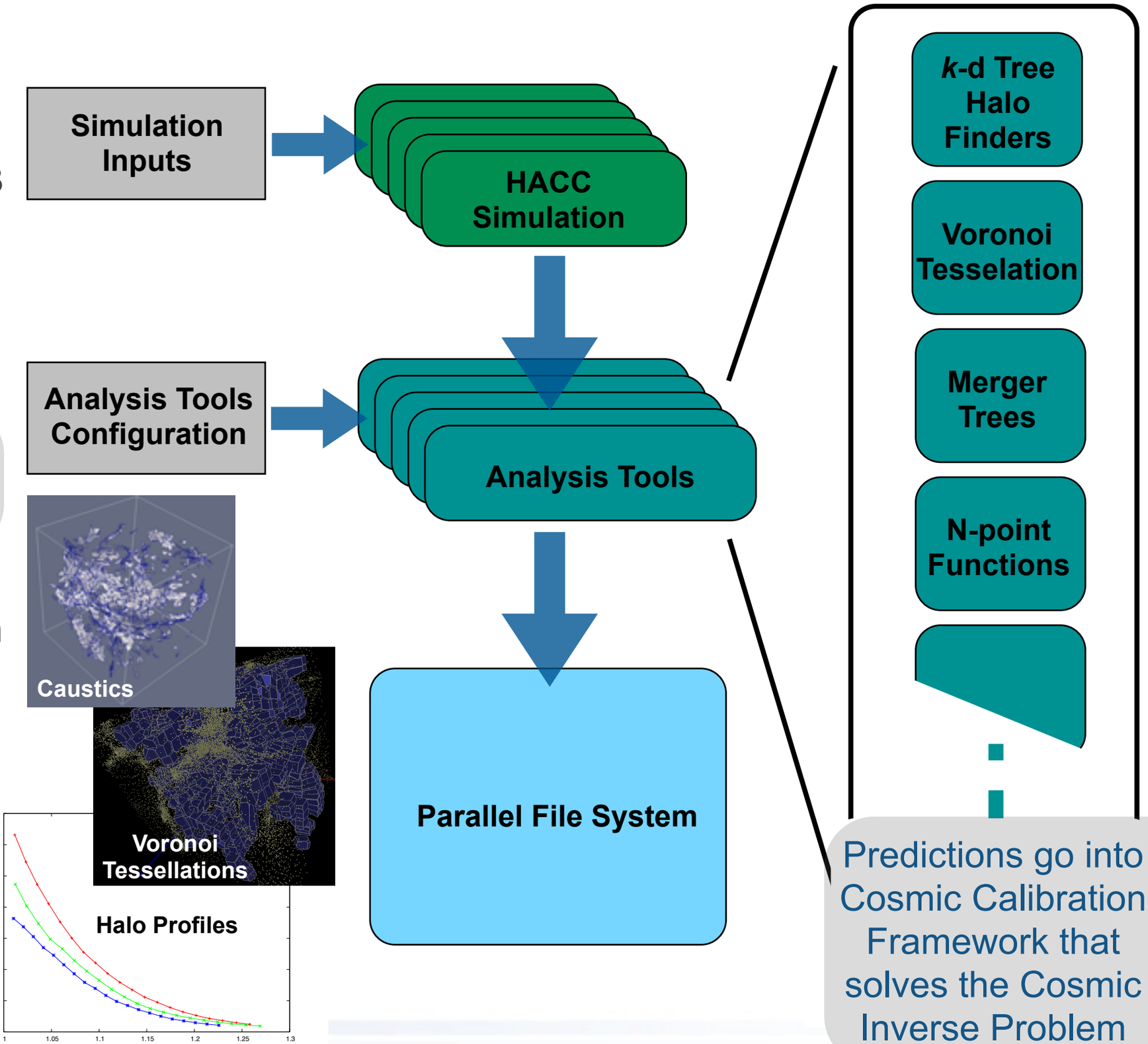
HACC force hierarchy (PPTreePM)



HACC: *Fast* In Situ Analysis

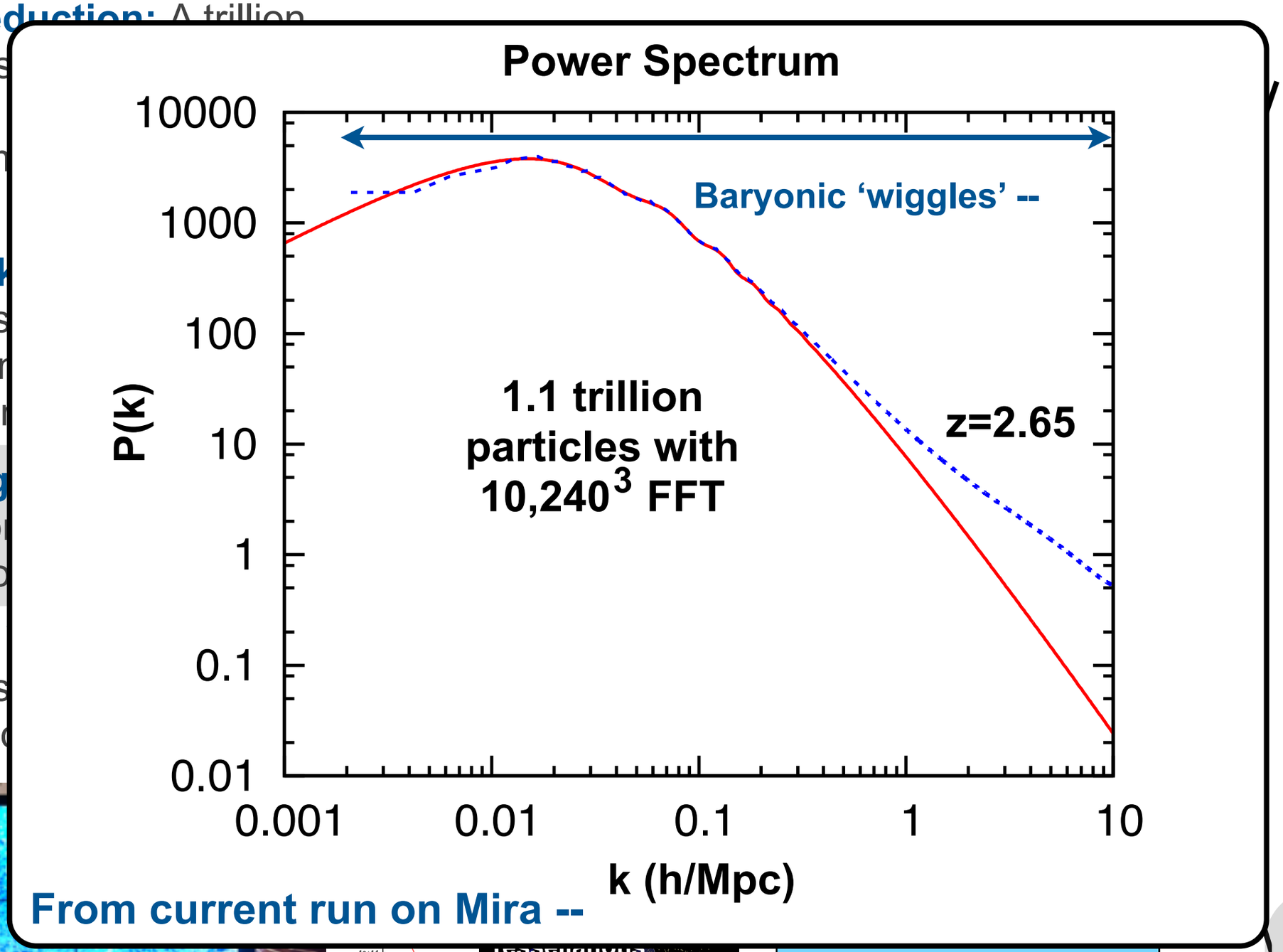
- **Data Reduction:** A trillion particle simulation with 100 analysis steps has a storage requirement of ~4 PB -- in situ analysis reduces it to ~200 TB
- **I/O Chokepoints:** Large data analyses difficult because I/O time > analysis time, plus scheduling overhead
- **Fast Algorithms:** Analysis time is only a fraction of a full simulation timestep
- **Ease of Workflow:** Large analyses difficult to manage in post-processing



