

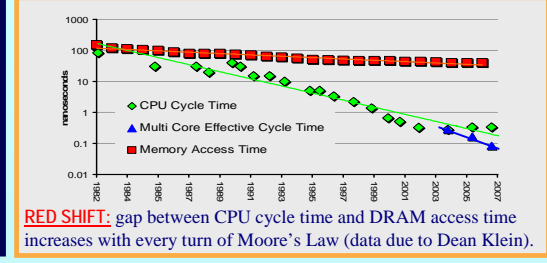
# Performance Modeling: Impact of Architecture Trends on Applications



A. Snavely, L. Carrington, B. de Supinski, J. Vetter

www.peri-scidac.org

Bringing scientific rigor to the prediction and understanding of performance on HPC systems.



## Four important architecture trends to mitigate Red Shift:

### COMPLEXITY OF PROCESSORS

**#1 MULTICORE:** CPU core counts are increasing.  
**IMPACT:** Adding cores may reduce bandwidth and effective size of shared cache while adding sockets may reduce bandwidth to main memory.  
**TOOL:** MAPS (Memory Access Pattern Signature)  
 ○ Measures rates at which a machine can load and store to different levels of their memory hierarchy.  
 ○ Captures rates dependent on memory access patterns and working-set size.  
 (sample MAPS curve below)

**#2 MANYCORE:** Systems augmented w/ accelerators, such as GPUs or FPGAs, are proliferating.  
**IMPACT:** Reasoning about cost / benefit of sending computation to accelerator is difficult.  
**TOOL: MetaSim Tracer**  
 ○ Captures an application's use of the processor and memory subsystem.  
**TOOL: Modeling Assertions**  
 ○ Also captures the application's memory behaviors; complements MetaSim, adding user-level knowledge.

### DEPTH AND BREADTH OF SYSTEMS

**#3 PETASCALE AND BEYOND:**  
**IMPACT:** Predicting scalability becomes critical-path for applications.  
**TOOL:** Communications Extrapolations  
 ○ AI techniques are used to predict the communication patterns of applications at unavailable CPU counts by taking a series of traces and extrapolating a larger trace.  
**TOOL: Modeling Assertions**  
 ○ Captures the app. communications.

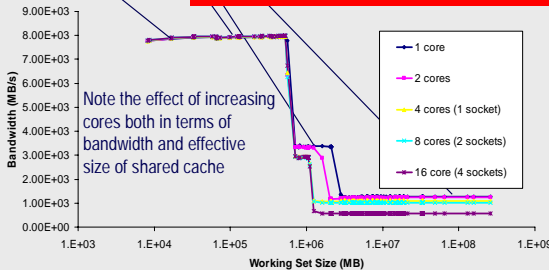
**#4 DEEP MEMORY HIERARCHIES:**  
**IMPACT:** Hiding latency becomes difficult.  
**TOOL:** Communications Convolver  
 ○ Extrapolated communications traces are combined with forecast memory performance models to yield performance predictions in advance of deployment.

**CONVOLUTION METHOD:** Mapping the memory usage needs of an application to the capabilities of a given machine.

Block #	% Mem. Ref.	Random Ratio	L1 Hit Rate	L2 Hit Rate	Data Set Location in Memory	Memory Bandwidth
180155	0.919	0.07	93.47	93.48	L1 Cache	4166.0
180153	0.027	0.00	90.33	90.39	Main Memory	1809.2
180160	0.023	0.00	94.81	99.89	L3 Cache	5561.3
S885	0.012	0.20	77.32	90.00	L1/L2 Cache	1522.6

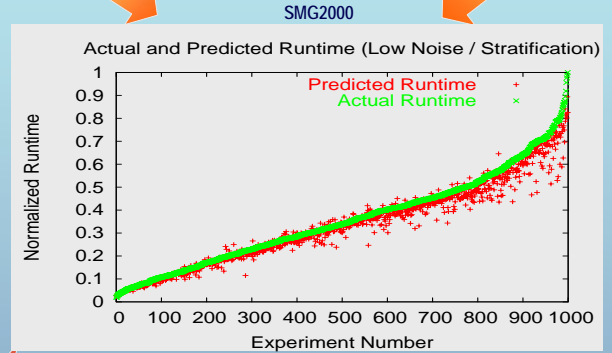
MetaSim Tracer

MAPS for MULTICORE OPTERON



**TOOL: MetaSim Convolver**

Automates the mapping of basic blocks' cache hit rates to their expected bandwidth on the MAPS curve.



**System Evaluation: The Multicore Tax:**

(Performance Predictions for Scientific Applications)

"If I upgrade my system from a given multicore level to a 2-fold increase in multicore level, will my applications run twice as fast on the same number of nodes?" MAPS curve suggests no.

**S3D performance predictions for 250T (Jaguar) per core per grid point as a function of chemistry (not counting communications)**

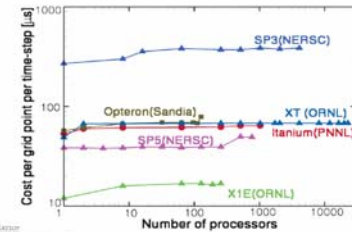
Chemical grid and opt	Time (μs)
H <sub>2</sub> 50 <sup>3</sup> orig	51
C <sub>2</sub> H <sub>4</sub> 50 <sup>3</sup> opt	132
C <sub>2</sub> H <sub>4</sub> 35 <sup>3</sup> opt	133
C <sub>2</sub> H <sub>4</sub> 18 <sup>3</sup> opt	172

S3D is a Combustion Joule Metric Code  
 ○ high-fidelity simulations of turbulent combustion with detailed chemistry  
 ○ optimized by PERI Tiger Team  
 ○ communications not significant to 10k+ cores with weak scaling (holding grid size per core constant)

**QUADCORE UPGRADE "TAX" = 5% FOR S3D**

### S3D's Parallel Scaling

Weak scaling test with CO/H<sub>2</sub> mechanism and 50<sup>3</sup> grid points per MPI thread

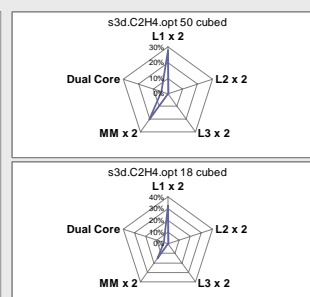
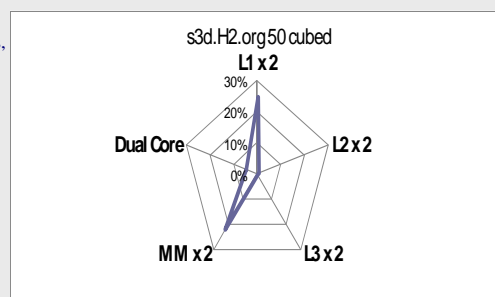


Data due to Ramanan Sankaran

### APPLICATION IN DEPTH: Sensitivity Studies

With careful manipulation of the convolution methods, with respects to the memory hierarchy, CPU rate, latency and bandwidth, we can carry out sensitivity studies. These studies are useful in explaining and quantifying performance differences.

Once a model has been confirmed to work for an application, as shown, the model can be used to explore which machine attributes will best benefit the application performance.



L1x2 – L1 cache BWx2, Lat./2  
 L2x2 – L2 cache BWx2, Lat./2  
 L3x2 – L3 cache BWx2, Lat./2  
 MMx2 – Main memory BWx2, Lat./2  
 Dual Core – Just two cores of QC (e.g. better main memory by ~20%)