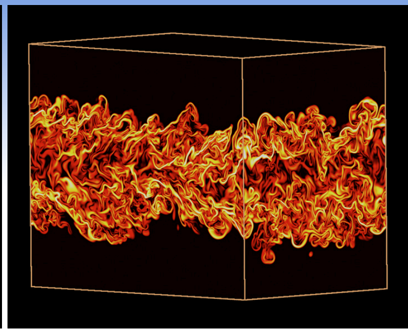
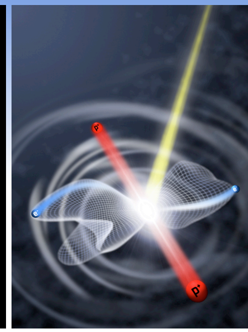
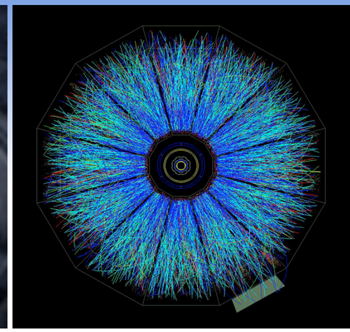
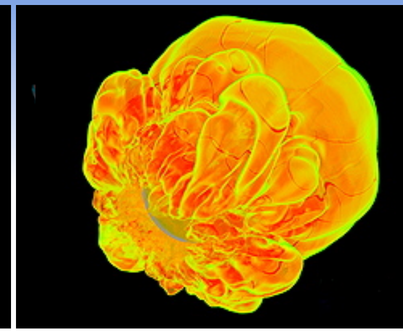






NERSC Workload Analysis and Benchmark Approach

Harvey Wasserman

NERSC Science Driven System Architecture Group

www.nersc.gov/projects/SDSA

August 27, 2008

Office of Science
U.S. DEPARTMENT OF ENERGY

NERSC

BERKELEY LAB

Acknowledgments

- **Contributions to this talk by many people:**

Bill Kramer
NERSC-6 Project Manager

Jim Craw
NERSC-6 Deputy Project Manager

Lynn Rippe
Procurement Lead

Katie Antypas

John Shalf

Tina Butler

Rei Lee

R.K. Owen

Science Driven System Architecture Group

- **Analyze requirements of broad scientific workload**
  - Benchmarking
  - Algorithm tracking
- **Track future trends in supercomputing architecture**
  - Assess emerging system technologies
- **Understand bottlenecks in current computing architecture**
  - Use the NERSC workload to drive changes in computing architecture.

http://www.nersc.gov/projects/SDSA

NERSC Strategy

- Increase user scientific productivity via a timely introduction of the best new technologies designed to benefit the broadest subset of the NERSC workload.*

- Question: what is that workload? How do we characterize it?

*H. Simon, et. al, Science-Driven Computing: NERSC's Plan for 2006–2010

Science Driven Evaluation

- Translate scientific requirements into computational needs and then to a set of hardware and software attributes required to support them.
- Question: how do we represent these needs so that we can communicate them to others?
  - Answer: a set of carefully chosen benchmark programs.

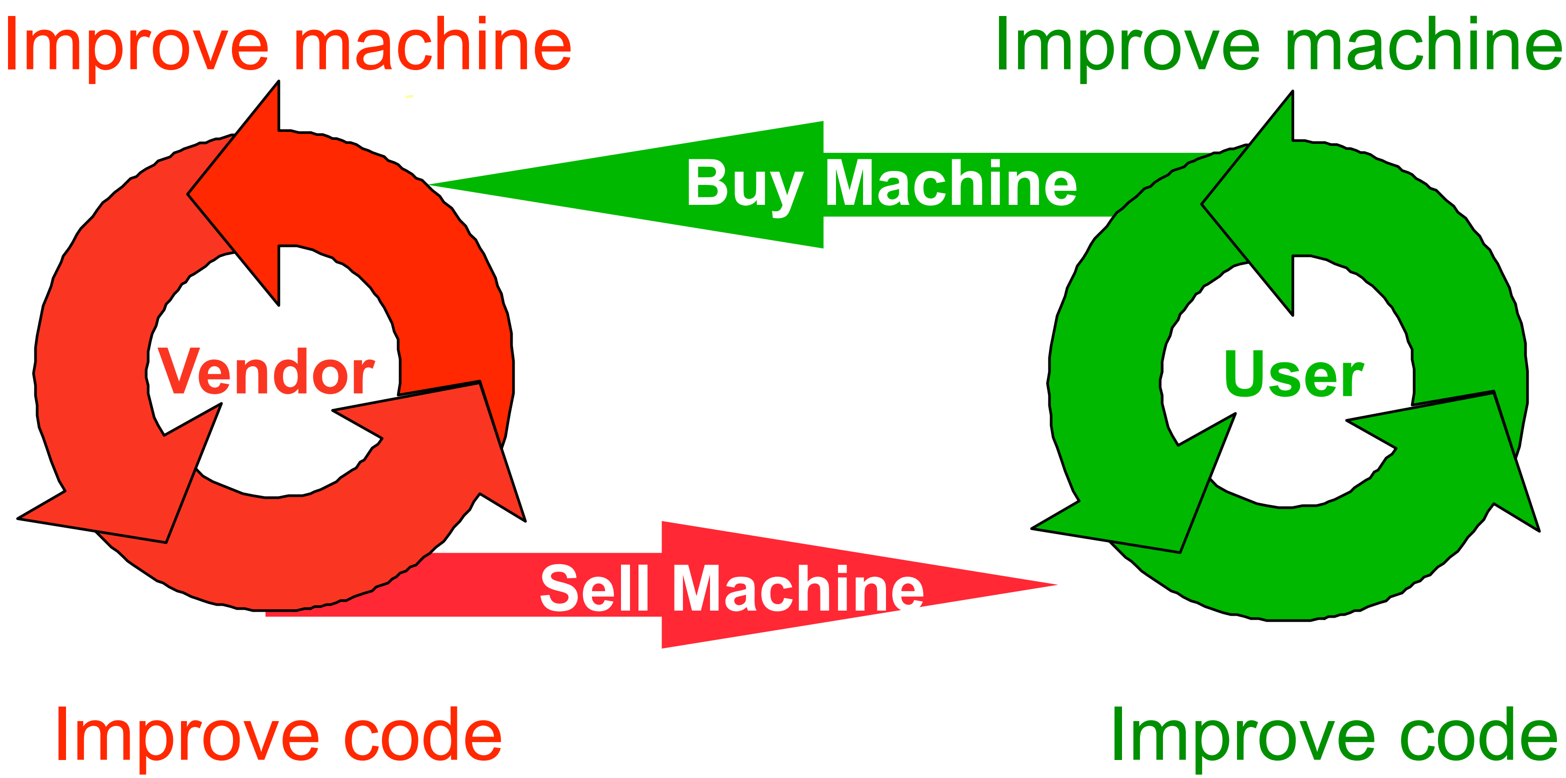
Thoughts

- "For better or for worse, benchmarks shape a field."
  - Prof. David Patterson, CS252 Lecture Notes, University of California, Berkeley, Spring, 1998.
- "Benchmarks are only useful insofar as they model the intended computational workload."
  - Bucher & Martin, LANL, 1982

NERSC Benchmarks Serve 3 Critical Roles

- Carefully chosen to represent characteristics of the expected NERSC-6 workload.
- Give vendors opportunity to provide NERSC with concrete performance and scalability data;
  - Measured or projected.
- Part of the acceptance test and a measure of performance throughout the operational lifetime of NERSC-6.

