

Design and evaluation of future computing architectures

*How is R&D in future technologies and architectures informed
by the needs of the scientific applications communities*

R. Scott Studham

Integrate core capabilities to deliver computing for frontiers of science

Develop and evaluate next-generation architectures with industry



[These are the vendors at Falls Creek Falls]

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY

Provide leadership-class computing resources for the Nation



Create math and CS methods to enable use of resources

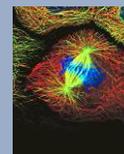
SciDAC
ISICs

Scientific
Applications
Partnerships

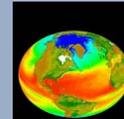
Modeling and
simulation
expertise

Transform scientific discovery through advanced computing

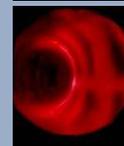
Computational End Stations



Biology



Climate



Fusion

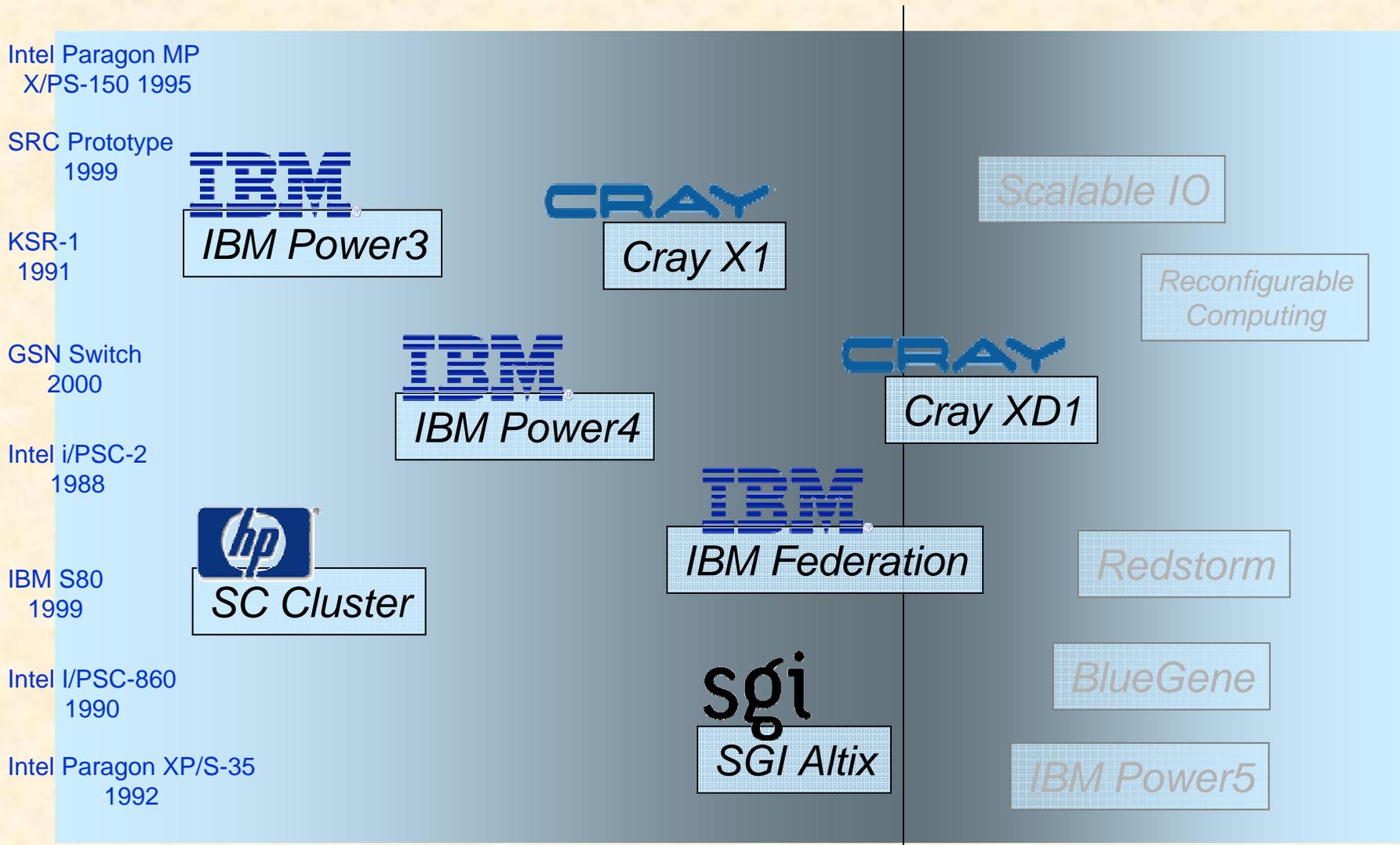


Materials



Industry/
other
agencies

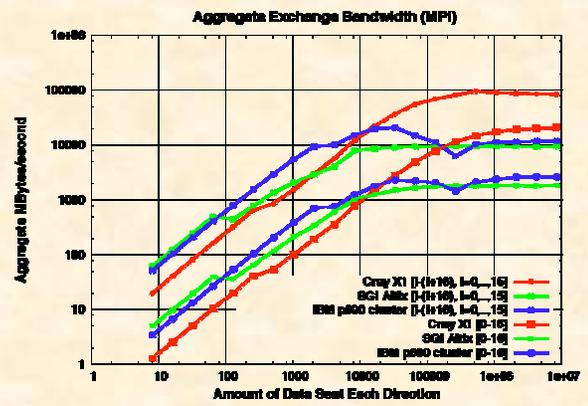
ORNL has a history of platform evaluations



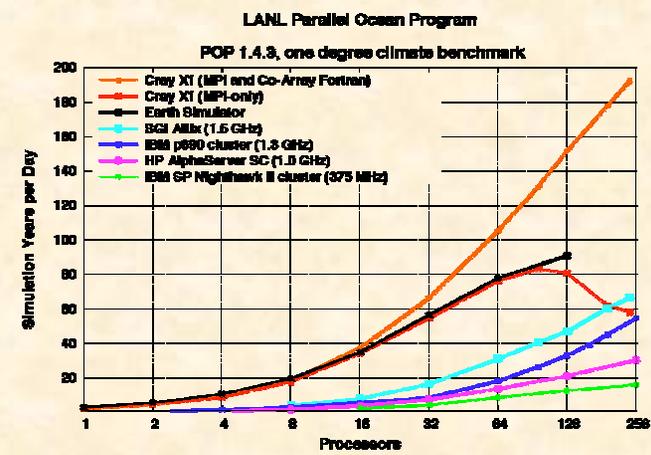
Evaluation Methodology

Open Environment

Microbenchmarks

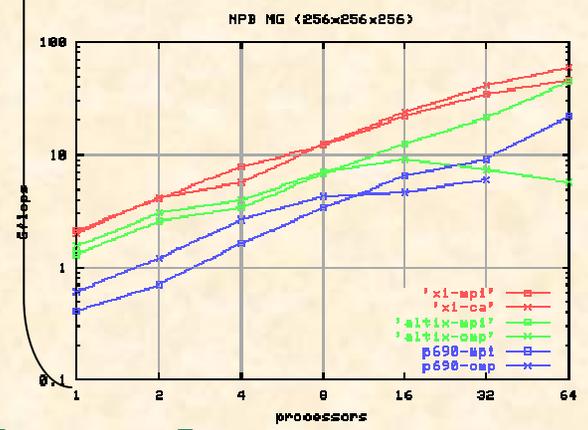


Applications

