Petascale Computing architectural requirements

William J. Camp, Director

Computation, Information, and Math Center

Sandia National Laboratories

Albuquerque, NM 87185

bill@sandia.gov

SOS-10 WORKSHOP

Maui,

March 2006

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Petascale Computing architectural requirements-- Environmental and Energy Sciences as an example

- What do we need in a petascale architecture for data-intensive simulation in 2009--2010 timeframe?
- Do we need radical new ideas?
- Can it be general purpose?
- Will we be able to afford it?

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Petascale Computing architectural requirements

- Data-intensive scientific simulation as a driver
- Environmental simulation incorporating historical and real-time data as the archetype
- Not the same as informatics and knowledge generation from large data sets
- Making computation really work for decision support
  - ***[the goal of computation should be insight]***

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

What is a good climate simulation engine?

Suppose we want to simulate the earth's climate

-at sufficient resolution

so that results are "converged"

-for long enough

decades, centuries, and millennia

-allowing for uncertainties

…of initial conditions

…in the physics and chemistry

…and in the boundary conditions

- incorporating measured data as constraints

-looking at the effectiveness of amelioration strategies

Sandia National Laboratories

so that results are “converged”

A Caveat:

“Converged” means that the results do not change significantly under additional mesh refinement for the entire time-span under study.

But the underlying equations are in fact highly sensitive to initial or boundary conditions.

Then if you simulate long enough you will find that you are never “converged.”

In practice, we converge coarse-grained observables-- this works

so that results are “converged”

In principle, we would look at how the evolution of key observables, e.g.

ice mass at the poles

distribution and fluctuations in precipitation

distribution and fluctuations in temperature

windspeed and storm intensity and frequency

change as we refine the mesh

This drives us to want to look at ensembles of increasingly resolved simulations-- this means we need a lot of ***capacity***

so that results are “converged”

Convergence means that we have to do very highly resolved studies

This drives the need for ***capability*** (do very large individual simulations rapidly)

It also poses requirements on algorithmic performance.

An algorithm that scales as $N^{4.33}$ -- as might an explicit spectral method-- will not scale

An algorithm that scales like $N^{1.33}$ -- as might an explicit spectral element method-- becomes preferable at high resolutions

…For Decades, Centuries, and Millennia

We need to model the past decades where we have extensive data

We need to model the past century, where we have reasonable data

We need to model the past millennium where we have some data

We need to project scenarios decades and centuries forward

This drives the requirement for ever more ***capability***

-allowing for uncertainties

Uncertain initial conditions mean that models need to run longer to lose memory of the initial conditions

Uncertainties in physics and chemistry mean that we have to do sensitivity analyses

Uncertainties in boundary conditions (locations, sizes and frequencies of volcanic events, fires, solar activities…)

These also drive the requirement for ever more ***capacity*** (do very many simulations simultaneously)

- incorporating measured data as constraints

Where we have measurements, we should use rigorous optimization techniques to choose parameters to obtain optimal, weighted overall agreement with observations.

While this can make use of surrogate methods, it still requires ***ensembles*** of calculations.

This drives the requirement for ever more ***capacity*** (do very many simulations simultaneously)

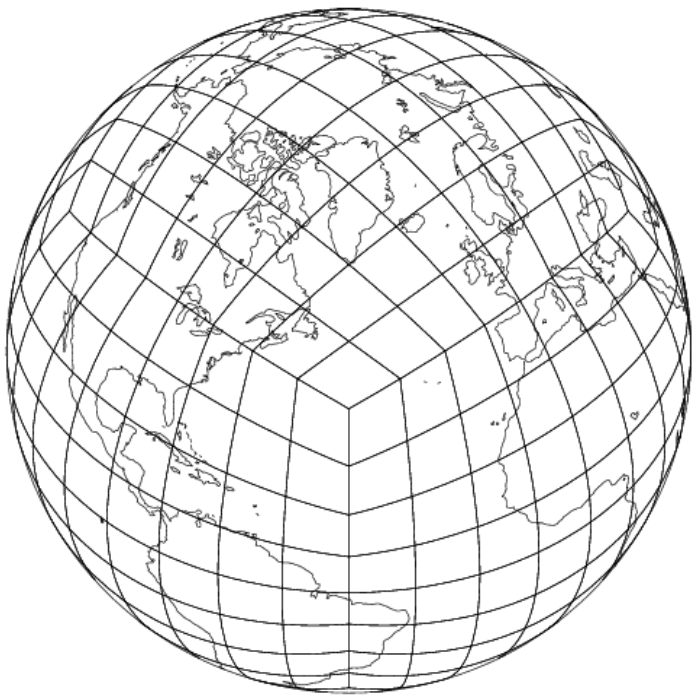
-Sizing the requirement

Capability: If we want to to do our most detailed runs in about one month and we want to couple atmosphere and ocean, we need a machine that will sustain petaflops-level computing (at current 1/10th degree/10KM finest resolution)

Capacity: we will want to do O(100) of these per year. This means the architecture must be affordable enough to provide ~10 copies world wide.

