EUROPEAN LABORATORY FOR PARTICLE PHYSICS

CN/95/14

25 September 1995

PC as Physics Computer for LHC ?

Sverre Jarp, Hong Tang, Antony Simmins

Computing and Networks Division/CERN
1211 Geneva 23 Switzerland
(Sverre.Jarp @ Cern.CH, Hong.Tang@Cern.CH, Antony.Simmins@Cern.CH)

Refael Yaari

Weizmann Institute
Israel
(FHYaari2@Weizmann.Weizmann.AC.IL)

PC as Physics Computer for LHC ?

Sverre Jarp, Antony Simmins, Hong Tang
Physics Data Processing Group/CN/CERN

R.Yaari
Weizmann Institute, Israel

CHEP-95, 21 September 1995, Rio de Janeiro, Brazil

Abstract

In the last five years, we have seen RISC workstations take over the computing scene that was once controlled by mainframes and supercomputers.

In this paper we will argue that the same phenomenon might happen again. A project, active since March this year in the Physics Data Processing group of CERN's CN division is described where ordinary desktop PCs running Windows (NT and 3.11) have been used for creating an environment for running large LHC batch jobs (initially the DICE simulation job of Atlas).

The problems encountered in porting both the CERN library and the specific Atlas codes are described together with some encouraging benchmark results when comparing to existing RISC workstations in use by the Atlas collaboration. The issues of establishing the batch environment (Batch monitor, staging software, etc.) are also covered.

Finally a quick extrapolation of commodity computing power available in the future is touched upon to indicate what kind of cost envelope could be sufficient for the simulation farms required by the LHC experiments.

1 - Introduction

The LEP (Large Electron Positron) accelerator experiments at CERN have satisfied their need for computing by deploying modern RISC/Unix workstation technology. On average, 1000 CERN Units[1] of computing power is sufficient to cover the needs for simulation, reconstruction and analysis. The next generation experiments, based on the planned accelerator, LHC (Large Hadron Collider), will demand computing capacities that are at least three orders of magnitudes higher. This implies that, whereas a farm of RISC workstations, providing 1,000 CERN Units, for LEP costs about one million dollars today, an adequate farm for LHC would cost about one billion dollars with today's prices. Since the large portions of

[1] A CERN unit is about 4 SPECint92. See references 1 and 2 for further details.

the computing capacity are needed only in some years from now, everybody expects the cost-effectiveness to increase, maybe with as much as a factor ten. The work described in this paper was initiated, however, because of the belief that only by using commodity computing components, such as standard Personal Computers, destined for the large home and business markets, will the High Energy Physics (HEP) community be sure to align itself with the best price/performance possible and reach the LHC computing requirements at an affordable cost of no more than ten million dollars.

2 - Choosing the Personal Computers

At CERN there are more than 2,000 PCs on site but few are being used for physics computing, basically because large portions of the machines are 486-based with maybe as little as 8 Megabytes of memory. At a planning meeting in CN in January it was proposed to evaluate the performance of simulation jobs on high-end Pentiums with adequate memory (16-32 MB) to demonstrate that they would provide market-leading price/performance. The target was $100/CU. The proposal raised two important issues. One was whether the effective performance of real-life jobs would match the peak SPECint performance numbers, measured by Intel under conditions that were "artificially" improved compared to real PCs in the market place. The recommendation was therefore to start with benchmarks on high-end PCs to discover the effective performance that HEP programs would obtain.

The second question concerned the operating system. PCs, although usually equipped at purchase time with Windows 3.11, enjoy a vast choice of operating systems. In addition to Windows 3, there were Windows/NT, OS/2 Warp, Linux, Solaris/Intel, Unixware, SCO/Unix to mention only a subset. This topic also created a lot of debate, basically because High Energy Physics is still deeply rooted in Unix (and VMS ?) and a move to other another system would make life of the average physicist even more complex.

Nevertheless, the system chosen for the evaluation, was Windows/NT with Windows 3.11 (now being superseded by Windows/95) as a second choice. The main reason for this choice was the fact, that Windows/NT is becoming a commodity operating system, and at the same time enjoys a modern design that includes clean 32-bit support, symmetric multiprocessing, pre-emptive multitasking, clustering, good networking for integration both the TCP/IP and Novell-based clusters, as well as a rich choice of compilers and PC applications. An additional argument, which has not yet been exploited, is the fact that Windows/NT has been ported to several architectures, including Alpha, MIPS, and PowerPC. One future possibility is therefore to construct clusters of cost-effective x86 and PowerPC-based systems with the exact same version of Windows/NT hiding the underlying differences in hardware architecture.