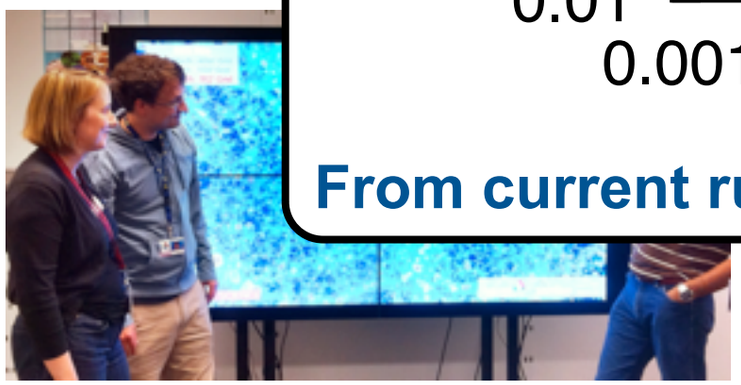


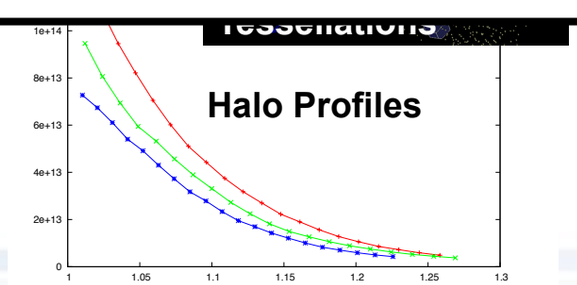
HACC: *Fast* In Situ Analysis

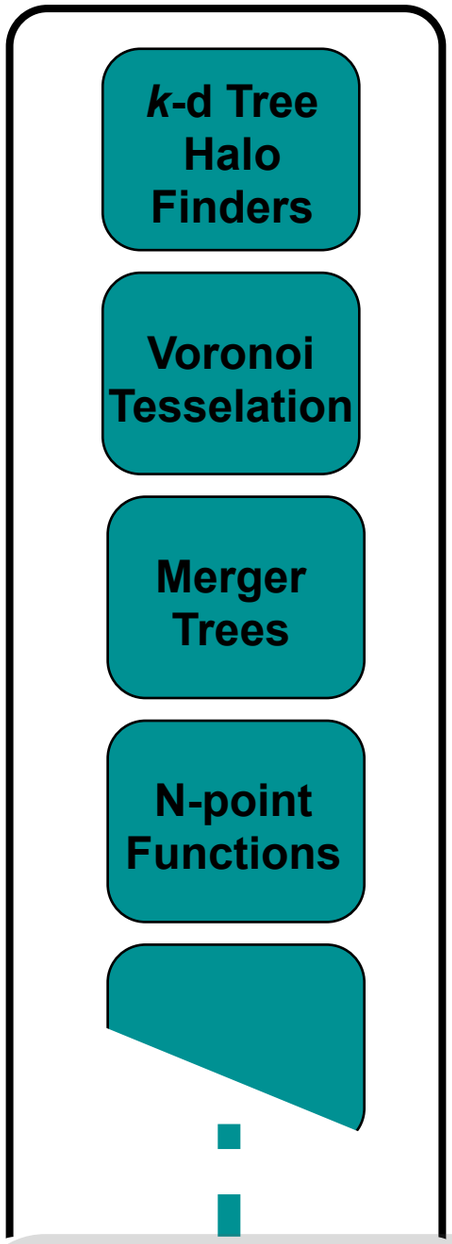
- **Data Reduction:** A trillion particle s
analysis
requirem
analysis
- **I/O Chok**
analyses
time > ar
schedulin
- **Fast Alg**
time is o
simulatio
- **Ease of**
analyses
post-proc



From current run on Mira --



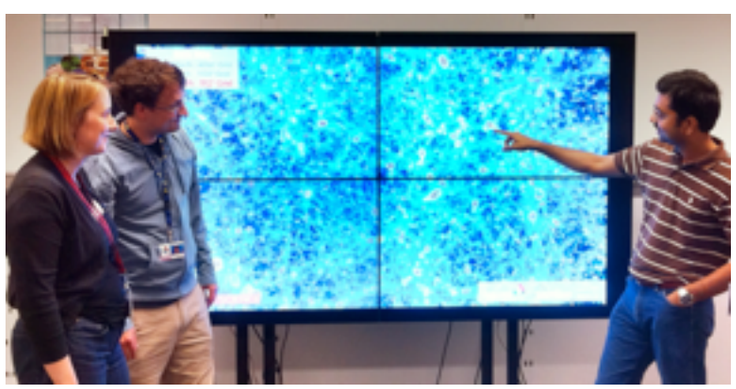


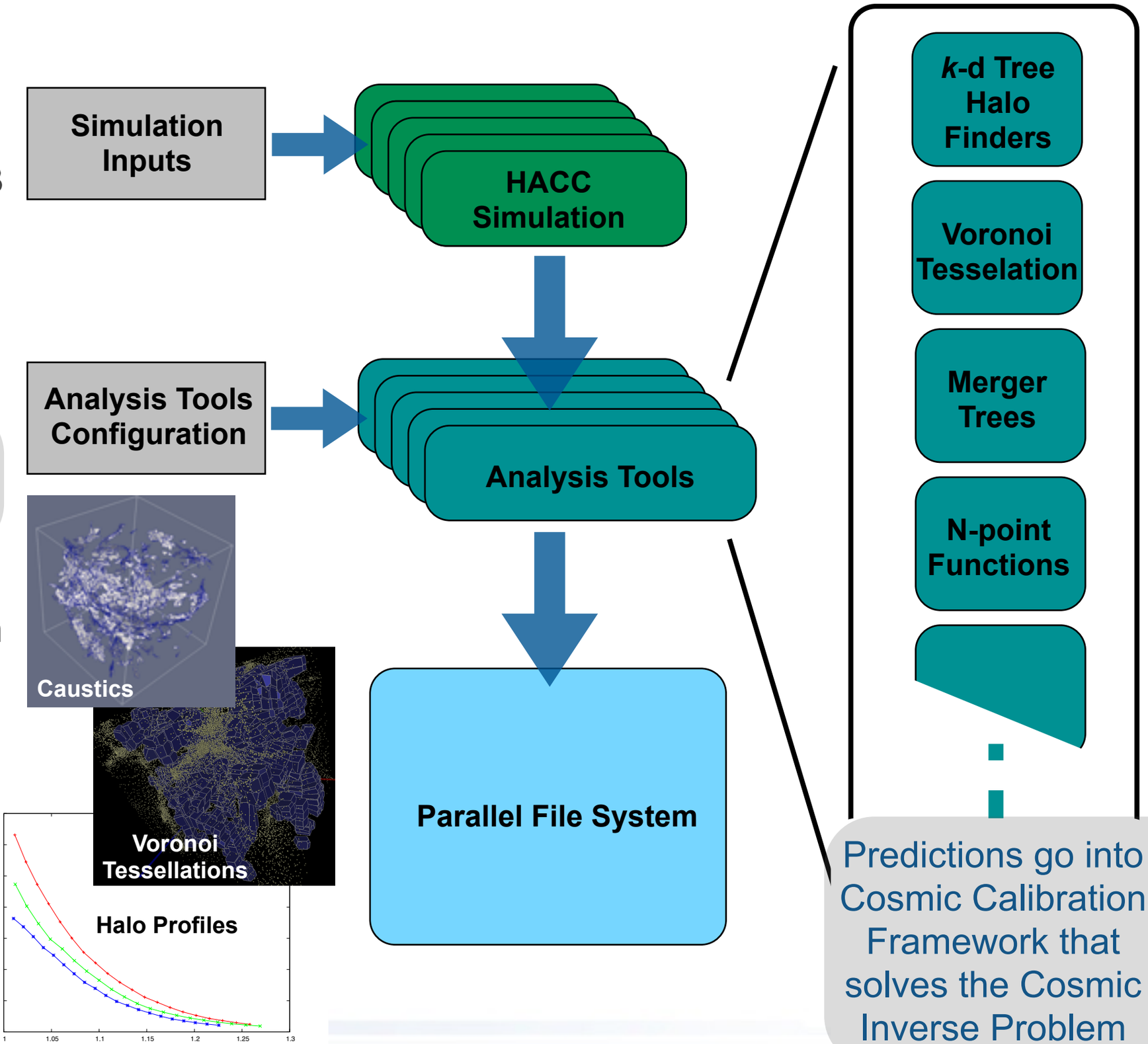


Predictions go into Cosmic Calibration Framework that solves the Cosmic Inverse Problem

