

Cluster Computing

Chip Watson

Jefferson Lab – High Performance Computing

Acknowledgements to

Don Holmgren, Fermilab, USQCD Facilities Project

Jie Chen, Ying Chen, Balint Joo, JLab HPC Group

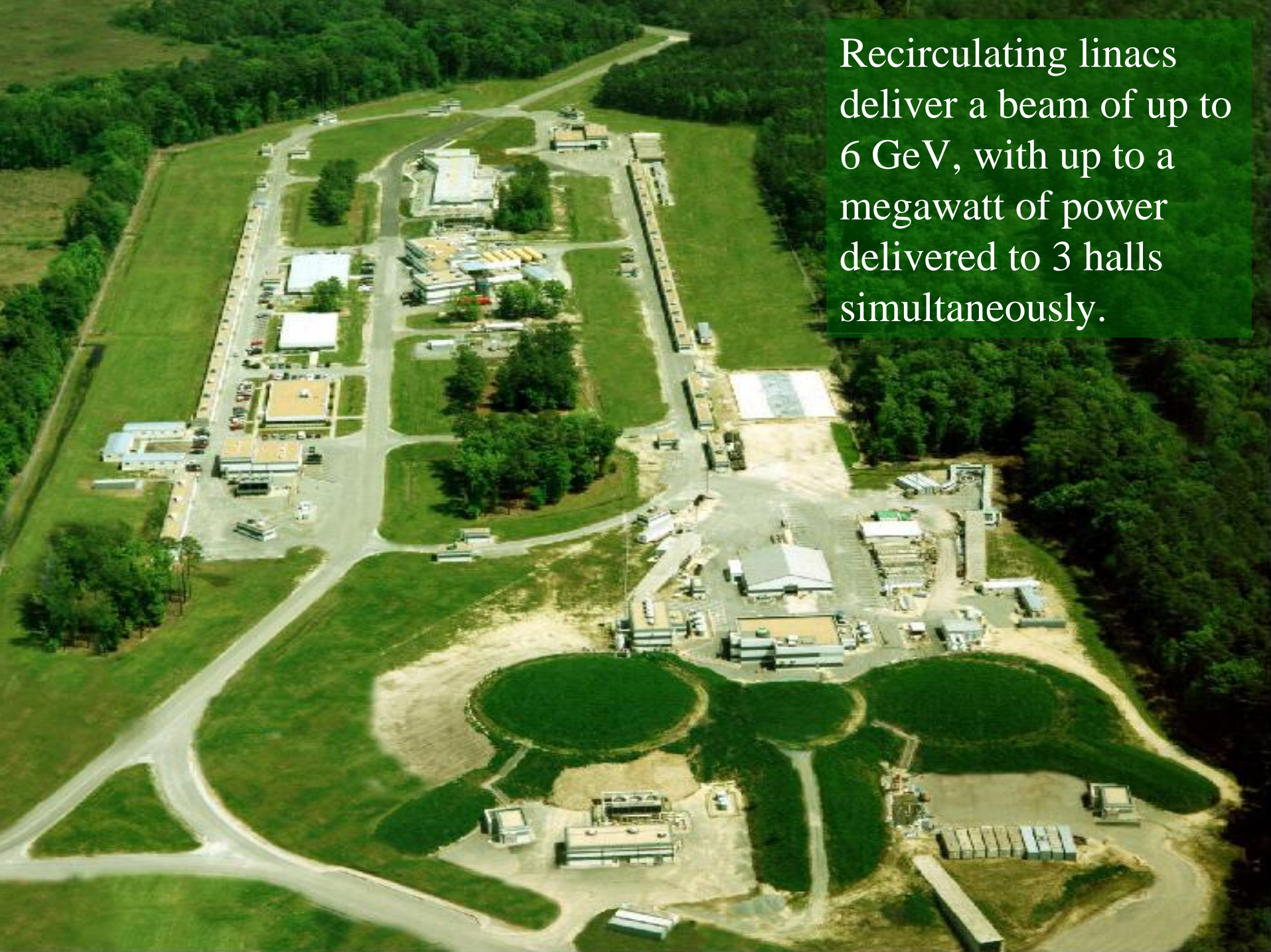
Distributed Computing (or, how this relates to controls)

The problem:

- 256+ computers, tightly coupled
- 50-100 Hz rep rate
- 2 GBytes / sec data rate
(a million waveforms a second)

Introduction

- Jefferson Lab is the premier Nuclear Structure laboratory in the world, with an international user community of over 1200 researchers from roughly two dozen countries.
- Its unique capabilities for exploring and elucidating the quark structure of matter are being used to test the validity of the Standard Model of the strong nuclear force.
- With experiments in three halls focusing on such fundamental topics as quark confinement, the proton spin crisis, and gluon excitations, Jefferson Lab acquires as much as a terabyte of experimental physics data per day.

An aerial photograph of a large industrial or research facility. The facility consists of numerous buildings of varying sizes, some with flat roofs and others with gabled roofs. There are several large parking lots filled with vehicles. The facility is surrounded by green grass and trees. A prominent feature is a large, circular green area in the lower center, which appears to be a landscaped area or a small park. The overall layout is organized and well-maintained.

Recirculating linacs
deliver a beam of up to
6 GeV, with up to a
megawatt of power
delivered to 3 halls
simultaneously.

Theory and Experiment

- The internal structure of the nucleon is a defining problem for hadron physics just as the hydrogen atom is for atomic physics. Observing that structure is the experiments' goal.
- Quantum ChromoDynamics is the fundamental theory of how quarks and gluons within the nucleus interact.

Theory and Experiment

- "The only known way to solve ... Quantum Chromodynamics (QCD) is a numerical solution on a discrete space-time lattice. Quantitative solution of QCD is essential to extract the full physics potential of present and proposed experiments at frontier nuclear physics facilities."
- These calculations are enormously difficult, requiring **teraflops-years** for the next set of problems.

