DARPA HPCS Overview
Productivity Evaluation

David Koester, Ph.D.
DARPA HPCS Productivity Team

HPCchallenge Benchmarks Panel
SC2004
12 November 2004

- This work is sponsored by the Department of Defense under Army Contract W15P7T-05-C-D001. Opinions, interpretations, conclusions, and recommendations are those of the author and are not necessarily endorsed by the United States

Outline

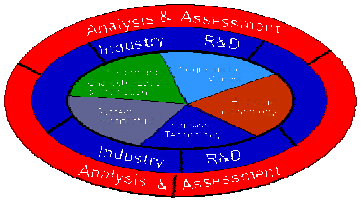
- Brief DARPA HPCS Overview
  - Impacts
  - Programmatics
  - HPCS Phase II Teams
  - Program Goals
  - HPCS Productivity Team Benchmarking Working Group
- Productivity Evaluation
  - Development Time Productivity Indicators
  - Publications on HPC Productivity
- Summary

High Productivity Computing Systems

➢ **Create a new generation of economically viable computing systems (2010) and a procurement methodology (2007-2010) for the security/industrial community**

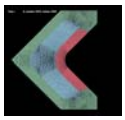
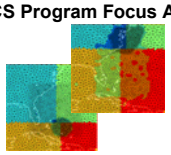
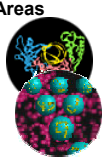
Impact:

- **Performance** (time-to-solution): speedup critical national security applications by a factor of 10X to 40X
- **Programmability** (idea-to-first-solution): reduce cost and time of developing application solutions
- **Portability** (transparency): insulate research and operational application software from system
- **Robustness** (reliability): apply all known techniques to **protect against outside attacks**, hardware faults, & programming errors



HPCS Program Focus Areas



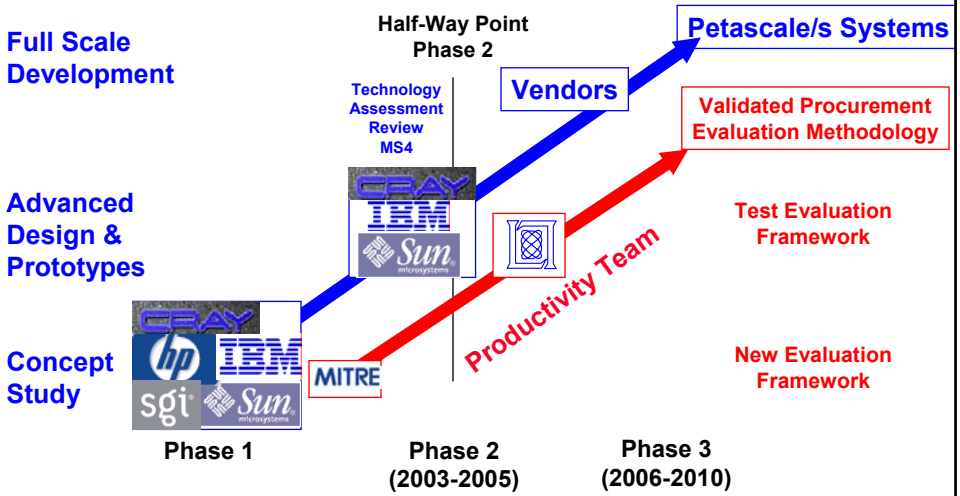
Applications:

- Intelligence/surveillance, reconnaissance, cryptanalysis, weapons analysis, airborne contaminant modeling and biotechnology

Fill the Critical Technology and Capability Gap
Today (late 80's HPC technology).....*to*.....Future (Quantum/Bio Computing)

High Productivity Computing Systems
-Program Overview-

➢ **Create a new generation of economically viable computing systems (2010) and a procurement methodology (2007-2010) for the security/industrial community**



HPCS Phase II Teams

Industry

IBM — PI: Elnozahy

Sun microsystems — PI: Mitchell

CRAY — PI: Smith

Mission Partners

Office of Science, U.S. Department of Energy; NASA; NSF; HPC Modernization Program; NNSA, National Nuclear Security Administration

Productivity Team (Lincoln Lead)

MIT Lincoln Laboratory — PI: Kepner

ISI — PI: Lucas

University of Maryland — PI: Basili

UCSD — PI: Benson & Snavely

ICL UT — PI: Dongarra

MITRE — PI: Koester

Oak Ridge, Los Alamos, Argonne, Lawrence Berkeley Lab — PIs: Vetter, Lusk, Post, Bailey