HACC: *Fast* In Situ Analysis

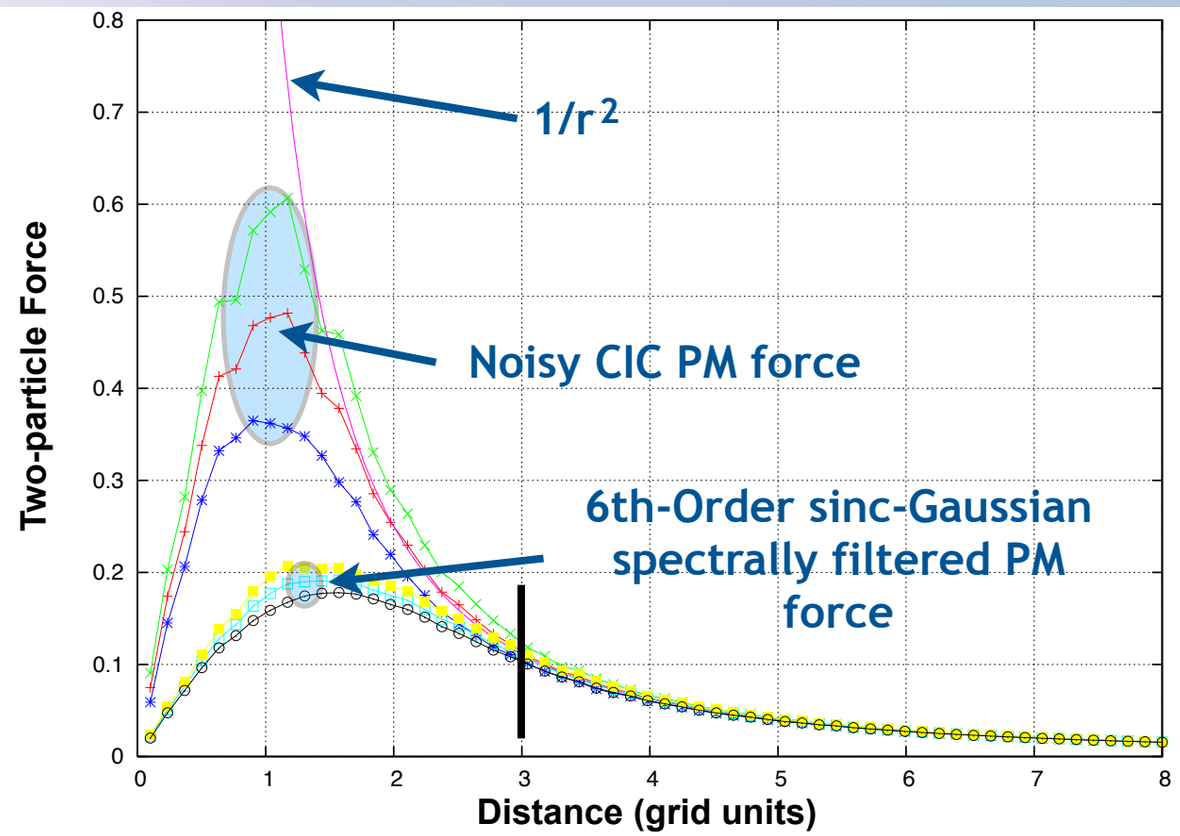
- **Data Reduction:** A trillion particle simulation with 100 analysis steps has a storage requirement of ~4 PB -- in situ analysis reduces it to ~200 TB
- **I/O Chokepoints:** Large data analyses difficult because I/O time > analysis time, plus scheduling overhead
- **Fast Algorithms:** Analysis time is only a fraction of a full simulation timestep
- **Ease of Workflow:** Large analyses difficult to manage in post-processing

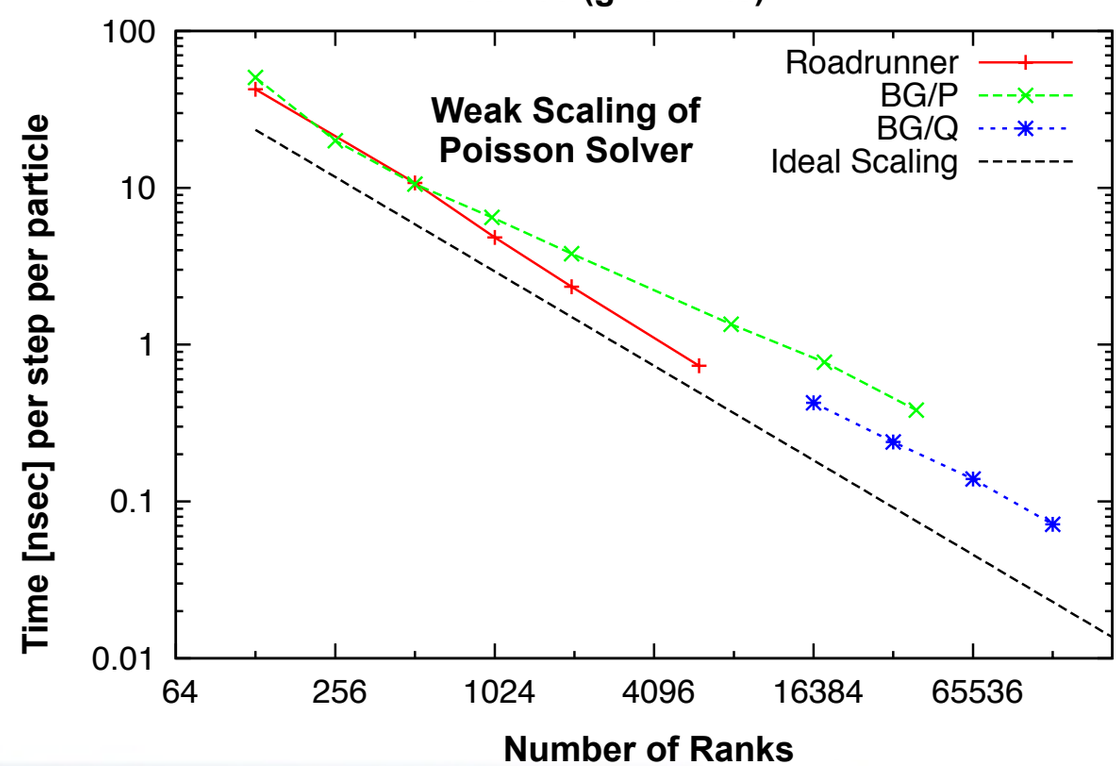




Splitting the Force: The Long-Range Solver

- **Spectral Particle-Mesh Solver:** Custom (large) FFT-based method -- uses (i) 6-th order Green function, (ii) 4th order spectral Super-Lanczos gradients, (iii) high-order spectral filtering to reduce grid anisotropy noise
- **Short-range Force:** Asymptotically correct semi-analytic expression for the difference between the Newtonian and the long-range force; uses a 5th order polynomial
- **Pencil-decomposed Parallel 3-D FFT:** Fast 3D-to-2D combinatorics, FFT performance theoretically viable to exascale systems; HACC scalability depends entirely on FFT performance
- **Time-stepping uses Symplectic Sub-cycling:** Time-stepping via 2nd-order accurate symplectic maps with 'KSK' for the global timestep, where 'S' is split into multiple 'SKS' local force steps





