Parallel Industrial Applications on Windows NT Clusters

K. Takeda[1], N.K. Allsopp[2], J.C. Hardwick[3], P.C. Macey[4], D.A.Nicole[5], S.J.Cox[5] and D.J.Lancaster[5]

[1] Department of Aeronautics and Astronautics, University of Southampton, UK

[2] Parallel Applications Centre, 2 Venture Road, Chilworth, Southampton, UK

[3] Microsoft Research, St George's House, 1 Guildhall Street, Cambridge, UK

[4] SER Systems Ltd, 39 Nottingham Road, Stapleford, Nottingham, UK

[5] Department of Electronics and Computer Science, University of Southampton, Southampton, UK

University of Southampton

Parallel Applications Centre

High Performance Computing Centre

Microsoft®
Where do you want to go today?®

SER
We Install Confidence

This project has been funded in part as ESPRIT project no. 24871, in the HPCN-TTN Network supported by the EC (see `http://www.hpcn-ttn.org/` *for details), and by the Engineering and Physical Sciences Research Council of the United Kingdom. The authors gratefully acknowledge the support of Microsoft Research, MPI Software Technology Inc. and Genias GmbH. We would also like to thank the HPVM team for their collaboration with us.*

ABSTRACT

In this poster, we discuss performance and deployment of PAFEC-FE VibroAcoustic, the first commercial parallel application to run using MPI on Windows NT. This finite-element/boundary element code is used by Celestion International to perform acoustic analysis of loudspeakers.

By utilising a cluster of existing office PCs running Windows NT, they have been able to improve turnaround time for their simulations from overnight to a couple of hours with little hardware investment. They are now able to run larger test cases than before in a scalable manner. This has significantly streamlined their design process. Another example of PAFEC-FE VibroAcoustic is in the design of sonobuoys, which requires the use of very large grids.

We present performance figures on a variety of Pentium II and Compaq/DEC Alpha-based commodity supercomputer systems using Ethernet, fast Ethernet and Myrinet interconnects. These systems represent configurations typically found in industry and at research institutions.

We describe problems and solutions related to the installation and operation of PAFEC-FE VibroAcoustic at Celestion International on a busy network.

PAFEC-FE VibroAcoustic

The PAFEC-FE VibroAcoustic system, developed by SER Systems Ltd and parallelised by PAC, combines finite-element (FE) and boundary-element (BE) methods and is used widely in industry for structural and vibro-acoustic analysis.

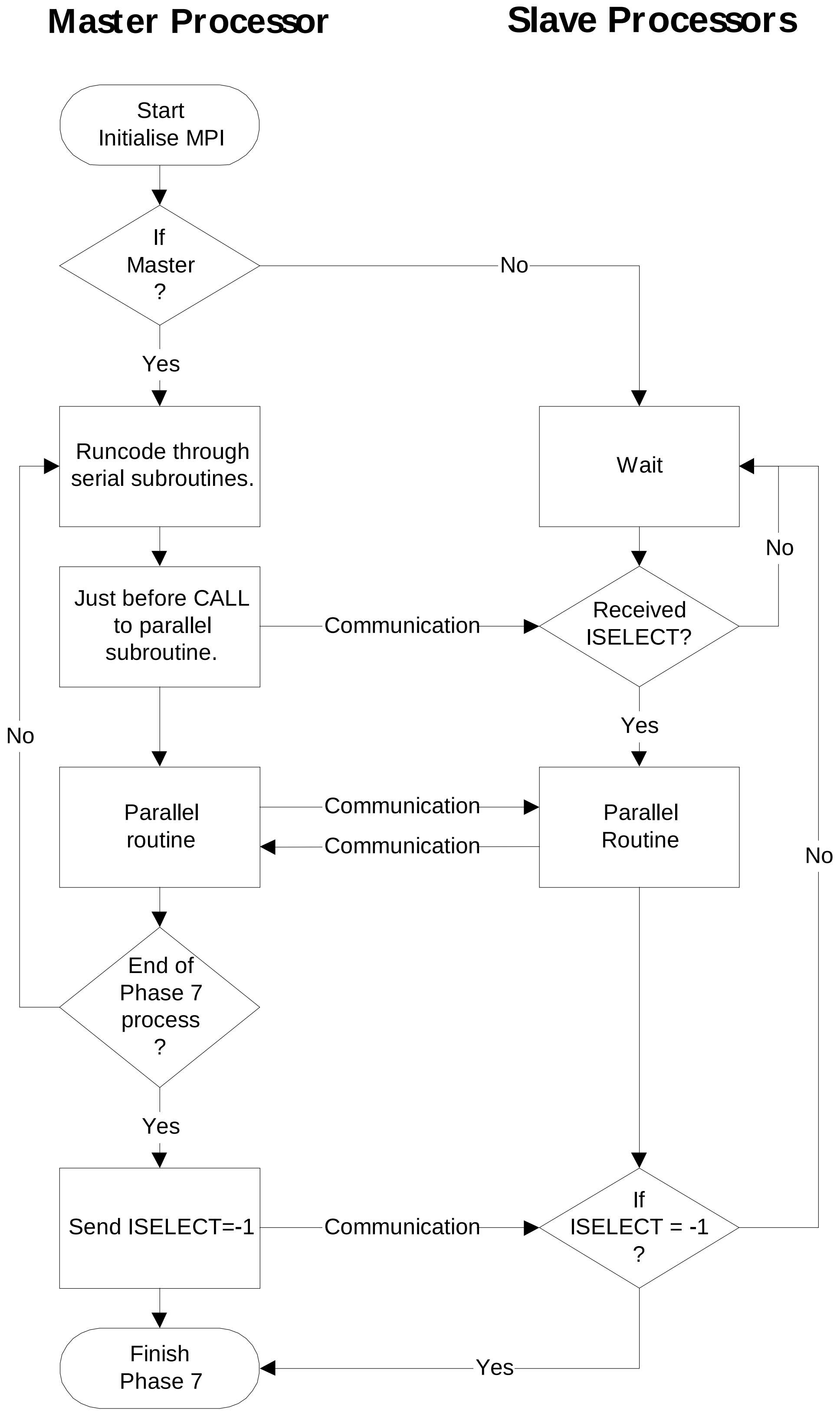
It consists mainly of FORTRAN code with some low-level machine-dependent parts written in C. It is large: there are over *18,000* subroutines containing several hundred thousand lines of FORTRAN in total. Original sections of the code were written in FORTRAN IV, more recently FORTRAN 77 and FORTRAN 90 have been used. The system is still being actively developed on Hewlett-Packard workstations running UNIX.