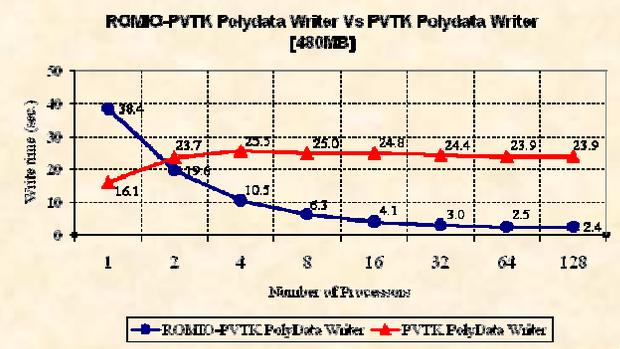


Report

Kernels



System Infrastructure & IO



I only have time to answer one of these questions.

- What are the new and exciting future technologies for microprocessors, interconnects, memory subsystems, storage, languages?
- What information do you need from the application-science communities to improve the effectiveness of your R&D activities?
- How should research in future computing architectures collaborate with capability-computing vendors, application scientists, and computer science and applied mathematics communities?
- What is the typical elapsed time between your “value added” in documenting and evaluating the performance of new architectures and your positive impact on scientific applications? How should this “time-to-market” inform R&D investment decisions in future technologies?
- How can the nation’s S&T agenda best affect the long-term actions and plans of the vendor community?
- How do the requirements of the leadership-class applications impact the R&D agenda in future technologies?
- How do we acknowledge and prepare for potentially revolutionary and disruptive computing technologies, (e.g., optical processors, reversible logic, or quantum computing)?

