An End-to-End Example of Performance Modeling in Action

Adolfy Hoisie

Center for Advanced Architectures and Usable Supercomputing (CAAUS)

Joint work with Kevin J. Barker, Kei Davis, Darren J. Kerbyson, Mike Lang, Scott Pakin, Jose Sancho, and many others

Performance and Architecture Laboratory (PAL)
Los Alamos National Laboratory

SOS13, March 2009
Hilton Head, SC

Outline

- **Performance analysis activities at Los Alamos**

- **Performance "in action":**
  - System design
  - Application Design
  - Performance prediction for assessment
  - Tool development
  - Performance acceptance testing

- **A few general remarks**

Performance and Architecture Lab

- **PAL is the performance analysis team at Los Alamos**
  - Measurement, Modeling, Simulation…
  - Novel modeling techniques developed and applied
- **Large-scale:**
  - Large-scale Applications + Large-scale System = Large-scale performance
- **Systems:**
  - ASCI (Q, purple, red-storm), ORNL (Jaguar), IBM BG/L, BG/P
  - PERCS (-> Blue Waters), Zia, Sequoia …
  - Analyze existing systems (or near-to-market systems)
  - Examine possible future systems (or subsystems such as circuit-switched optical networks)
- **Recent work includes:**
  - Optimization of ASCI Q, OS Noise (SC'03)
  - Blue Gene/L (SC04)
  - Large-scale Optical Circuit Switch network (SC05)
  - System Comparison: BG/L, Red Storm, ASC Purple (SC06)
  - Architecture and performance of Roadrunner (SC08)
  - Early processor analysis: Barcelona, PowerXCell-8i, Nehalem

Performance Modeling

- Novel methodologies developed by PAL at Los Alamos in the last decade
- Models encapsulate performance of entire apps on full systems
- Our modeling approach is “application centric”
- Models are predictive, and highly accurate
- The application workload considered already is large and diverse (NNSA, SC, DARPA, NSF, etc)
- Models were validated on most of the large supercomputers in the last decade
- Models are our tools for performance analysis
- We apply modeling to :

Performance Modeling at LANL

Model

Maintenance → **System updates**
Quantify impact on performance

Optimization → **Improvements**
Quantify impacts prior to implementation

Installation → **Is the machine working?**
Performance should be expected (ASCI Q, Purple, Roadrunner…)

Procurement → **Which system should I buy?**
Systems unavailable to measure (e.g. roadrunner)

Implementation → **Small prototype available**
What will be the performance of a N Petaflops system?

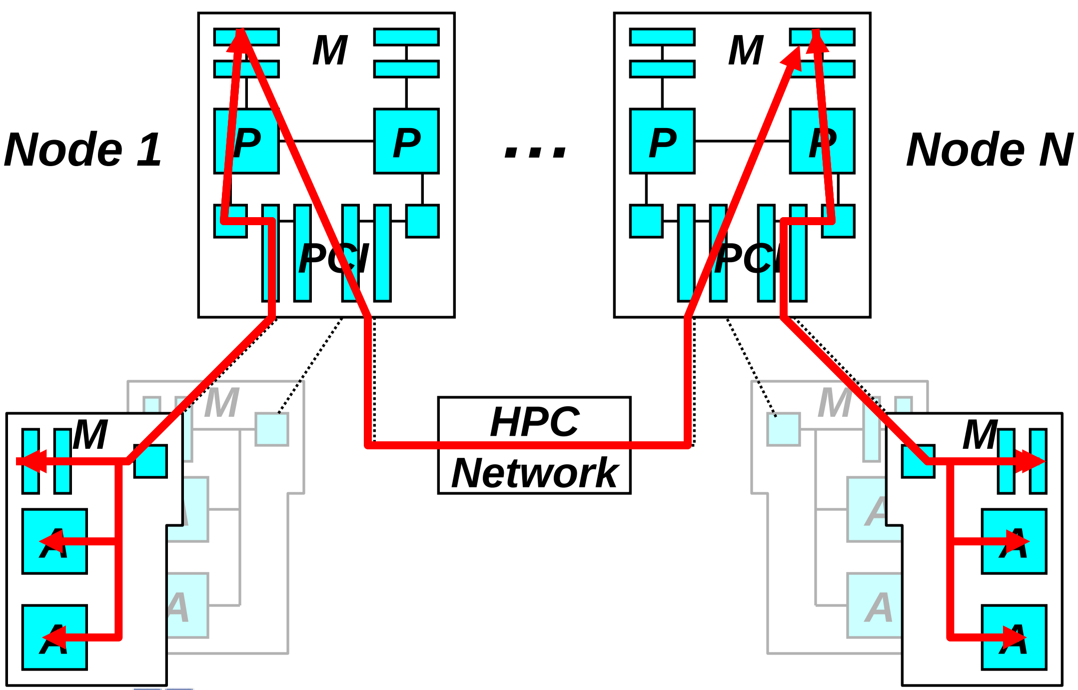
Design → **System unavailable for measurement**
What will be the performance of IBM PERCS be in 2010?
What will be the performance of BG/P?
Exascale system design

Outline

- Performance analysis at Los Alamos
- **Performance "in action":**
  - System design
  - Application Design
  - Tool development
  - Performance prediction for assessment
  - Performance acceptance testing
- A few general remarks

Pre-Roadrunner, Circa 2005
Two-level Heterogeneous System

- Compute nodes (e.g., with 2-sockets)
- HPC interconnection network (e.g., Infiniband)
- Accelerators placed in each node (e.g., PCI based)



1) Start-up

Node → Accelerator

2) Process on accelerator

3) Inter-node communication

Accelerator → Node →

HPC Network → Node

→ Accelerator

4) Repeat 2 (& 3)

