High Performance Computing: Crays, Clusters, and Centers. What Next?[1]

Gordon Bell and Jim Gray

August 2001

Technical Report

MSR-TR-2001-76

Revised September 17

Microsoft Research
Microsoft Corporation
301 Howard Street, #830
San Francisco, CA, 94105

[1] : This work has been submitted for publication to the Communications of the ACM. Copyright may be transferred without further notice and the publisher may then post the accepted version. A version of this article appears at http://research.microsoft.com/pubs/

High Performance Computing: Crays, Clusters, and Centers. What Next?

Gordon Bell and Jim Gray

{GBell, Gray} @ Microsoft.com
Bay Area Research Center
Microsoft Research

August 2001

Abstract: *After 50 years of building high performance scientific computers, two major architectures exist: (1) clusters of "Cray-style" vector supercomputers; (2) clusters of scalar uni- and multi-processors. Clusters are in transition from (a) massively parallel computers and clusters running proprietary software to (b) proprietary clusters running standard software, and (c) do-it-yourself Beowulf clusters built from commodity hardware and software. In 2001, only five years after its introduction, Beowulf has mobilized a community around a standard architecture and tools. Beowulf's economics and sociology are poised to kill off the other architectural lines – and will likely affect traditional super-computer centers as well. Peer-to-peer and Grid communities* are beginning to *provide significant advantages for embarrassingly parallel problems and sharing vast numbers of files. The Computational Grid can federate systems into supercomputers far beyond the power of any current computing center. The centers will become super-data and super-application centers. While these trends make high-performance computing much less expensive and much more accessible, there is a dark side. Clusters perform poorly on applications that require large shared memory. Although there is vibrant computer architecture activity on microprocessors and on high-end cellular architectures, we appear to be entering an era of super-computing mono-culture. Investing in next generation software and hardware supercomputer architecture is essential to improve the efficiency and efficacy of systems.*

Introduction: Vectors and Clusters

High performance comes from parallelism, fast-dense circuitry, and packaging technology. In the 1960's Seymour Cray introduced parallel instruction execution using parallel (CDC 6600) and pipelined (7600) function units, and by 1975 a vector *register* processor architecture (Cray 1). These were the first production "supercomputers". By 1982 Cray Research had synthesized the multiprocessor (XMP) structure and vector processor to establish the modern supercomputer architecture. That architecture worked extremely well with FORTRAN because the innermost loops could be carried out by a few pipelined vector instructions, and multiple processors could execute the outermost loops in parallel. Several manufacturers adopted this architecture for large machines (e.g. Fujitsu, Hitachi, IBM and NEC), while others built and delivered mini-supercomputers aka "Crayettes" (Alliant, Ardent, and Convex) in the early 1980s. In 2001 Cray-style supercomputers remain a significant part (10%) of the market and are vital for applications with fine grain parallelism on a shared memory (e.g. legacy climate modeling and crash codes.) Single node vector supers have a maximum performance. To go beyond that limit, they must be clustered.

It has been clear since the early 1980's that clusters of CMOS-based killer-micros would eventually challenge the performance of the vector-supers with much better price-performance and an ability to scale to thousands of processors and memory banks. By 1985, companies such as Encore and Sequent began building shared memory multi-micro-processors with a single shared bus that allowed any processor to access all connected memories. Combining a cache with the microprocessor reduced memory traffic by localizing memory accesses, and by providing a mechanism to observe all memory transactions. By *snooping* the bus transactions, a single coherent memory image could be preserved. Bell predicted that all future computers or computer nodes would be *multis* [Bell, 1985]. A flurry of new *multi* designs emerged to challenge custom bipolar and ECL minicomputers and mainframes.

A cluster is a single system comprised of inter-connected computers that communicate with one another either via a message passing; or by direct, inter-node memory access using a single address space. In a cluster – inter-node communication is 10-1000 times slower than intra-node memory access. Clusters with over 1000 processors were called massively parallel processors or MPPs. A *constellation* connotes clusters of nodes with more than 16 processor "multis". However, parallel software rarely exploits the shared memory aspect of nodes, especially if it is to be portable across clusters.