The PAFEC-FE VibroAcoustic system consists of a suite of programs called *phases*. There are 10 phases in total, phases 1 to 6 perform the pre-processing of the element data of the model to be analysed. Phase 7 carries out the solution of the Finite-Element (FE) and Boundary Element (BE) equations, and is the numerically intensive part of the whole analysis. Phases 8 to 10 handle the plotting and visualisation of the calculated data. Information passes between the phases via files.

Porting the code from UNIX to Windows NT is described in the companion paper at this conference, **Porting Legacy Engineering Applications onto Distributed NT Systems,** Nick Allsopp, Tim Cooper, P. Ftakas, *Parallel Applications Center;* and Patrick Macey, *SER Systems, Ltd.*

In this poster we describe the implementation, performance and deployment of parallel PAFEC-FE VibroAcoustic on Windows NT clusters.

Parallelisation Strategy



A Master-Slave paradigm is used and due to the complexity of the data storage within the PAFEC-FE VibroAcoustic code it was decided to concentrate on parallelising only the numerically intensive sections of the code (*Phase 7)*, shown in the flowchart.

This means that the master processor follows the original serial route through the analysis. As we are dealing with a small cluster of NT workstations, the management of communications is a small task compared with the computation to be done. For this reason the parallel algorithms have been written such that the master acts as a slave process during the numerically intensive sections of the code.

When one of these sections is reached all of the slave processors, as well as the master processor, perform a similar amount of work. This means that the system is load balanced within the parallel sections of the code whilst all of the I/O is dealt with by the master processor only.

Cluster Configurations

In this section we discuss the performance of three different Windows NT 4.0 clusters, each of which represents a different computational environment that might be found in industry.

1. A cluster of eight DEC/Compaq Alpha 500MHz 21164 PCs with 256MB of RAM each, connected by switched Fast Ethernet, and using MPI/Pro and HPVM (which we have ported to Alpha socket networking). This represents a cluster optimized for floating-point arithmetic.
2. A cluster of sixteen dual-processor 300MHz Pentium II PCs with 384MB of RAM each, connected by switched Ethernet and Myrinet, and using MPI/Pro and HPVM. This cluster *(Figure 1)* is optimized for fine-grained problems with a high-performance system-area interconnect.
3. A cluster of four dual-processor 450MHz Pentium II PCs with 128MB of RAM each, connected by switched Ethernet and Fast Ethernet, and running MPI/Pro and WMPI. This represents a "found cluster" that in industry might be composed of existing office machines.