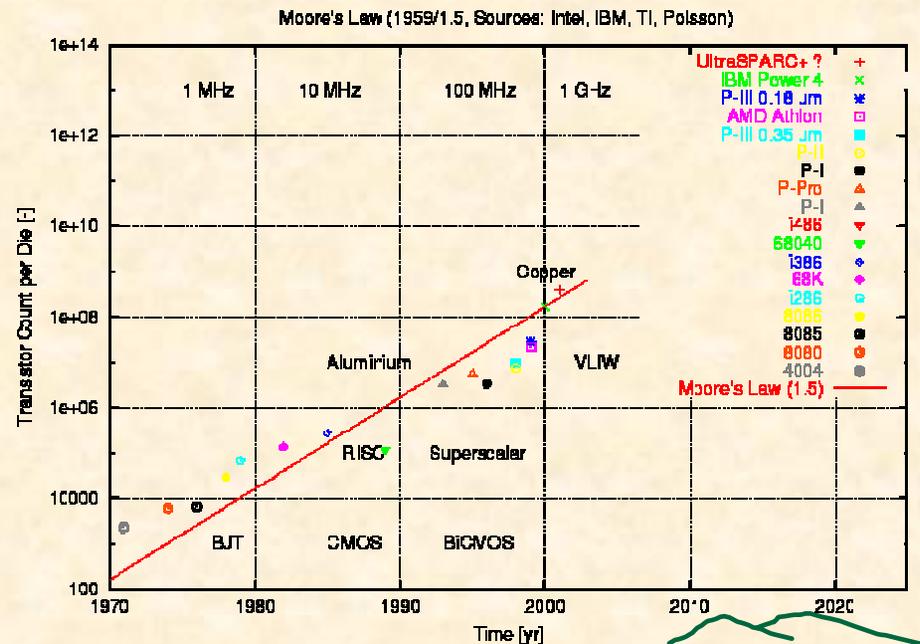
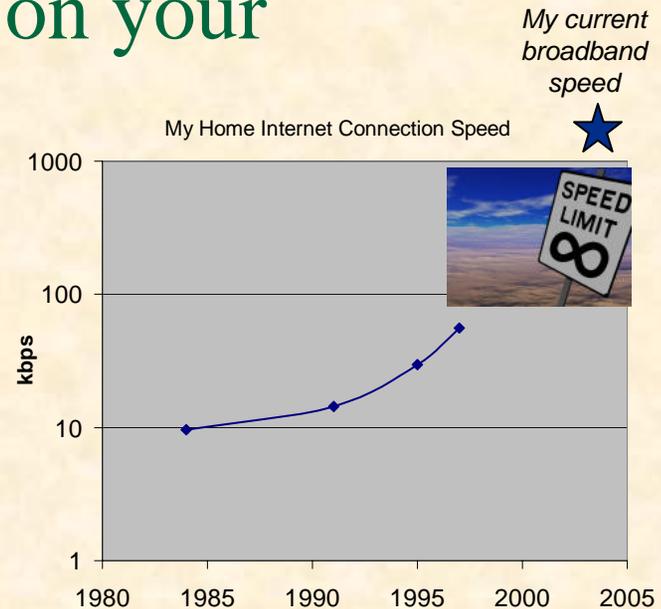
Predicting the future depends on your understanding of the present.

In the mid-90's if you were to ask me what my internet connection speed was going to be in 2004 I would have said 56kpbs because that is the physical limit of the phone line.

Many people have predicted the end of Moore's law in 2010-2020.

1) Ignore the speed limit when telling us what you want.

OAK RIDGE NATIONAL LABORATORY
U. S. DEPARTMENT OF ENERGY



UT-BATTELLE

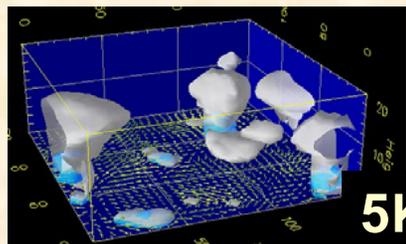
The challenge is to understand domain needs and think of new technologies that will bring them to reality.

When asked, “What would you do with a 100TF supercomputer” the computational climate community answered:

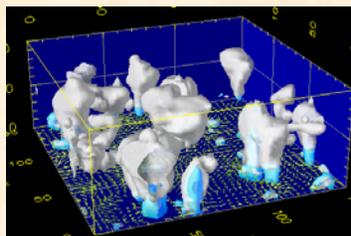
Ensembles with increased resolution and physics [SCALES Report]

When asked, “What are the scientific challenges for the climate community over the next 10-20 years” the climate community answered:

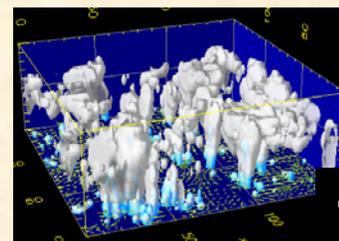
High resolution global cloud models with coupled carbon cycle impacts. [Report on the CCSM Atmosphere Model Working Group Meeting]