Why Measure Performance?

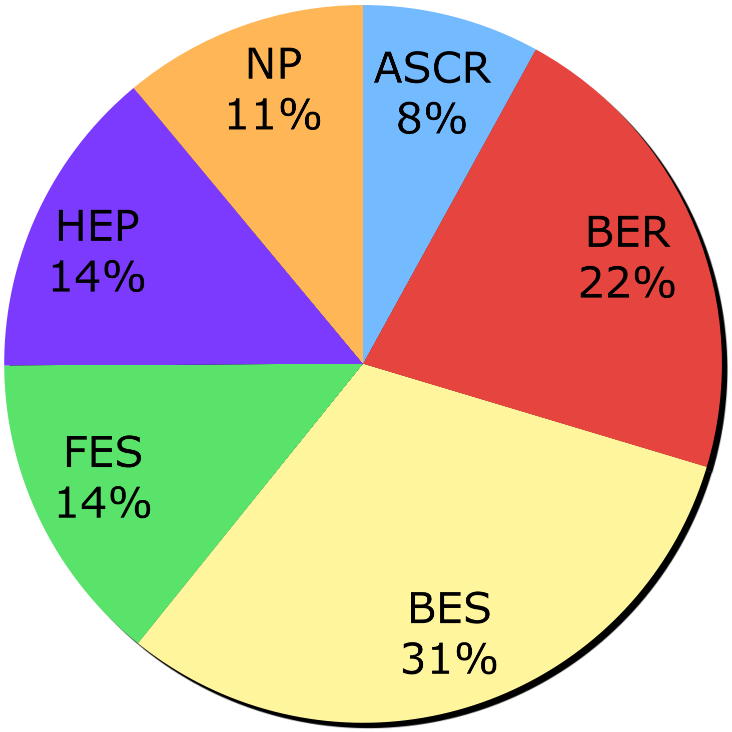


Overarching goal: improve the state of computer architecture

Workload Analysis

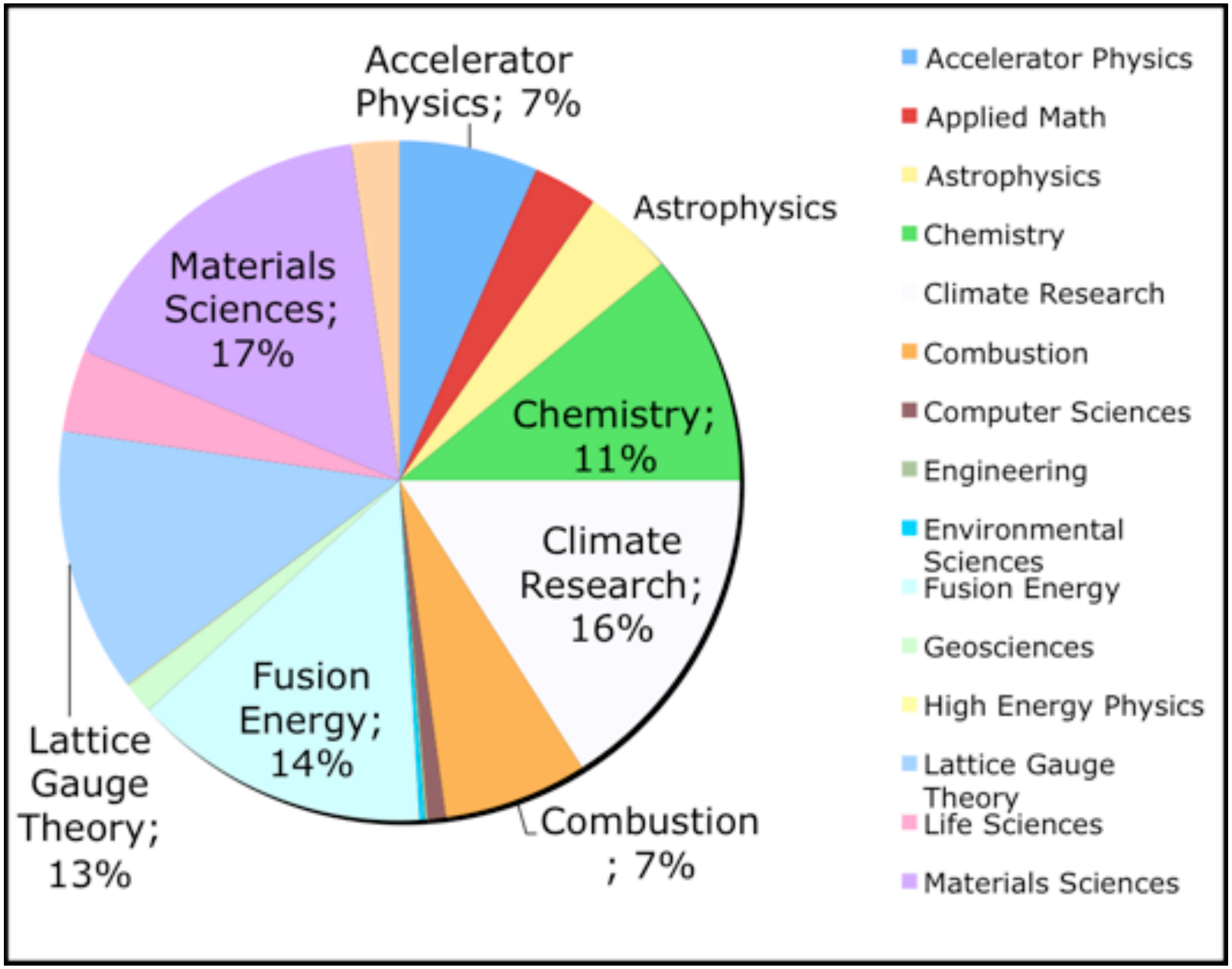
- **Understand D.O.E. Office of Science computational requirements**
  - Augment with anticipated algorithm / science / technology trends
  - Workshop reports, individual discussions, etc.
- **NERSC Workload overview**
  - **~3000 users**
  - **300 - 400 projects representing a broad range of science and algorithms**
  - **~700 codes (>2 codes per project on average)**
  - **15 science areas for 6 D.O.E Office of Science divisions.**