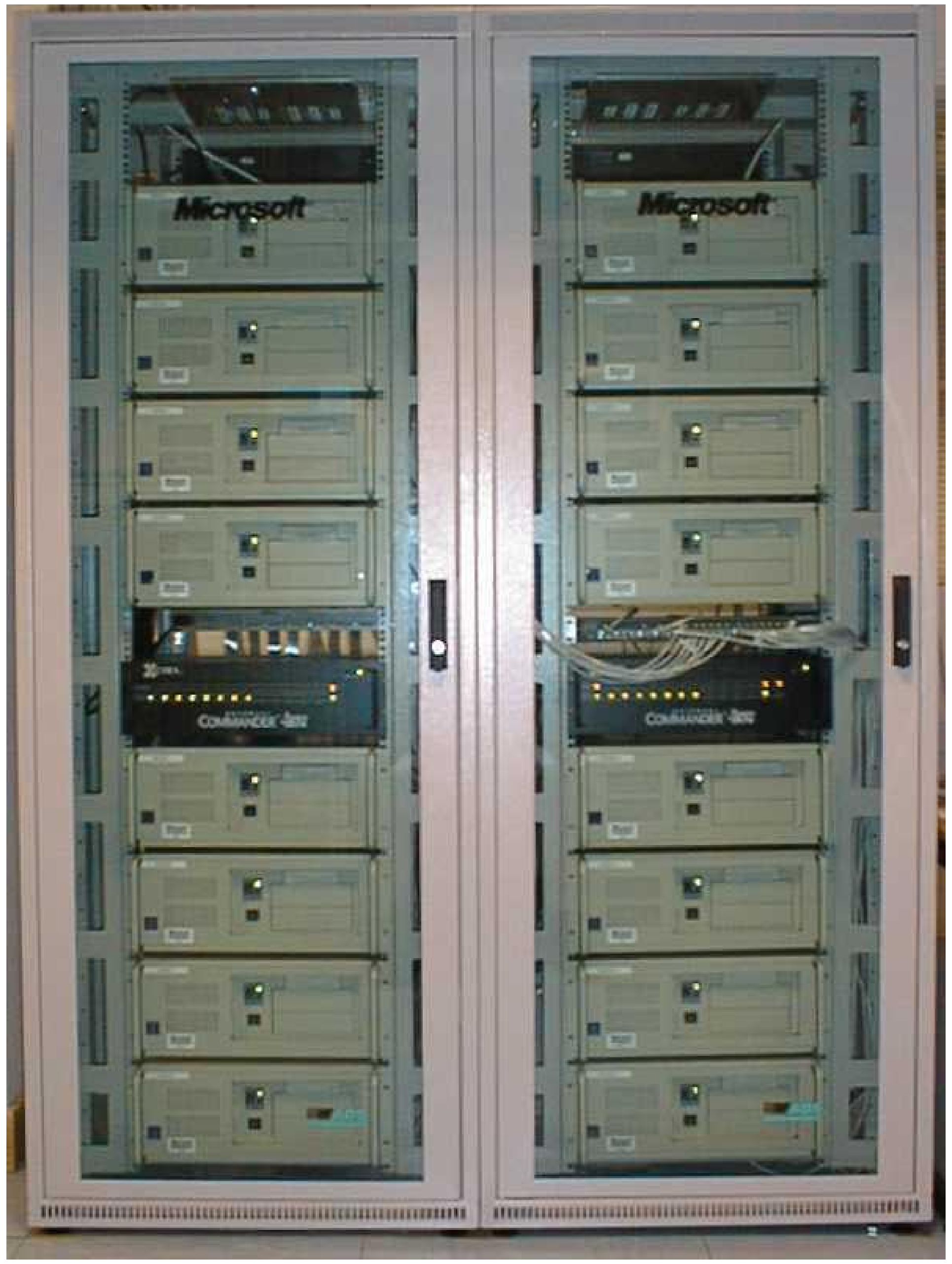


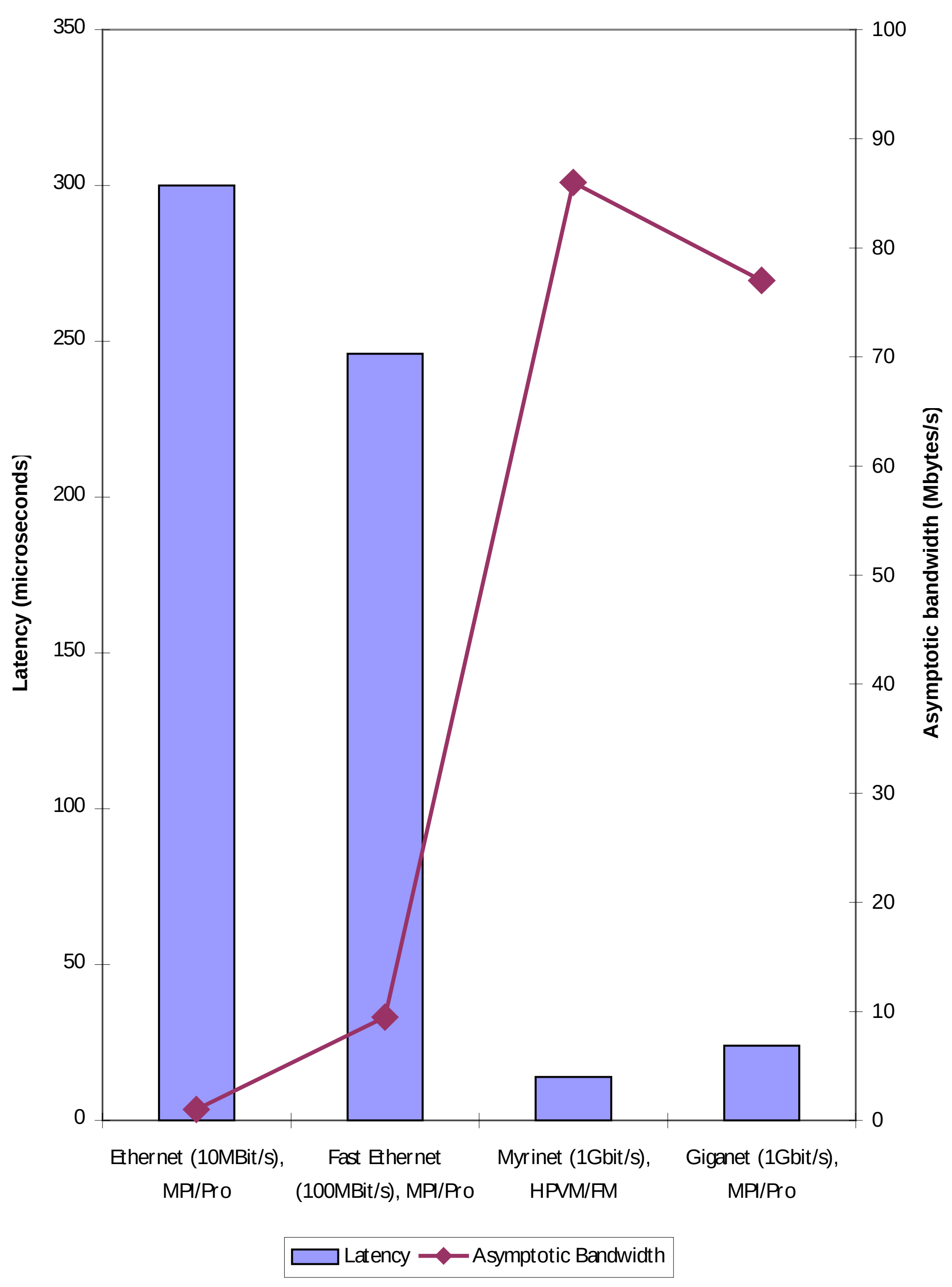
Figure 1. Microsoft Research (Cambridge) Cluster. Sixteen dual 300MHz PIIs connected by Myrinet and switched fast Ethernet

MPI under Windows NT

Most Unix implementations of the Message-Passing Interface (MPI) are derived from the MPICH code base from Mississippi State University and Argonne National Labs. MPICH has been ported to Windows NT resulting in three different implementations.