UCSB, CSAIL, Ohio State, CodeSourcery — PIs: Gilbert, Edelman, Ahalt, Mitchell

HPCS Phase II Teams

Industry

IBM — PI: Eln

Sun microsystems

CRAY

Mission P

Office

Productiv

MIT Linc
Laborat
PI: Kepner

PI: Lucas

PI: Basili

PI: Benson & Snavely

ICL UT — PI: Dongarra

MITRE — PI: Koester

Oak Ridge, Los Alamos, Argonne, Lawrence Berkeley Lab — PIs: Vetter, Lusk, Post, Bailey

UCSB, CSAIL, Ohio State, CodeSourcery — PIs: Gilbert, Edelman, Ahalt, Mitchell

Productivity Team Working Groups

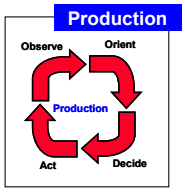
- Development Time Experiments
- Execution Time Modeling
- Benchmarks
- Programming Models and Definitions
- Test and Spec Environment
- Workflows, Models and Metrics
- Existing Codes Analysis

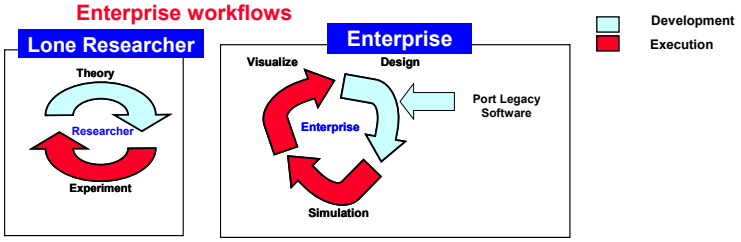
Slide-6
SC2004
HPCC Panel

MITRE | MIT Lincoln Laboratory | ISI

HPCS Program Goals
Productivity Goals

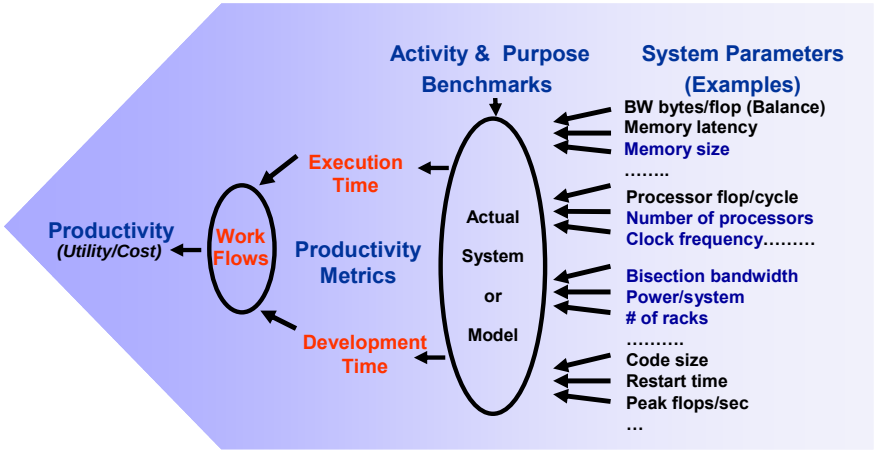
- HPCS overall productivity goals:
  - Execution (sustained performance)
    - 1 Petaflop/s (scalable to greater than 4 Petaflop/s)
    - Reference: Production workflow
  - Development
    - 10X over today's systems
    - Reference: Lone researcher and Enterprise workflows





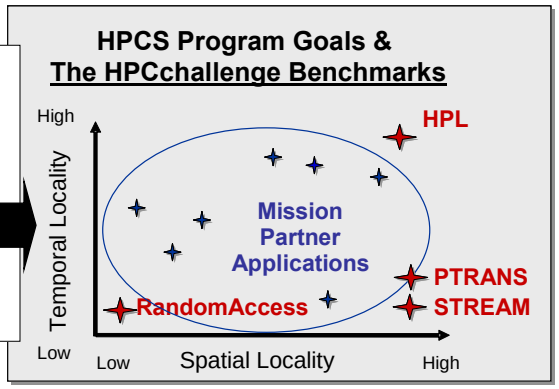
10x improvement in time to first solution!