Clusters and HPC

- Parallel Computing
 - We can't buy a fast enough (single processor) computer, so we need to use multiple CPU's
 - Take a "divide and conquer" approach to science
- The motivation to use clusters is two-fold:
 - Assemble a large computational resource
 - Achieve teraflops performance without spending > \$4M

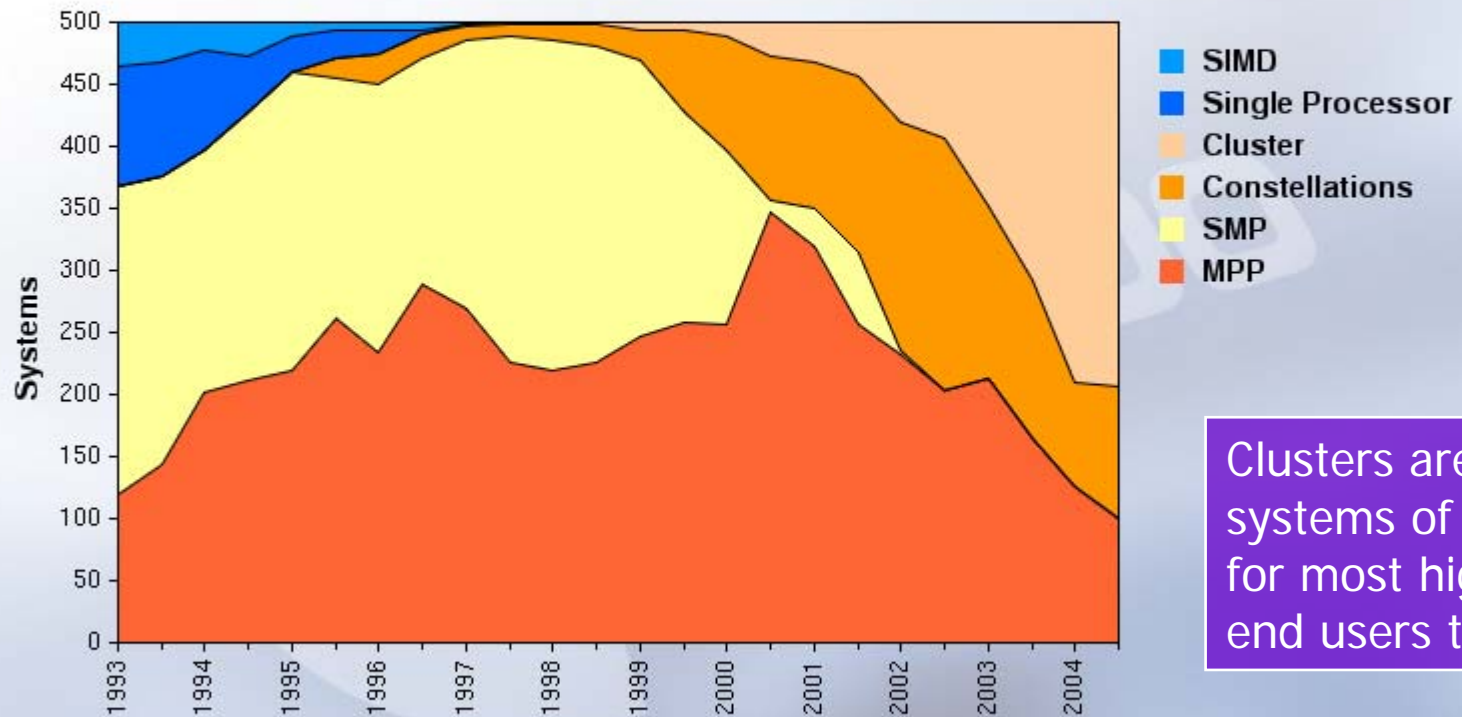
Clusters and HPC (2)

■ Relevant facts

- Moore's Law delivers increases in processor price performance of the order of 60% per year
- A high volume market has driven the cost of CPUs and components extremely low, with newer components available every few months, allowing increased capability each year at constant investment
- Home video gaming has encouraged the development of multi-media extensions; these small vector processors on commodity processors can deliver super-scalar performance, exceeding 8 Gflops sustained on a Pentium 4 – scaling this to a cluster is the challenge!
- Cluster interconnects are maturing, allowing ever larger clusters to be constructed from semi-commodity parts

Commodity Clusters

- Why commodity?
 - rapid incorporation of latest technology
 - low cost driven by mass market
 - mature, productive environment for science
- High end capacity, not extreme capability
 - goal is most science for a given investment
 - 2nd tier of Branscomb's pyramid meets most needs
 - scalability (capability) at any cost is not good for science
 - goal is not a perfectly balanced architecture (although that is nice);
 - multiple machines is a valid solution
 - most of the largest machines run space- or time-partitioned anyway
 - extended parameter studies run easily on multiple clusters



Clusters are the systems of choice for most high end users today

Challenges to Cluster Computing

- Clusters (as compared to large SMPs) face certain architectural challenges:
 - Distributing work among many processors requires communications (no shared memory)
 - Communications is slow compared to memory R/W speed (both bandwidth and latency)
- The importance of these constraints is a strong function of the application, and of how the application is coded (strategy).

Standard Objections

- Too much heat
 - Power and A/C are just money; so include that in the science/dollar calculation (<20% effect)
 - Mobile computing & home PC market will help to constrain power use into the future (Intel is reacting now)
- Not scalable
 - Few single applications need to (or can afford to) scale above \$1M - \$2M per application instance sustained year round (US LQCD is running multiple instances in 2005)
 - clusters today easily exceed this threshold
 - scaling is slightly non-linear, but that is just another cost in the science/dollar calculation

Cluster Design Approach

- Optimize for a single (large) problem
 - e.g. LQCD running asqtad action OR LQCD running dwf action
 - If you try to cover all of the possible computer parameter space, (as in dealing with many science fields) you have wasted money on every problem that runs... better to have multiple optimized clusters
- Leverage whatever the market produces
 - SSE = short vector processors
 - market volume wins (cost matters)

Cluster Parameters

■ CPU

- raw floating point performance (clock speed, dispatch rate)
- cache size & latency
- memory bandwidth, latency, size

■ SMP capability

- cache coherency
- memory: shared bus, crossbar, NUMA

■ Network fabric

- bandwidth, latency
- topology: blocking behavior or bottlenecks

Most Important for LQCD

Parameters:

- Raw floating point performance (need, but can't afford, petaflops)
- Memory bandwidth (high bytes / flop)
- Network bandwidth (to scale to teraflops)
- Network latency (scalability)

Design goal:

highest sustained science performance for \$1M - \$4M

Irrelevant goals:

% peak, node cost, link cost, ...(any other single metric)

Best choice today

Intel IA-32

■ Advantages:

- SSE allows 4 flops / cycle, single precision
- ~10 GFlops sustained on tiny LQCD kernel, in-cache SU(3) algebra (very high efficiency in L1 cache)
- huge volume market, low price per node

■ Disadvantage:

- Memory bandwidth can't keep up
- Dual processors (Xeon) use a shared bus (useless)

■ Dual core now under evaluation

- May deliver more sustained flops / dollar at same bandwidth

Understanding the Requirements: LQCD Behavior

- Regular 4D problem (space time)
 - some problems use additional pseudo dimensions for numerical convergence, e.g. 5D domain wall action
- Periodic boundary conditions
 - Maps well onto a mesh (torus) machine topology
- Characterization of LQCD algorithms
 - Sparse, banded diagonal matrix inversion; each element is an SU3 complex matrix, increasing the floating point cost
 - Algorithm splits into forward, backward phases: rotation on mesh; latency tolerance: 80% overlap possible
 - Frequency of global operations (barriers) is low for some numerical approaches

Cluster Architectures

Different network architectures suit different applications.