5) Finalize

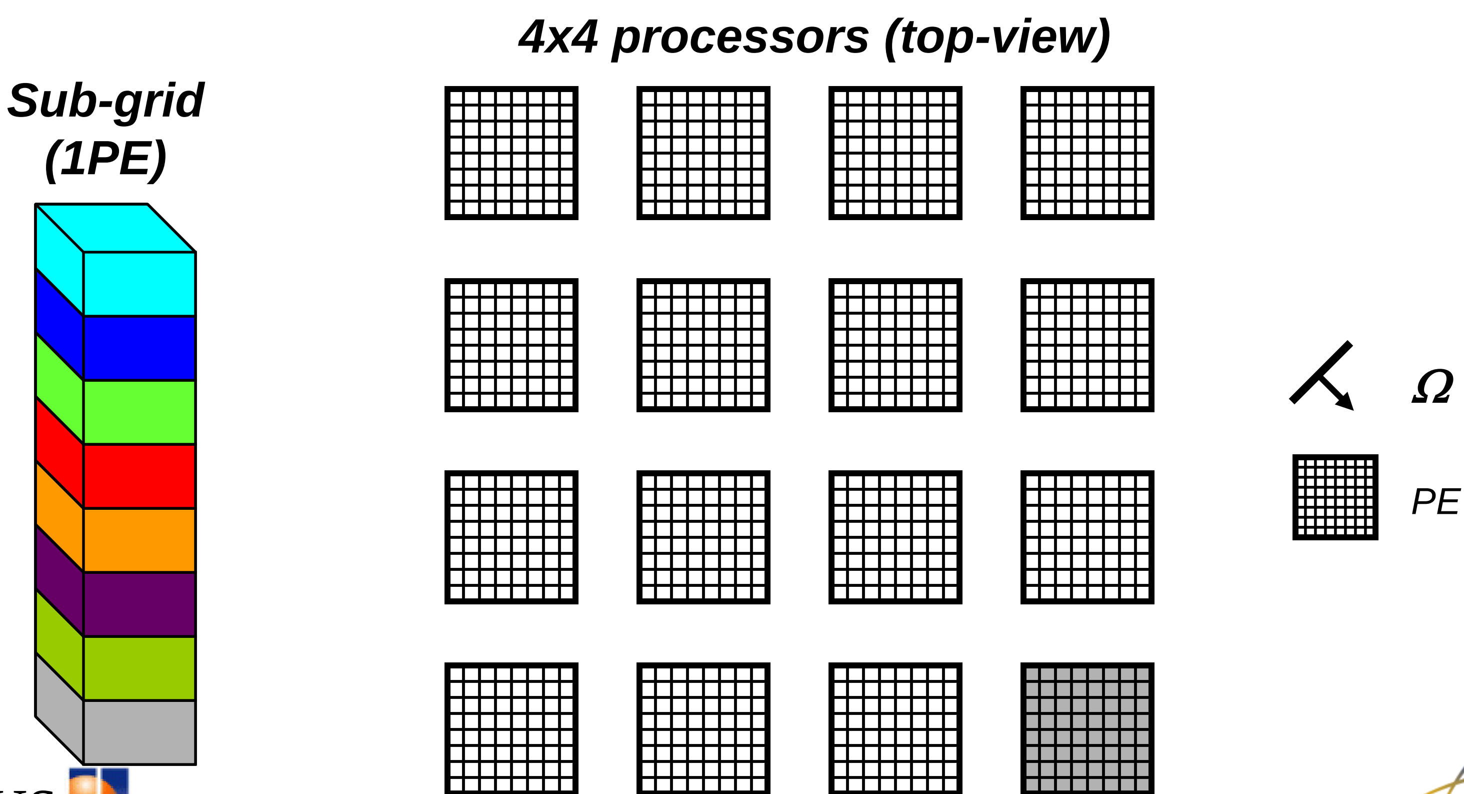
Accelerator → Node

CAAUS
Center for Advanced Architectures
And Usable Supercomputing

Los Alamos
NATIONAL LABORATORY
EST. 1943

Sweep3D: Wavefront Parallelization

- **Available parallelism is limited**
- **3-D grid is typically parallelized in only 2-D**
  - Column (without blocking) gives poor efficiency
  - Blocking used to increase parallel efficiency



Characteristics of Wavefronts

- **A pipeline in several dimensions, with 2-D parallelization:**
  - pipeline length = $P_x + P_y - 2$ (for one direction)
- **Blocking factor, *B = K-planes per block*, increases parallel efficiency**
- **Basic performance model uses pipeline length and blocking:**

$$T_{cycle} = \underbrace{\frac{K}{B}}_{\text{Number of K blocks in column}} . \left(\underbrace{B.T_c}_{\text{Time to process one K-plane}} + \underbrace{4.T_{msg}(B)}_{\text{Communication time per block}} \right) + \overbrace{\left(P_x + P_y - 2 \right)}^{\text{Pipeline length}} . \left(\underbrace{B.T_c}_{\text{Time to process one K-plane}} + \underbrace{2.T_{msg}(B)}_{\text{Communication time per block}} \right)$$

- **Pipeline effect minimized when B = 1**
- **Parallel overhead (message time) minimized when B = K, and**
  - In general, block size decreases with scale