In this poster we present results using MPI/Pro produced by MPI Software Technology Inc. on Ethernet and fast Ethernet, which supports both TCP/IP and the VIA standard for system-area networks. The Fast Messages project provides HPVM, which supports MPI and other user-level protocols on top of TCP/IP and Myrinet. Results presented here for Myrinet cluster use HPVM. PaTENT WMPI is also available for IA32 and recently for Alpha-based systems. For a detailed comparison of these MPI implementations using real application benchmarks we refer the reader to our paper, *Takeda et al*, **An Assesment of MPI Environments for Windows NT**, *Proc. PDPTA'99*, July 1999.

Low Level MPI performance under Windows NT



Parallel Performance I - Small Test Case

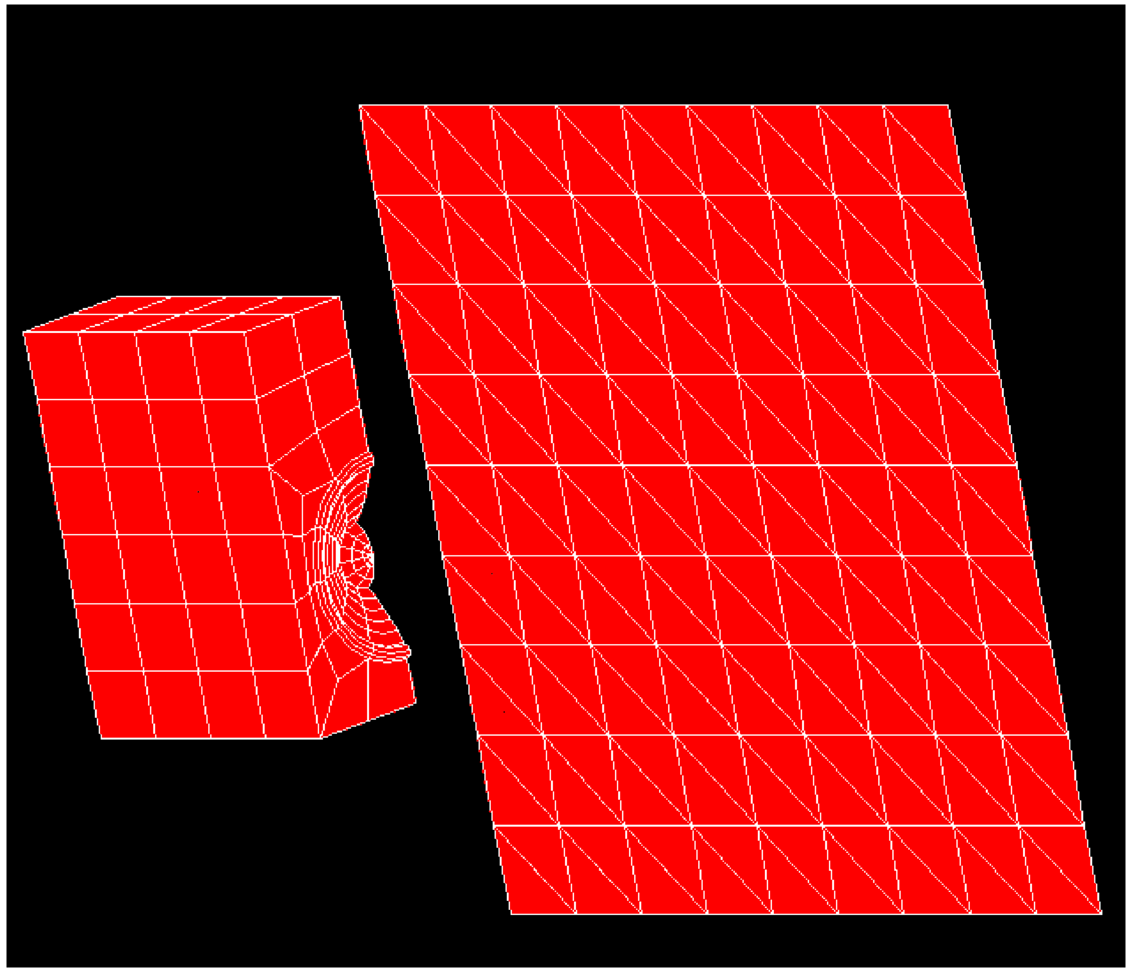


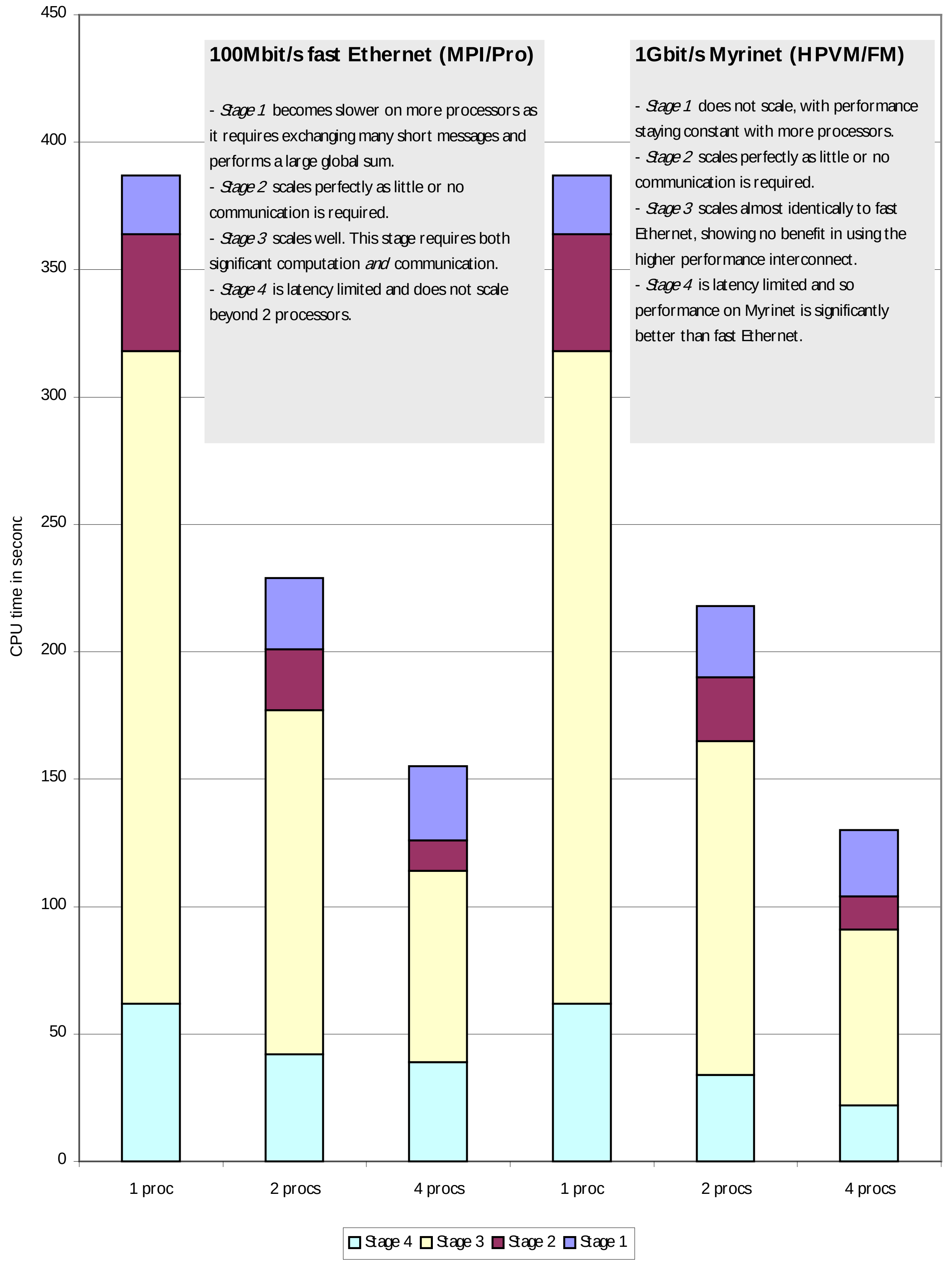
Figure 2. This test case is representative of a typical loudspeaker system under test. The system is a half model. Its parameters are 141 structural elements, 1917 structural degrees of freedom and 914 acoustics degrees of freedom (interior and exterior). Pressure field shown at 125 Hz.