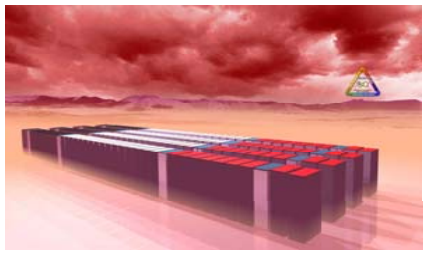
In fact, doing resolution studies, uncertainty estimation, elimination of sensitivity to IC's, and optimization of ability to reproduce observation, drive a need for many times this capacity.

Spectral Element Atmospheric Model (SEAM)



- Sandia is collaborating with NCAR on the development of NCAR's Spectral Element Atmospheric Model (SEAM)
- Spectral elements replace spherical harmonics in horizontal directions
- High order (p=8) finite element method with efficient Gauss-Lobatto quadrature used to invert the mass matrix.
- Two dimensional domain decomposition leads to excellent parallel performance.

SEAM Benchmarks (aqua planet)



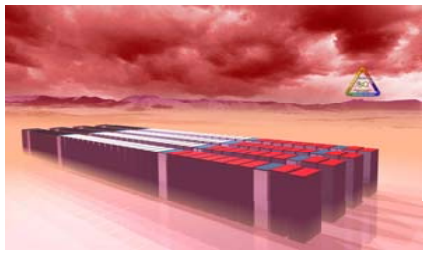
Red Storm

Blue Gene Livermore

Resolution	NCPU	Teraflops	Computer
40 km	10K	4.60	RS
	10K	1.25	BGL
10 km	10K	5.25	RS
	10K	1.90	BGL
	64K	3.20	BGL

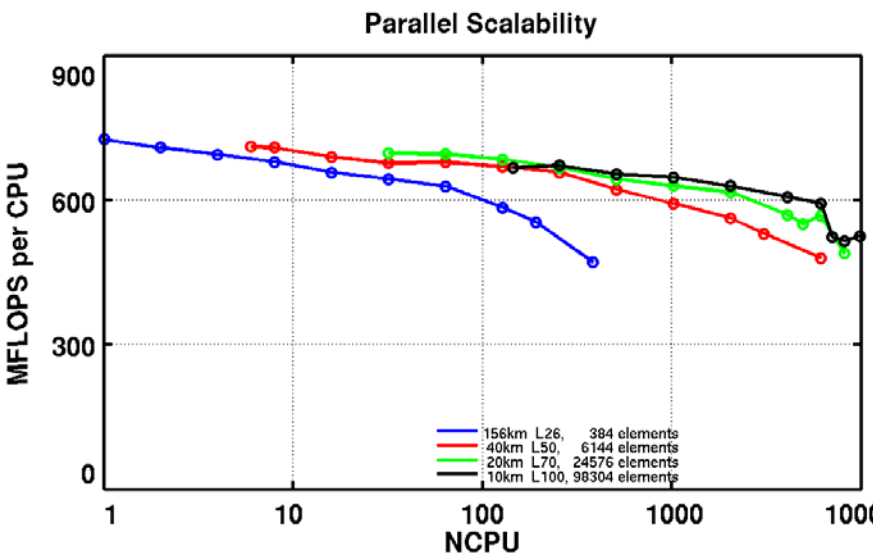
Sandia National Laboratories

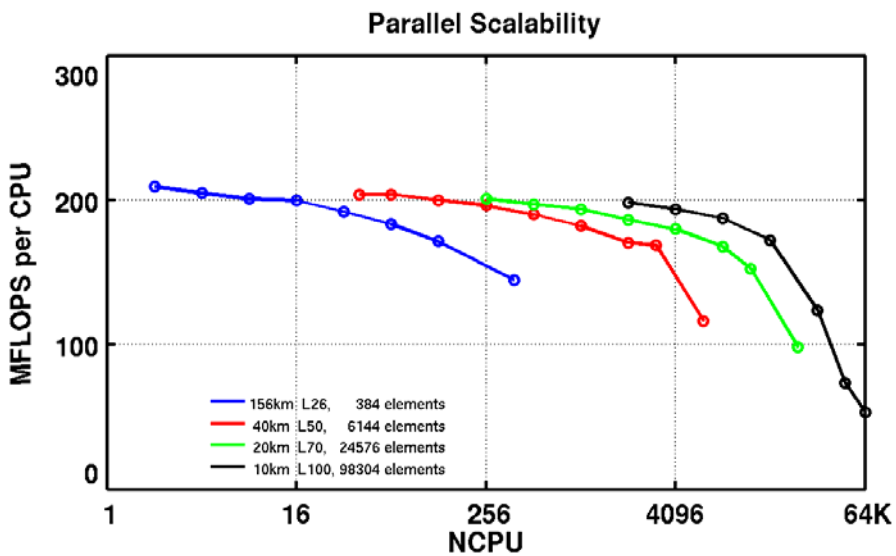
SEAM Benchmarks (aqua planet)