HPCS Program Goals
Productivity Framework

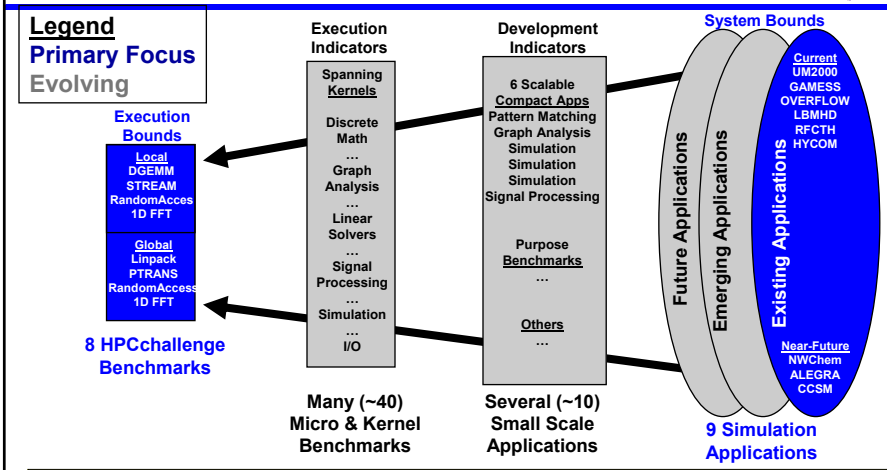


HPCS Program Goals Hardware Challenges

- General purpose architecture capable of:
  - Subsystem Performance Indicators
    1) 2+ PF/s LINPACK
    2) 6.5 PB/sec data STREAM bandwidth
    3) 3.2 PB/sec Bisection bandwidth
    4) 64,000 GUPS

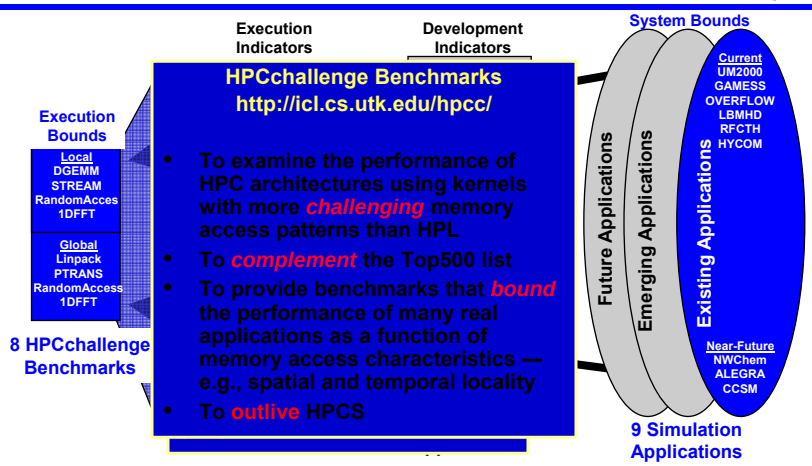


HPCS Benchmark Spectrum



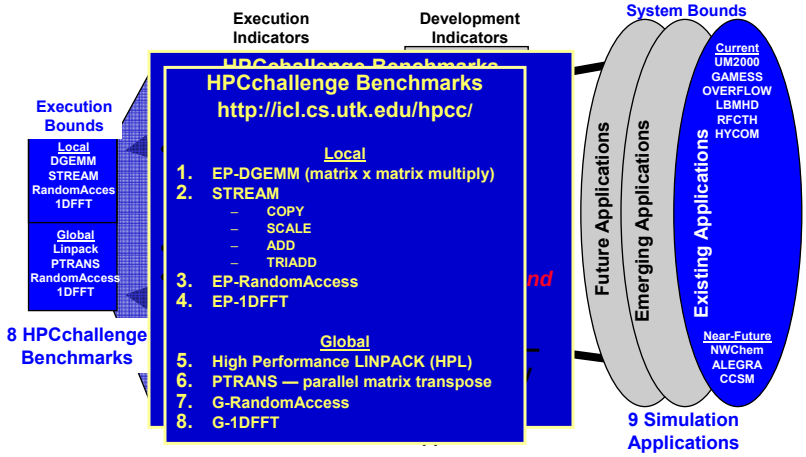
- Spectrum of benchmarks provide different views of system
  - HPCchallenge pushes spatial and temporal boundaries; sets performance bounds
  - Applications drive system issues; set legacy code performance bounds
- Kernels and Compact Apps for deeper analysis of execution and development time