3. Choosing the compilers and porting the codes

The next issue was the choice of compilers. Although HEP programs are mainly written in FORTRAN-77, parts of the libraries are in C, so it was important to choose a set of compilers that allowed mixing of these two languages. The initial choice fell on the Watcom/Powersoft compilers (version 9.5[2]) which were also said to provide good code optimisation, a feature thought highly relevant to the benchmarking exercise. The compilers were also said to be robust, a feature that matters when one plans the compilation of up to one million lines of code.

With the FORTRAN compiler installed on two HP Vectra PCs, it was possible to start porting the CERN benchmarks (See Ref. 1 for details). This proved relatively easy, given that the programs (about 20,000 lines in total) have no dependency on the underlying operating system and do not contain any C code. Results are reported in section 4.

The next task was to port the Geant Example 1 (GEXAM1) which is often used as a quick (although not always 100% representative) benchmark of simulation jobs. This task was tougher than the first one since GEXAM1 depends on a fair number of routines from the CERN library, mainly in FORTRAN but also in C. It also depends on ZEBRA, which in turn depends on the endian-ness of the computer. A source version of the necessary CERN library routines, extracted for other benchmarking purposes some years ago, was used as a base. After a fair bit of work, also to understand the options of the C and FORTRAN compilers, code optimisation and language mixing, we were able to report the first successful results.

The third task was to port the full CERN library to Windows/NT on x86 architecture. Although this sounds like a major task it turned out that the work done on GEXAM1, including the understanding of ZEBRA on little-endian systems and language mixing, plus the fact that a library port had already been done to the first version of Windows/NT on DEC Alpha two years earlier[3], were both of great help. Since the plan was only to run batch versions of the physics programs and not interactive versions, HIGZ and other graphics libraries were not ported.[4] Nevertheless, the total amount of code was close to 340,000 lines in six libraries (HBOOK, ZEBRA, KERNLIB, GEANT, and GENLIB and FFREAD). A few routines, that have more relevance to Unix than Windows/NT, i.e. *getuid*, *getgid*, *readlink* and *kill*, were left unconverted.

The Watcom FORTRAN library does not for some strange reason contain the MVBITS routine, so we had to add it by hand. Similarly some routines had slightly different names and calling convention, so, for instance, LSHIFT(I,J) became ISHL(I,J) and RSHIFT(I,J) became ISHL(I,-J). Passing character strings between the FORTRAN and C compilers, WATCOM use the "DESCriptor" option by default

[2] The most recent version is 10.5 which CERN has beta-tested over the last few months.
[3] This work was carried out by Valery Fine/Dubna and the changes made were added to the library source files at CERN.
[4] These libraries are now in the process of being ported by IN2P3, Marseilles.

and pass the pointer to a string descriptor containing a pointer to the data and the length of data in two fullwords.

The final effort in the long series of porting exercises was the transfer of the full Atlas simulation program, DICE, to the PC. Once more, the size of the code was far from trivial, with over 120,000 lines in three main libraries (DICE, SLUG, and GENZ). In this porting exercise there were several problems:

- DICE events are very time-consuming, on the PC full HIGGS events take about 20 minutes each. When problems were discovered in event N, it was impossible to restart from the beginning and trace one's way through hours and hours of computing. The common remedy to this problem is to write out the seeds used for the first random number of each event and simply instrument the job to restart from the seeds of event N. In doing so, we discovered two problems. The first one was the fact that DICE did not call GRNDM (the GEANT random number generator) for all its random numbers, but also RNDM, so even though we restarted with the seeds of event N, we did not reproduce the same event (and the error did not reoccur). This was solved by changing DICE to consistently use the same number generator everywhere. The second problem was that the trace option, /TR, of the FORTRAN compiler influenced the accuracy of floating-point calculations, so that the switching on of this option, led to a different route through the simulated detector, with again the avoidance of an error seen previously.
- Having tackled the problem of reproducibility, we solved two floating point errors caused by the fact that the x86 architecture is stricter on floating-point exceptions than, for instance, the HP architecture and compiler. One polycone defined in the subroutine, COPEDF2, had the last two planes in the same z position, and this caused a divide by zero exception. Moving one plane by an epsilon solved the problem. In GLANDZ, which is the GEANT energy loss straggling routine, a comparison was done with an invalid floating point number (after an overflow) and this was overcome by adding a correction from the latest 3.21 correction cradle.