Red Storm

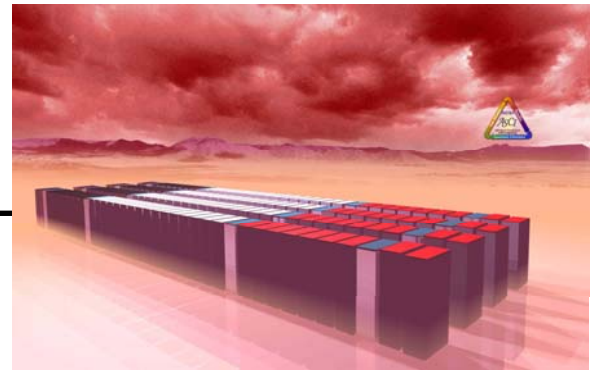
Blue Gene Livermore

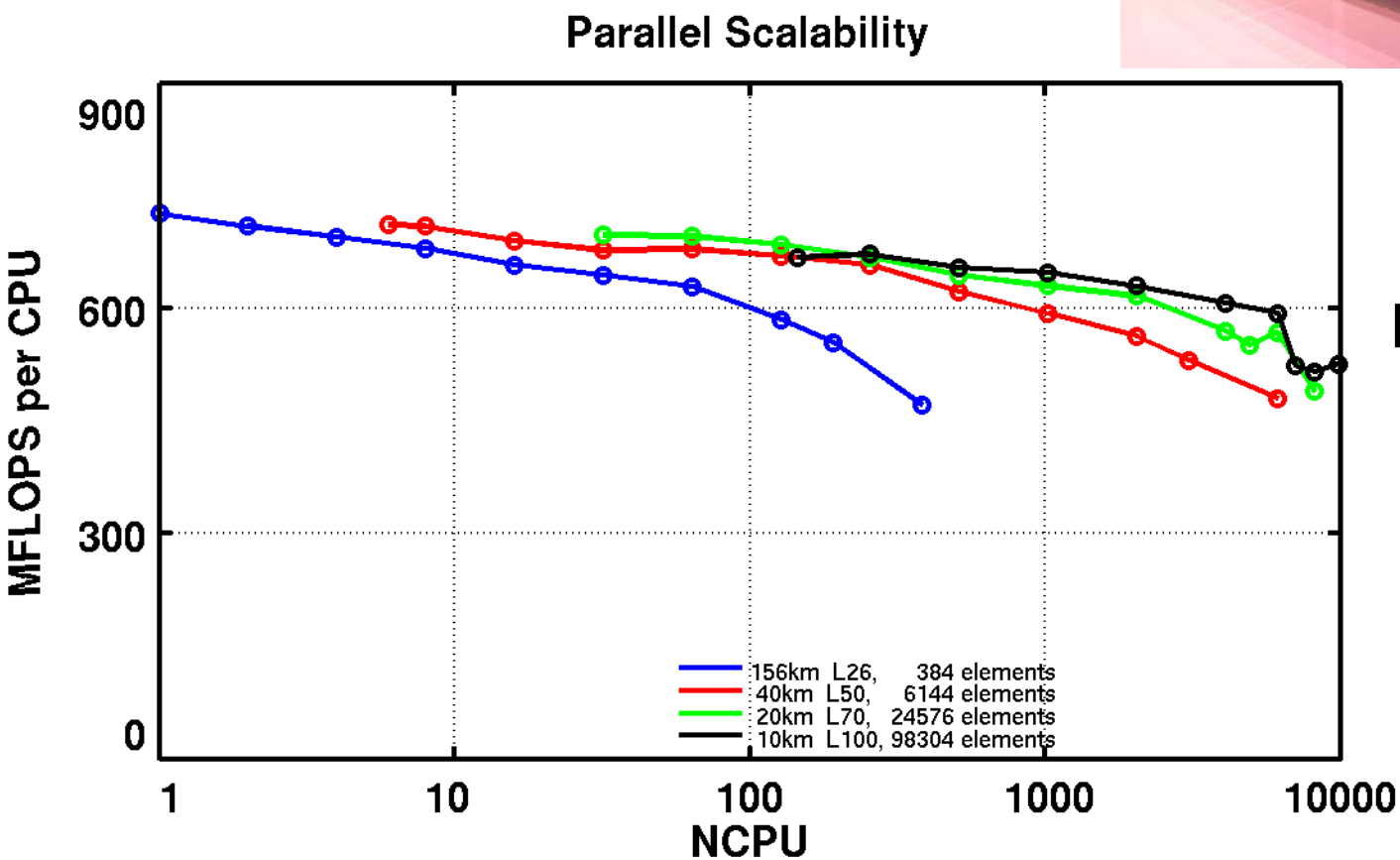




Sandia National Laboratories

SEAM on Red Storm



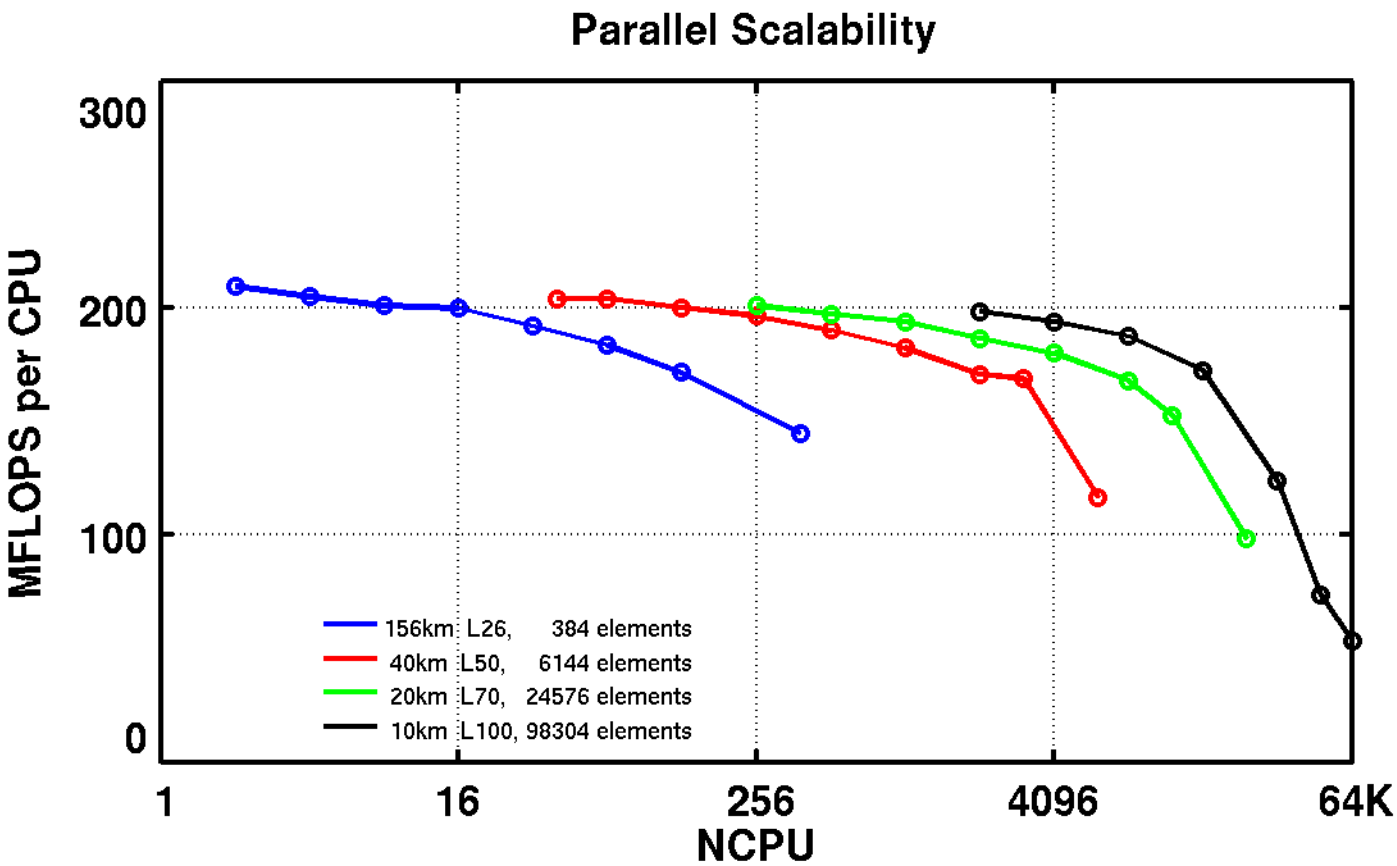


Max: 5TF

Performance of 4 fixed problem sizes, on up to 6K CPUs. The annotation gives the mean grid spacing at the equator (in km) and the number of vertical levels used for each problem.

Sandia National Laboratories

SEAM on BG/L

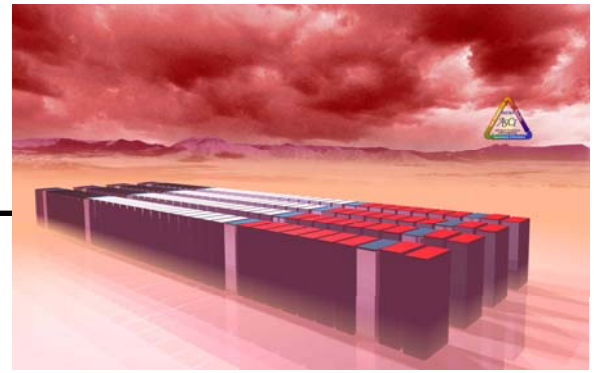


Max: 4TF

Performance of 4 fixed problem sizes, on up to 64K CPUs. The annotation gives the mean grid spacing at the equator (in km) and the number of vertical levels used for each problem.