NERSC 2008 Allocations By DOE Office

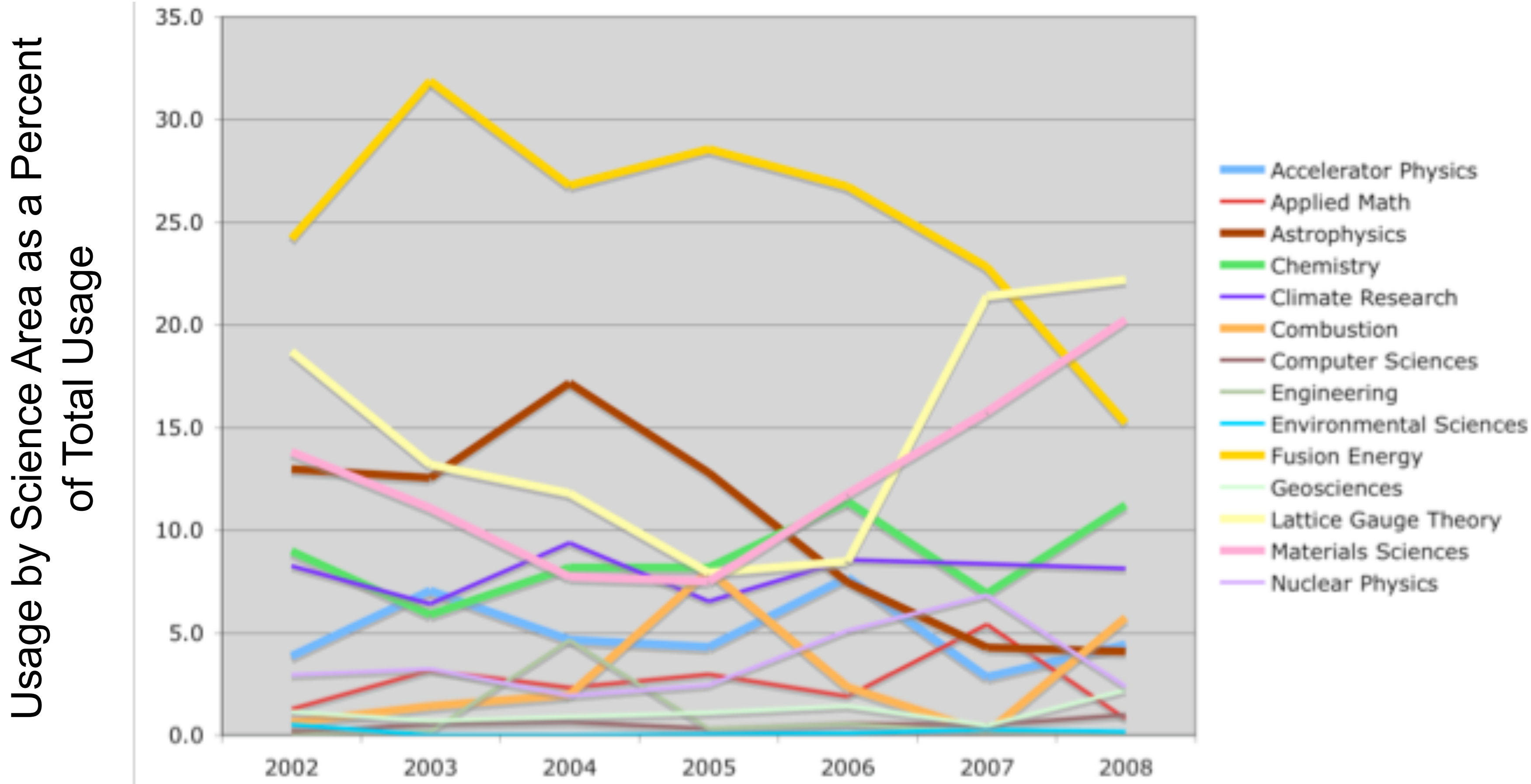


ASCR	Advanced Scientific Computing Research
BER	Biological & Environmental Research
BES	Basic Energy Sciences
FES	Fusion Energy Sciences
HEP	High Energy Physics
NP	Nuclear Physics

NERSC Allocations 2008
By Science Area



Allocation History



NERSC workload is diverse and time varying.

NERSC Workload Analysis

- Now we “drill down” to the code level.
- What follows are some case studies:
  - Science area
  - Science driver
  - Code distribution
  - Best benchmark choice
  - Some characteristics of the benchmarks.

Example: Climate Modeling

• CAM dominates CCSM computational requirements.

• FV-CAM increasingly replacing Spectral-CAM in future CCSM calculations.

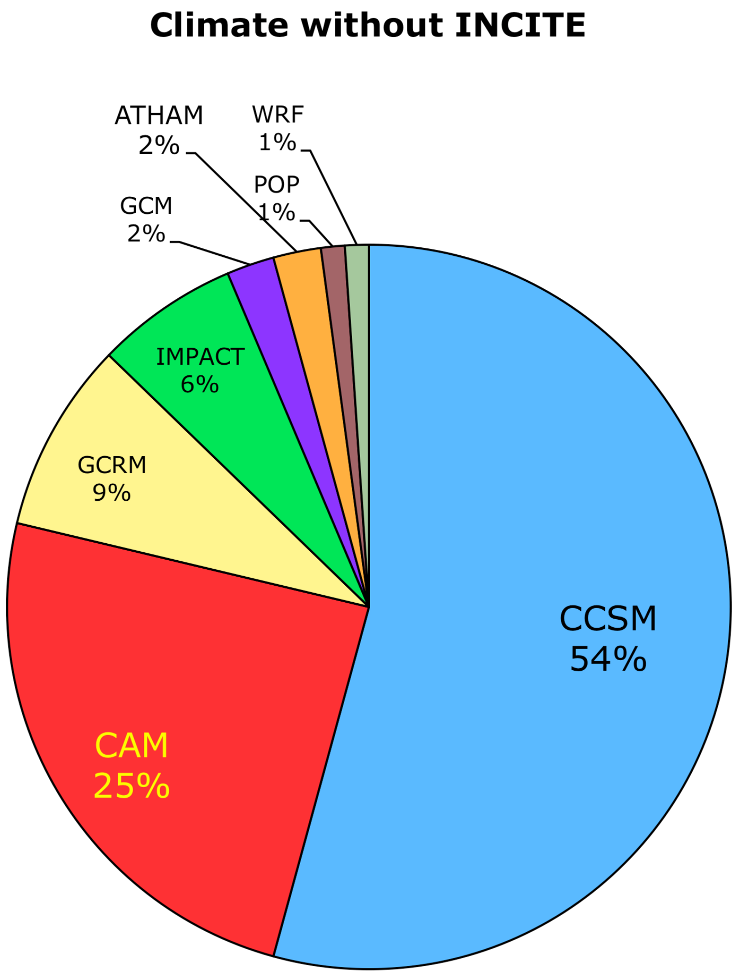
• Drivers:

    • Critical support of U.S. submission to the Intergovernmental Panel on Climate Change (IPCC).

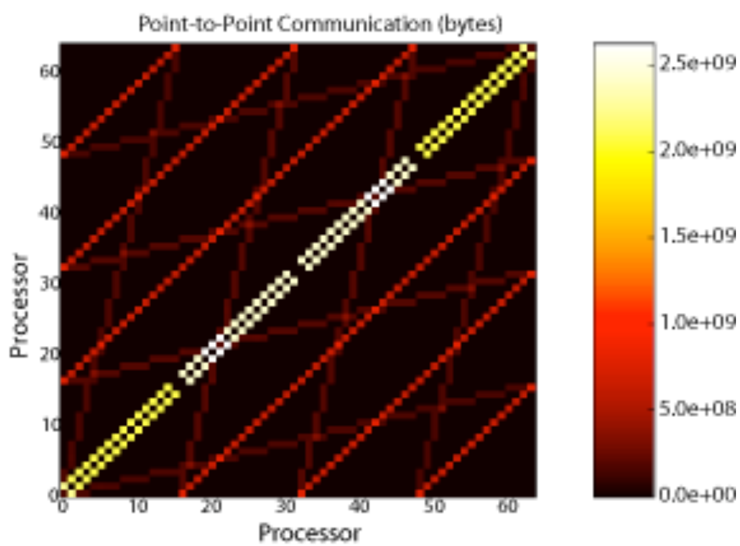
        • Schedule coincident with arrival of NERSC-6 system.

    • V & V for CCSM-4

• Focus on ensemble runs - 10 simulations per ensemble, 5-25 ensembles per scenario, relatively small concurrencies.