Wavefront Performance When Using Accelerators

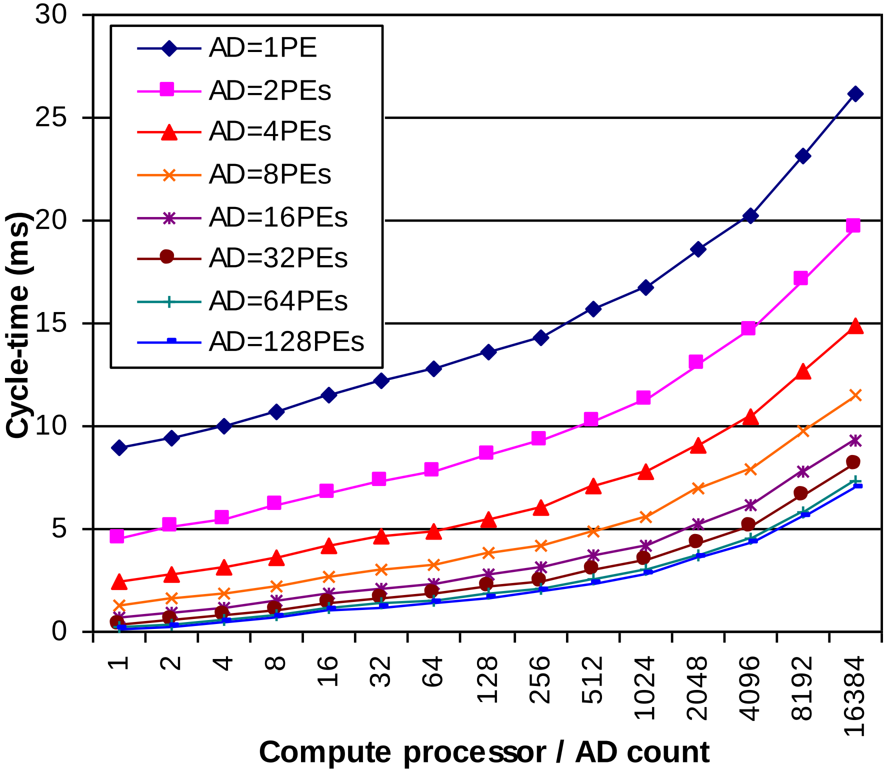
- **Block processed on the accelerator**
  - To process a block we have a pipeline on the accelerator

$$B.T_c = \frac{B}{B'}.\left(B' T_{AC} + 4.T_{msgAC}(B')\right) + \left(P_x^{'} + P_y^{'} - 2\right).\left(B'.T_{AC} + 2.T_{msgAC}(B')\right)$$

  - *(P'_x+P'_y-2)* is the pipeline length of the accelerator
  - *B'* is the blocking factor on the accelerator (usually 1)
  - T_{AC} is the compute time on the accelerator

- **Increases the pipeline length by factor *(P'_x+P'_y-2)***
- **Effect of accelerator pipeline is minimized when *B* is large**

Expected Performance (Run Time)



- **Assumptions (hypothetical system):**
  - Weak-scaling
  - 16x8x1000 sub-grids
  - Processing time per cell = 70ns
  - Inter-PE (on Accelerator)
    - » **Bandwidth =1GB/s,**
    - » **Latency = 50ns**
  - Inter-node (MPI)
    - » **Bandwidth = 1.6GB/s,**
    - » **Latency = 4μs**

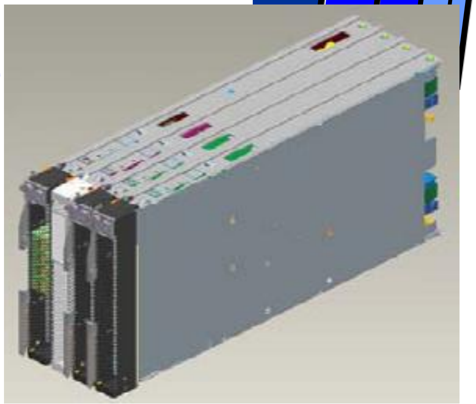
- ***At largest scale, 16,384 compute processors & 16,384 accelerators***
  - *Performance improvement is ~3.5x when using Accelerators with 128x more PEs*

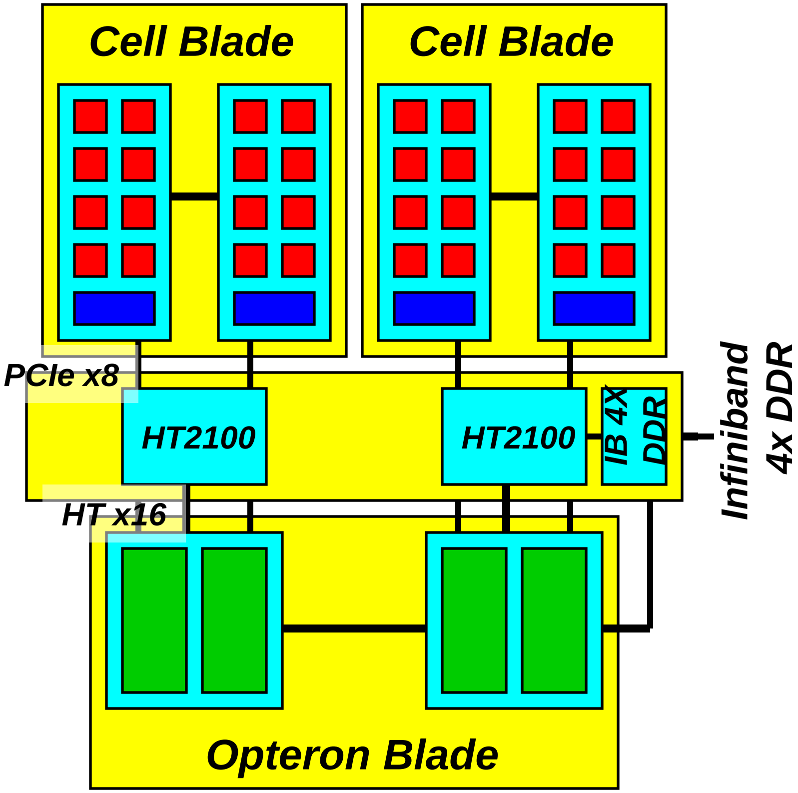
"A Performance Analysis of Two-Level Heterogeneous Processing Systems on Wavefront Algorithms", D.J. Kerbyson, A. Hoisie, Unique Chips and Systems, CRC Press, pp. 259-290, 2008.