Sandia National Laboratories

SEAM Estimates on Red Storm



Resolution	Simulation Rate
40km	7-30 yr/day
10km	0.1-0.3 yr/day

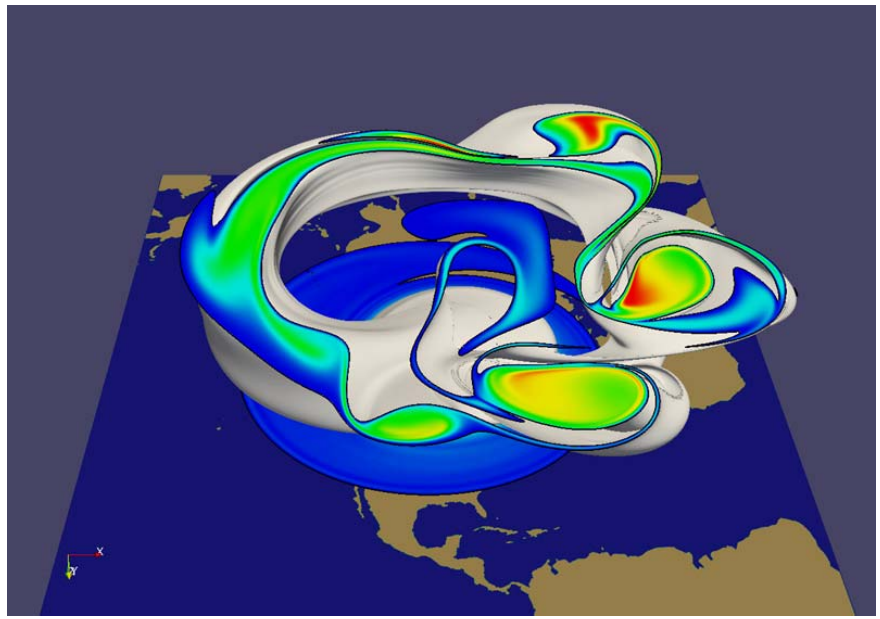
Estimated simulation rates for full atmospheric model (extrapolated from Aqua planet benchmark results)

Sandia National Laboratories

Climate Simulation: Science Run

Polar Vortex:
Strong circumpolar jet that traps air over the south pole.

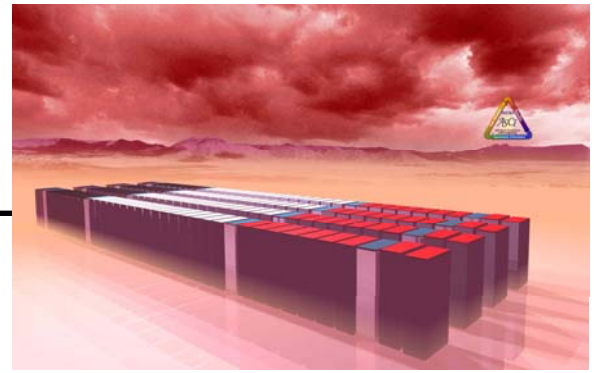
Computation:
1B grid points
Integrated for 288,000 timesteps using 7,200 CPUs

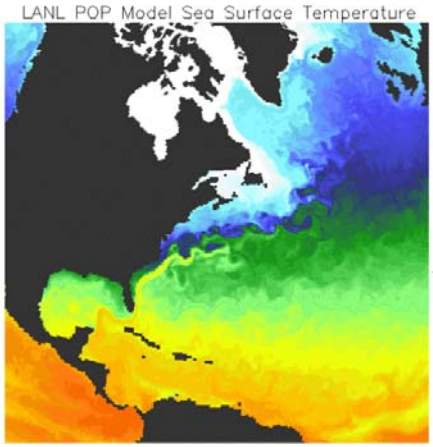


Contours of iso-surfaces over the North Pole. Relates to Ozone balance.

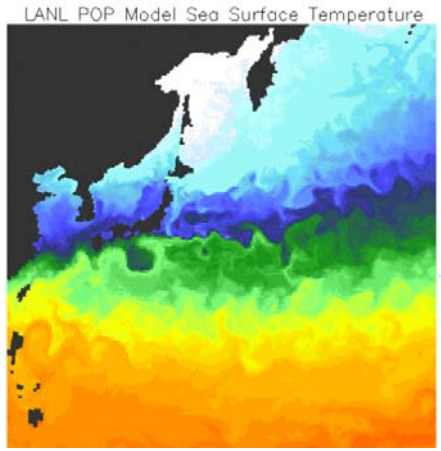
Sandia National Laboratories

POP Science Run on Red Storm





N Atlantic

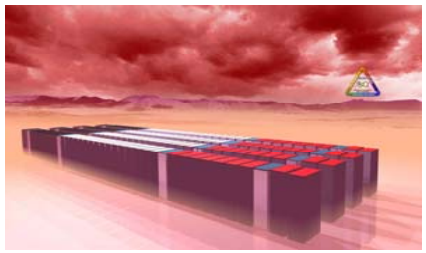


NW Pacific

- LANL's POP (Parallel Ocean Program) Model
- In collaboration with Mat Maltrud (LANL), we performed two 10 year simulations on 5000 processors.
- Resolution: 1/10 th of a degree (3600x2400x40) 350M grid points.
- I/O: 1.5TB
- Improved results for Gulf Stream separation, NW corner, Agulhas rings and Kuroshio current