HPCS Benchmark Spectrum
HPCchallenge Benchmarks



- HPCchallenge pushes spatial and temporal boundaries; sets performance bounds
- Available for download http://icl.cs.utk.edu/hpcc/

HPCS Benchmark Spectrum
HPCchallenge Benchmarks



- HPCchallenge pushes spatial and temporal boundaries; sets performance bounds
- Available for download http://icl.cs.utk.edu/hpcc/

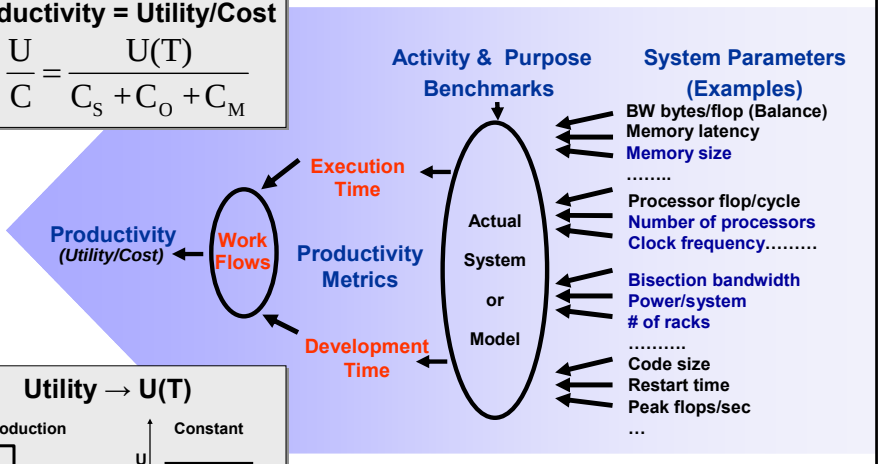
Outline

- Brief DARPA HPCS Overview
  - Impacts
  - Programmatics
  - HPCS Phase II Teams
  - Program Goals
  - HPCS Productivity Team Benchmarking Working Group
- **Productivity Evaluation**
  - **Development Time Productivity Indicators**
  - **Publications on HPC Productivity**
- **Summary**

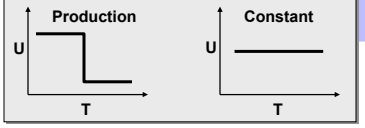
HPCS Program Goals Productivity Framework

Productivity = Utility/Cost

$$\Psi \equiv \frac{U}{C} = \frac{U(T)}{C_S + C_O + C_M}$$



Utility → U(T)

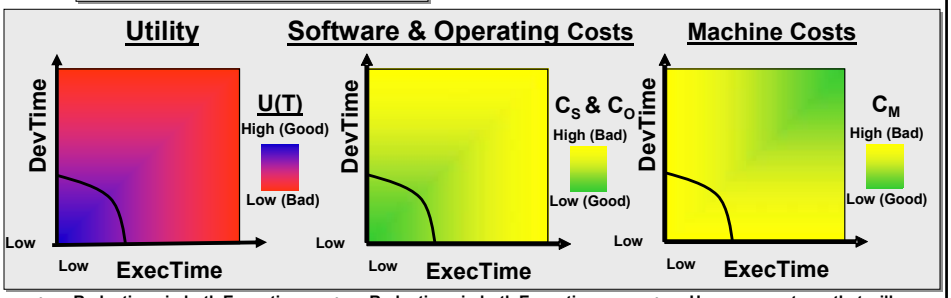


Productivity Factors
Execution Time & Development Time

Productivity = Utility/Cost

$$\Psi \equiv \frac{U}{C} = \frac{U(T)}{C_S + C_O + C_M}$$

- Utility and some Costs are relative to
  - Workflow (WkFlow)
  - Execution Time (ExecTime)
  - Development Time (DevTime)



- Reductions in both Execution Time and Development Time contribute to positive increases in Utility
  - Utility generally is inversely related to time
  - Quicker is better
- Reductions in both Execution Time and Development Time contribute to positive decreases in Software and Operating costs
  - Reduction in programmer costs
  - More work performed over a period
- However, systems that will provide increased utility and decreased operating costs may have a higher initial procurement cost
  - Need productivity metrics to justify the higher initial cost