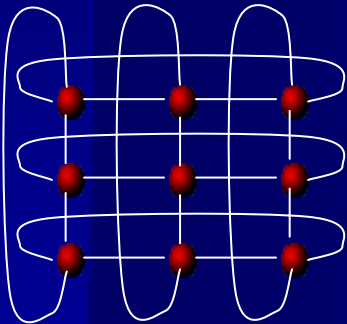
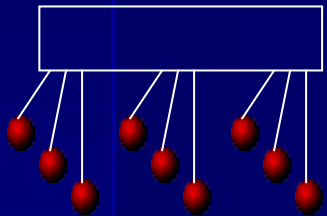
– Switched network:

- General parallel computing platform
- Any-to-any communication paths

– Multi-dimensional Mesh Connections (torus):

- Good platform for nearest neighbor communications.
- Potentially higher total bandwidth per node

– Lattice QCD requires primarily nearest neighbor and some global communication.



Cluster Architectures (2)

■ Switched

- **Ethernet**: modest bandwidth, high latency, low cost
- **Myrinet**: better bandwidth, lower latency, semi-commodity = moderate cost
- **Infiniband**: very good bandwidth, lower latency, emerging technology = moderate cost, falling rapidly
- **Quadrics**: very good bandwidth, low latency, high cost

■ Mesh

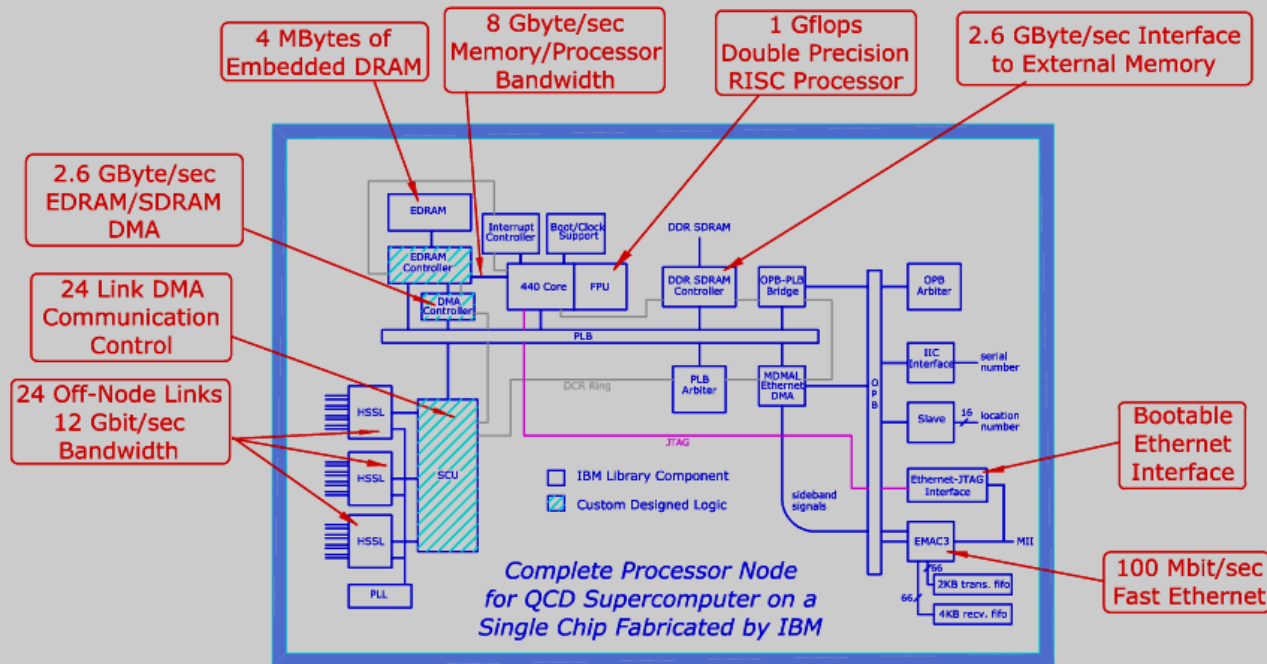
- **Eliminates the cost of the switch**; Achieves high aggregate bandwidth through multiple links
- Still suffers from ethernet's higher latency
- Less flexibility in configuring the machine

LQCD and Mesh Machines

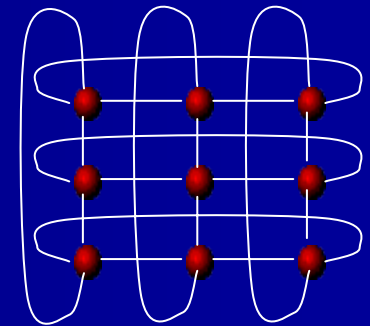
LQCD has a history of using mesh architectures

- QCDSP (DSP based, w/ custom I/O chip)
- APPE (Italian-German design)
- QCDOC (QCD On a Chip) – balanced design, but frozen technology

QCDOC ASIC DESIGN



Mission-critical, custom logic (hatched) for high-performance memory access and fast, low-latency off-node communications is combined with standards-based, highly integrated commercial library components.



- Each node takes a 4D sub lattice, a portion of the problem.
- Lattice must be a multiple of the number of nodes in each direction.

Designing a Cluster

The overriding metric is science per dollar, or \$/Mflops

- Peak is irrelevant
- Percent of peak is irrelevant
- Petaflops are needed (eventually), so cost matters!

Intel IA-32 is a good building block

- SU3 matrix multiplies can be mapped onto the SSE registers and instructions
- Core 2 Duo achieves > 3 flops / Hz / core for problems resident in L2 cache.
- Memory bandwidth is a severe constraint
 - Dual Xeons do not have double the performance of single Xeon for non cache resident problems (Opterons & NUMA does better)
- I/O (parallel computing) is also a strong constraint (chipsets)

SciDAC Prototype Clusters

JLab has been building a sequence of cluster prototypes which allow us to track industry developments and trends, while also deploying critical compute resources.

Myrinet + Pentium 4

- 128 single 2.0 GHz P4 (Summer 2002)

Gigabit Ethernet Mesh + Pentium 4

- 256 (8x8x4) single 2.66 GHz P4 (Fall 2003)
- 384 (8x8x6) single 2.8 GHz P4 (Fall 2004)

Infiniband + Pentium 4

- 256 single 3.0 GHz P4-D (Winter 2006)

2002: 128 Node Cluster @ JLab



Myrinet

**2 GHz P4
1U, 256Mb**



Pushing \$/Mflops Down

- Moore's Law helps, but what if I/O is 50% of the cost? Myrinet in 2002, and Infiniband in 2003-2004 were high performance, but also high cost
- GigE NICs fell in price as they became commodity, but large high performance switches were still NOT commodity
- Dual gigE cards made 3D meshes possible and cost effective...