Sandia National Laboratories

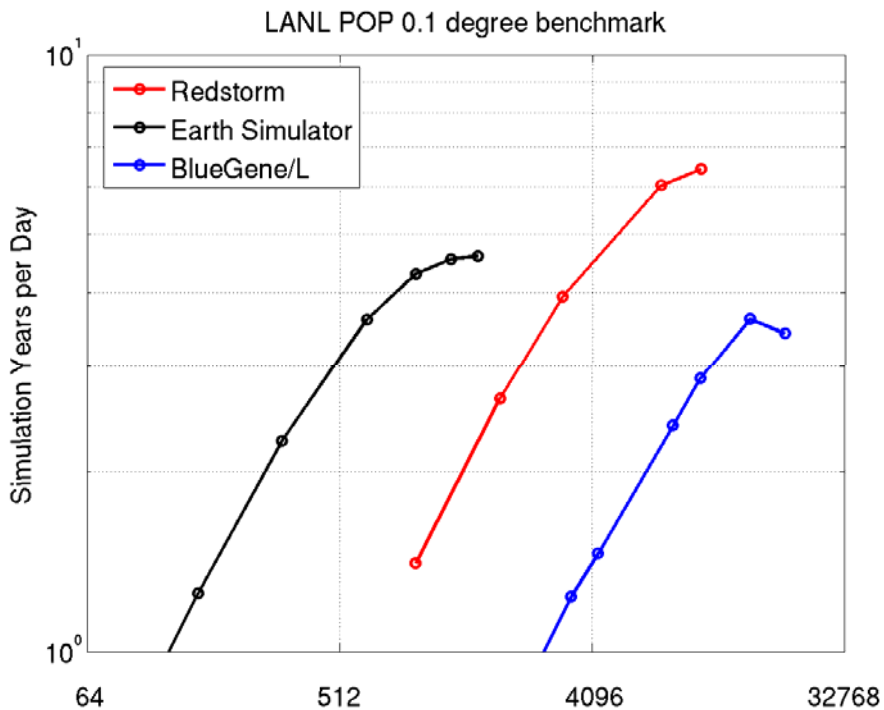
POP Benchmarks (1/10 degree Ocean)



Red Storm

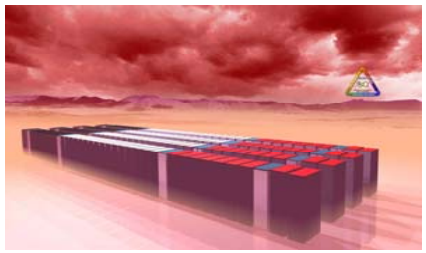
Blue Gene Livermore

Earth Simulator



Sandia National Laboratories

POP Benchmarks (1/10 degree Ocean)



Red Storm

Blue Gene Livermore

Earth Simulator

Real time Sim Time	NCPU	Computer
1 yr/day	700	RS
	3000	BGL
	128	ES
3 yr/day	2000	RS
	8000	BGL
	500	ES
6 yr/day	8000	RS
	Not possible	BGL
	Not possible	ES

Sandia National Laboratories

	Performance Comparison of Some Current Large HPC Systems					
	Columbia (SGI ALTIX)	Earth Simulator (NEC)	ASC Purple (IBM)	ASC Blue Gene Light (IBM)	ASC Red Storm (Cray/Sandia)	Minimum Red Storm Advantage
System Operational Time Frame	2004	2002	2005	2005	2005	
Theoretical Peak (TF)	60.96	40.96	77.82	367.00	43.52	
Number of Processes	10240	5120	10240	131072	10848	
HPL (TF)	51.87	35.86	63.39	280.6	36.19	
HPCC Baseline (Normalized to System Peak TF))	12.95	Results Not Yet Available	77.82	183.50	41.40	None
HPL (TF/TF)	0.72		0.74	0.44	0.80	1.1
G-PTRANS (B/MF)	1,400		7,100	1,800	48,000	6.8
G-Random Access (B/MF)	30.4		17.4	28.7	197.	6.5
G-FFTE (F/MF)	3,530		10,800	11,900	270,000	22.7
G-STREAM Triad (B/MF)	0.309		0.709	0.292	1.053	1.5

Sandia National Laboratories

	Comparison of the Architecture of Some Current Large HPC Systems				
System Characteristics	Columbia (SGI ALTIX)	Earth Simulator (NEC)	ASC Purple (IBM)	ASC Blue Gene Light (IBM)	ASC Red Storm (Cray/Sandia)
System Operational Time Frame	2004	2002	2005	2005	2005
Theoretical Peak (TF)	60.96	40.96	77.82	367.00	43.52
Number of Processors	10240	5120	10240	131072	10880
Processor	Intel Itanium 2 @ 1.6 GHz	NEC Vector @ 500 MHz	IBM Power 5 @ 1.9 GHz	IBM PPC 440 @ 700 MHz	AMD Opteron @ 2.0 GHz
Topology	Hypercube	640 x 640 Cross Bar	Omega Switch 3 Levels	3-D Torus 32 x 32 x 64	3-D Mesh 27 x 16 x 24
Compute Node Operating System	LINUX	UNIX	UNIX	LWK	LWK (compute nodes) and LINUX (Service and I/O nodes)
Programming model	Cache coherent NUMA (+Message Passing)	UMA Vector + Message passing	NUMA Superscalar + Message passing	UMA Superscalar + Message passing	Superscalar + Message passing

Sandia National Laboratories

Petascale Design Goals

- Balanced System Performance - CPU, Memory, Interconnect, and I/O.
- Usability - Functionality of hardware and software meets needs of users for Massively Parallel Computing.
- Scalability - System Hardware and Software scale, single cabinet system to >50,000 processor system.
- Reliability - Machine stays up long enough between interrupts to make real progress on completing application run (at least 50 hours MTBI), requires full system RAS capability.
- Upgradeability - System can be easily upgraded with a processor swap during expected lifetime.
- Space, Power, Cooling - High density, relatively low power system.
- Price/Performance - Excellent performance per dollar, use high volume commodity parts where feasible-- drives us to Intel or AMD.