The small test case used for performance benchmarking of the PAFEC-FE VibroAcoustic code is of half a loudspeaker cone, shown in *Figure 2*, and was supplied by Celestion International. This was the largest test case that Celestion were able to run on their individual workstations, but by using a domain decomposition parallelisation approach they are now able to tackle much larger systems.

Performance of *Phase 7* of PAFEC-FE VibroAcoustic is shown here. This is split into 4 stages:

1. Merge contributions from finite elements and reduce resulting shared sparse stiffness matrix.
2. Form boundary element matrices.
3. Reduce boundary element matrices - the most numerically intensive stage.
4. Perform Gaussian elimination with partial pivoting.

Parallel Performance of Loudspeaker Case on 300MHz PII cluster using fast Ethernet and Myrinet



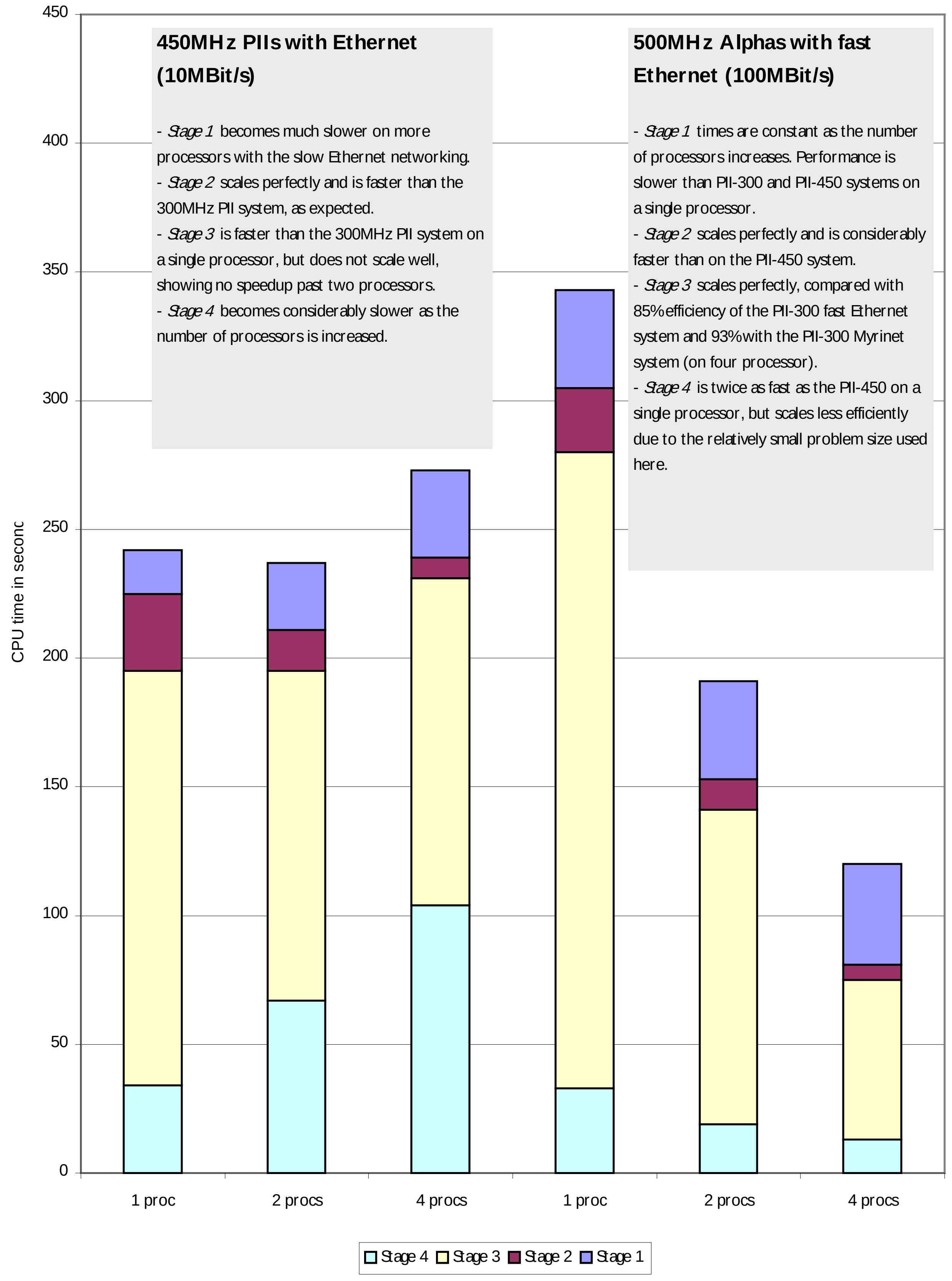
100Mbit/s fast Ethernet (MPI/Pro)