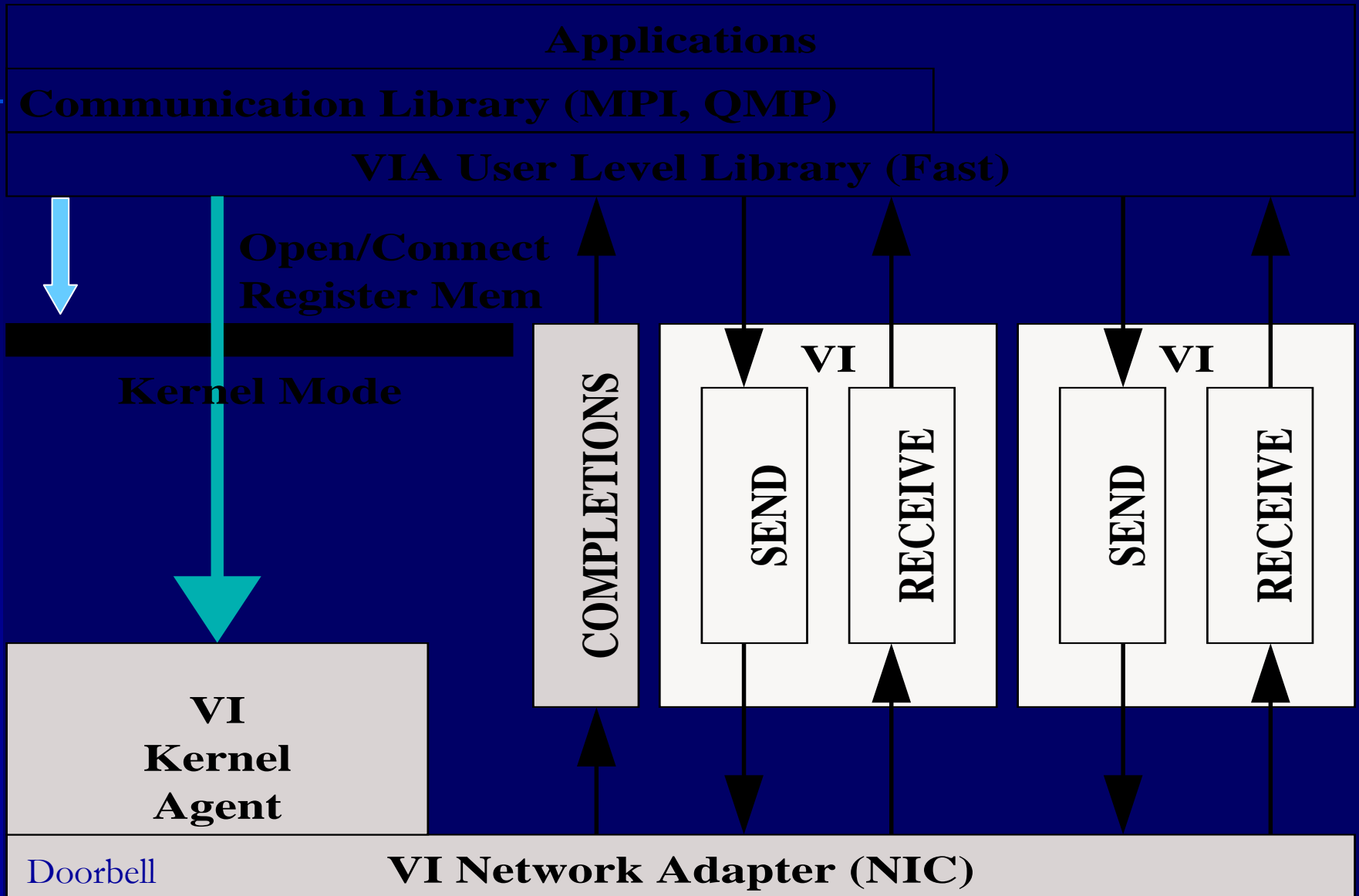
What about GigE?

- Cost in 2003:
 - Myrinet: \$1400 per node (switch + NIC) up to 256, \$1800 for 1024
 - GigE mesh: \$450 per node (3 dual gigE cards, 1 per dimension)
- Needed efficient user space code:
 - TCP/IP consumes too much of the CPU

Communication Software

- User level networking (ULN)
 - Remove OS from critical path of sending/receiving
 - Better latency and higher bandwidth
 - Vendor supplied: GM
 - Research software: FM, Unet
 - Industrial Standard: VIA
 - Sample Implementations: M-VIA, Berkeley VIA