IBM's Cell version of Sweep3D

- One of the first applications optimized for the Cell architecture
- Published at IPDPS 2007
- Reported significant speedups over microprocessor performance
- Implemented a master-worker paradigm
- Bounded by available memory bandwidth, lots of DMAs
- Mapping well to the way in which accelerated systems (Roadrunner) were envisioned to be used
- Compared to our version of Sweep3D through modeling

Roadrunner@Los Alamos: some peak-performance numbers

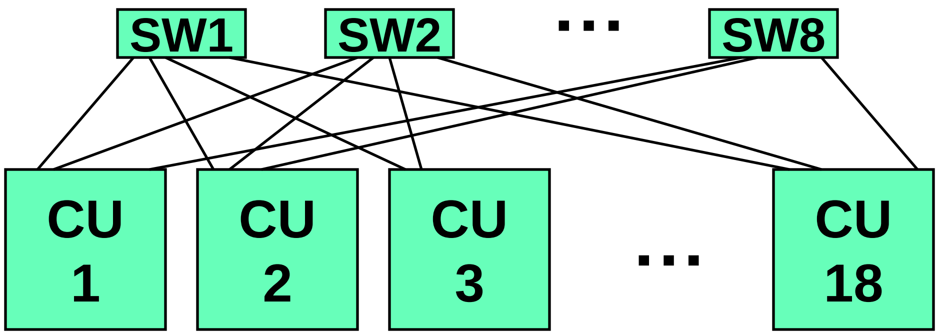




- **4x PowerXCell 8i (3.2GHz) = 4x (PPU + 8 SPUs)**
  - SPEs (per cell) = 102.4 Gflop/s (DP)
  - PPE (per cell) = 6.4 Gflops/s (DP)
- **4x AMD cores (1.8GHz)**
  - AMD = 3.6 Gflop/s (DP) / core
- **Cell <-> AMD**
  - Bandwidth = 2.0GB/s + 2.0GB/s
  - Latency ~1.5µs
- **AMD <-> AMD (inter-node)**
  - Bandwidth = 2.0GB/s + 2.0GB/s
  - Latency ~ 1.5µs

CAAUS
Center for Advanced Architectures
And Usable Supercomputing

Essential Roadrunner System Peak Performance Parameters



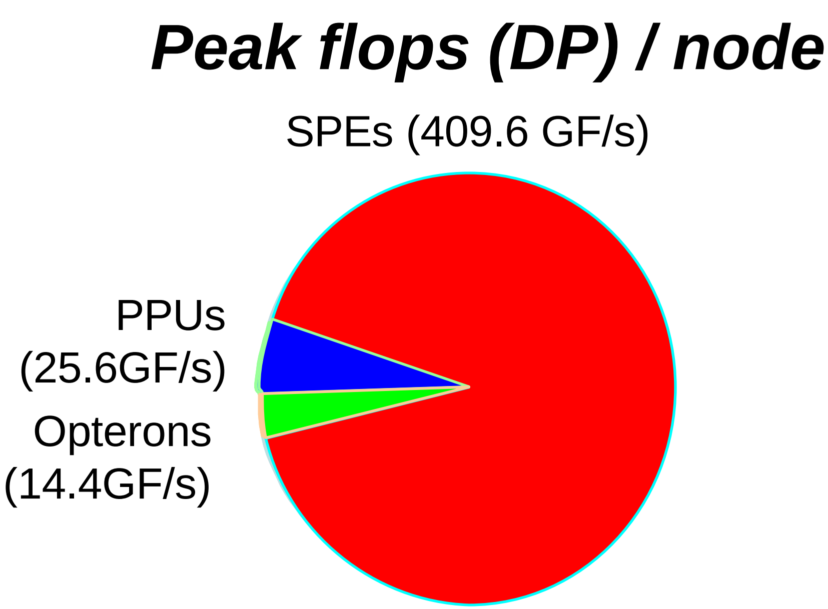
- System = 18 CU = 3240 triblades
  = 12960 (AMD cores + cell eDP
- Peak DP flops = 1.33Pf/s
- Memory capacity=77 TBytes
- Peak memory bandwidth (cells) = 0.277PB/s

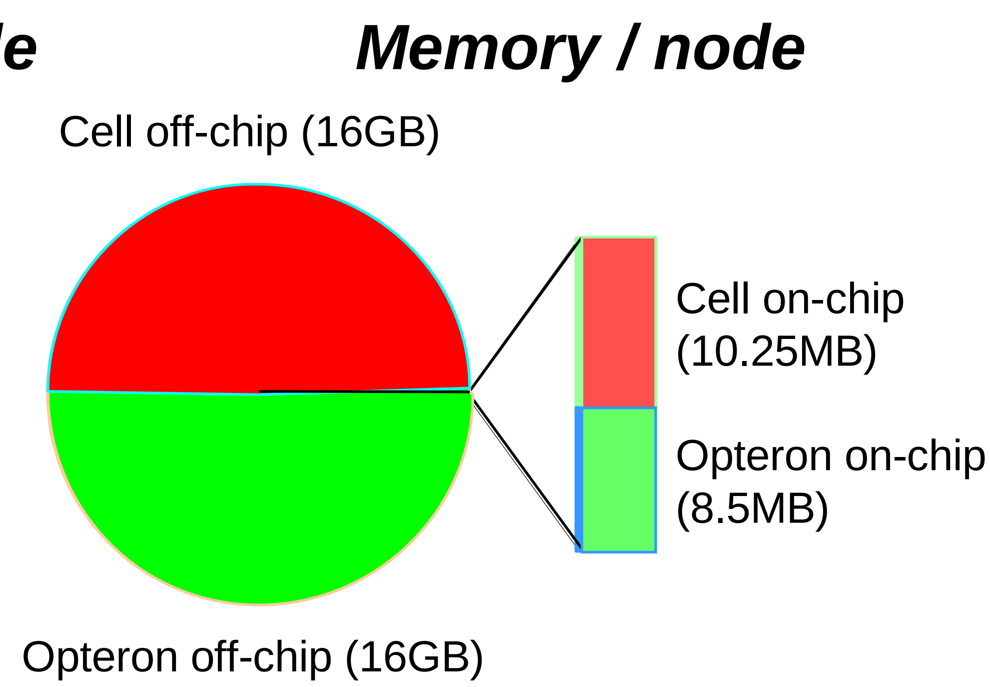
CAAUS
Center for Advanced Architectures
And Usable Supercomputing