Development Time
Productivity Indicators

- Several key indicators which can be applied directly or indirectly to HPCchallenge, CompactApps, Full App, and Classroom Experiments
- Actual User *Performance Achieved*
  - Direct: timing of user code
  - Indirect: paper analysis of code/features => connection to workflows
- *Effort* required
  - Direct: measure time to implement/modify code
  - Indirect: software lines of code (SLOC)
- *Expertise* level required
  - Direct: fraction of users who can achieve a certain level of performance
  - Indirect: paper analysis of code/features => connection to workflows, number experts of needed

- Many additional factors are important
- Performance, Effort and Expertise were mentioned the most

Strawman Development Time Productivity Formula

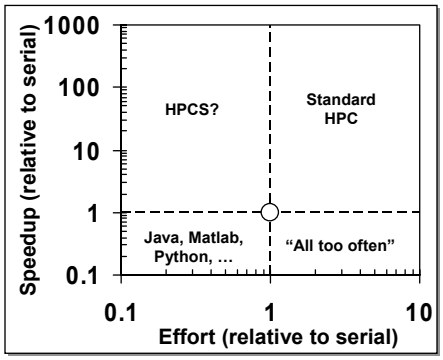
$$\text{Dev Time Productivity} = \frac{\text{Relative Speedup}}{\text{Relative Effort}}$$

$$\text{Speedup} = \frac{\text{Parallel Performance}}{\text{Serial Performance}}$$

$$\text{Relative Effort} = \frac{\text{Parallel SLOC}}{\text{Serial SLOC}}$$

- Dev Time Productivity = Utility/Effort
  - Units: speedup per relative effort
- Utility = median user speedup
  - Compared to serial on workstation
- Effort = relative time to implement
  - Compared to serial on workstation

- Simplest way to combine currently measurable quantities
- Too simplistic?

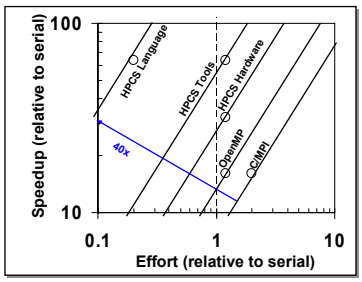


Hypothetical Formula Usage

- Consider Application implemented using various approaches

	Speedup			
Approach	Median	Expert	Effort	Productivity
C/MPI on a 128 CPU cluster	16	100	2	8
OpenMP on Shared Memory	16	100	1.2	13.3
HPCS hardware	32	200	1.2	26.3
HPCS performance tools	64	200	1.2	53.3
High Level Language	64	200	0.2	320

- Max HPCS development productivity benefit 320/8 = 40x



Special Issue on "HPC Productivity"

- *International Journal of High Performance Computing Applications*, Volume 18, Number 4, Winter 2004 (November)

1. "HPC Productivity: An Overarching View" Jeremy Kepner
2. "Software Project Management and Quality Engineering Practices for Complex, Coupled Multi-Physics, Massively Parallel Computational Simulations: Lessons Learned from ASCI" Doug Post and Richard Kendall
3. "A Framework for Measuring Supercomputer Productivity" Marc Snir and David A. Bader
4. "Productivity Metrics and Models for High Performance Computing" Thomas Sterling
5. "A Strategy for Measuring the Productivity of Programming Interfaces" Ken Kennedy, Charles Koelbel and Rob Schreiber
6. "Performance Metrics Based on Computation Action" Robert W. Numrich
7. "Measuring HPC Productivity" Stuart Faulk, Philip Johnson, Adam Porter, Walter Tichy, and Lawrence Votta
8. "Purpose-Based Benchmarks" John L. Gustafson
9. "Productivity in HPC" David J. Kuck
10. "HPC Productivity Model Synthesis" Jeremy Kepner

- Inventing a new field

Outline

- Brief DARPA HPCS Overview
  - Impacts
  - Programmatics
  - HPCS Phase II Teams
  - Program Goals
  - HPCS Productivity Team Benchmarking Working Group
- Productivity Evaluation
  - Development Time Productivity Indicators
  - Publications on HPC Productivity
- Summary

Summary

- **Create a new generation of economically viable computing systems (2010)**

– Impacts	– Hardware Challenges
▪ Performance	▪ 2+ PF/s LINPACK
▪ Programmability	▪ 6.5 PB/sec STREAM bandwidth
▪ Portability	▪ 3.2 PB/sec Bisection bandwidth
▪ Robustness	▪ 64,000 GUPS

- **Create a new procurement methodology based on Productivity (2007-2010)**