Splitting the Force: The Long-Range Solver

$$G_6(\mathbf{k}) = \frac{45}{128}\Delta^2 \left[\sum_i \cos\left(\frac{2\pi k_i \Delta}{L}\right) - \frac{5}{64}\sum_i \cos\left(\frac{4\pi k_i \Delta}{L}\right) + \frac{1}{1024}\sum_i \cos\left(\frac{8\pi k_i \Delta}{L}\right) - \frac{2835}{1024}\right]^{-1}$$

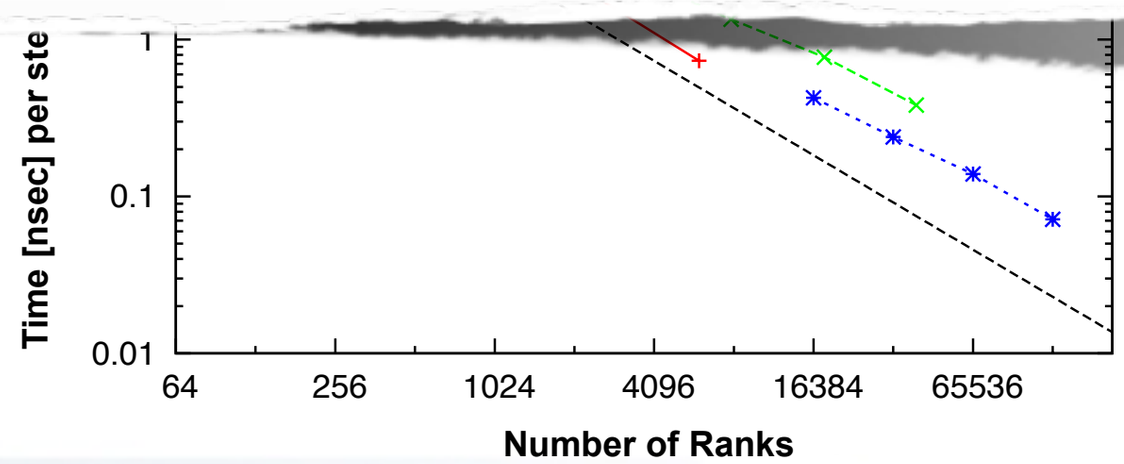
$$\left.\frac{\Delta f}{\Delta x}\right|_4 = \frac{4}{3}\sum_{j=-N+1}^{N} iC_j \mathrm{e}^{(2\pi jx/L)} \frac{2\pi j\Delta}{L}\frac{\sin(2\pi j\Delta/L)}{2\pi j\Delta/L} - \frac{1}{6}\sum_{j=-N+1}^{N} iC_j \mathrm{e}^{(2\pi jx/L)} \frac{2\pi j\Delta}{L}\frac{\sin(4\pi j\Delta/L)}{2\pi j\Delta/L}$$

where the C_j are the coefficients in the Fourier expansion of f

$$S(k) = \exp\left(-\frac{1}{4}k^2\sigma^2\right)\left[\left(\frac{2k}{\Delta}\right)\sin\left(\frac{k\Delta}{2}\right)\right]^{n_s}$$

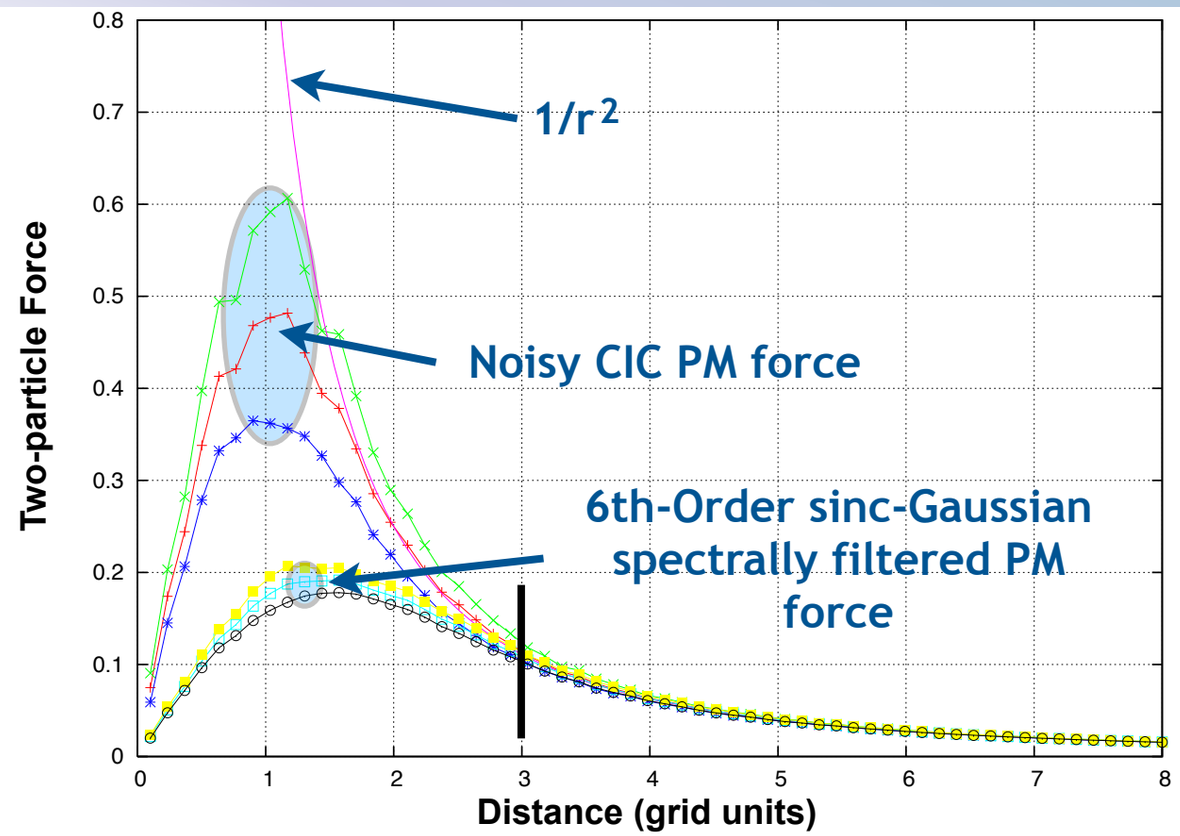
$$f_{grid}(r) = \frac{1}{r^2}\tanh(br) - \frac{b}{r}\frac{1}{\cosh^2(br)} + cr\left(1+dr^2\right)\exp\left(-dr^2\right) + e\left(1+fr^2+gr^4+lr^6\right)\exp\left(-hr^2\right)$$