5KM



2KM

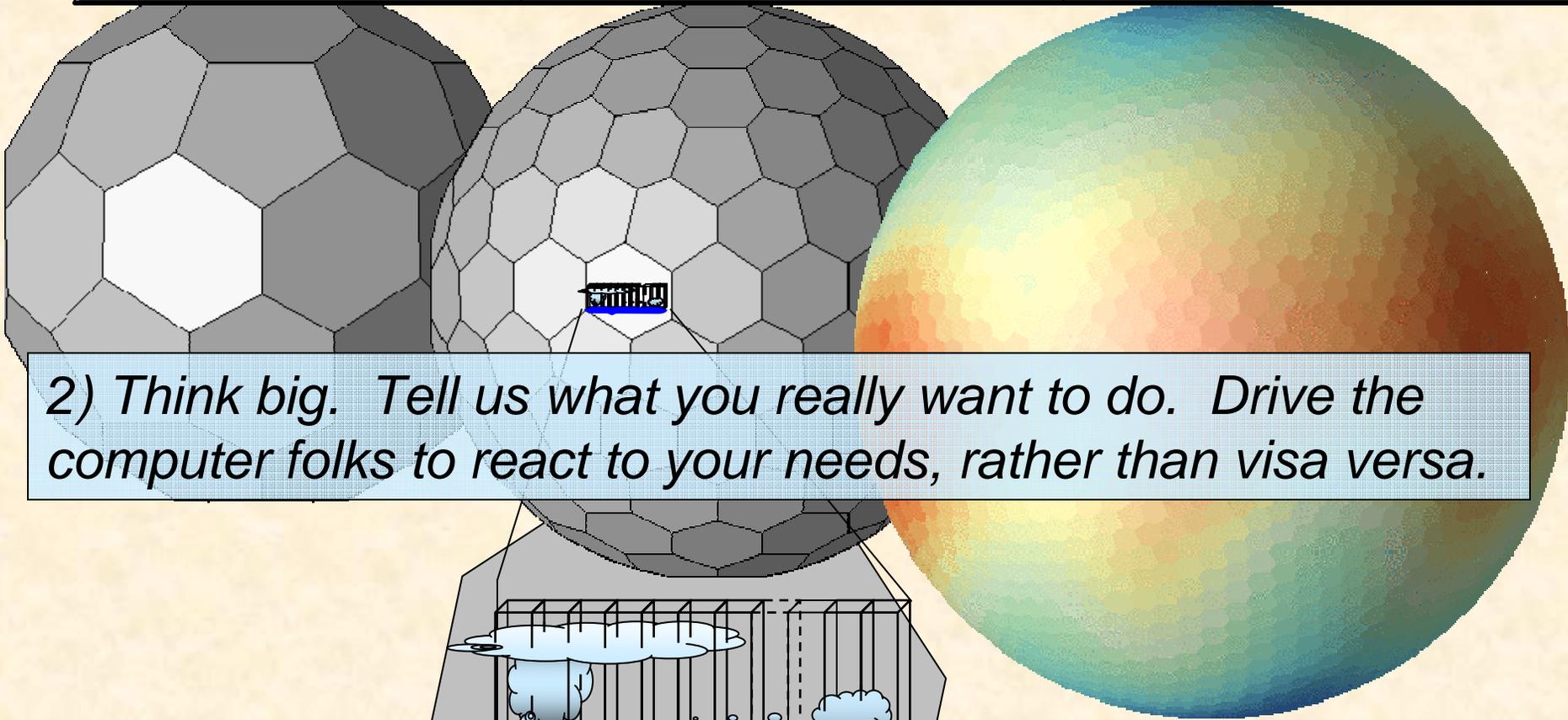


1KM

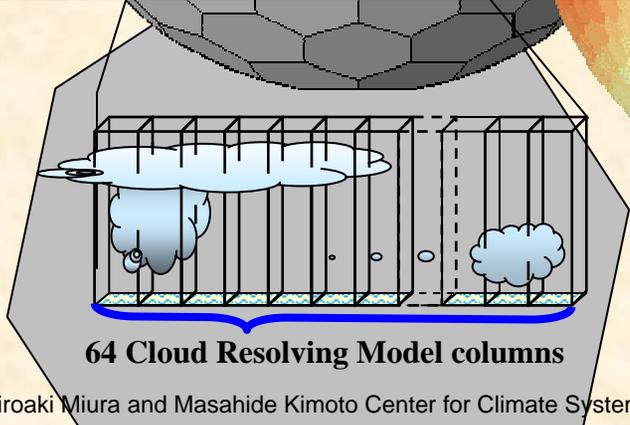
Now

Near Future
(2005-2020)

Far Future
(2020+)



2) Think big. Tell us what you really want to do. Drive the computer folks to react to your needs, rather than visa versa.