- *Stage 1* becomes slower on more processors as it requires exchanging many short messages and performs a large global sum.
- *Stage 2* scales perfectly as little or no communication is required.
- *Stage 3* scales well. This stage requires both significant computation *and* communication.
- *Stage 4* is latency limited and does not scale beyond 2 processors.

1Gbit/s Myrinet (HPVM/FM)

- *Stage 1* does not scale, with performance staying constant with more processors.
- *Stage 2* scales perfectly as little or no communication is required.
- *Stage 3* scales almost identically to fast Ethernet, showing no benefit in using the higher performance interconnect.
- *Stage 4* is latency limited and so performance on Myrinet is significantly better than fast Ethernet.

Parallel Performance of Loudspeaker Case on 450MHz PII and 500MHz 21164 Alpha clusters using MPI/Pro



450MHz PIIs with Ethernet (10MBit/s)

- *Stage 1* becomes much slower on more processors with the slow Ethernet networking.
- *Stage 2* scales perfectly and is faster than the 300MHz PII system, as expected.
- *Stage 3* is faster than the 300MHz PII system on a single processor, but does not scale well, showing no speedup past two processors.
- *Stage 4* becomes considerably slower as the number of processors is increased.

500MHz Alphas with fast Ethernet (100MBit/s)

- *Stage 1* times are constant as the number of processors increases. Performance is slower than PII-300 and PII-450 systems on a single processor.
- *Stage 2* scales perfectly and is considerably faster than on the PII-450 system.
- *Stage 3* scales perfectly, compared with 85% efficiency of the PII-300 fast Ethernet system and 93% with the PII-300 Myrinet system (on four processor).
- *Stage 4* is twice as fast as the PII-450 on a single processor, but scales less efficiently due to the relatively small problem size used here.