With these corrections and an improved understanding of floating-point accuracy and exception handling on the x86 architecture we were ready to move on to some real benchmarking work.

4. Benchmark results

A relatively full set of benchmarks is provided in Appendix I.

In Table 1 it can be seen that the desired goal of 20 CERN Units have not yet been reached. The best number until now is 17.7 CU from a 133 MHz Pentium, a little behind the 19.9 Units provided by the HP/712/80MHz workstation[5]. Given a PC at $3,000[6] this puts the current cost of a CERN Unit well under 200 dollars, but the goal of $100/CU will probably only be reached with Pentiums at 180 MHz or

[5] Comparisons are often made to this HP workstation because it was the workstation of choice for the most recent increase of CERN's Central Simulation Farm.

[6] Configuration with 16 MB of memory, 500 MB disk, and no monitor.

the forthcoming new generation of x86 chips, dubbed P6. It should nevertheless be stressed that the PC has right away put itself at the top of the list of cost-effectiveness, possibly sharing the number one position with systems based on IBM's latest PowerPC 604 chip, another commodity processor.

In Table 2 we see even more encouraging results. The GEXAM1 benchmark scales well when using higher frequency processors and/or fast pipelined SRAM caches. The time per event (2.37 seconds) is practically identical to the HP/712/80. It is also very close to the 200 MHz SGI Challenge and the IBM/SP2 system which uses RS6000/390s as nodes. The PowerPC/604 with 133 MHz processor, 256 KB L2 cache, fast math libraries and a tool to optimise the cache is twice as fast as the PC/133. Without the fast math library and the optimisation tool the timing is 1.62 seconds which is still very respectable (See ref. 3). Both the PC and the PowerPC are far behind DEC's latest Alpha chip, the 21164 at 300 MHz, used in the DEC/8400 multiprocessor system or the DEC 600 5/300 workstation, but the cost of these high-end systems is also quite different.

In Table 3 the DICE results from various PCs are documented. For comparisons one needs to know that a full HIGGS event[7] simulation takes 485 seconds on a HP/735/99 and 956 on a HP/712/80. The best PC result is 1103 seconds obtained with a 133 MHz processor with 512 KB asynchronous cache. Relatively speaking the result from a 100 MHz Pentium with a 1 MB cache (1150.5) is much more impressive, because the difference in frequency is 33% but the result is only 4% slower. The results from a PC/133 with 256 KB fast pipelined cache is even more surprising (it is 1% slower than the PC/100), underlining once more the importance of large (and not just fast) caches. The cache effect comes from the ZEBRA banks heavily used in large simulation programs and have already been discussed in previous HEP code analyses (See Ref. 4). An interesting detail in Table 3 is the fact that the initialisation of DICE always scales with the clock frequency, but not the events themselves. In the moment of writing, efforts are underway to test the 133 MHz Pentium with a 1 MB cache. In any case, the PCs are already with the current results quite a bit more cost-effective than the HP systems used at CERN for CSF phase 1 and 2.

One attractive feature which is offered in some PC configurations, such as the HP/XU Vectras, is the possibility to install a second processor. Table 4 lists the throughput increase obtained for various runs of GEXAM1 and DICE. As can be seen, an increase of at least 1.6 was obtained with the result that the PC cost-effectiveness improved considerably given that a second processor can normally be purchased for only $500 - $1000, depending on the model.

In concluding this section, it must be said that all benchmarks with the exception of SMP tests, have been run with both Windows/NT and Windows 3.11. An effort to replace Windows 3.11 by Windows/95 is now underway. This should allow us to use identical binary files for the two environments[8], whereas we now use different binaries (built from identical object files).

[7] DICE results are based on the average of 10 full HIGGS events with input from tape LH0111.

[8] Stop press: This works as expected.

5. Optimising the hardware

As can be seen from the benchmark results in Appendix I, hardware features such as enhanced caches and additional processors can improve our price/performance ratio by quite a large amount. Our hope is therefore that, although PCs are bound to follow the mass market rules, motherboards may continue to be configurable in a cost-effective way. Manufacturers that solder the cache directly on to the board miss an opportunity to offer flexibility to a larger community in spite of an immediate saving of a few dollars.