Tandem introduced its 16-node, uni-processsor cluster architecture in 1975, followed in 1983 by Digital VAXClusters and the Teradata's 1,024 node database machine. This was followed by the IBM Sysplex and SP2 in the early 90s. By the late-90's most manufacturers had evolved their micro-based products to be clusters or multicomputers (Bell and Newell, 1971) -- the only known way to build an arbitrarily large, scalable, computer system. In the late 1990s, SGI pioneered large, non-uniform memory access (NUMA) shared memory clusters.

In 1983 ARPA embarked on the Strategic Computing Initiative (SCI) to research, design, build, and buy exotic new, scalable, computer architectures. About 20 research efforts and 40 companies were funded by ARPA to research and build scalable computers to exploit the new technologies. By the mid-90s, nearly all of these efforts had failed. The main benefit was increased effort in scalability and parallelism that helped shift the market to coarse grain parallelism required by a cluster.

Several other forces aided the transition to the cluster architecture. They were "helped" by exorbitant tariffs and by policies that prevented US government agencies from purchasing Japanese supercomputers. Low cost clusters empowered users to find an alternative to hard-to-use, proprietary, and expensive alternatives.

The shift from vectors to micro-based clusters can be quantified by comparing the Top500 machines in 1993 with 2001[2]. Clusters and constellations from Compaq, Cray, HP, IBM, SGI, and SUN comprise 90% of the TOP500. IBM supplied 42% of the 500, including the fastest (12.3 Tflops peak with 8192 processors) and slowest (96 Gflops peak with 64 processors). Vector supercomputers, including clustered supers from Fujitsu, Hitachi, and NEC comprise only 10%. NEC's 128 processor clustered vector supercomputer operates at a peak of 1.28 Tflops. Based on the ratio of their peak speeds, one vector processor is equal to 6-8 microprocessors. Although supers peak advertised performance (PAP) is very expensive, their real applications performance (RAP) can be competitive or better than clusters on some applications. Shared memory computers deliver RAP of 30-50% of the peak advertised performance (PAP). Clusters typically deliver 5-15%. [Bailey and Buzbee].

[2] The Top500 is a world-wide roster of the most powerful computers as measured by Linpack. See www.Top500.org.

Table 1.Comparison of computer types in the Top500 between 1993 and 2001

Type	1993		2001			
	Number	Vendors	Number	Vendors	New	Defunct*
Scalar	133	9	450	6	3	6
Vector	332	4	50	3	0	1
SIMD**	35	1	0	0	0	1

*Either computer or the company producing it has ceased to exist.
**Single Instruction stream, Multiple Data-operations. An architecture with 16-64 thousand units to exploit VLSI that was abandoned as microprocessors overtook it.

High performance computing has evolved into a small, stable, high-priced market for vector supers and constellations. This allows suppliers to lock customers in to a unique hardware-software environment e.g. PowerPC/Linux or SPARC/Solaris. Proprietary environments allow vendors to price systems at up to $30K per microprocessor versus 3K$ per slice for commodity microprocessors, and to maintain the margins needed to fund high-end, diseconomies of scale.

Enter Beowulf: Commercial Off-the-shelf hardware and software

The 1993 Beowulf Project goal was to satisfy NASA's requirement for a one Gflops workstation costing less than $50,000. The idea was to use commercial off-the-shelf (COTS) hardware and software configured as a cluster of machines. In 1994 a 16-node $40,000 cluster built from Intel 486 computers achieved that goal. In 1997, a Beowulf cluster won the Gordon Bell performance/price Prize. By 2000, several thousand-node Beowulf computers were operating. In June 2001, 28 Beowulfs were in the top500 and the Beowulf population is estimated to be several thousand. High schools can now buy and assemble a Beowulf using the recipe "How to Build a Beowulf" [Sterling, et al 2001].