Parallel Performance II - Large Test Case

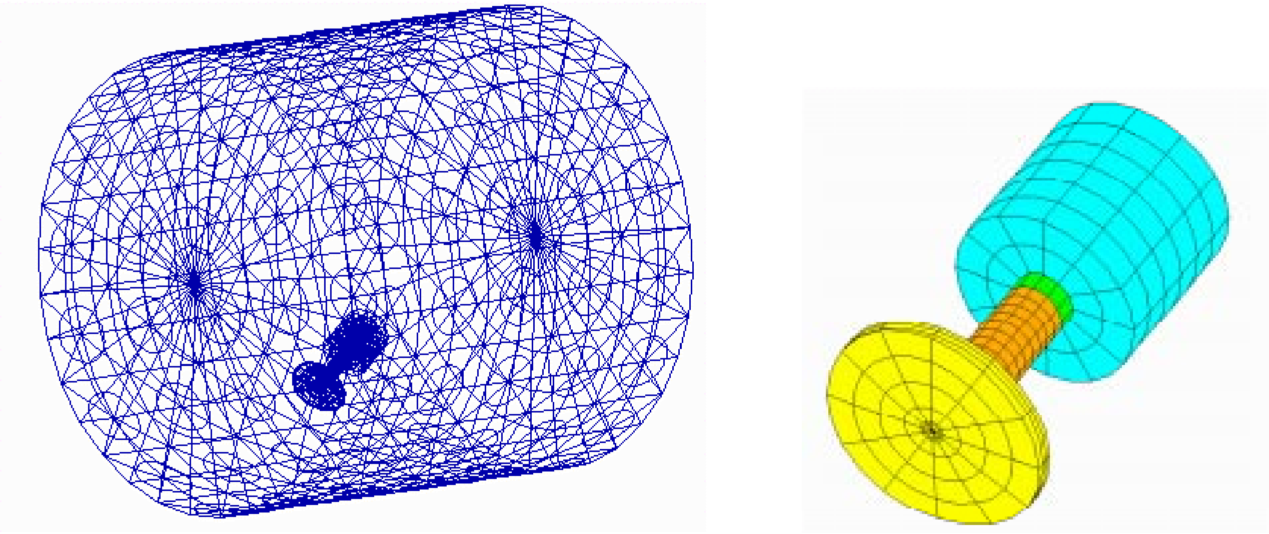


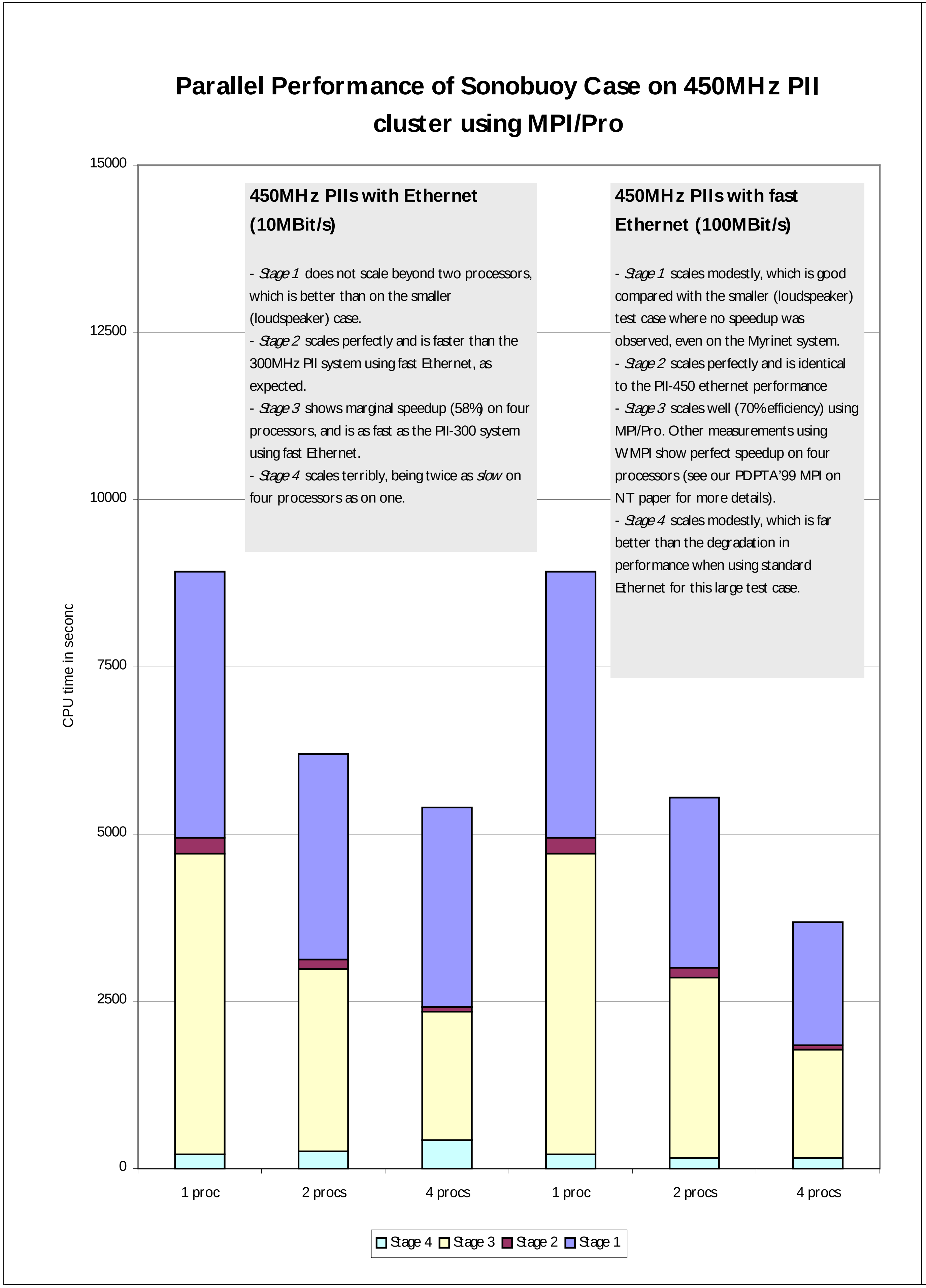
Figure 3. The large PAFEC-FE test case is a finite/boundary element model of a sonobuoy, consisting of a cylindrical baffle, diameter 0.32m, axial length=0.31m (left) containing a piston transducer (right). The piston head is 0.04m consisting of 10923 structural degrees of freedom and a front size 813. The coloration shows an applied electrical excitation.

While the Loudspeaker test case supplied by Celestion was the largest previously run by them on a single machine, it proved too small to efficiently utilise a cluster of PCs.

The Sonobuoy test case shown in *Figure 3* is significantly larger than the loudspeaker case and takes many hours to run on a single machine.

Problems of this size are now feasible due to the domain decomposition approach taken in parallelising PAFEC-FE VibroAcoustic. The user is constrained by the global amount of memory available in the cluster, rather than being restricted to the amount on a single machine. This greatly enhances the designer’s ability to perform analyses that might not otherwise be possible.

As the problem size is large, we expect scalability to improve when compared with the smaller test case. Indeed, the results show that scalability is achieved on all test systems, even when using 10Mbit/s Ethernet.