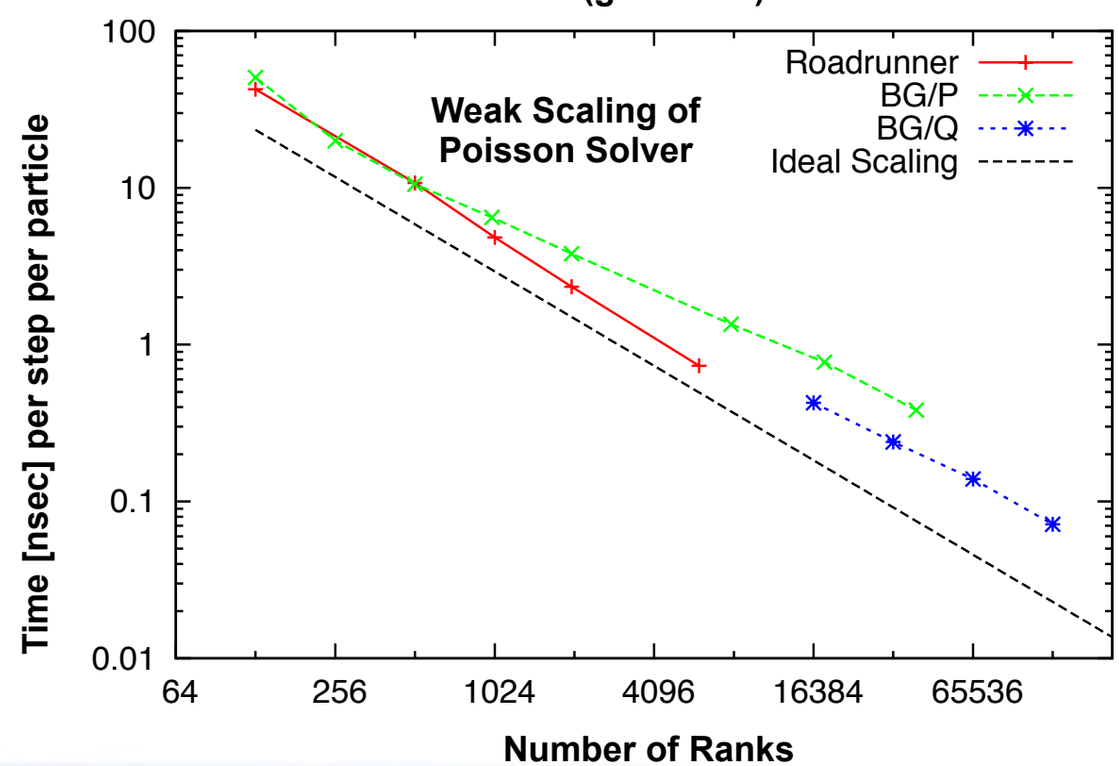
- **Time-stepping uses Symplectic Sub-cycling:** Time-stepping via 2nd-order accurate symplectic maps with ‘KSK’ for the global timestep, where ‘S’ is split into multiple ‘SKS’ local force steps



Splitting the Force: The Long-Range Solver

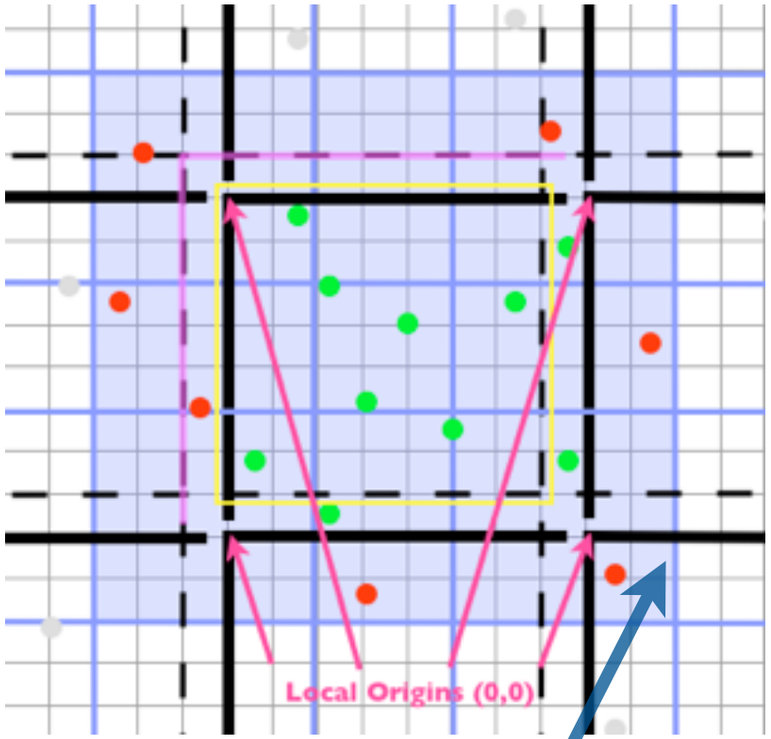
- **Spectral Particle-Mesh Solver:** Custom (large) FFT-based method -- uses (i) 6-th order Green function, (ii) 4th order spectral Super-Lanczos gradients, (iii) high-order spectral filtering to reduce grid anisotropy noise
- **Short-range Force:** Asymptotically correct semi-analytic expression for the difference between the Newtonian and the long-range force; uses a 5th order polynomial
- **Pencil-decomposed Parallel 3-D FFT:** Fast 3D-to-2D combinatorics, FFT performance theoretically viable to exascale systems; HACC scalability depends entirely on FFT performance
- **Time-stepping uses Symplectic Sub-cycling:** Time-stepping via 2nd-order accurate symplectic maps with 'KSK' for the global timestep, where 'S' is split into multiple 'SKS' local force steps



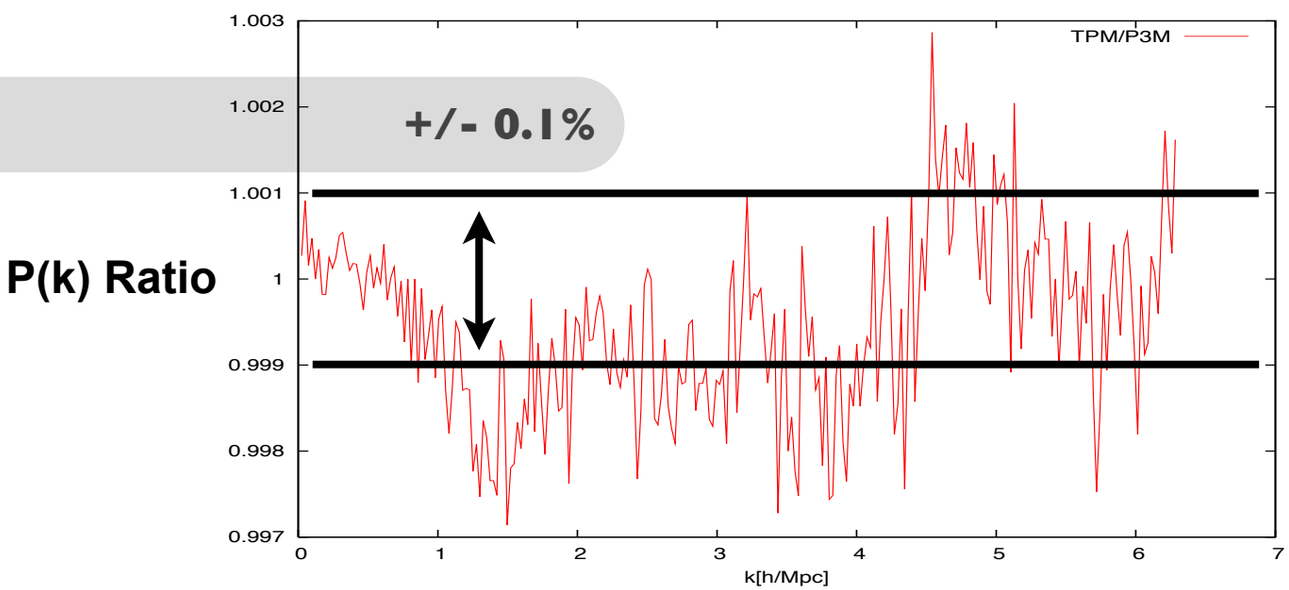


Particle Overloading and Short-Range Solvers

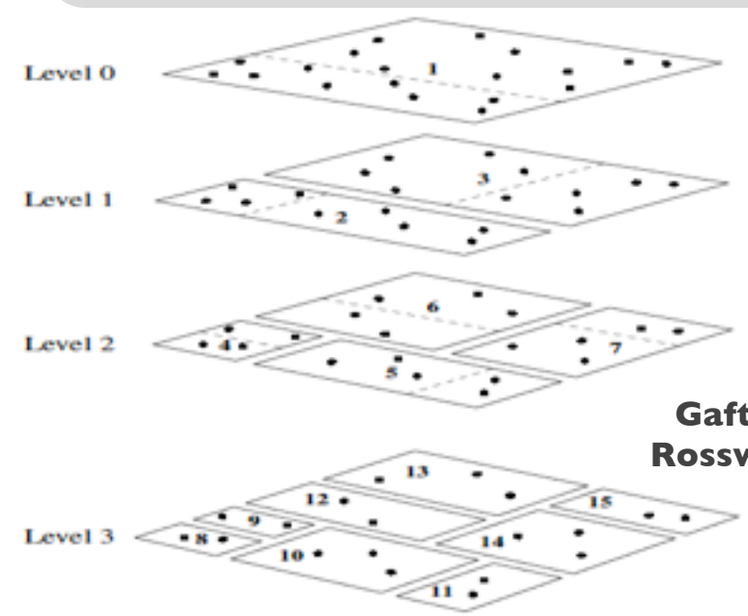
- **Particle Overloading:** Particle replication instead of conventional guard zones with 3-D domain decomposition -- minimizes inter-processor communication and allows for swappable short-range solvers
- **Short-range Force:** Depending on node architecture switch between P3M and PPTreePM algorithms (pseudo-particle method goes beyond monopole order), by tuning number of particles in leaf nodes and error control criteria, optimize for computational efficiency
- **Error tests:** Can directly compare different short-range solver algorithms