Los Alamos
NATIONAL LABORATORY
EST. 1943

Relative capacities: Opterons & Cells

- 90% of the peak flops in the SPEs on the Cell
- Equal main memory between the Cells and Opterons

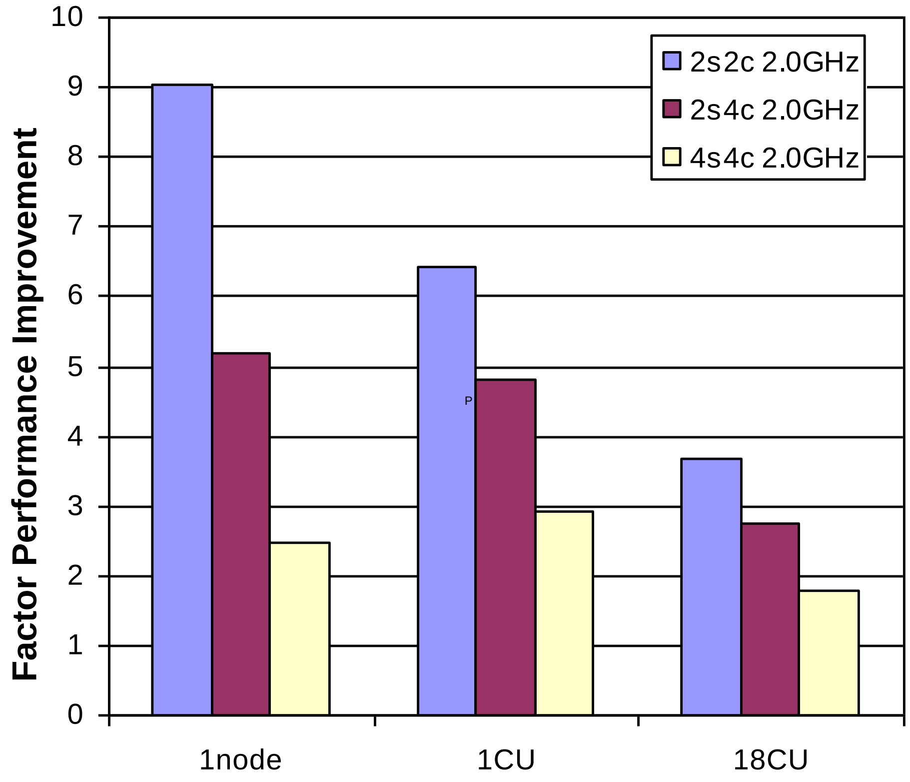




Outline

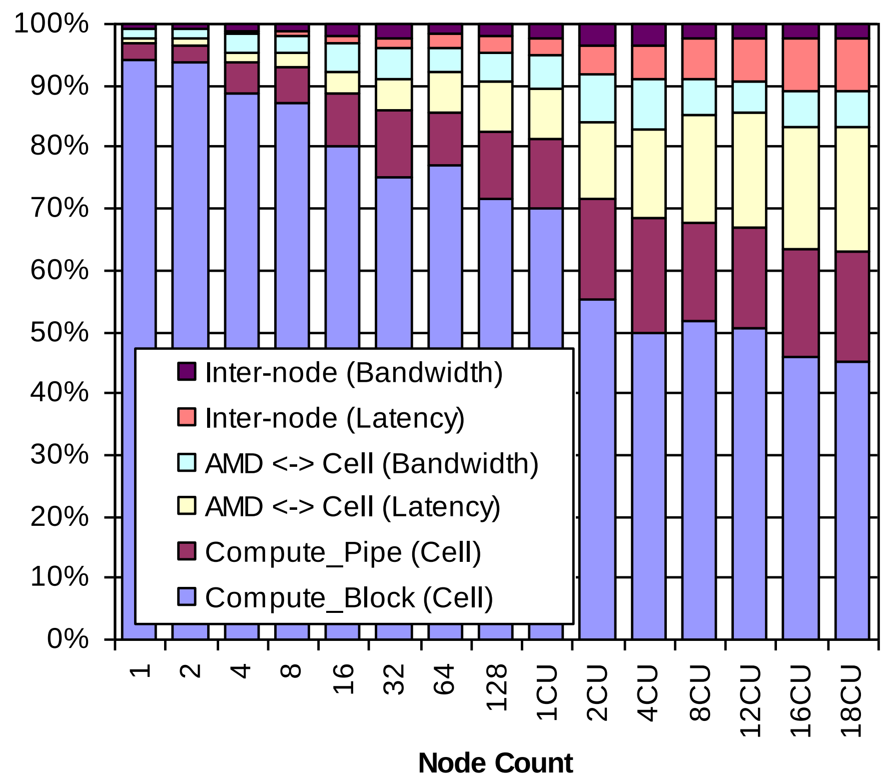
- Performance analysis at Los Alamos
- Performance "in action":
  - System design
  - Application Design
  - Performance prediction for assessment
  - Tool development
  - Performance acceptance testing
- A few general remarks