CAM Characteristics

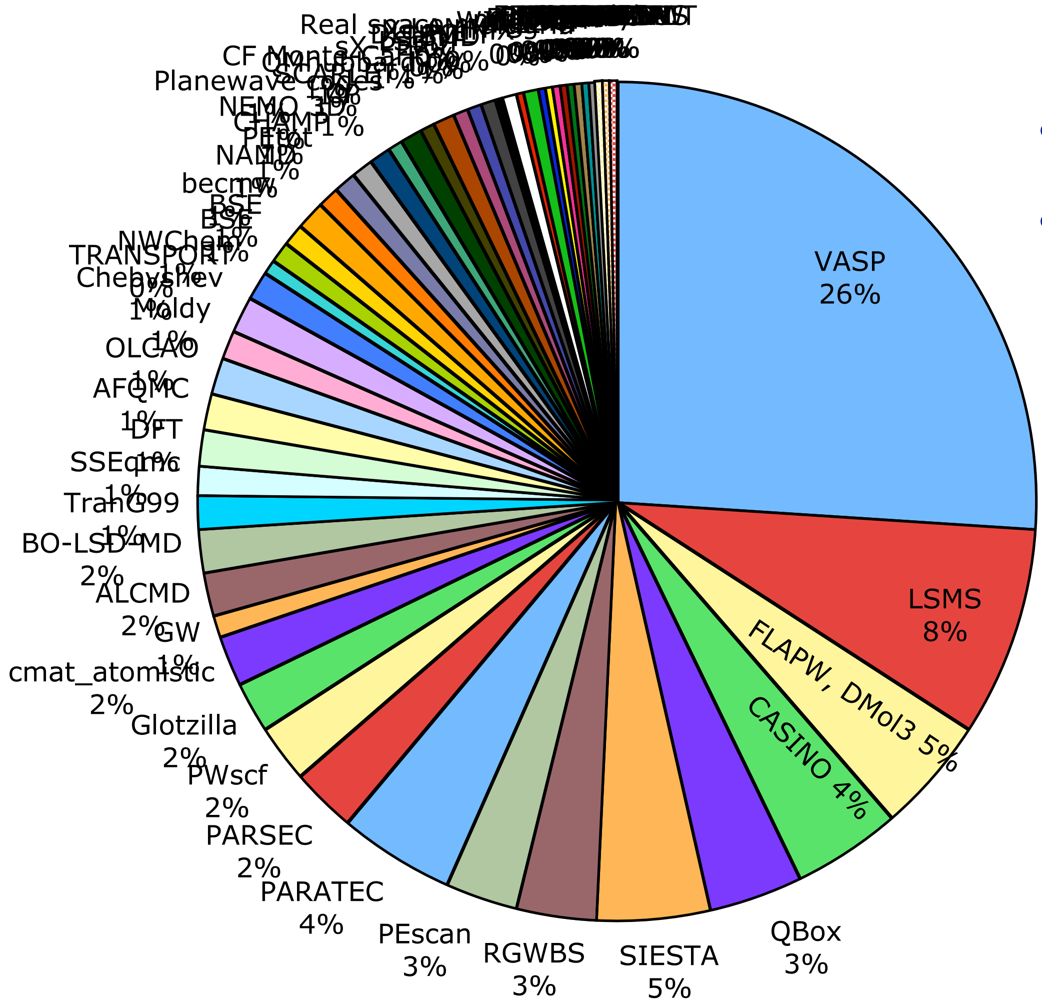


- **Unusual interprocessor communication topology – stresses interconnect.**
- **Relatively low computational intensity – stresses memory subsystem.**
- **MPI messages in bandwidth-limited regime.**
- **Limited parallelism requires faster processors (counter to current microprocessor architectural trends).**

*Computational intensity is the ratio of # of Floating Point Operations to # of memory operations.

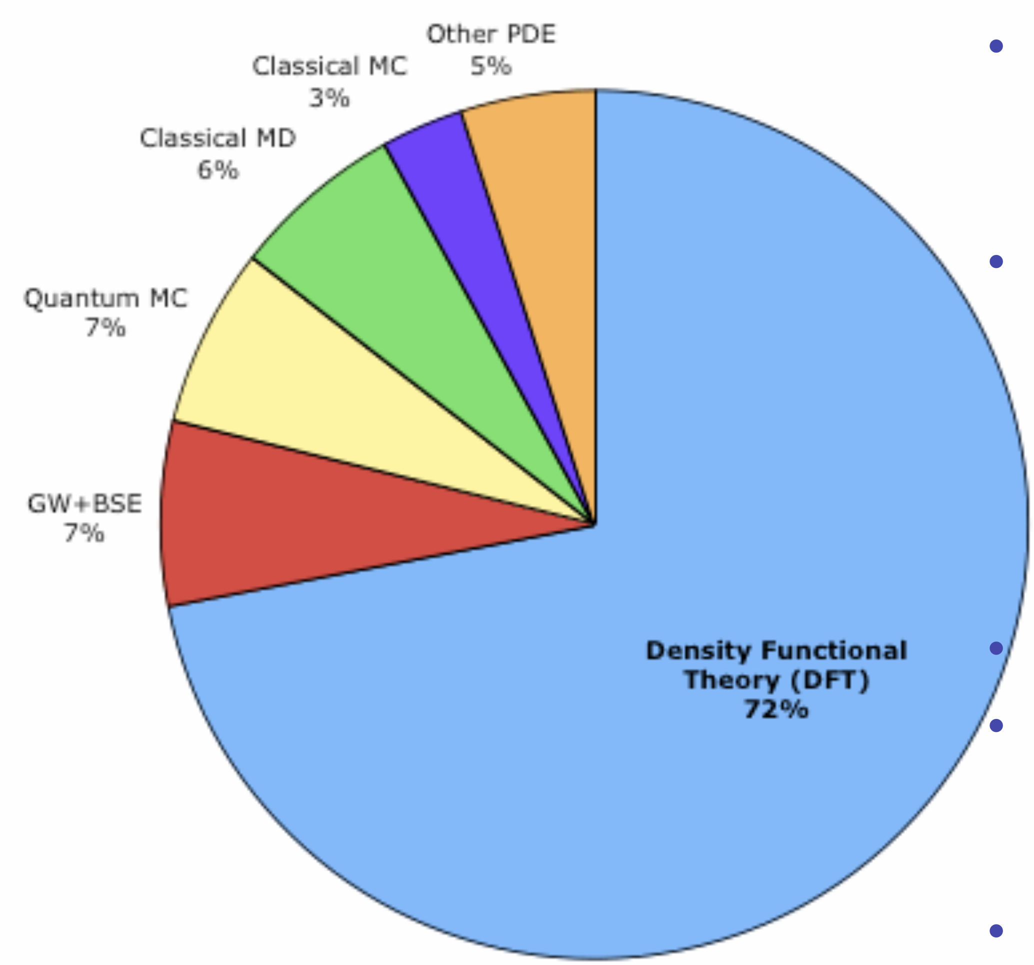
Material Science by Code

- **7,385,000 MPP hours awarded**
- **62 codes, 65 users**
- **Typical code used in 2.15 allocation requests**



	Code	MPP Hours	Percent	Cumulative%
1	VASP	1,992,110	26%	26%
2	LSMS	600,000	8%	34%
3	FLAPW, DMol3	350,000	5%	39%
4	CASINO	312,500	4%	43%
5	QBox	262,500	3%	46%
6	SIESTA	346,500	5%	51%
7	RGWBS	232,500	3%	54%
8	PEscan	220,000	3%	57%
9	PARATEC	337,500	4%	61%
10	PARSEC	182,500	2%	64%
	Other	167,300	34%	66%