Beowulf is mostly about software. Beowulf clusters' success stems from the unification of public domain parallel tools and applications for the scientific software community. It builds on decades of parallel processing research and on many attempts to apply loosely coupled computers to a variety of applications. Some of the components include:

- message passing interface (MPI) programming model
- parallel virtual machine (PVM) programming, execution, and debugging model
- parallel file systems
- tools to configure, schedule, manage and tune parallel applications (e.g. Condor, the Maui scheduler, PBS)
- higher-level libraries e.g. Linpack, BLAS

Beowulf's enabled do-it-yourself cluster computing using commodity microprocessors -- the Linux/GNU or Windows 2000 operating system, plus tools that have evolved from the research community. This standard software platform allows applications to run on many computer types – thereby fosters competition (and avoids lock-in). Most importantly Beowulf is a convergent architecture that will run over multiple computer generations, and hence protects application investment. Beowulf fosters a community of users with common language, skills, and tools, but with diverse hardware. Beowulf is the alternative to vector supercomputers and proprietary clusters normally found in centers.

Centers: Haven't We Seen this Movie?

Over time, we have seen computation and data migrate from central facilities when no low cost facilities were available, then to distributed VAX minicomputers in the early 80s, then back to a few large NSF and state-supported centers with personal computers for access in the mid-80s, then to fewer, large again back to centers in the late '90's, and now back to build-it-yourself clusters.

Beowulf's economics have important socio-economic-political effects. Now individuals and laboratories believe they can assemble and incrementally grow *any-size* super-computer *anywhere* in the world. The decision of where and how to compute is a combination of: cost, performance, availability (e.g. resource allocation, application program, ease of access, and service), the applications focus and dataset support, and the need or desire for individual control.

Economics is a key Beowulf advantage -- the hardware and software is much less expensive. Centers add a cost factor of 2 to 5. Center's costs are explicit: space (e.g. air conditioning, power, and raised floors for wiring and chilled air ducts), networking, and personnel for administration, system maintenance, consulting, etc. A center's explicit costs are implicit when users build and operate their own "centers" because home grown centers ride "free" on their organizational overhead that includes space, networks, and especially personnel.

Sociology is an equally important Beowulf advantage. Its standards-setting and community nature, though not usually part of the decision, eliminates a barrier because users have access to both generic and profession-specific programs and talent that centers try to provide. Furthermore, a standard platform enables a market for programs and enhanced technical recognition.

The situation is similar to the late 70s when VAX was introduced and "Cray" users concluded that it was more productive and cost effective to own and operate their own, smaller, focused centers. Scientists left centers because they were unable to get sufficient computing power compared to a single user VAX. Although the performance gap between the VAX and a centers "Cray" was a factor of 5-10 and could be 100, the performance per price was usually the reverse.

By the mid 80s, government studies bemoaned the lack of supercomputer centers and super-computer access for university scientists. These researchers were often competing to make breakthroughs with their counter-parts in extremely well funded Dept. of Energy Labs. The various DOE labs had been given the mandate with the Advanced Strategic Computing Initiative (ASCI) to reach 10 Teraflops and Petaflops (10^{12} and 10^{15} floating-point operations per second, respectively) levels in 2001 in order to fulfill their role as the nation's nuclear stockpile steward.

In response NSF established five centers in 1985. Keeping all of the expensive supercomputer centers at the leading edge was neither affordable nor justified, especially in view of the relatively small number of users. To be competitive a center has to be among the world's largest computers (about two orders of magnitude larger than what a single researcher can afford).

In 1999, in response to these realities, NSF reduced the number of supercomputing centers to two. This concentrated enough funding to achieve several Teraflops at each center. The plan was that each year or so, one of the two centers would leapfrog the other with new technology to keep centers at the forefront and provide services that no single user could afford. In 2001, NSF seemed to have forgotten all this[3] and created a third center -- or at least they funded the CPU and memory *with what turned out to be the last, Alpha cluster and inherently an orphan..* Storage was unaffordable! The next act is predictable easy to predict: NSF will under-fund all three centers – and then eventually discontinue one of them. The viability of individual centers decreases as more centers dilute funding.