Roadrunner Performance Comparison for Sweep3D



Sweep3D: Profiling

- **Where is the time being spent ?**
  - ~63% Compute on Cell
  - ~20% Latency (Cell <-> AMD)
  - ~5% Bandwidth (Cell <-> AMD)
  - ~8% Latency (Infiniband)
  - ~3% Bandwidth (Infiniband)



- **Pipeline unavoidable**
- **Latency dominates communication (Cell <-> AMD is major component)**

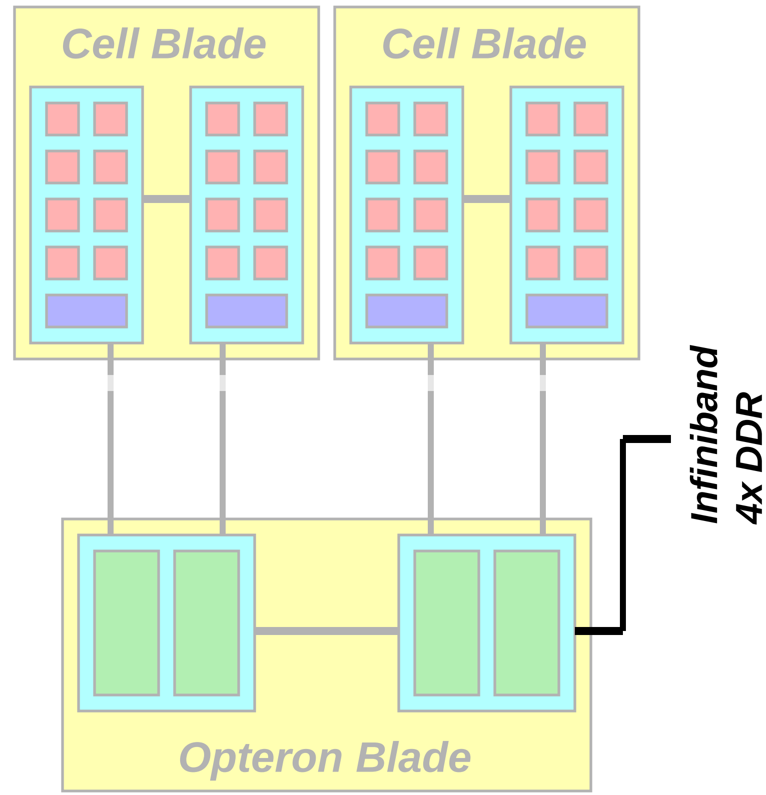
- **Uses 'probable' HW parameters**

Outline

- Performance analysis at Los Alamos
- Performance “in action”:
  - System design
  - Application Design
  - Performance prediction for assessment
  - Tool development
  - Performance acceptance testing
- A few general remarks

Roadrunner: Usage

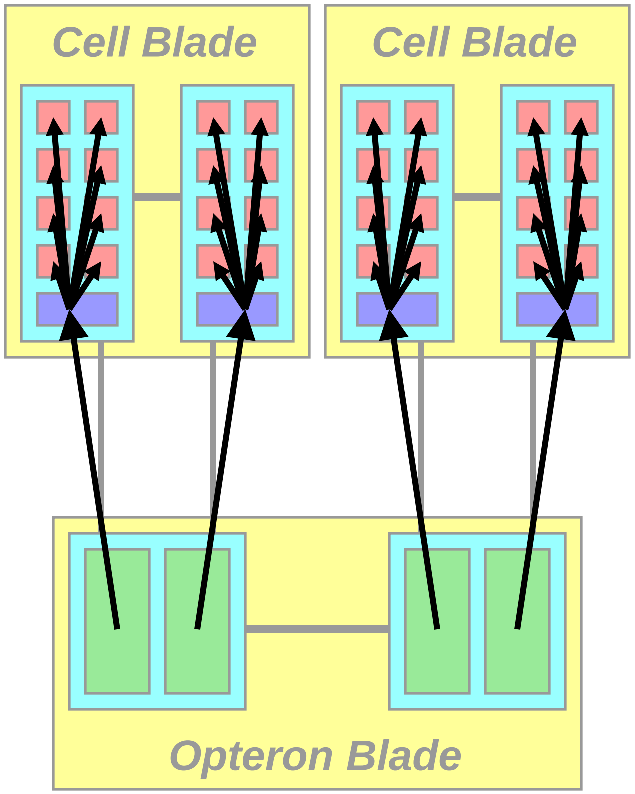
- **Non-hybrid (Opteron only)**
  - Codes run without modification
- ***Hybrid (Opteron and Cell)***
  - *Code* ***performance hotspots*** *ported to the Cell*
  - *Also incremental porting*
- ***Cell-centric (Cell only)***
  - *Need support for communications between Cells*
    - » ***Between PPEs***
    - » ***Between SPEs (e.g. CML)***