Materials Science by Algorithm



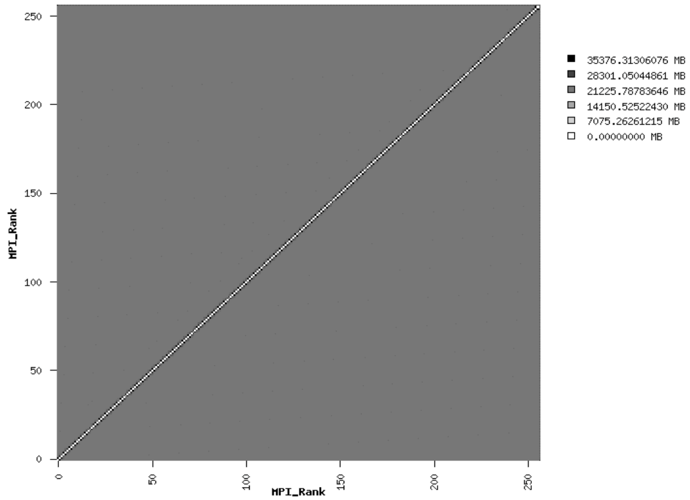
Analysis by Lin-Wang Wang

- Density Functional Theory codes
  - >70% of the MatSci. workload!
  - Majority are planewave DFT.
- Common requirements for DFT:
  - 3D global FFT
  - Dense Linear Algebra for orthogonalization of wave basis functions and calculating pseudopotential
- Dominant Code: VASP
- Science driver: nanoscience, ceramic crystals, novel materials, quantum dots, …
- Similar Codes (planewave DFT)
  - Qbox, PARATEC
  - PETOT/PESCAN

PARATEC: Parallel Total Energy Code

- **Authors**: LBNL + UC Berkeley.
- **Relation to NERSC Workload**
  - Represents / captures the performance of a wide range of codes (VASP, CPMD, PETOT, QBox).
  - 70% of NERSC MatSci computation done via Plane Wave DFT codes.
- **Description**: Planewave DFT; calculation in both Fourier and real space; has custom 3-D FFT to transform between.
- **Coding**: 50,000 lines of Fortran90; uses SCALAPACK / FFTW / BLAS3; vectorizable version;
- **Parallelism**: fine-grain parallelism over DF grid points via MPI.
- **NERSC-6 tests**: strong scaling on 256 and 1024 cores.
- **Profile**: all-to-all data transpositions dominate communications time; Good differentiation between systems.
- **Special**: Also used for NSF Trac-I/II benchmarking.

Paratec Characteristics



	256 cores	1024 cores
Total Message Count	428,318	1,940,665
16 <= MsgSz < 256		114,432
256 <= MsgSz < 4KB	20,337	1,799,211
4KB <= MsgSz < 64KB	403,917	4,611
64KB <= MsgSz < 1MB	1,256	22,412
1 MB <= MsgSz < 16MB	2,808	

- All-to-all communications
- Strong scaling emphasizes small MPI messages.
- Overall rate dominated by FFT speed and BLAS.
- Achieves high per-core efficiency on most systems.
- Good system discrimination.

Other Application Areas

- **Fusion: 76 codes**
  - 5 codes account for >50% of workload: OSIRIS, GEM, NIMROD, M3D, GTC
  - Further subdivide to PIC (OSIRIS, GEM, GTC) and MHD (NIMROD, M3D) code categories
- **Chemistry: 56 codes for 48 allocations**
  - Planewave DFT: VASP, CPMD, DACAPO
  - Quantum Monte Carlo: ZORI
  - Ab-initio Quantum Chemistry: Molpro, Gaussian, GAMESS
  - Planewave DFT dominates (but already covered in MatSci workload)
  - Small allocations Q-Chem category add up to dominant workload component
- **Accelerator Modeling**
  - 50% of workload consumed by 3 codes VORPAL, OSIRIS, QuickPIC
  - Dominated by PIC codes

code	MPP Award	Percent	Cumulative%
OSIRIS	2,112,500	11%	11%
GEM	2,058,333	11%	22%
NIMROD	2,229,167	12%	34%
M3D	1,921,667	10%	45%
GTC	1,783,333	10%	54%

Code	Award	Percent	Cumulative%
ZORI	695,000	12%	12%
MOLPRO	519,024	9%	21%
DACAPO	500,000	9%	29%
GAUSSIAN	408,701	7%	36%
CPMD	396,607	7%	43%
VASP	371,667	6%	49%
GAMESS	364,048	6%	56%

Code	MPP Award	Percent	Cumulative%
VORPAL	1,529,786	33%	33%
OSIRIS	784,286	16%	49%
QuickPIC	610,000	13%	62%
Omega3p	210,536	4%	66%
Track3p	210,536	4%	70%

Benchmark Selection Criteria

- **Coverage**
  - Cover science areas
  - Cover algorithm space
- **Portability**
  - Robust ‘build’ systems
  - Not an architecture specific implementation
- **Scalability**
  - Do not want to emphasize applications that do not justify scalable HPC resources
- **Open Distribution**
  - No proprietary or export-controlled code
- **Availability of Developer for Assistance/Support**

NERSC-6 Application Benchmarks

Benchmark	*Science Area*	*Algorithm Space*	*Base Case Concurrency*	*Problem Description*	*Lang*	*Libraries*
CAM	Climate (BER)	Navier Stokes CFD	56, 240 Strong scaling	D Grid, (~.5° resolution); 240 timesteps	F90	netCDF
GAMESS	Quantum Chem (BES)	Dense linear algebra	384, 1024 (Same as Ti-09)	DFT gradient, MP2 gradient	F77	DDI, BLAS
GTC	Fusion (FES)	PIC, finite difference	512, 2048 Weak scaling	100 particles per cell	F90	
IMPACT-T	Accelerator Physics (HEP)	PIC, FFT	256,1024 Strong scaling	50 particles per cell	F90	
MAESTRO	Astrophysics (HEP)	Low Mach Hydro; block structured -grid multiphysics	512, 2048 Weak scaling	16 32^3 boxes per proc; 10 timesteps	F90	Boxlib
MILC	Lattice Gauge Physics (NP)	Conjugate gradient, sparse matrix; FFT	256, 1024, 8192 Weak scaling	8x8x8x9 Local Grid, ~70,000 iters	C, assemb.	
PARATEC	Material Science (BES)	DFT; FFT, BLAS3	256, 1024 Strong scaling	686 Atoms, 1372 bands, 20 iters	F90	Scalapack, FFTW

NERSC
NATIONAL ENERGY RESEARCH SCIENTIFIC COMPUTING CENTER

Algorithm Diversity

Science areas	Dense linear algebra	Sparse linear algebra	Spectral Methods (FFT)s	Particle Methods	Structured Grids	Unstructured or AMR Grids
Accelerator Science		X	X	X	X	X
Astrophysics	X	X	X	X	X	X
Chemistry	X	X	X	X		
Climate			X		X	X
Combustion					X	X
Fusion	X	X		X	X	X
Lattice Gauge		X	X	X	X	
Material Science	X		X	X	X	

NERSC users require a system which performs adequately in all areas

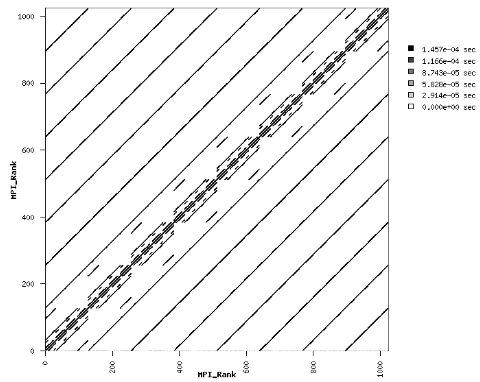
Office of Science
U.S. DEPARTMENT OF ENERGY