High Level Architecture

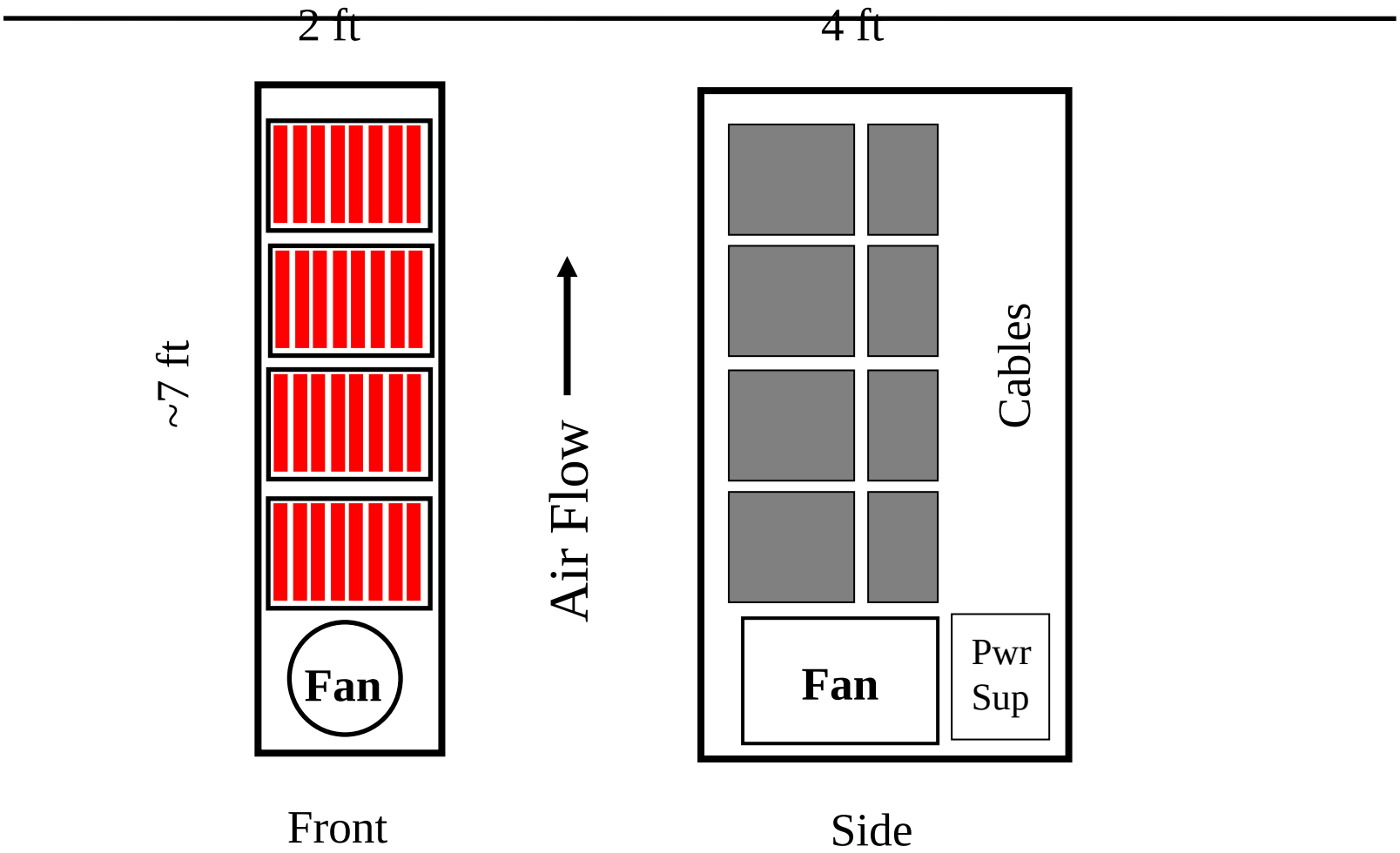
- True MPP, designed to be a single system from both a hardware and a system software perspective.
- Distributed memory MIMD parallel supercomputer.
- Fully connected 3-D mesh (probably a torus) interconnect. Each processor socket has a high performance connection to the primary communication network.
- Hardware is functionally partitioned between service and I/O nodes, compute nodes, and RAS and system management.
- Partitioned OS - Full OS (Linux) on service and I/O nodes, LWK OS on compute nodes, full OS on system management and RAS workstation, and embedded OS for RAS nodes.
- Separate RAS network used for system management and monitoring,

High Level Architecture

- Source based routing in primary communication network.
- All nodes are diskless.
  - There is a single boot image for each software partition.
  - Boot images are loaded from a boot node over the primary communication network.
  - Service and I/O nodes have a shared root file system.
  - All I/O is through the service and I/O partition
- Vertical blades and a “passive” backplane are used.
- Cabinets are nominally 2 ft wide by 4 ft deep by ~7 ft high.
- System is air cooled, although, depending on power per cabinet a air water heat exchanger might be considered.