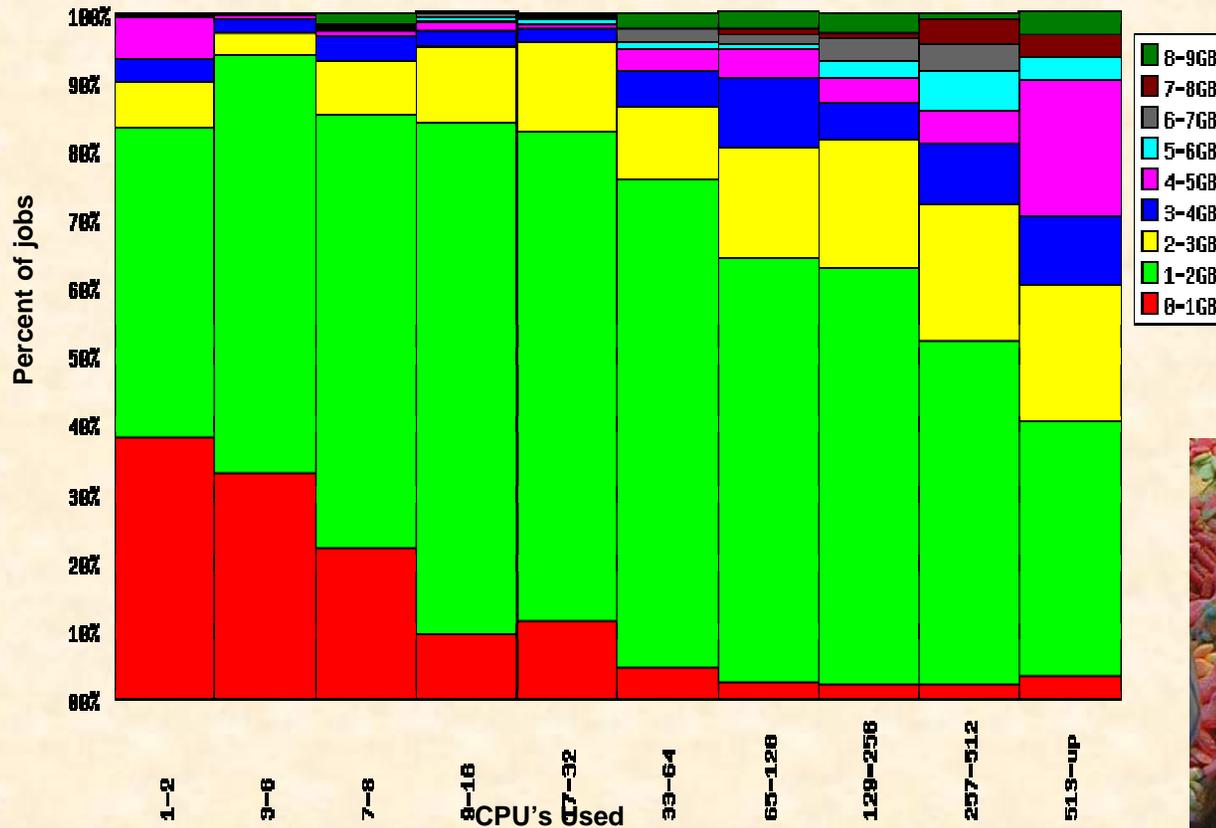
64 Cloud Resolving Model columns



Pictures by: Hiroaki Miura and Masahide Kimoto Center for Climate System Research, University of Tokyo

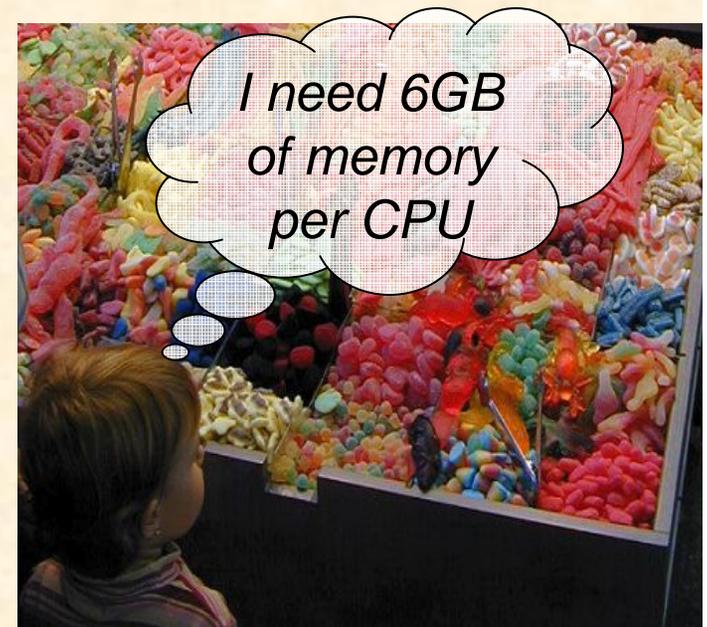
Future systems need to be driven off of real requirements