BERKELEY LAB

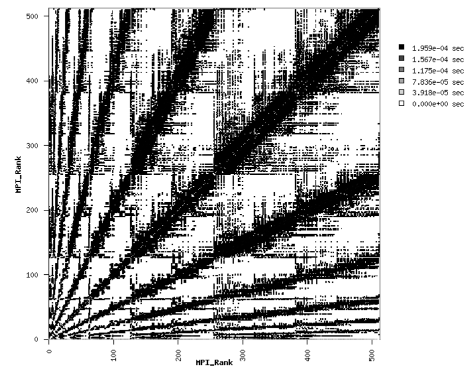
N6 Benchmarks Coverage

Science areas	*Dense linear algebra*	*Sparse linear algebra*	*Spectral Methods (FFT)s*	*Particle Methods*	*Structured Grids*	*Unstructured or AMR Grids*
Accelerator Science		X	X IMPACT-T	X IMPACT-T	X IMPACT-T	X
Astrophysics	X	X MAESTRO	X	X	X MAESTRO	X MAESTRO
Chemistry	X GAMESS	X	X	X		
Climate			X CAM		X CAM	X
Combustion					X MAESTRO	X AMR Elliptic
Fusion	X	X		X GTC	X GTC	X
Lattice Gauge		X MILC	X MILC	X MILC	X MILC	
Material Science	X PARATEC		X PARATEC	X	X PARATEC	

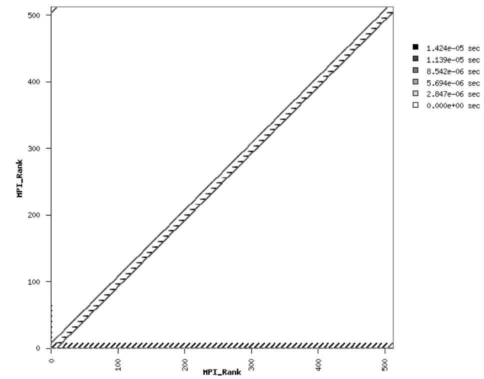
Benchmark Communication Topology



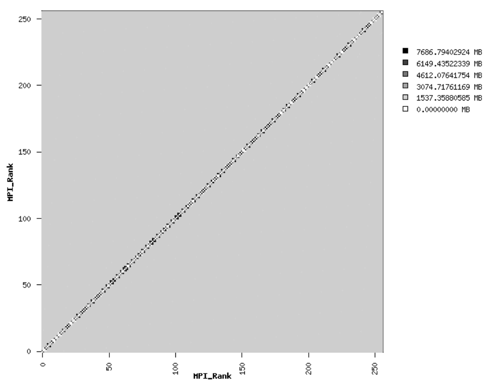
MILC



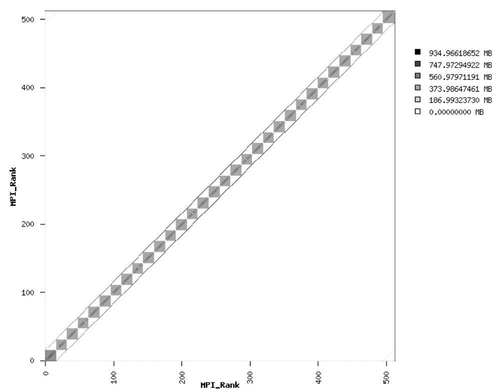
MAESTRO



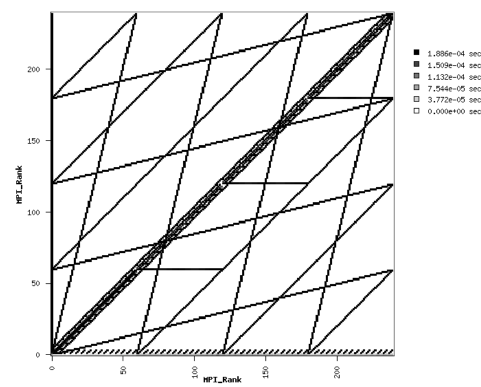
GTC



PARATEC



IMPACT-T



CAM

Summary: CI & %MPI

	CAM	GAMESS	GTC	IMPACT-T	MAESTRO	MILC	PARATEC
CI*	0.67	0.61	1.15	0.77	0.24	1.39	1.50
Cray XT4 %Peak per Core (largest case)	13%	12%	24%	14%	5%	14%	44%
Cray XT4 %MPI Medium	29%		4%	9%	20%	20%	27%
Cray XT4 %MPI Large	35%		6%	40%	20%	23%	64%
Cray XT4 %MPI ExtraL	n/a	n/a	n/a	n/a	n/a	30%	n/a
Cray XT4 Avg Msg Size Med	113K	n/a	1 MB	35KB	2K	16KB	34KB

*CI is the computational intensity, the ratio of # of Floating Point Operations to # of memory operations.

Traditional Sources of Performance Improvement are Flat-Lining

- **New Constraints**
  - 15 years of *exponential* clock rate growth has ended
- **But Moore's Law continues!**
  - How do we use all of those transistors to keep performance increasing at historical rates?
  - Industry Response: #cores per chip doubles every 18 months *instead* of clock frequency!