Overload Zone (particle 'cache')



HACC Force Algorithm Test: PPTreePM vs. P3M

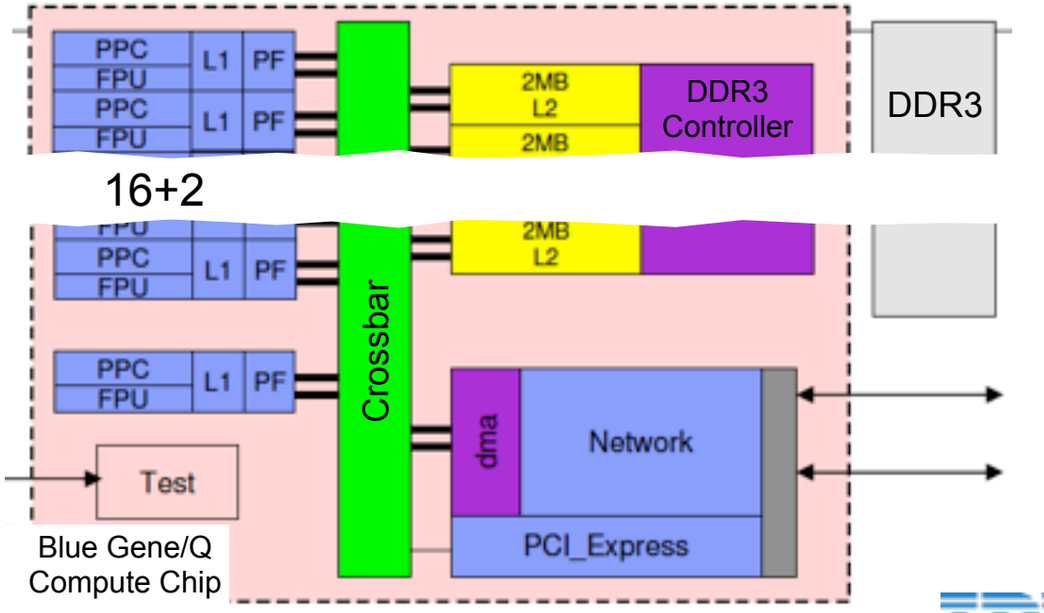


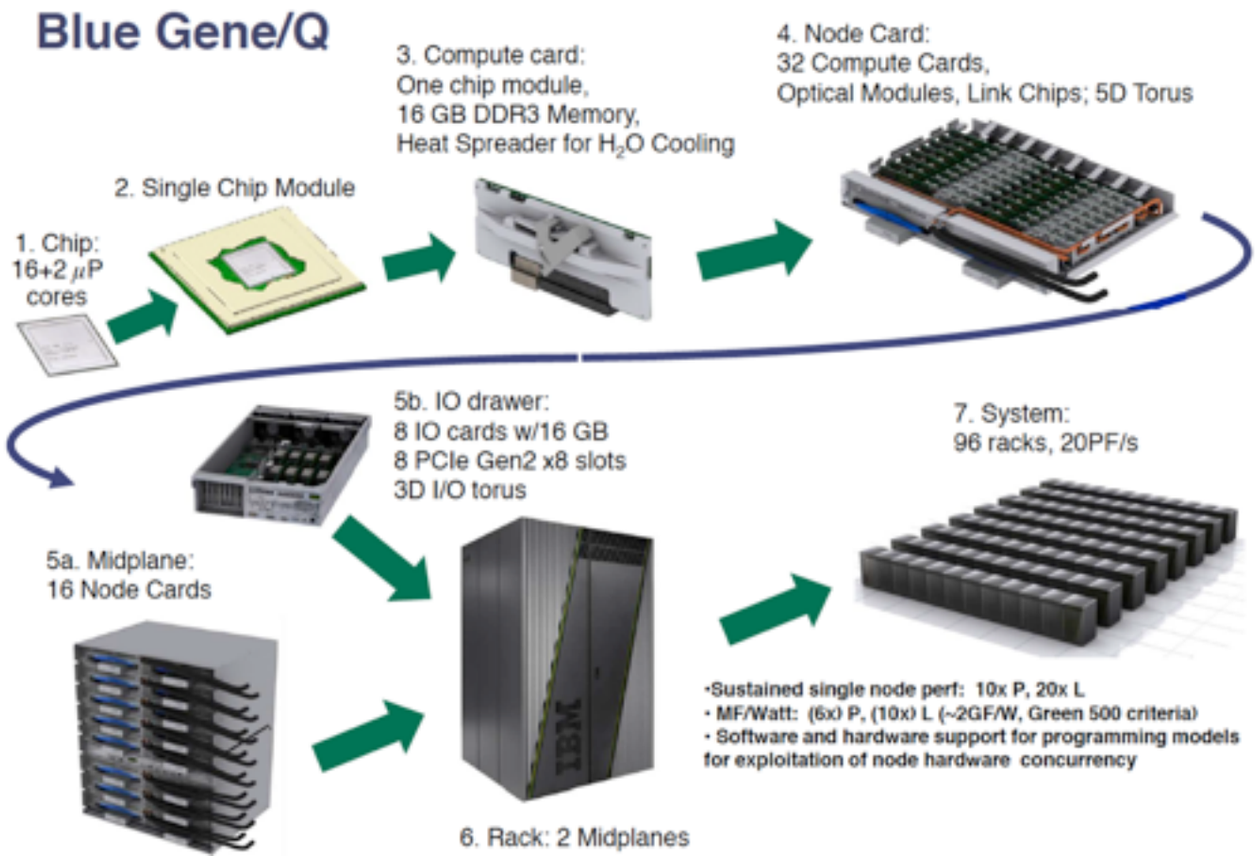
Gafton and Rosswog 2011

RCB Tree Hierarchy

HACC: BG/Q Implementation I

- **BG/Q Basics**
  - **BQC:** SoC design with 16+1 1.6 GHz PowerPC A2 processor cores with SIMD quad FPU (4 FMAs per cycle, 204.8 GFlops), connecting to an L2 cache (32 MB) via a crossbar switch, 563 GB/s internal bisection BW, 16 GB RAM external memory
  - **Network:** 5-D torus network, 10 comm links/ compute node at peak aggregate BW of 40 GB/s, 31 hops on 96 racks -- latency of 3 micro sec
  - **I/O:** 8 I/O nodes/rack; 240 GB/s peak on 48 racks (35 PB storage at ALCF); currently seeing >80% of peak on large read/writes
  - **Programming Model:** Two-tiered programming model, message passing plus shared memory (MPI + OpenMP, --), on one node can run 64 processes with 1 thread per process to 1 process with 64 threads





HACC: BG/Q Implementation II

- **HACC BG/Q Algorithms:**

  1) Long-range force with base HACC FFT-based SPM (excellent performance)

  2) Short-range force: Particle-Particle + RCB Tree + highly tuned force kernel

- **Data Locality:** At rank-level, enforced by particle overloading, at tree-level use the RCB grouping to organize particle memory buffers (all P-P interactions are in nearby leaf nodes, this also increases accuracy)

- **Tree Build/Walk Minimization:** Every particle has an interaction list -- constructing this is an overhead ('treebuild'); reduce tree depth in two ways: (i) rank-local trees, (ii) shortest possible hand-over scale, (iii) bigger P-P component than is usual, using the optimized force kernel