VIA Architecture



Assembly coding of inner numerical kernels

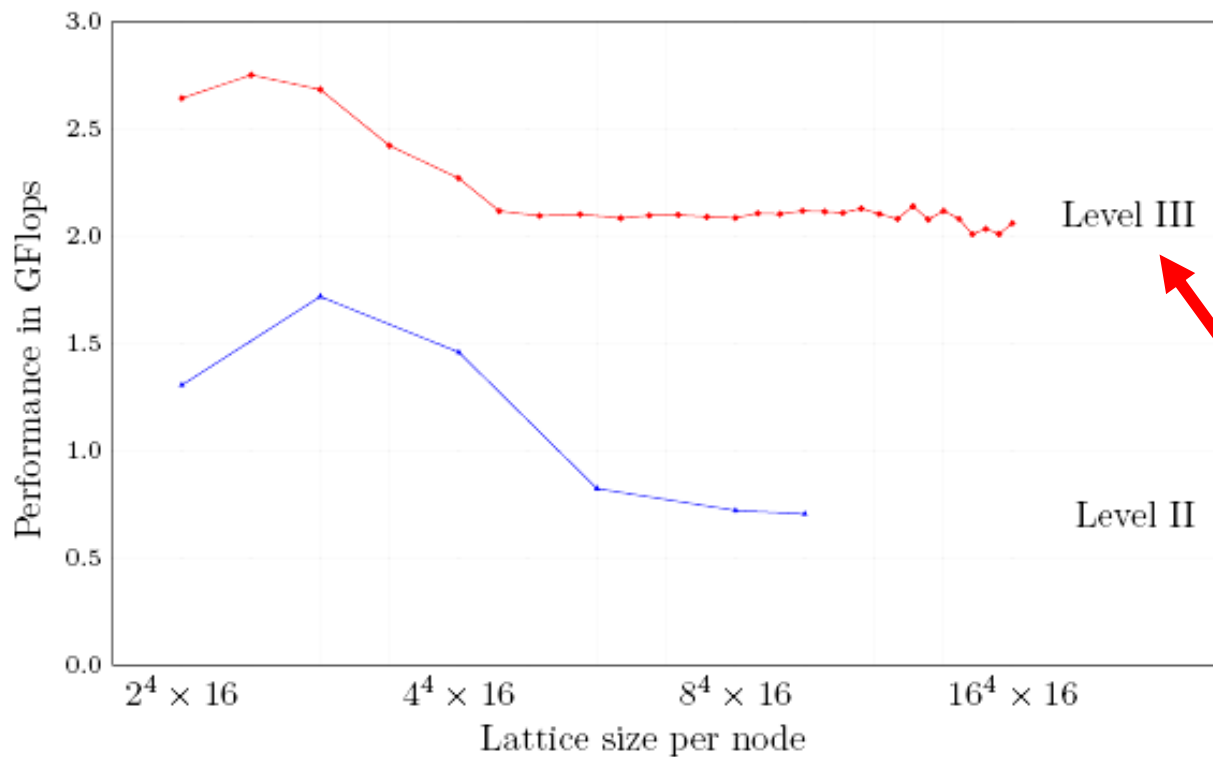
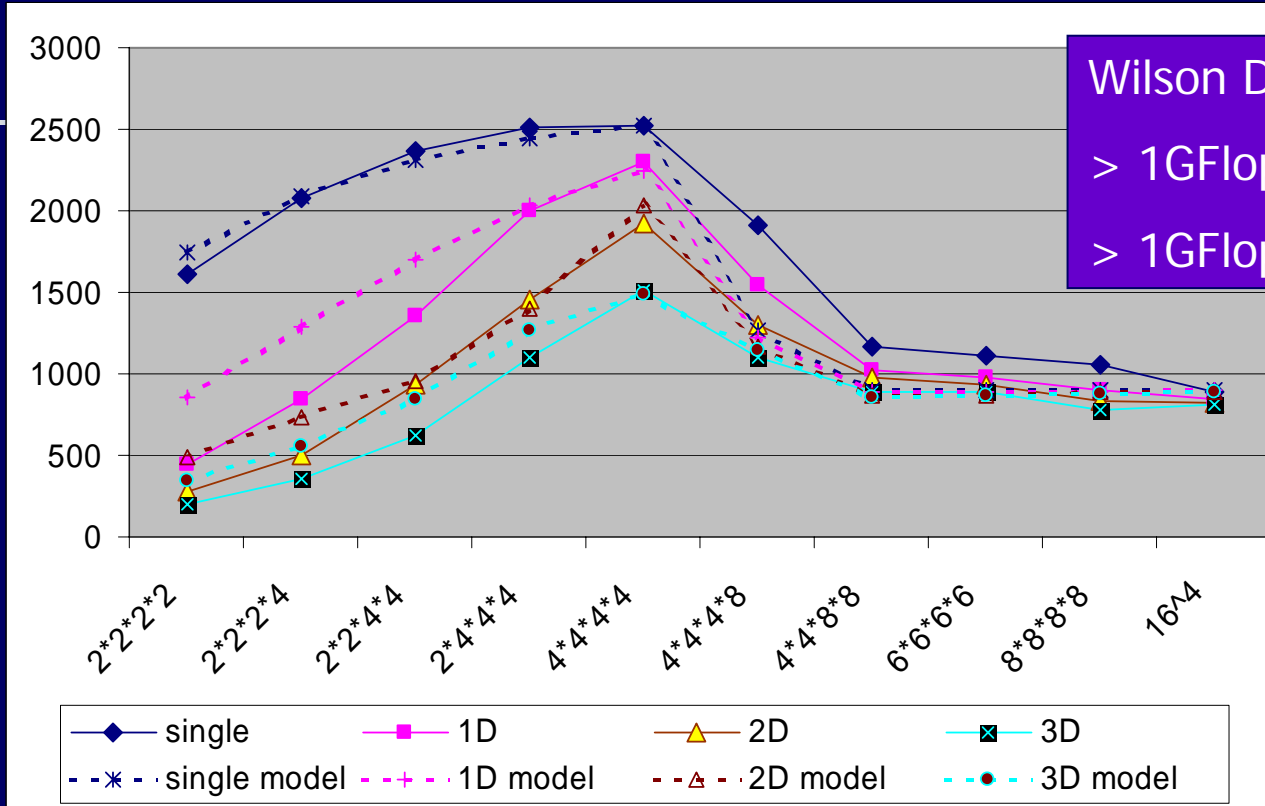


Figure 1: Comparison of DWF CG performance on a single 2.66 Ghz P4 node as a function of lattice size for $L_s = 16$ using Level II code in Chroma and using Level III SSE code optimized to minimize memory bus traffic.

SciDAC software optimizations: goal was best performance, memory bandwidth bound; vectorized in 5th dimension

Modeling 2.0 GHz P4 + Myrinet Cluster Performance : 2002



- 2 GHz, 400 MHz fsb (~1/2 of today's chips)
- Model includes CPU in- and out-of-cache single node performance, PCI and link bandwidth, latency, etc.
- Moderately simple model predicts cluster performance pretty well.

Case Study 1: 2004

Cost Optimized at < 1\$M

Jefferson Lab 4g cluster

Goal: design a cluster for under \$1M to maximize science.

Approach: work with less expensive network, since extreme scalability is not needed at this investment level.

Solution:

- 3.0 GHz P4
- 5d gigE mesh (collapse to 3d, flexible dimensions)
- lean memory (512 MB)
- low performance disk
- 384 nodes, \$700K
- > 500 GFlops sustained
- \$1.33/MFlops in 2004 for LQCD domain wall fermions

384 Node 2004 GigE Mesh Cluster



SciDAC LQCD prototype

\$1.3 / MFlops DWF, single prec

Historical Performance Trends – Single Node

MILC Improved Staggered Code (“Asqtad”)

Processors used:

- Pentium Pro, 66 MHz FSB
- Pentium II, 100 MHz FSB
- Pentium III, 100/133 FSB
- P4, 400/533/800 FSB
- Xeon, 400 MHz FSB
- P4E, 800 MHz FSB