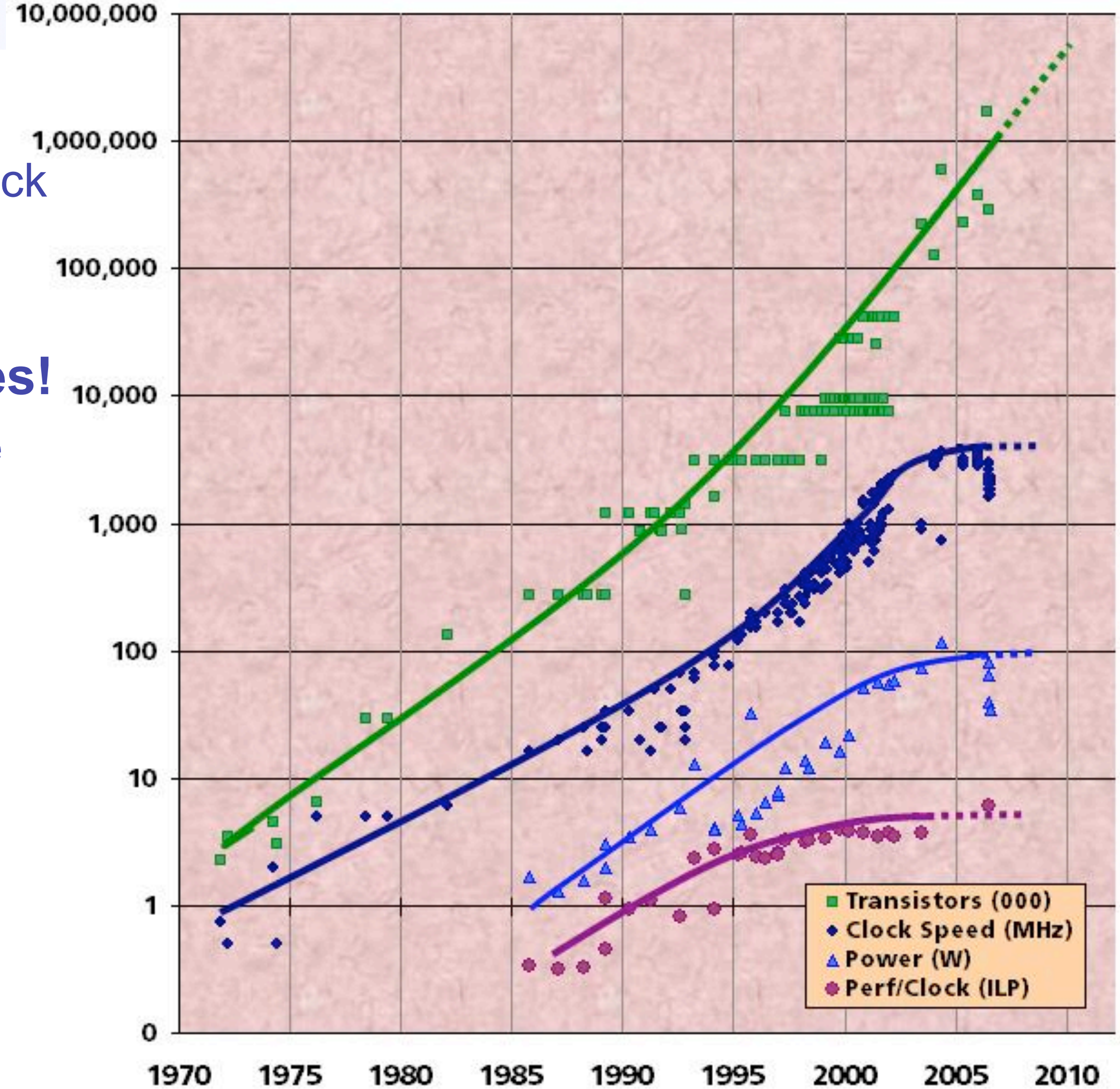


Figure courtesy of Kunle Olukotun, Lance Hammond, Herb Sutter, and Burton Smith

Response to Technology Trends

- **Parallel computing has thrived on weak-scaling for past 15 years**
- **Flat CPU performance increases emphasis on strong-scaling**
- **Benchmarks changed accordingly**
  - **Concurrency:** *Increased 4x over NERSC-5 benchmarks*
  - **Strong Scaling:** *Input decks emphasize strong-scaled problems*
  - **Implicit Methods***: Added MAESTRO application benchmark*
  - **Multiscale**: *Added AMR Poisson benchmark*
  - **Lightweight Messaging:** *Added UPC FT benchmark*

Summary So Far

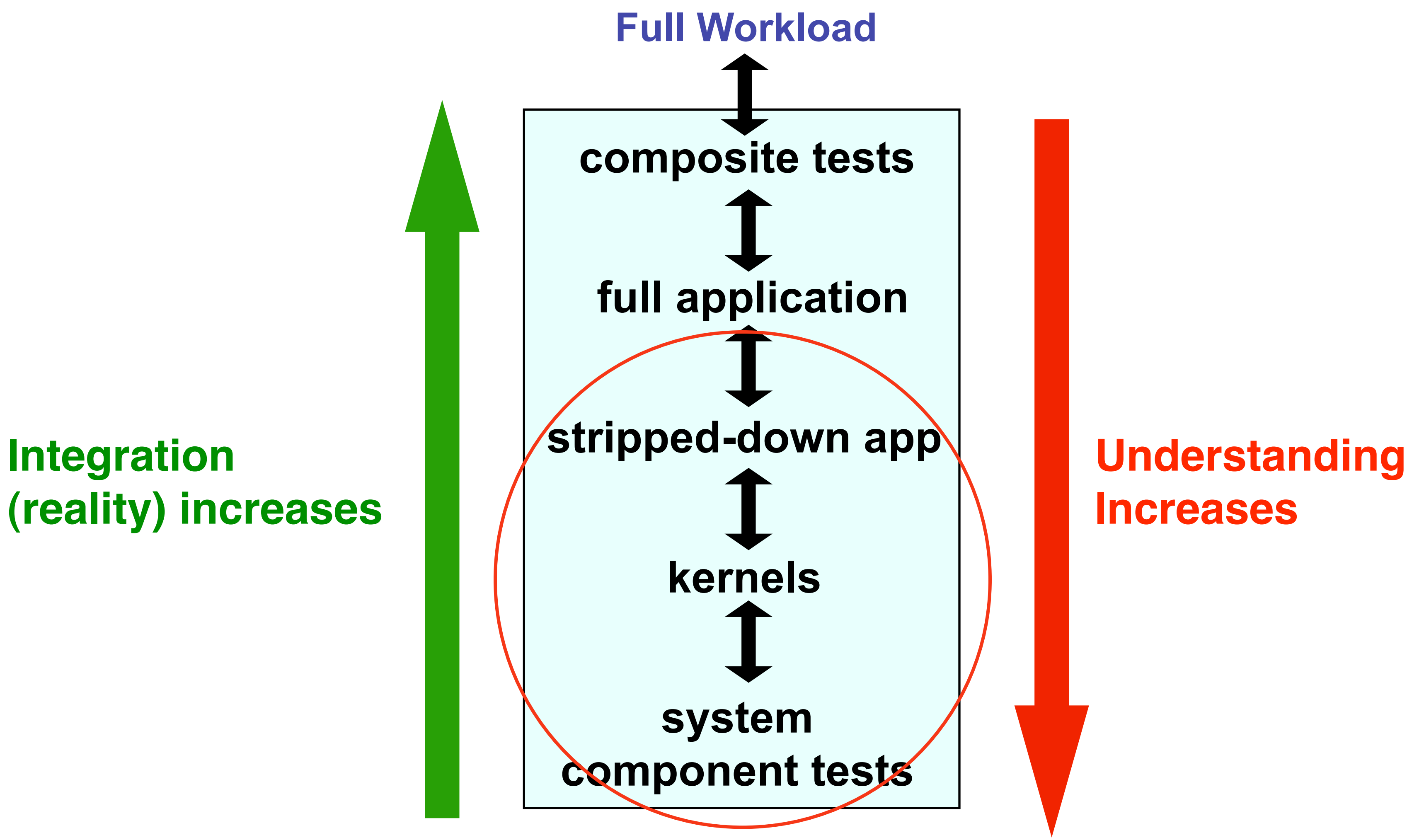
- Codes represent important science and/or algorithms <u>and</u> architectural stress points such as CI*, message type/size/topology.
- Codes provide a good means of system differentiation during acquisition and validation during acceptance.
- Strong suite of scalable benchmarks (256-8192+ cores).