- **Force Kernel:** Because of the compactness of the short-range interaction, the kernel can be represented as

$$f_{SR} = (s+\epsilon)^{-3/2} - f_{grid}(s)$$

  where

$$s = \mathbf{r}\cdot\mathbf{r}, \quad f_{grid}(s) = poly[5](s)$$

- **Kernel Evaluation:** This consists of three parts: (i) Filtering, (ii) Inverse square root evaluation, and (iii) Polynomial evaluation

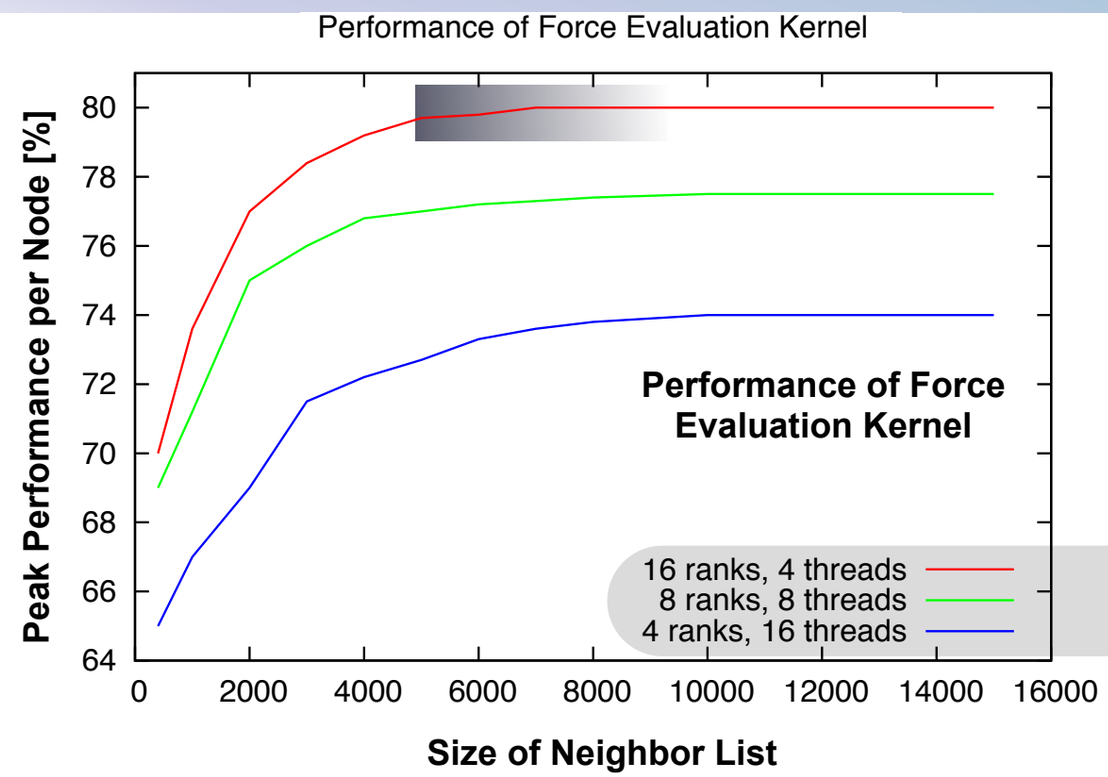
HACC: BG/Q Implementation III

- **RCB 'Fat Leaf' Tree:**
- Vectorize kernel evaluation by evaluating the force of every neighbor of each particle at once -- every particle has an independent interaction list of particles and is processed within a separate thread; every particle on a leaf node shares the interaction list, therefore all lists are the same size, automatically balancing the computational threads
- The generated neighbor list has particle data stored in SOA ('structure of arrays') format; each array is contiguous and properly aligned -- preparation of the data structure for the force evaluation kernel
- Within the force evaluation kernel, explicit use is made of the dcbt instruction and stream prefetch policy to load the data, all with vector instructions
- **Force Evaluation (~5% in FFT, ~10% in treebuild/CIC, etc., >80% in force kernel):**
- Short-range condition test is included in the force evaluation, vectorizing the entire computation
- Hide BG/Q instruction latency (6 cycles) by spacing dependent instructions, 2-fold loop unrolling, and running 4 OpenMP threads per core
- Register pressure for the 32 vector FP registers is a key design constraint
- Performance: filtering -- two fsel QPX instructions (50% of peak), s evaluation -- 4 adds and 6 FMAs (80% of peak), inverse square root -- rsqrt QPX plus one Newton iteration with 3 FMAs (80% of peak), poly[5] -- 6 FMAs (100% of peak); total 80% of peak (theoretical max 81%)
- Very high Flop/byte ratio = 206/64

HACC: BG/Q Implementation IV

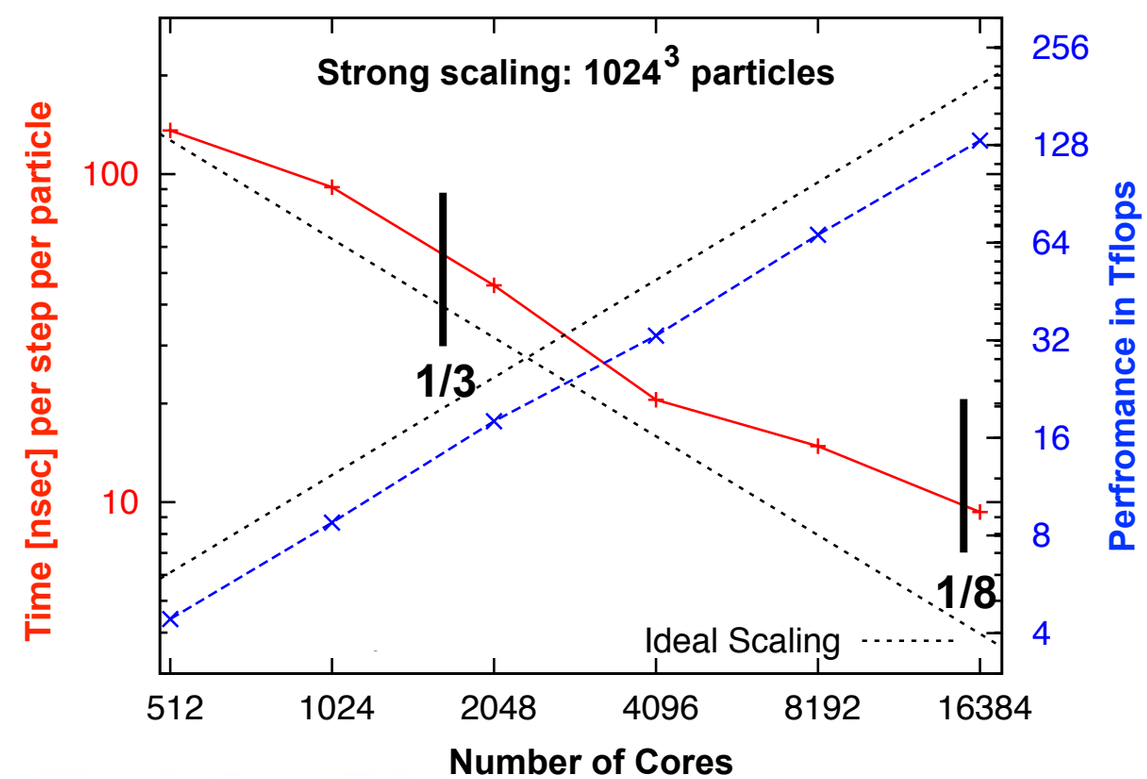
MPI Ranks vs. OpenMP Threads:

- Reasonably stable performance is attained at 4 hardware threads/core, the maximum possible
- A relatively large number of OpenMP threads is compatible with good performance; 16X4 is somewhat better than 8X8 in this test case



Strong Scaling (Readiness for the Future):

- Cosmological simulations are typically (close to) memory-bound -- HACC is designed to run at >50% memory utilization on the BG/Q to about a factor of 3 less
- The performance is very good even at very low memory footprint
- Effective particle push time is degraded only slightly due to the particle replication overhead (billion particles on 16384 cores)