Memory footprint per node during FY04 on 11.4TF HPCS2 system at PNNL



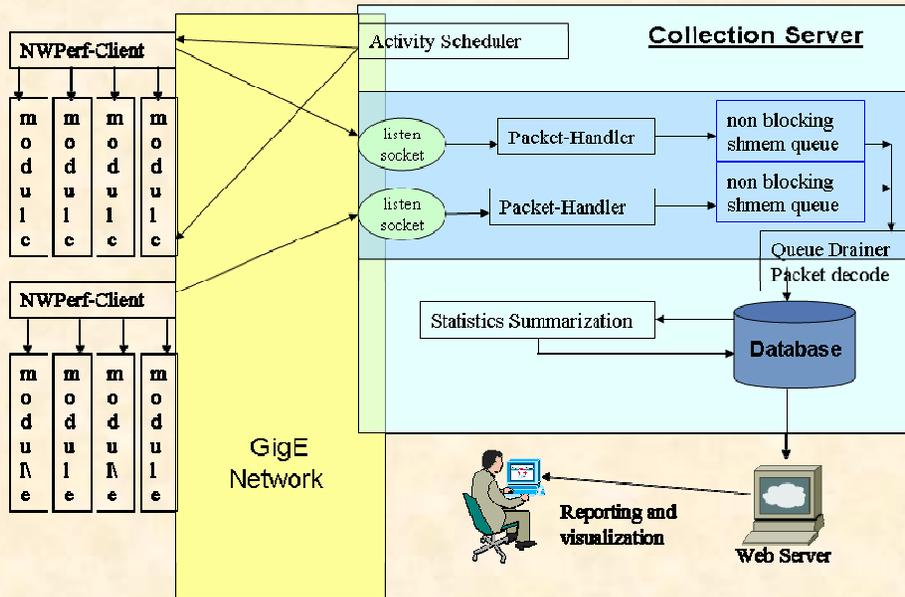
Most jobs use <50% of available memory (max avail is 6-8G)

Large jobs use more memory.



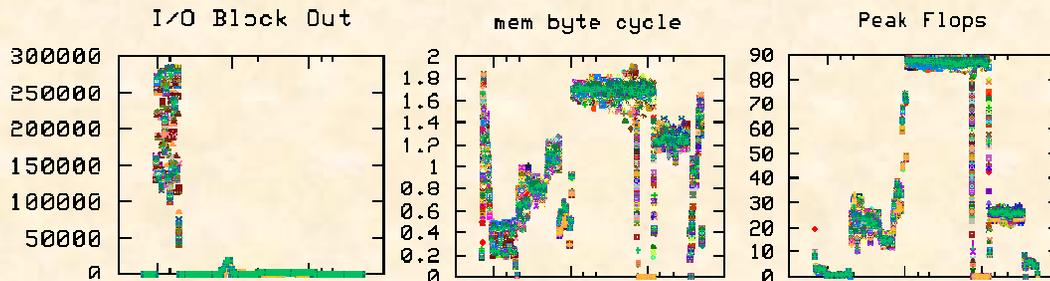
3) *Be realistic with your requirements*

So, we wanted to develop a tool to profile EVERY code and get an unbiased assessment of the real needs.

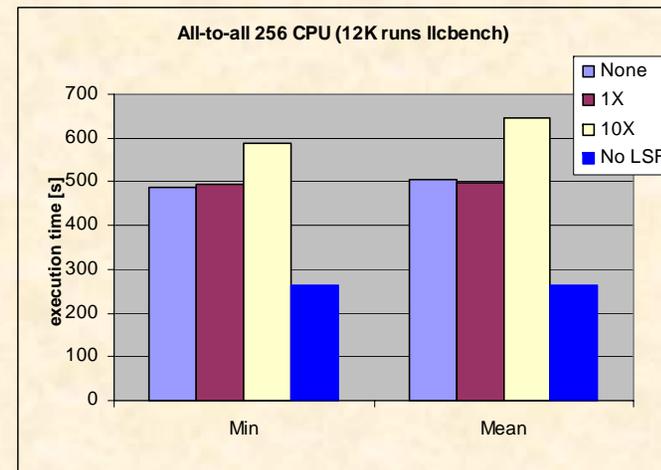


27 metrics are collected on all nodes once per minute

- Hardware Performance Counters including: Flops, Memory Bytes/Cycle, Total Stalls
 - Local Scratch Usage (obtained via `fstat()`)
 - Memory swapped out (total), swap blocks in and out
 - Memory free, used, and used as system buffers
 - Block I/O in, and out
 - Kernel Scheduler CPU allocation to user, kernel, and idle time
 - Processes running, and blocked
 - Interrupts, and Context Switches per second.
- Lustre I/O (Shared global Filesystem)