*CI = Computational Intensity, # FLOPs / Memory references

Other NERSC Benchmark Tests

Validation Efforts

Use a Hierarchy of Tests



Lower-Level Benchmarks

CODE	PURPOSE / DESCRIPTION
STREAM	Single- and multi-core memory bandwidth.
FCT	Full-Configuration Test, run a single app over all cores; FFT mimics planewave DFT codes.
PSNAP	FWQ operating system noise test.
NAS PB serial & 256-way MPI	Serial application performance on a single **packed** node; measures memory BW/ computation rate balance and compiler capabilities. **Packed** means all cores run.
NAS PB UPC	Measure performance characteristics not visible from MPI for FT benchmark.
Multipong	NERSC MPI PingPong for “latency” and BW, nearest- and furthest nodes in topology; also intra-node.
AMR Elliptic	C++/F90 LBNL Chombo code; proxy for AMR Multigrid elliptic solvers; 2 refinement levels; weak scaling with geometry replication; very sensitive to OS noise;

Validation & Benchmark Efforts

- **XT4: DC & QC / CNL**
- **IBM p575 / AIX**
- **BG/P / LWK**
- **Sun QC Opteron + InfiniBand / Linux**
- **IBM Power6 / AIX**
- **SiCortex MIPS / Linux**
- **SGI DC Itanium / NumaLink**
- **Compilers PGI / Intel / PathScale / XLF**
- **Profiles from CrayPat and NERSC's IPM**

Composite Performance Metrics

Benchmark Hierarchy

Full Workload	
composite tests	SSP, ESP
full application	CAM, GTC, MILC, GAMESS, PARATEC, IMPACT-T, MAESTRO
stripped-down app	AMR Elliptic Solve
kernels	NPB Serial, NPB Class D, UPC NPB, FCT
system component tests	Stream, PSNAP, Multipong, IOR, MetaBench, NetPerf

34

Sustained System Performance (SSP)

- Aggregate, un-weighted measure of <u>sustained</u> computational capability relevant to NERSC's workload.

- Geometric Mean of the processing rates of seven applications multiplied by *N*, # of cores in the system.
  - Largest test cases used.

- Uses floating-point operation count <u>pre</u>determined on a reference system by NERSC.

$$\text{SSP in TFLOPS} = \frac{N * \sqrt[7]{\prod_i P_i}}{1000}$$

NERSC-6 Composite SSP Metric

The largest concurrency run of each full application benchmark is used to calculate the composite SSP metric

NERSC-6 SSP

CAM	GAMESS	GTC	IMPACT-T	MAESTRO	MILC	PARATEC
240p	1024p	2048p	1024p	2048p	8192p	1024p

For each benchmark measure

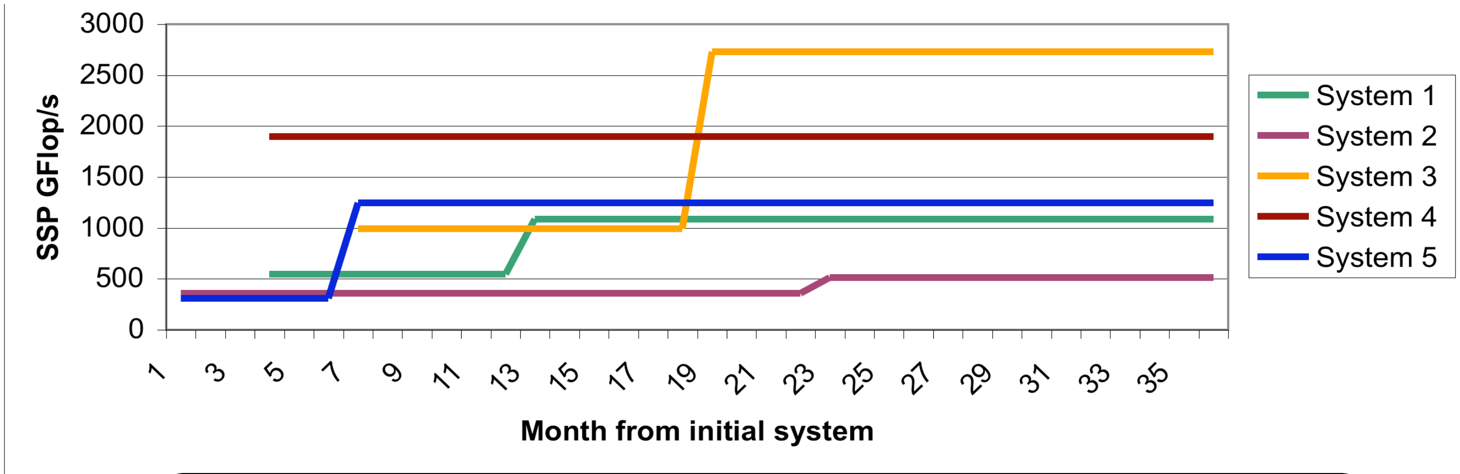
- *FLOP counts on a reference system*
- *Wall clock run time on various systems*

Key Point - Sustained System Performance (SSP) Over Time

- Integrate the SSP over a particular time period.
- SSP can change due to
  - System upgrades, Increasing # of cores, Software Improvements
- Allows evaluation of systems delivered in phases.
- Takes into account delivery date.
- Produces metrics such as SSP/Watt and SSP/$

$$Value_s = \frac{Potency_s}{Cost_s}$$

SSP Over 3 Year Period for 5 Hypothetical Systems



Area under SSP curve, when combined with cost, indicates system 'value'

I, II, III, IV, V

Example of N6 SSP on Hypothetical System

Hypothetical N6 System			Results	
	Tasks	System Gflopcnt	Time	Rate per Core
CAM	**240**	**57,669**	**408**	0.589
GAMESS	**1024**	**1,655,871**	**2811**	0.575
GTC	**2048**	**3,639,479**	**1493**	1.190
IMPACT-T	**1024**	**416,200**	**652**	0.623
MAESTRO	**2048**	**1,122,394**	**2570**	0.213
MILC	**8192**	**7,337,756**	**1269**	0.706
PARATEC	**1024**	**1,206,376**	**540**	2.182
GEOMETRIC MEAN				0.7

Rate Per Core = Ref. Gflop count / (Tasks*Time)

Flop count measured on reference system

Measured wall clock time on hypothetical system

Geometric mean of 'Rates per Core'

SSP (TF) = Geo mean of rates per core * # cores in system/1000

N6 SSP of 100,000 core system = 0.7 * 100,000 /1000 = 70

N6 SSP of 200,000 core system = 0.7 * 200,000 /1000 = 140

Allows vendors to size systems based on benchmark performance

Benchmarking Methodology

Performance Obligations

- A selected vendor at NERSC is required to meet benchmark performance levels reported in the RFP response as a condition of acceptance…
  - and throughout the life of the subcontract.

- Includes all applications (with all inputs), all lower-level tests, SSP, and other tests, with strict constraints on variability across runs…
  - both in dedicated mode and production mode.

Base Case for Application Runs

- Primary basis for comparison among proposed systems.
- Limits the scope of optimization.
  - Modifications only to enable porting and correct execution.
- Limits allowable concurrency to prescribed values.
- MPI only for all codes (even if OpenMP directives present).
- Fully packed nodes.
- Libraries okay (if generally supported).
- Hardware multithreading okay, too.
  - Expand MPI concurrency to occupy hardware threads.

Optimized Case for Application Runs

- Allow the Offeror to highlight features of the proposed system.
- Applies to seven SSP apps only, all test problems.
- Examples:
  - Unpack the nodes;
  - Higher or lower concurrency than corresponding base case;
  - Hybrid OpenMP / MPI;
  - Source code changes for data alignment / layout;
  - Any / all of above.
- Caveat: number of tasks used to calculate SSP must use the total number of processors blocked from other use.

Summary

- Workload-based evaluation.
- Appropriate aggregate metrics.
- Formal methodology for tests, with stringent requirements based on proposed system.
- Wide range of tests from all levels of the benchmark hierarchy.

Scientists Need More Than Flop/s

- Performance — How fast will a system process work if everything is working well
- Effectiveness — What is the likelihood that users can get the system to do their work
- Reliability — The system is available to do work and operates correctly all the time
- Consistency — How often will the system process users' work as fast as it can
- Usability — How easy is it for users to get the system to go as fast as possible