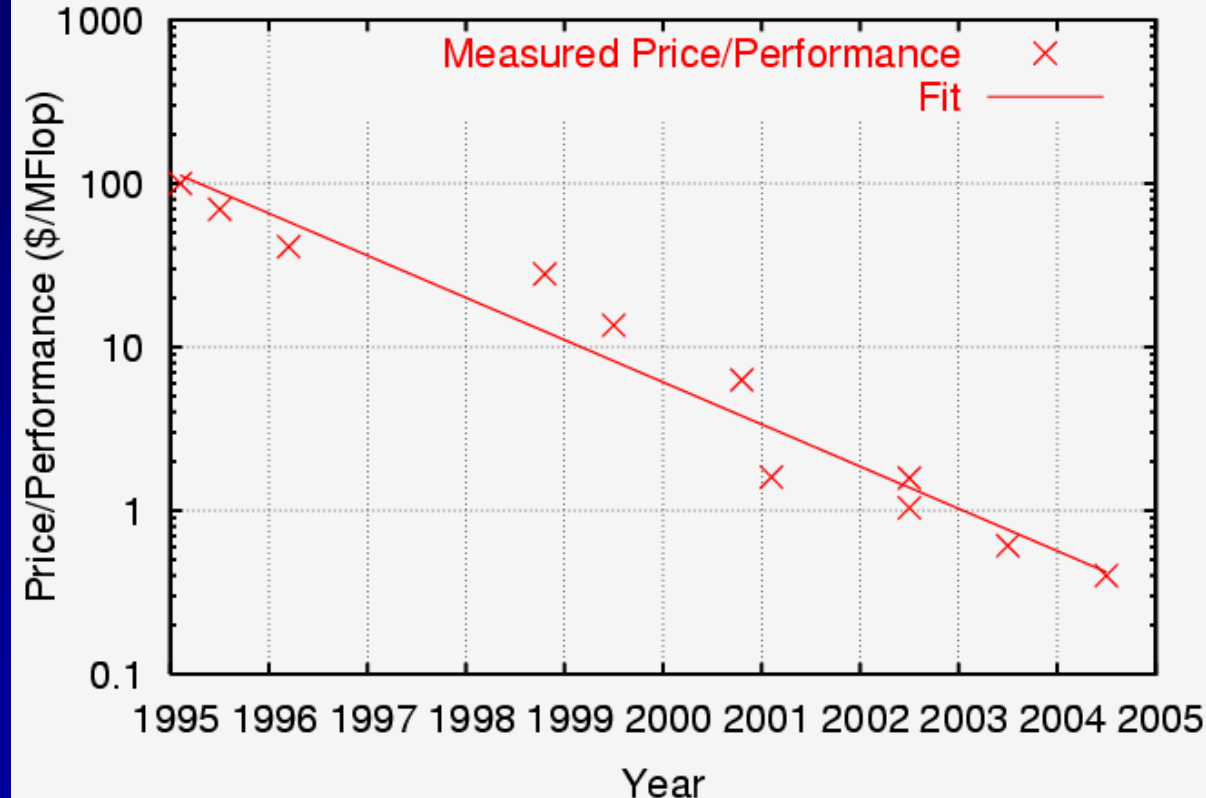
Performance range:

- 48 to 1600 MFlop/sec
- measured at 12^4

Doubling times:

- Performance: 1.88 years
- Price/Perf.: 1.19 years !!

Price/Performance vs Year of MILC Asqtad on Intel x86



Source: FNAL

Major future trends

Will these trends continue? Yes.

- Multi-core: SMP on-chip

- multiplies issue rate / clock cycle
- exacerbates memory bandwidth issues
- 2006: dual, 2007: quad

- Memory bus

- Intel going to 1333 now, 1600 next year, going from shared bus to crossbar (dual bus) in future years
- Opteron NUMA went from DDR to DDR-2, DDR-3 next year

- Cache size

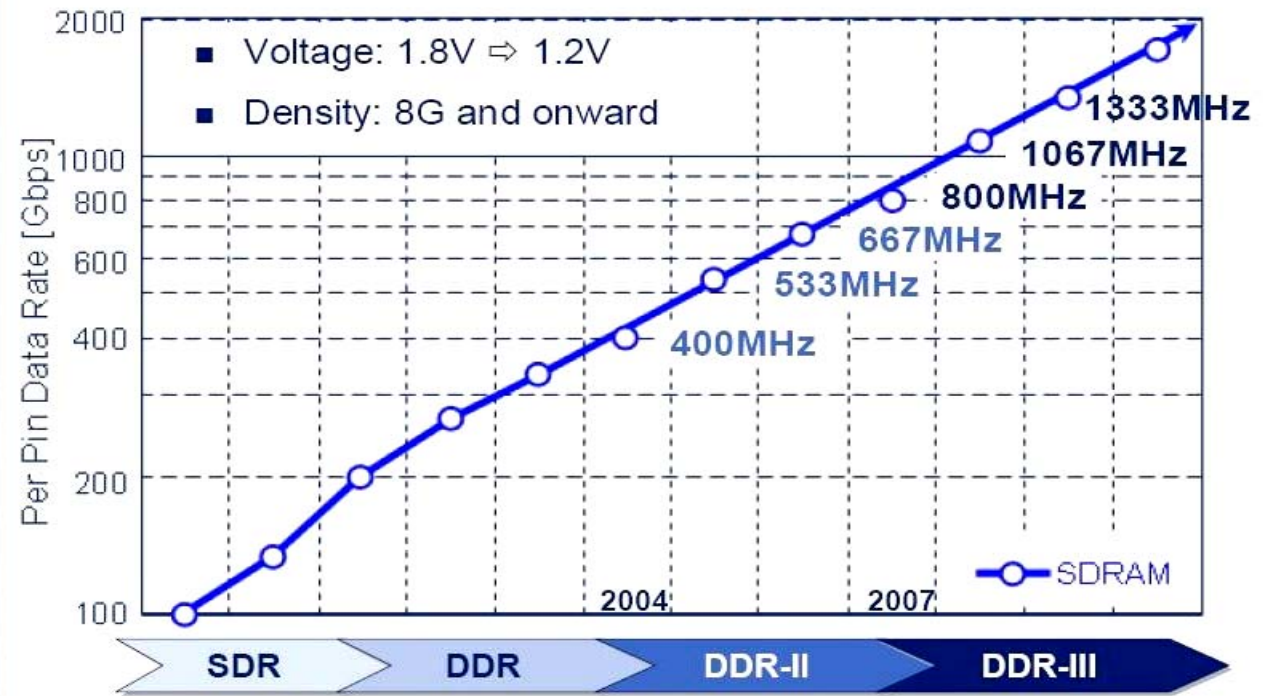
- 4MB today, 8MB next year ...
- is cache resident LQCD on the horizon ?

Memory speed roadmap



Never stop thinking

Roadmap for DDR III



High Speed Links (this decade)

■ Infiniband

- Infiniband 4x delivers 10 Gb/sec bi-directional bandwidth (total 2 GBytes/sec) at very low cost on PCI-Express
- 3 - 4 usec latency (good enough for \$4M machine)
- network cost per node is falling rapidly (now < \$700) and shows promise of falling considerably further
- DDR (20 Gb/s) links are now becoming mainstream
- next? 4x QDR, 12x QDR? (>100 GBytes/sec)

■ 10 gig ethernet

- will put price/performance pressure on Infiniband
- latencies will be higher, but good enough for smaller clusters once price falls

Case Study 2: 2006 Cost Optimized at \$1M

Winter 2006 Infiniband cluster

Goal: design a cluster for ~ \$1M to maximize LQCD.

Approach: use new, inexpensive Infiniband 4x NIC.