CAAUS
Center for Advanced Architectures
And Usable Supercomputing

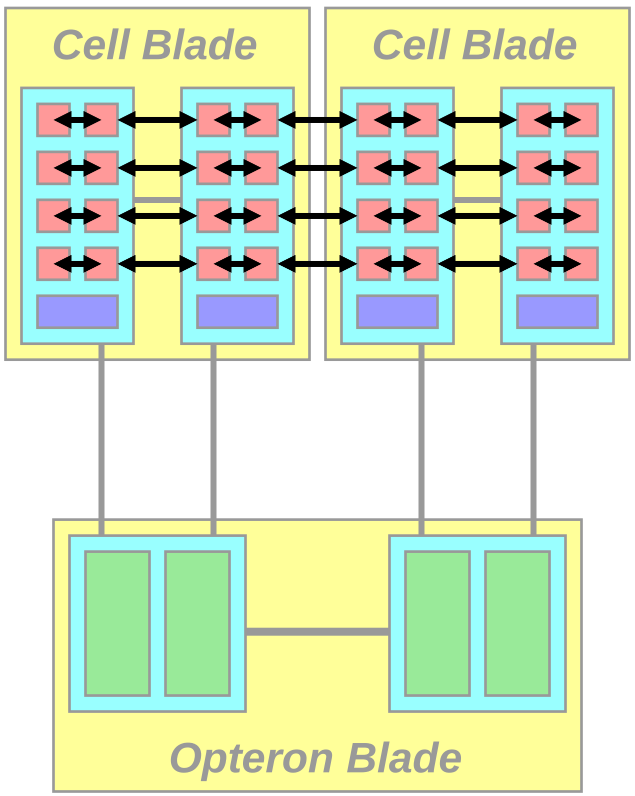
Los Alamos
NATIONAL LABORATORY
EST. 1943

Hybrid (general accelerator approach)



- One MPI rank per Opteron
- SPE = accelerator
- Opterons see each other and their local SPEs
- Opteron pushes work (data) to SPEs and receives results
- DaCS
  - Data Communication and Synchronization for Opteron <-> Cell
- libSPE (or ALF) for SPE work management

SPE-centric (Cell-Messaging-Layer)



- **One MPI rank per SPE**
- **Opteron = NIC (& extra storage)**
- **SPEs see each other and their local Opteron**
  - SPEs communicate directly with other SPEs
  - PPE provides support
- **MPI subset, currently:**
  - blocking MPI pt2pt & collectives
  - Small memory footprint
- **"Cluster of 100,000 SPEs"**

Cell Messaging Layer (CML)

- **Facilitates porting MPI codes to the Cell**
  - Based on lessons learned in getting Sweep3D ported
  - Leverages modeling results
  - Generalizes approach that worked for Sweep3D to other codes
- **Provides a familiar programming model to application developers**
  - One MPI rank per SPE across all of Roadrunner
  - "Cluster of 100,000 SPEs"
  - PPEs & Opterons used only for comm. across node boundaries
  - No need to rethink application's domain decomposition
  - No hybrid programming (but may still need to transfer data between main memory and local store)

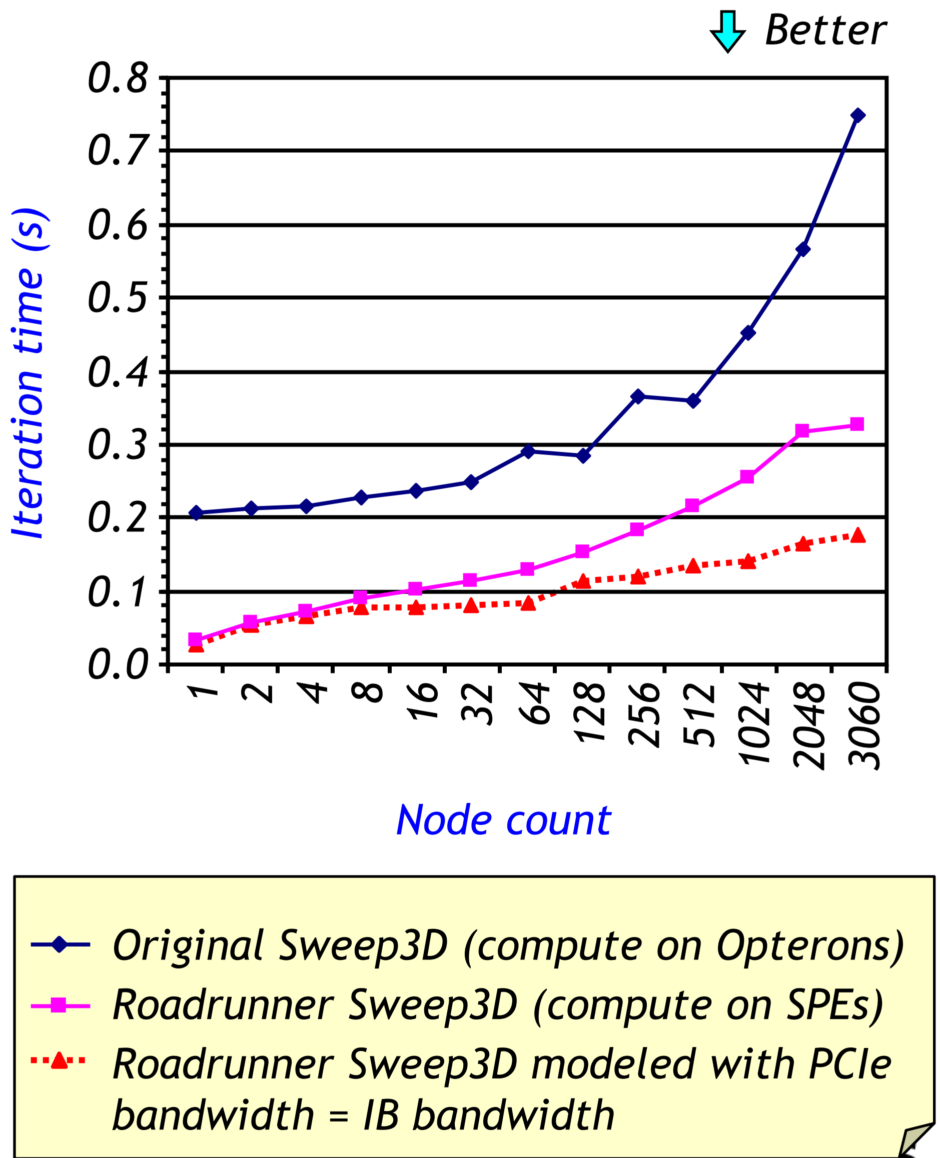
Other CML Characteristics

- **"Reverse acceleration" model**
  - SPE process can invoke code on its PPE (today) and Opteron via remote procedure call interface
  - Sweep3D uses this for memory allocation and I/O
- **Fast!**
  - Intra-cell send/receive latency of 0.272 µs (870 clock cycles)
  - Intra-cell send/receive bandwidth of 24.1 GB/s (94% of peak)
  - .843 µs/22.3 GB/s within a blade
- **Open-sourced (GPL license)**

  http://cellmessaging.sourceforge.net/
- **For more information see:**
  - Scott Pakin. *Receiver-initiated Message Passing over RDMA Networks*. IPDPS 2008, Miami, FL, April 2008.