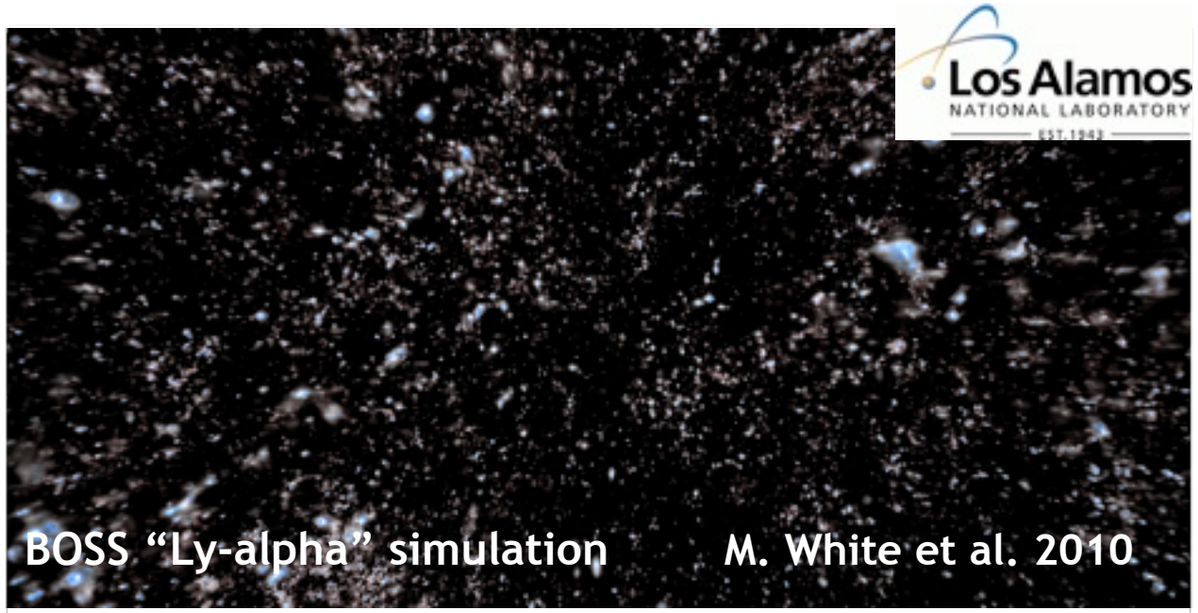
HACC: BG/Q Performance Table

Mira

Sequoia

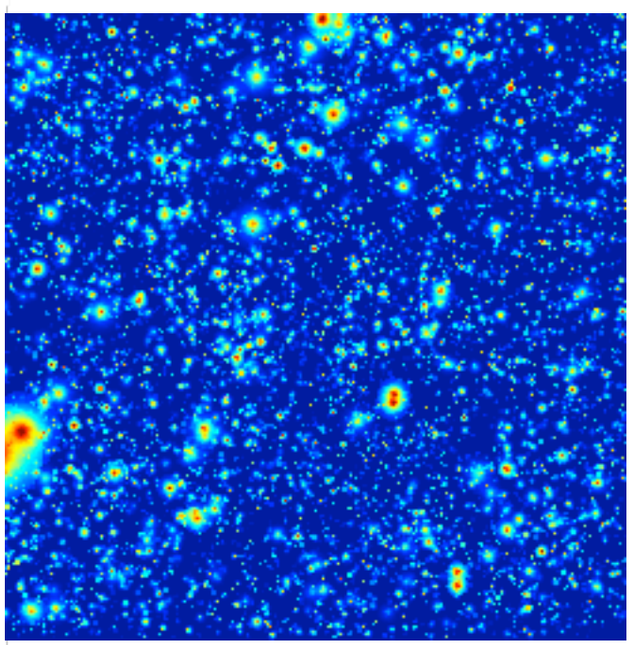
Cores	N_p	L [Mpc]	Time/Substep /Particle [nsec]	Memory [MB/rank]	Total Perf. [PFlops]	Peak [%]
2,048	1600^3	1814	41.2	377	0.018	69.00
4,096	2048^3	2286	19.2	380	0.036	68.59
8,192	2560^3	2880	10.0	395	0.072	68.75
16,384	3200^3	3628	5.19	376	0.144	68.50
32,768	4096^3	4571	2.88	414	0.269	69.02
65,536	5120^3	5714	1.46	418	0.576	68.64
131,072	6656^3	6857	0.74	377	1.16	69.37
262,144	8192^3	9142	0.3	346	2.27	67.70
393,216	9216^3	9857	0.2	342	3.39	67.27
524,288	10240^3	11429	0.16	348	4.53	67.46
786,432	12288^3	13185	0.12	415	7.02	69.75
1,572,864	15360^3	16614	0.06	402	13.94	69.22

HACC Science



Roadrunner view (halos) of the Universe at z=2 from a 64 billion particle run (9 runs on one weekend)

- **Science Projects:** Diverse set of precision cosmology science projects on multiple machines
- **Cosmic Emulation:** Simulation campaign for next-generation cosmological inverse problems, Mira Early Science Project
- **Cosmic Simulations meet Big Data:** Argonne initiative, LSST collaboration

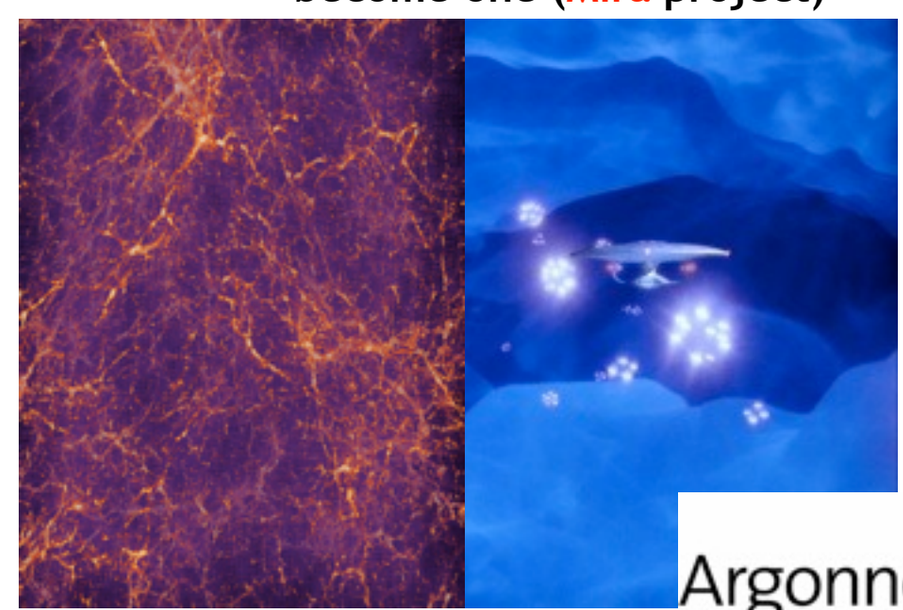


Sunyaev-Zel'dovich sky maps for ACT and SPT

New simulation completed on Hopper at NERSC

Bhattacharya, Das et al. in prep.

The Outer Rim -- where thoughts, time, and space become one (Mira project)



Acknowledgements

- **ANL:** To Susan Coghlan, Paul Messina, Mike Papka, Rick Stevens, and Tim Williams for their support. To the ALCF Ops team, in particular, Paul Rich, Adam Scovel, Tisha Stacey, and William Scullin who kept things going. To Ray Loy and Venkat Vishwanath for essential work on system troubleshooting and parallel I/O
- **IBM:** To Dewey Dasher for helping with system access and to Bob Walkup for running HACC on prototype systems
- **LANL:** To Andy White for for his encouragement and the fateful email! To Jim Ahrens and Pat McCormick for many discussions
- **LLNL:** To Brian Carnes, Kim Cupps, David Fox, and Michel McCoy and the LLNL Sequoia team for support beyond the call of duty