Solution:

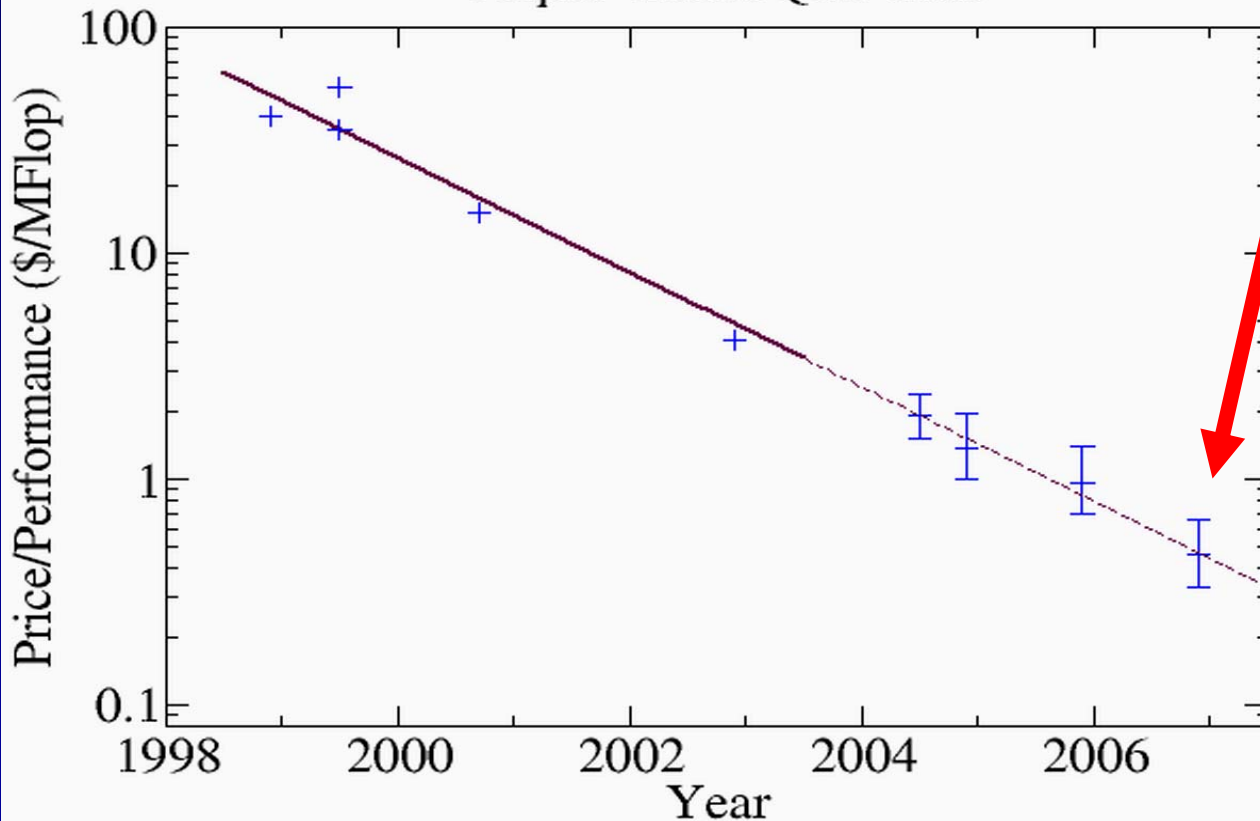
- 3.0 GHz Pentium-D, 800 front side bus
- PCI-Express Infiniband 4x NIC
- 18 nodes / 24 port switch (arranged as $2^4 + 2$, 2:1 oversubscribed)
- 1 GB memory
- low performance disk
- 320 nodes, \$600K
- 0.6 TFlops sustained
- \$1/MFlops
(matches custom machines for single precision in their 1st year!)

Coming soon...

Winter 2007:

- dual core P4
- 1066 MHz FSB ("fully buffered DIMM technology")
- PCI-Express
- Infiniband
- \$1400 + \$600 (system + network per node)
- 4.0 GFlop/node, based on faster CPU, higher memory bandwidth

Cluster Performance Trends
"Asqtad" Lattice QCD Code



High Speed Links - 2

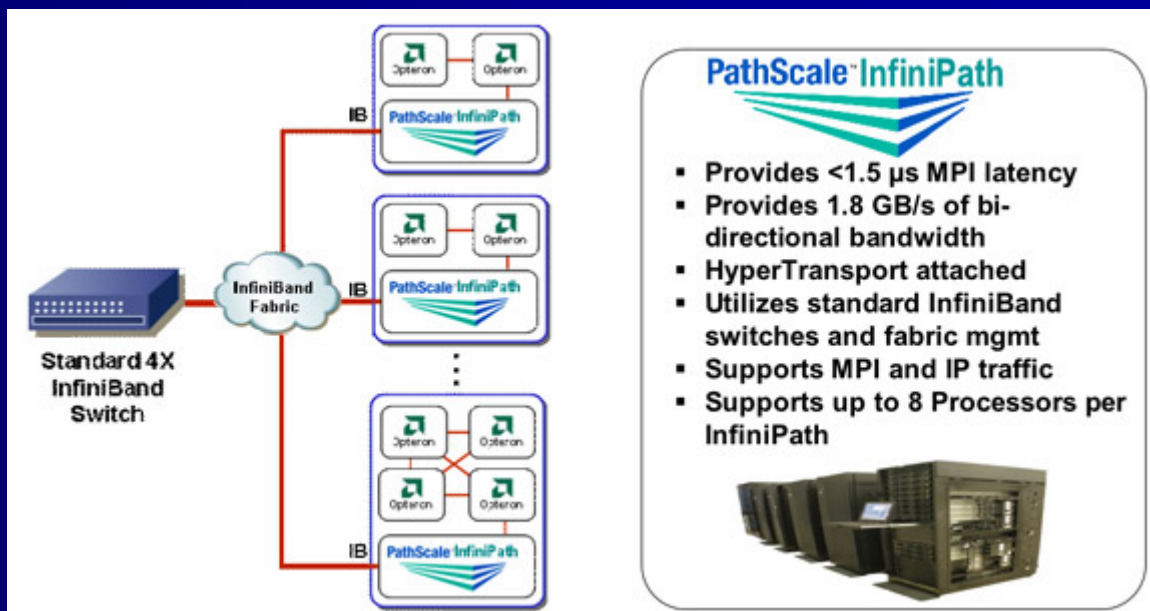
Pathscale Infinipath

- hypertransport to infiniband bridge
- 1.5 usec latency, 1+1 GBytes / second (growing)
- optimized for short messages ($n_{1/2} = 600$ bytes)
- direct from processor to I/O without going through memory!!!
- \$70 / chip

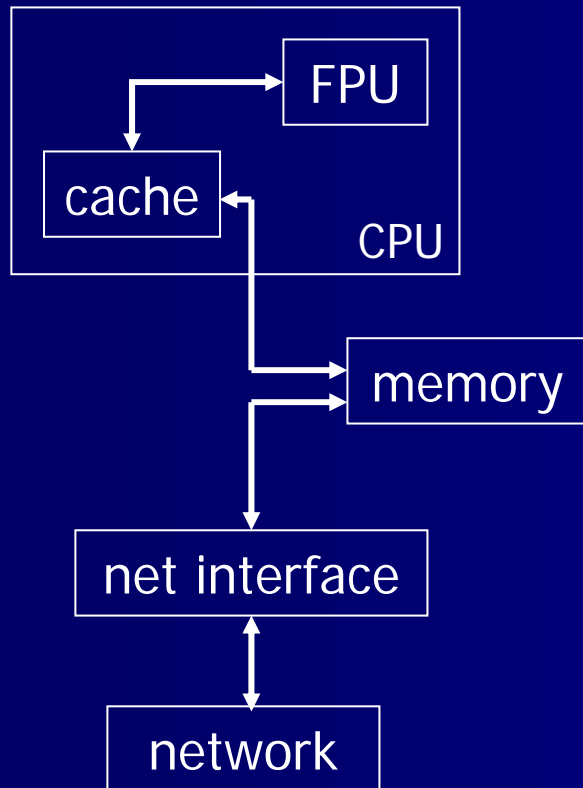
but...