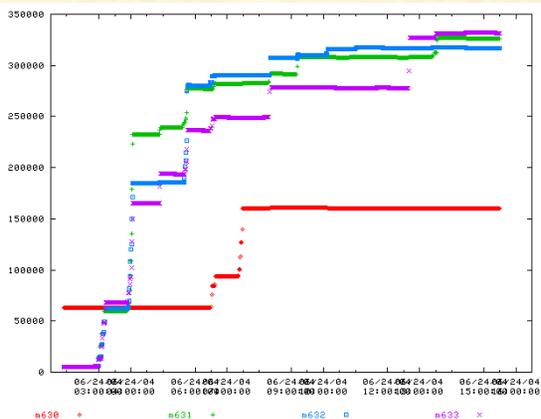


The 3 graphs are from the same 3 day 600CPU run

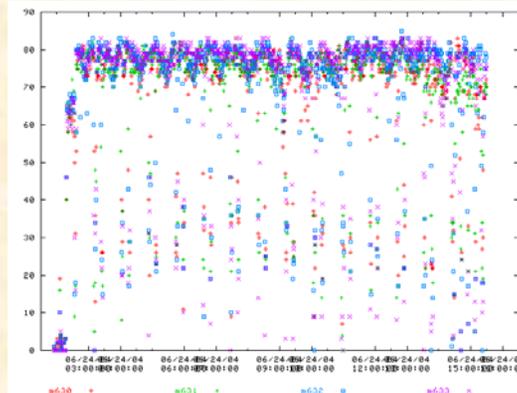


Finding Problem Jobs

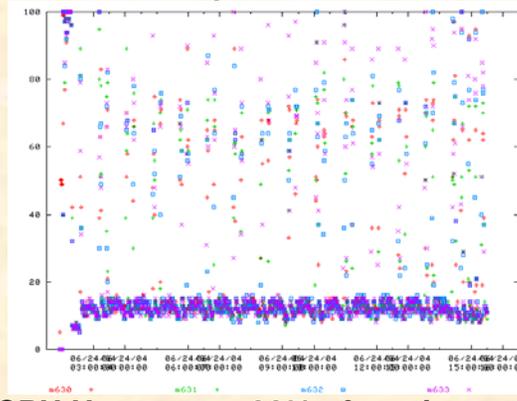
Found by running:
 db=> select jobid, avg
 from job_average_detail
 where avg > 50
 and point = ('cpu_user'



Job went into Swap

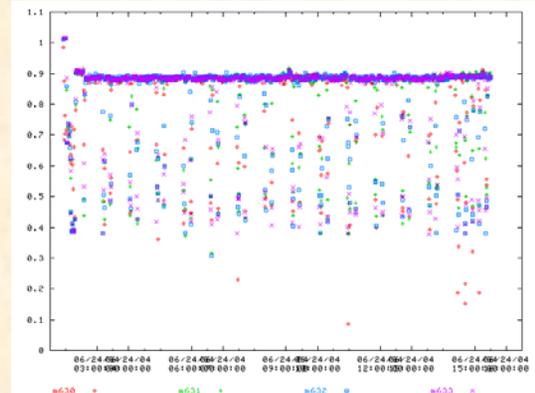


CPU Kernel space 70% of runtime

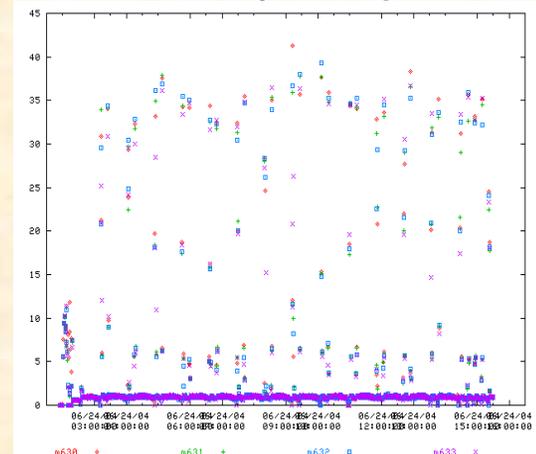


CPU User space 20% of runtime
 (over 10% idle)

High kernel space versus user space CPU – usually indicative of floating point assists (non normalized floating point operations)