  Bottom to top air flow.

Cabinet Layout



Sandia National Laboratories

Network Interface Chip (NIC)/Router Characteristics

Smart NIC

- **Off-load most message passing work - OS bypass**
- **Custom hardware to support high message throughput**
- **Global address support for remote memory access**

7 port router - Full crossbar

Source based routing

Circuit Switched

4 virtual networks

- **Large MPI messages**
- **Small MPI messages**
- **Load/Store remote memory access**
- **Control messages**

NIC/Router Performance

- Link Bandwidth ~ 30 GB/s in each direction for each port
- Processor to NIC Bandwidth ~ 30 GB/s in each direction
- MPI Latency ~500 ns for neighboring node
- Router Latency - <20 ns per hop
- Message Throughput - >15M messages per second each direction

System Address Map

Cache coherent boundary is the node boundary including the processor socket associated NIC.

The node size will not be more than

4--6 processor sockets (one CPU board). It may be a single processor socket.

Global address map will be provided through the NIC.

Topology

- The system topology is a fully connected 3-D mesh.
- The exact topology depends on the system size and the final cabinet and card cage configurations.
- We expect that 128--192 sockets per cabinet will be a good design point

Some Possible System Topologies

A Peak Petaflops system - ~16,500 processor sockets, each socket is 64 GF/s

For 128 processor sockets per cabinet

- 26x20x32 (X,Y,Z)
- 16,640 processor sockets
- 5 rows of 26 cabinets + Service and I/O cabinets

For 192 processor sockets per cabinet

- 22x24x32 (X,Y,Z)
- 16,896 processor sockets
- 4 rows of 22 cabinets + Service and I/O cabinets

For sustained Petaflops, 4 times as big a system is needed!

System Partitioning

- Service and I/O nodes - Full OS
  - Login Nodes
  - Boot Node/s
  - I/O Nodes
  - Other Dedicated Nodes - Allocator, Batch System, Meta-Data, Apps, etc.
- Compute nodes - LWK OS
- RAS System
  - Hard partition
  - Separate, dedicated network
- System is configured at boot time from tables that determine what nodes will be booted with which OS.

Reliability, Availability, Serviceability (RAS)

- Single system image for system management
- Dedicated RAS network
- System designed to prevent single point failures
  - redundant hardware for components that have high failure rates
- Real-time monitoring and tracking of all significant hardware components
- Tracking of all errors, recovered and non-recovered

Reliability, Availability, Serviceability (RAS)

- Automatic failover for all disk controllers and I/O nodes
- 100 hour MTBI for system (~16,500 processor sockets)
- 50 hour MTBI for applications on full system
- Recovered Bit Error Rate (BER) less than 1 bit in 10^{22}
- Full system boot in under 15 minutes
- Hot swapping of failed hardware components
- Field Replaceable Unit diagnostics

Topology Reconfiguration Under Error

RAS system provides feedback to Allocator to remove failed nodes from available pool of nodes.

RAS modifies source based routing algorithm to route around failed NIC/Router chips or failed interconnect boards. Failure of multiple nodes can be tolerated.

Failure of multiple interconnect boards or full cabinets is problematic.

The latter will require a system shutdown for repair.

System Software

- OS Partitioning
  - Service and I/O nodes - Full service OS (LINUX)
  - Compute Nodes - Light Weight Kernel (LWK) OS
  - SMW - Full service OS (LINUX)
  - RAS Nodes - Embedded or Real-time like OS (LINUX)
- Service side App support
- Heterogeneous App support
- Parallel file system, Unix File System, TCP/IP, NFS

System Software

- System resource sharing - spacing sharing of compute nodes, time-sharing and space sharing of service nodes
- Standard libraries - I/O, Math, MPI
- Batch and interactive processing - Include a batch processing tool
- Tools - Compilers (Fortran, C, C++), Debugger, Performance Monitor, etc.
- System administration - Resource accounting, user access control, etc.

Mechanical Architecture

- Board, Box, Cabinet, Complex (multi-cabinet) Architecture
  - Custom, but simple, boards
  - Custom packaging within an industry-standard cabinet
  - COTS parts except for NIC/Router chip and VRMs
  - All high-value parts are socketed, not soldered
  - Physical scalability

Mechanical Architecture

Cabling

- **Optical**
- **Very simple cabling**

Thermal Architecture/Hierarchy

Base design: air cooling with Bottom to top air flow -

CFD analysis must be done on a board level, a card cage level, a cabinet level, and a full system level

Intercoolers are possible-- need to analyze cost of manufacturing and ownership

Timeframe and Cost

First customer ship is possible in to early 2009 --based on Intel or AMD Architecture.

Cost: First system will cost O($100--125M) plus R&D costs [O($50M)]