Sweep3D on Roadrunner

- **All compute done on SPEs**
- **PPEs and Opterons used as smart NICs**
  - (Remember: 91% of performance on SPEs)
- **Same basic data structures and control flow as conventional Sweep3D**
  - Cell Messaging Layer provides MPI for SPEs
  - One MPI rank per SPE
    - » **Treat Roadrunner as a 97,920-SPE cluster**
- **2X improvement at scale**
  - *Expect 4X with feasible SW modifications*



Outline

- Performance analysis at Los Alamos
- Performance "in action":
  - System design
  - Application Design
  - Performance prediction for assessment
  - Tool development
  - Performance acceptance testing
- A few general remarks

A Taste of Performance Acceptance Testing Criteria

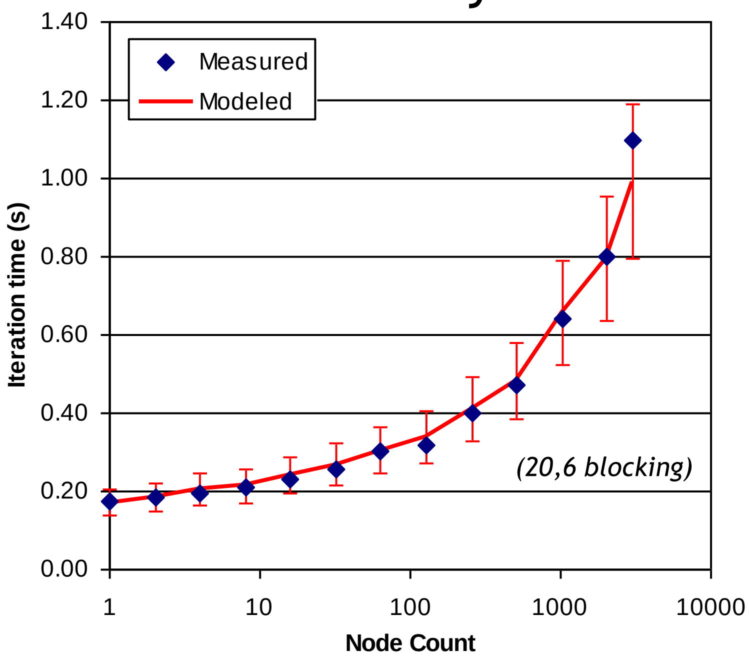
- Communication latency (node-to-node)
- Communication bandwidth
- MPI_Allreduce collective latency with an 8-byte payload
- Application performance within 20% of the modeled performance

4. Application Performance – Sweep3D

Test goal: application performance within 20% of the modeled performance

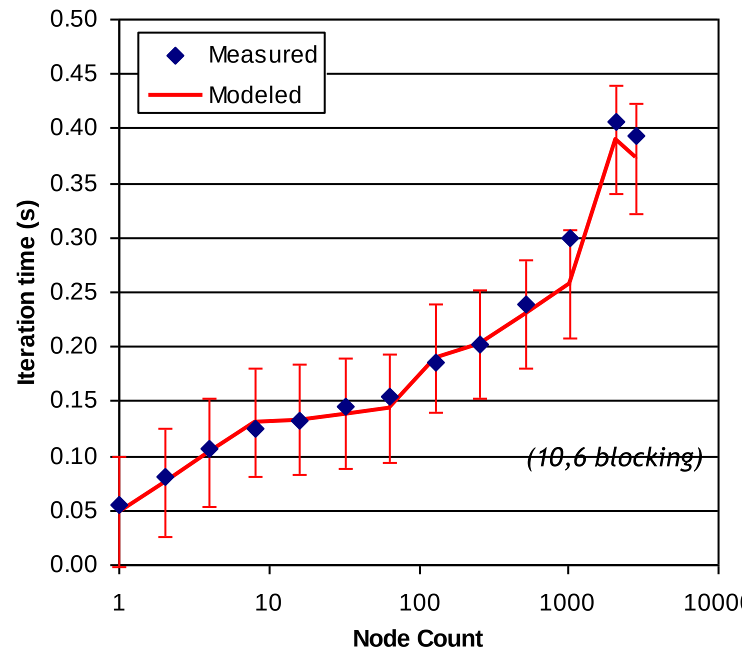
20x10x400 on each Opteron Core, or 5x5x400 on each SPE

AMD only



Max difference: 11%

Hybrid (PAL-Sweep3D, CML)



Max difference: 16%

CAAUS
Center for Advanced Architectures
And Usable Supercomputing

Los Alamos NATIONAL LABORATORY EST.1943

A few general remarks

- Performance analysis at Los Alamos: methodology development and active application to real life problems
- Performance analysis applied in practice: end-to-end example: system and application design, analysis, prediction and acceptance testing
- Performance modeling works!

Shameless plug

- Special issue of IEEE Computer on “Extreme scale computing”, scheduled for November 2009.
- Call for papers to be widely distributed soon
- Widely encompassing coverage of topics: architecture, networking, performance, reliability, power efficiency, programming models, etc.