Memory seems to have less effect on the benchmark results. Most tests have been done with 16 or 24 MB of standard 70 ns DRAM memory. EDO-RAM, for instance, does seemingly not improve results on cache-based system and a GEXAM1 benchmark run on a 133 MHz Pentium with 256 KB of pipelined cache and 16 MB of EDO-RAM at KEK in Japan, showed that when the L2 cache was disabled, performance dropped to merely 50%.

Disks have not been assessed either and most disks in use are standard extended IDE drives. Since problems of slow access are likely to slow down interactive compilation and program execution considerably, this area will be studied more intensively in the future.

6. Adding integration software

A set of stand-alone PCs, even if they are clustered together through the native capabilities of Windows/NT, would be of limited interest. A series of efforts to integrate the PCs into the other CERN computing environments have therefore been undertaken in parallel to the program porting and benchmark testing.

Windows/NT systems can flexibly be integrated with CERN's NICE (Novell network) and this feature allows for immediate access to printing, mail services (should you want your user agent on your PC), plus additional disk space and file sharing with Windows-based PCs. One amusing consequence of the last item, was to make the Windows-version of the GEXAM1 benchmark available on one of the NICE servers and walk around to people's offices asking them to run the benchmark of their machines by simply clicking on an executable in the File Manager. This offered us a quick understanding of the performance of other systems, including 486s which we had decided not to use[9].

Given that CERN's main infrastructure, such as the CORE batch environment, is based on RISC/Unix servers, it was felt to be of great importance to integrate our PCs into this environment.

The fist step was to install a freeware product, called SAMBA[10], that runs on a Unix server and exports Unix file systems (local or NFS mounted) to Windows/NT

[9] The benchmark results did not make us change our opinion !

[10] SAMBA was developed by A.Tridgell at the Australian National University. It can be obtained via the ftp address: "nibmus.anu.edu.au:/pub/tridge/samba".

clients. This product installed easily (no local tailoring needed) and gave us access to the Atlas staging pool by NFS mounting the relevant file systems. As a result, there was no longer a need to keep the DICE input file locally.

The second step was even bolder (and more ingenious ?) and consisted of a port of the SHIFT software to Windows/NT. This software can be considered as a kind of middleware that offers access to the SHIFT stager through commands such as stagein, stageqry, stageput, and stageclr, and also the Remote File I/O routines for optimised file access without requiring NFS mounts. The problem was that this software is heavily Unix dependent. We deployed the same strategy as before and picked up from Internet a public utility, named DOWNHILL[11], that implements about 80 Unix APIs to ease the porting of Unix software to NT. This was a good start, but we later discovered that we also needed the Microsoft Visual C++ compiler, which implements the Windows/NT system calls more thoroughly than Watcom and also happened to be the compiler used by the original implementor of DOWNHILL. Some additional routines, such as *inet_netof* and *syslog*, were added in from the BSD 4.4 distribution in order to cover all the APIs required by the SHIFT software. Some routines, such as *fork*, *ioctl*, and *symlink* have no direct counterpart in Windows/NT so we changed the functionality accordingly.

The combination of the DOWNHILL port, allowing us to run stagein commands and SAMBA accessing files from the real staging pool, makes the PCs almost fully integrated with SHIFT. The remaining bit is to use the RFIO routines for access instead of SAMBA and this work is well underway.

The PCs need some kind of batch monitor and we have successfully tested Intergraph's version of NQS. This version of NQS does not offer integration with the NQS version run in SHIFT on the Unix CPU servers, but a port of CERN's NQS++ should allow job submission and retrieval for physicists wherever they reside. Furthermore, IBM and Platform Computing have expressed interest in producing ports of the LoadLeveler and the Load Sharing Facility, respectively, but these products are not yet available for testing.

Yet another freeware product, rlogin, provides remote login between machines as long as no windowing commands are being issued. Such commands will always send the window display to the local host, a feature not very useful for the remote user.

7. Conclusions

In a short period of time, we have demonstrated that PCs for Physics Computing are becoming ready for prime time. Benchmark results are encouraging, propelling PCs right to the top of the list of cost-effective systems. Configuration flexibility help assure the right hardware for the demanding HEP batch jobs. Integration software has allowed us to become participants in both the PC NICE environment and the SHIFT UNIX environment. Especially the latter is vital for batch processing but it is only when people realise that PCs are first of all for desk-top use, that they

[11] DOWNHILL was developed by G.Knauss at NETCOM. It is available from "ftp.netcom.com:/pub/kn/knauss/DOWNHILL_1.2.tar.Z".

also appreciate the importance of a good integration with the HEP batch environment.