- limited to AMD (today)

Hypertransport is a bus which can link CPUs and I/O devices, and is the native SMP bus for Opterons.

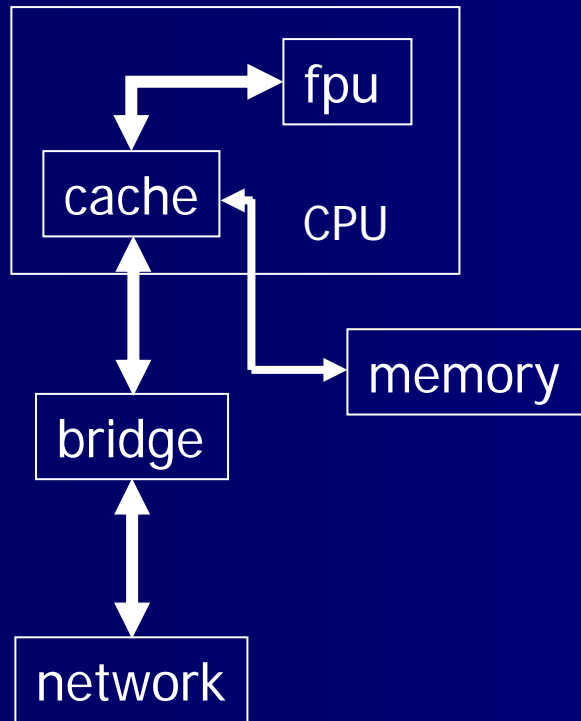


Classical memory bottleneck...



- Even for cache resident problem sizes, message data must cross the memory bus twice
- This limits network performance to $\frac{1}{2}$ memory speed
- If message buffers must be built (scatter / gather), even more memory bandwidth is consumed in I/O

Getting around the bottleneck



- the bridge chip sits in the processor's address space
- data can be written directly to the network, bypassing memory
- for multi-threading chips, one thread could do I/O
- bandwidth limit is now no longer limited to memory speed, and I/O need not consume memory bandwidth

Closely watched Intel alternative

AMD Opteron

■ Advantages:

- NUMA architecture gives linear SMP scaling
- Hypertransport on-chip could use PathScale HCA
- Memory interface scales with chip speed (faster CPU means faster front side bus)

■ Disadvantages:

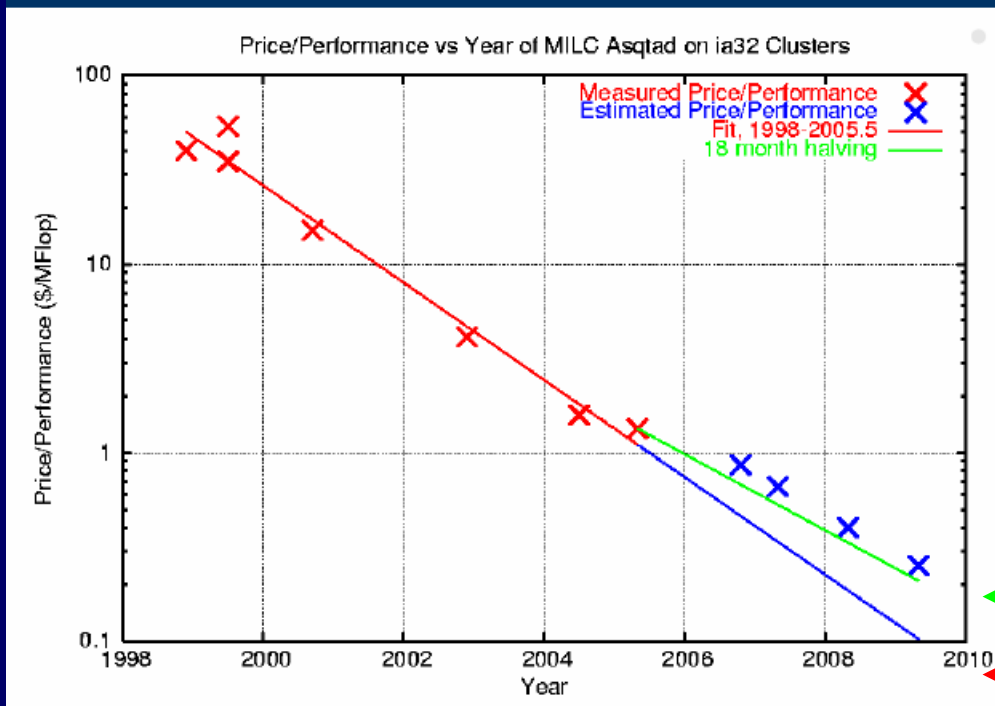
- issue rate (flops/cycle) seems lower
but...
- quad core will help deliver missing flops

Next events in clusters

- Multi-core (2, then 4, then...)
- Faster front side bus (1066, ... 1600)
- NUMA or switched memory buses
 - surely Intel will eventually do something!
- Tight coupling (low latency) of processor to external I/O
 - PathScale is just the first instance

4 Year Extrapolations

Performance Milestones - FY06-FY09



- Measured and estimated asqtad price/performance
 - Blue crosses derive from our "deploy" milestones
 - Green line is Moore's Law with 18 month doubling time

← Conservative

← Trend line

Revolutions

While you can't schedule them, they do happen

- SSE: commodity vector processing
- Multi-core CPUs
- Pathscale (direct bridge from chip to network)

Ideal cluster

4 years from now

Simple (low risk) extrapolations:

- SMP node with NUMA architecture, 4-16 core CPUs
- Bridge chip to Infiniband QDR, <1 usec latency
- On-chip cache sufficient to hold real physics calculations

Result:

- Memory bandwidth no longer as severe a limit
- Clusters of 4K processors, 16K cores, 10's of TFlops, less than \$0.1/Mflops

Application of HPC Technology to Accelerator Controls

- Infiniband fabrics vastly outperform ethernet
 - One tenth latency, 10x bandwidth
 - Creates potential for moving many calculations from real time front end (difficult environment) to Linux hosts
- Cost is still a constraint
 - GigE is "free"
 - Infiniband is ~\$600 / node (less for small systems)

BUT

 - Cost is falling, and could become nearly free as chips get integrated onto motherboards