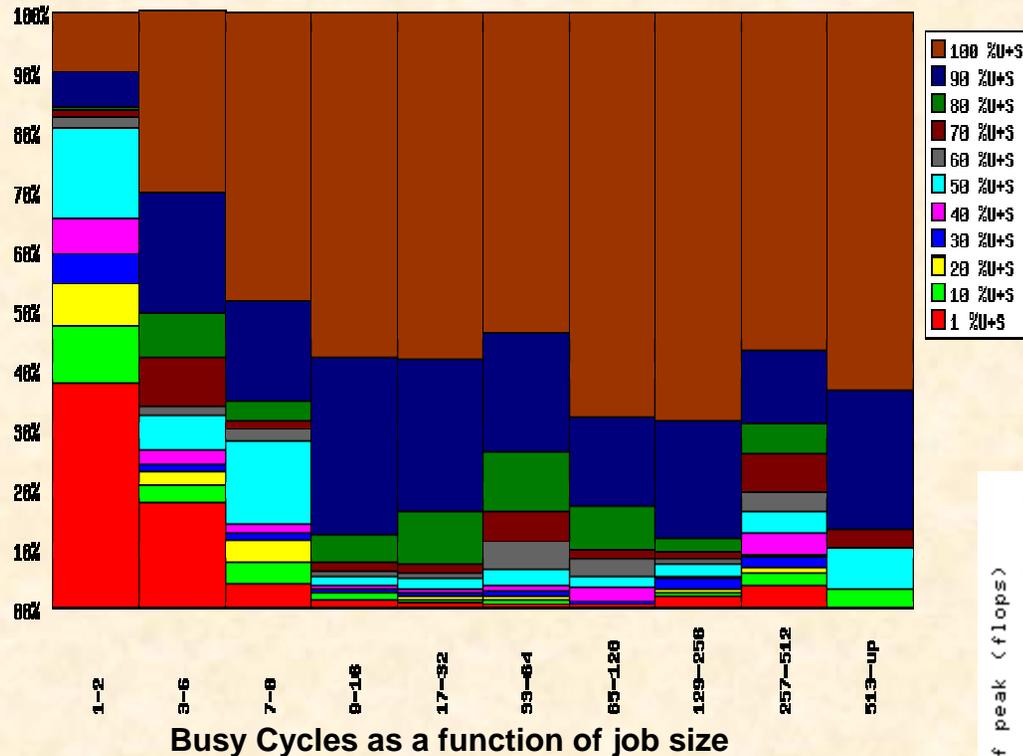


Over 80% of cycles experienced a stall



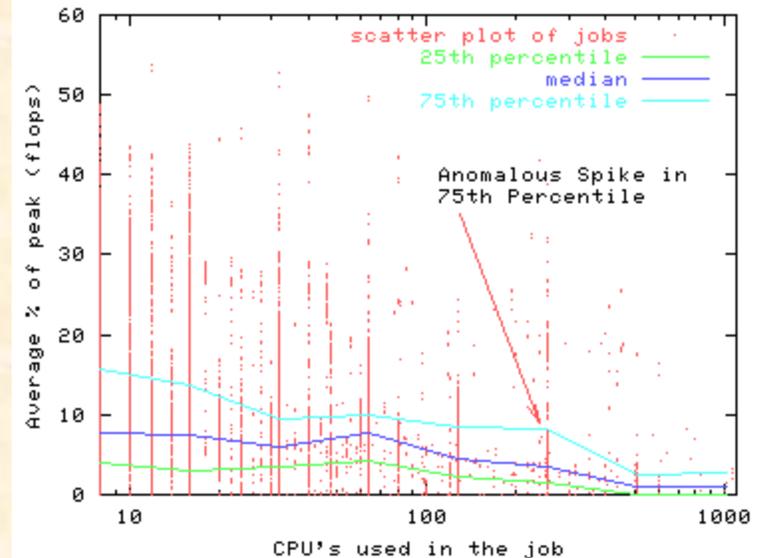
End result less than 3% of peak Flops probably could do much better

Aggregate Results



10% of the >256CPU jobs have the CPU scheduled for idle >50% of the time.

The median sustained performance for jobs over 256CPU's is <5% efficiency.



Sustained Performance as a function of CPU count

A mix of usage patterns for the same computer

By analyzing the computer we discovered some disturbing trends:

- Most jobs use <50% of available memory (max avail is 6-8G)
- 10% of the >256CPU jobs have the CPUs scheduled for idle >50% of the time.
- The mean sustained performance for jobs over 256CPU's is <5% efficiency.

However, the high impact science results all used the majority of the system:

- CCSD(t) of Cetane sustained >5TF and used over 5TB of RAM
- MP2 of H2O20 sustained 61GB/s of IO and 6TB of RAM

The highly efficient use by a few seasoned users typifies the need for user coaching and better vetting before allowing access to HPC resources.

Summary

- What information do you need from the application-science communities to improve the effectiveness of your R&D activities?
 - *Ignore the speed limit when telling us what you want.*
 - *Think big. Tell us what you really want to do. Drive the computer folks to react to your needs, rather than visa versa.*
 - *Be realistic with your requirements.*
- Evaluation of utilization patters of existing platforms leads to as many new insights as the evaluation of emerging technologies.
 - *The general community can benefit from coaching on how best to use the new systems*

I couldn't resist rubbing the crystal ball



Question #1: What are the new and exciting future technologies for:

Microprocessors – Sockets will each have many CPU's resulting in clusters with $O(100K)$ CPU's.

Interconnects – Commodity channel IO (user space communications for Ethernet).

Memory subsystems – Compilers that can support logic in the DIMM (FBD)

Storage - Scalable IO that is addressable from multiple hosts as if local.