Some centers claim a role with constellations built from large shared memory multiprocessor nodes. Each of these nodes is more powerful than a Beowulf cluster of commodity PC uni- or dual-processors.

The centers idea may already be obsolete in light of Beowulfs, computational Grids, and peer-to-peer computing. Departmental Beowulfs are attractive for a small laboratory because they give low-overhead dedicated access nearly the **same** capability a large center provides. A center typically allocates between 64 and128 nodes to a job[4], comparable to the Beowulf that most researchers can build in their labs (like their VAXen two decades earlier). To be competitive, a supercomputer center needs to have at least 1,000 new (less than two years old) nodes, large data storage for each user community, and some asset beyond the scope of a small laboratory.

We believe that supercomputer centers may end up being fully distributed computation brokers – either collocated with instrumentation sites as in the case of the astronomy community, or centers to support peer-to-peer computing

[3] Although NSF is an independent agency that is directly funded by congress, it is subject to varying political winds and climate that include congress persons, conflicting centers and directorate advisory committees, and occasionally its own changing leadership.

[4] At a center (with approximately 600 SP2 processors), one observed: 65% of the users ran on more than 16 processors; 24% on more than 32; 4% on more than 64; 4% on more than 128; and 1%on more than 256.

e.g. www.seti.org averaging 10 teraflops from 1.6 million participants who donate their computer time, or www.Entropia.com that brokers fully-distributed-problems to internet PCs.

We see two possible futures from super-computer centers:

1. **Exotic**: An application centric vector or cellular supercomputer www.research.ibm.com/BlueGene for an area like weather prediction to run apps that users have been unable to convert to a Beowulf architecture or Japan's Earth Observation Research Center Simulator www.eorc.nasda.go.jp.

2. **Data Center**: a concentration of peta-scale datasets (and their applications) in one place so that users can get efficient and convenient access to the data. The various NASA Data Access Archives and Science Data Centers fit this model. The Data Center becomes increasingly feasible with an Internet II delivering 1-10 Gbits per second.

Both these models cast the supercomputer center as the steward of a unique resource for specific application domains.

Paths to PetaFlops Computing

The dark side to Beowulf commodity clusters is they perform poorly on applications that require large shared memory. We are concerned that traditional super-computer architecture is dead and that we are entering a supercomputer mono-culture. At a minimum we recommend increased investment in research on ultra-high-performance hardware-software architectures including new programming paradigms, user interfaces, and especially peta-scale distributed databases.

In 1995 a group of eminent architects outlined approaches that would achieve a petaops by 2010 [Sterling, et al 1995]. Their recommendation was three interconnected machines: (1) a 200 Teraflops multi-threaded shared memory architecture; (2) a 10,000- 0.1 Teraflops nodes; and (3) a 1 million 1 Gflops processor in memory nodes. Until recently, Sterling had been pursuing data-flow architectures with radical packaging and circuit technology. IBM's BlueGene is following the third path (a million gigaflops chips) to build a petaflops machine by 2005 geared to protein folding and other embarrassingly parallel tasks with limited memory needs (it has mips:megabyte ratio of 20:1 versus 1:1). IBM is also considering a better balanced machine codenamed Blue Light. Only a small number of unconventional experimental architectures e.g. Berkeley's processor-in-memory are being pursued.

Because custom system-on-a-chip experiments are so complex and the tools so limited, we can only afford a few such experiments.

Next generation Beowulfs represent the middle path. It has taken 25 years to evolve the crude clusters we have today. The number of processors has stayed below a maximum of 10,000 for at least five years, with very few apps able to utilize more than 100 processors. By 2010, the cluster is likely to be the principal computing structure. Therefore research programs that stimulate cluster understanding and training are a good investment for laboratories that depend on the highest performance machines. Sandia's computational plant program is a good example of this (http://www.cs.sandia.gov/cplant/).