PCs for batch should allow unprecedented "clonability". Small HEP institutes will be able to establish farms without investments in the hundred-thousand-dollar range, and physicists may even consider small farms in their offices without major investments or complications shold they so desire.

Windows/NT has lived up its promise of being a modern operating system with the correct set of features for HEP needs. With its capability of providing an identical user interface on multiple hardware architectures, it seems easy and promising to mix PCs with DEC Alpha systems or IBM/Motorola PowerPC systems. HEP's old strategy of always buying the most cost-effective hardware, can then be fully deployed in this environment as well. A LHC research proposal, P61, has recently been put forward to initiate a broad collaboration amongst several HEP institutes based on these premises (See Ref. 5).

Not everything has been solved yet. This report has mentioned several items that are still pending or need refinement. Our Windows/NT version of the library is not yet an officially supported port and the graphics parts are not yet completed. Some basic reliability tests have been done, we recently ran a DICE job for 8 days non-stop, but further testing is clearly required, especially if large clusters of PCs (or PowerPC systems) are desired.

Power PC systems and P6 systems (especially with dual processors) are soon bound to bring us below the limit of one hundred dollars per CERN Unit. Each architecture should see its performance double over the next two years. Furthermore, the likelihood that the joint HP/Intel P7 chip (or the next IBM/Motorola chip) comes in at an even higher performance level, offers additional credibility to a strategy based on rapid and thorough deployment of commodity technology to help the LHC experiments and HEP in general enjoy unprecedented cost-performance numbers. The dream of a one million CERN Unit computing facility at affordable cost well before LHC start-up will then begin to come true.

References

[1] "Benchmarking Computers for HEP", Eric McIntosh, CN/92/13.

[2] "Is there risk with RISC ?", S.Jarp, Proceedings of CHEP-92, Annecy.

[3] Private communication from T.Bell/IBM.

[4] "Quo Vadis Code Optimisation in HEP", S.Jarp, Proceedings of CHEP-94, S.F.

[5] "HEP Processing using Commodity Components (HEP pc)", M.Delfino et al, CERN/LHCC/95-46, P61/LCRB.

Appendix I

The full set of current benchmarks comparing various PCs and RISC workstations.

Table 1. CERN Unit measurements for selected PCs and RISC workstations.

System	CERN Units
HP-750 (66 MHz)	13.4
PC (133 MHz)	17.7
DEC-3400 (225 MHz)	18.0
HP-712 (80 MHz)	19.9
IBM-RS/590 (67 MHz)	27.1
HP-735 (99 MHz)	28.1
SGI-Challenge (200 MHz)	29.8
HP-735 (125 MHz)	35.4
DEC-3900 (275 MHz)	43.9

Table 2. GEXAM1 timings (in seconds) for selected PCs and RISC workstations.

System	Seconds /event
PC/90 (256 KB Asynch. SRAM)	4.00
PC/133 (256 KB Asynch. SRAM)	2.85
PC/133 (256 KB Pipelined SRAM)	2.37
HP/9000/712 (80 MHz)	2.32
SGI/Challenge (200 MHz)	2.29
IBM CERNSP (67 MHz)	2.21
PowerPC/604 (w/cache optimisation tool)	1.15
HP/735/125	1.09
DEC/8400	0.59

Table 3 DICE timings (in seconds) for selected PCs.

Phase	PC/90 (with 256 KB L2)	PC/100 (with 1 MB L2)	PC/133 (with SPL 256 KB L2)	PC/133 (with 512 KB L2)
Initialisation	224	200	150	151.3
Event 1	782	607	645	579
Event 2	1940	1568	1572	1502
Event 3	1112	834	868	795
Event 4	1263	901	939	863
Event 5	1359	965	989	923
Event 6	1313	937	1003	890
Event 7	1260	898	912	886
Event 8	2350	1668	1706	1606
Event 9	3475	2480	2471	2404
Event 10	895	647	690	602
Average:	1574.9	1150.5	1169.5	1103

Table 4 Symmetric Multi-Processing tests for a HP PC/90 MHz.

Test type	Single job (seconds)	Parallel job (seconds)	Throughput ratio
Gexam1/315 (10 events)	40.3	49.1	1.64
Gexam1/321 (10 events)	48.7	60.7	1.60
DICE/Muon (100 events)	877.4	1003.1	1.75
DICE/HIGGS (1 event)	953.2	1136.0	1,68