Future Investments

Continued investments to assure that Moore's Law will continue to be valid underlies all of our assumptions about the future. Based on recent advances and predictions, progress is likely to continue for at least another decade. Assuming continued circuit progress, performance will come from a hierarchy of computers starting with multiprocessors on a chip. For example, several "commodity" chips with multiple processing units are being introduced that will operate at 20 Gflops. As the performance of single, multiprocessor chips approaches 100 Gflops, a petaflops machine will *only* need 10,000 units.

On the other hand, it is hardly reasonable to expect a revolutionary technology within this time period because we see no laboratory results for near term revolution. Certainly petaflops performance will be achieved by special purpose computers like IBM's Blue Gene project, but they stand alone.

SGI builds a shared memory system with up to 256 processors and then clusters these to form a constellation. But this architecture is low-volume and hence expensive. On the other hand, research into high speed interconnections such as Infiniband™, may make the SGI approach a commodity. It is entirely possible that huge cache-only memory architectures might emerge in the next decade. All these systems require good locality because on-chip latencies and bandwidth are so much better than off-chip. A processor-in-memory architecture or multi-system on a chip will no doubt be part of the high-performance equation.

In 2001, the world's 500 top computers consist of about 100,000 processors, each operating at about one gigaflops. Together they deliver slightly over 100 teraflops.

Seti@home does not run Linpack, so does not qualify in the top500. But Seti@home averages 13 Tflops, making it more powerful than the top 3 of the top500 machines combined. This suggests that GRID and peer-to-peer computing using the Internet II is likely to remain the world's most powerful supercomputer.

Beowulfs and Grid computing technologies will likely merge in the next decade. When multi-gigabit LANs and WANS become ubiquitous, and when message passing applications can tolerate high latency, the Grid becomes a Beowulf. So all the LAN-based PCs become Beowulfs – and together they form the Grid.

Progress has been great in parallelizing applications that had been challenging in the past (e.g. n-body problems). It is important to continue on this course to parallelize applications heretofore deemed the province of shared memory multiprocessors. These include problems requiring random variable access and adaptive mesh refinement. For example, automotive and aerodynamic engineering, climate and ocean modeling, and applications involving heterogeneous space remain the province of vector multiprocessors. It is essential to have "the list" of challenges to log progress – unfortunately, the vector-super folks have not provided this list.

Although great progress has been made by computational scientists working with computer scientists, the effort to adopt, understand, and train computer scientists in cluster and constellation parallelism has been minimal. Few computer science departments are working with their counter-parts in other scientific disciplines to explore the application of these new architectures to scientific problems.

Acknowledgments

An early draft of this paper was circulated and received considerable discussion and constructive criticism. We are especially indebted to David Bailey (NERSC), Bill Buzbee (NCAR), Ken Kennedy (Rice U.), David Lifka (Cornell Theory Center), Ken Miura (U. Kysuhu), George Spix (Microsoft), Tom Sterling (CalTech), Peter Denning (GWU) Erich Strohmaier (LBL), Tadashi Watanabe (NEC) for their constructive comments. It is controversial; so it was not possible for us to accept all comments and suggestions.

References

Bailey, D.H. and W. Buzbee. Private communication.

Bell, C.G. and A. Newell, Computer Structures, McGraw Hill, New York 1971.

Bell, C. G., "Multis: A New Class of Multiprocessor Computers", Science, Vol. 228, pp. 462-467 (April 26, 1985).

Bell, G., "Ultracomputers: A Teraflop Before Its Time", Communications of the ACM, Vol. 35, No. 8, August 1992, pp 27-45.

Earth Observation Research Center www.eorc.nasda.go.jp

Foster, I. and Kesselman, C. editors *The Grid: Blueprint for a New Computing Infrastructure*, Morgan Kaufman, San Francisco, 1999.

IBM BlueGene web site www.research.ibm.com/BlueGene

seti@Home web site www.seti.org

Sterling, Thomas; Paul Messina; and Paul H. Smith, *Enabling Technologies for Petaflops Computing*, MIT Press, Cambridge, MA, July 1995

Sterling, T. *Beowulf PC Cluster Computing with Windows and Beowulf PC Cluster Computing with Linux*, MIT Press, Cambridge, MA, 2001.