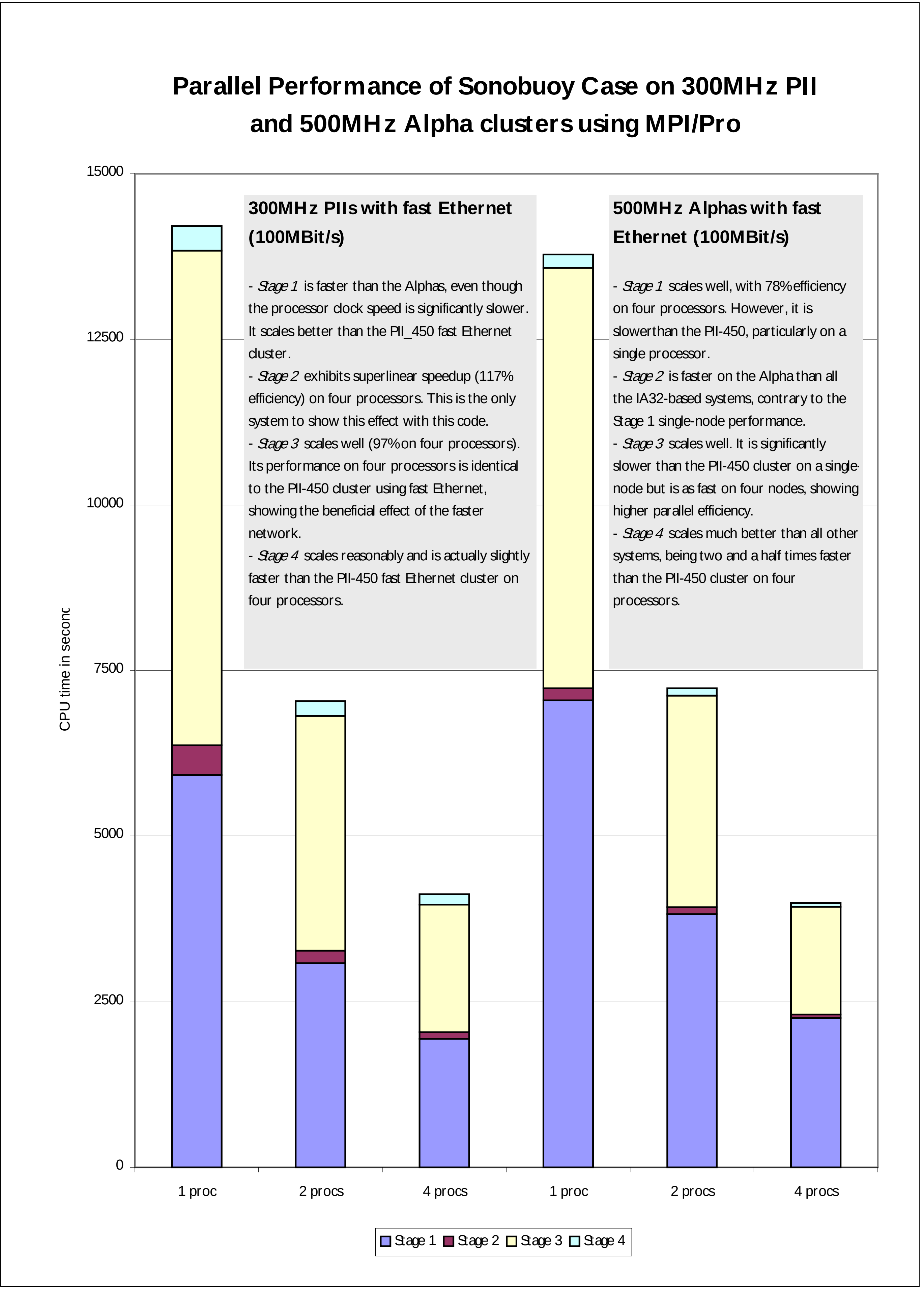
Parallel Performance of Sonobuoy Case on 450MHz PII cluster using MPI/Pro

450MHz PIIs with Ethernet (10MBit/s)

- *Stage 1* does not scale beyond two processors, which is better than on the smaller (loudspeaker) case.
- *Stage 2* scales perfectly and is faster than the 300MHz PII system using fast Ethernet, as expected.
- *Stage 3* shows marginal speedup (58%) on four processors, and is as fast as the PII-300 system using fast Ethernet.
- *Stage 4* scales terribly, being twice as *slow* on four processors as on one.

450MHz PIIs with fast Ethernet (100MBit/s)

- *Stage 1* scales modestly, which is good compared with the smaller (loudspeaker) test case where no speedup was observed, even on the Myrinet system.
- *Stage 2* scales perfectly and is identical to the PII-450 ethernet performance
- *Stage 3* scales well (70% efficiency) using MPI/Pro. Other measurements using WMPI show perfect speedup on four processors (see our PDPTA'99 MPI on NT paper for more details).
- *Stage 4* scales modestly, which is far better than the degradation in performance when using standard Ethernet for this large test case.



Parallel Performance of Sonobuoy Case on 300MHz PII and 500MHz Alpha clusters using MPI/Pro

300MHz PIIs with fast Ethernet (100MBit/s)

- *Stage 1* is faster than the Alphas, even though the processor clock speed is significantly slower. It scales better than the PII_450 fast Ethernet cluster.
- *Stage 2* exhibits superlinear speedup (117% efficiency) on four processors. This is the only system to show this effect with this code.
- *Stage 3* scales well (97% on four processors). Its performance on four processors is identical to the PII-450 cluster using fast Ethernet, showing the beneficial effect of the faster network.
- *Stage 4* scales reasonably and is actually slightly faster than the PII-450 fast Ethernet cluster on four processors.

500MHz Alphas with fast Ethernet (100MBit/s)

- *Stage 1* scales well, with 78% efficiency on four processors. However, it is slowerthan the PII-450, particularly on a single processor.
- *Stage 2* is faster on the Alpha than all the IA32-based systems, contrary to the Stage 1 single-node performance.
- *Stage 3* scales well. It is significantly slower than the PII-450 cluster on a single-node but is as fast on four nodes, showing higher parallel efficiency.
- *Stage 4* scales much better than all other systems, being two and a half times faster than the PII-450 cluster on four processors.

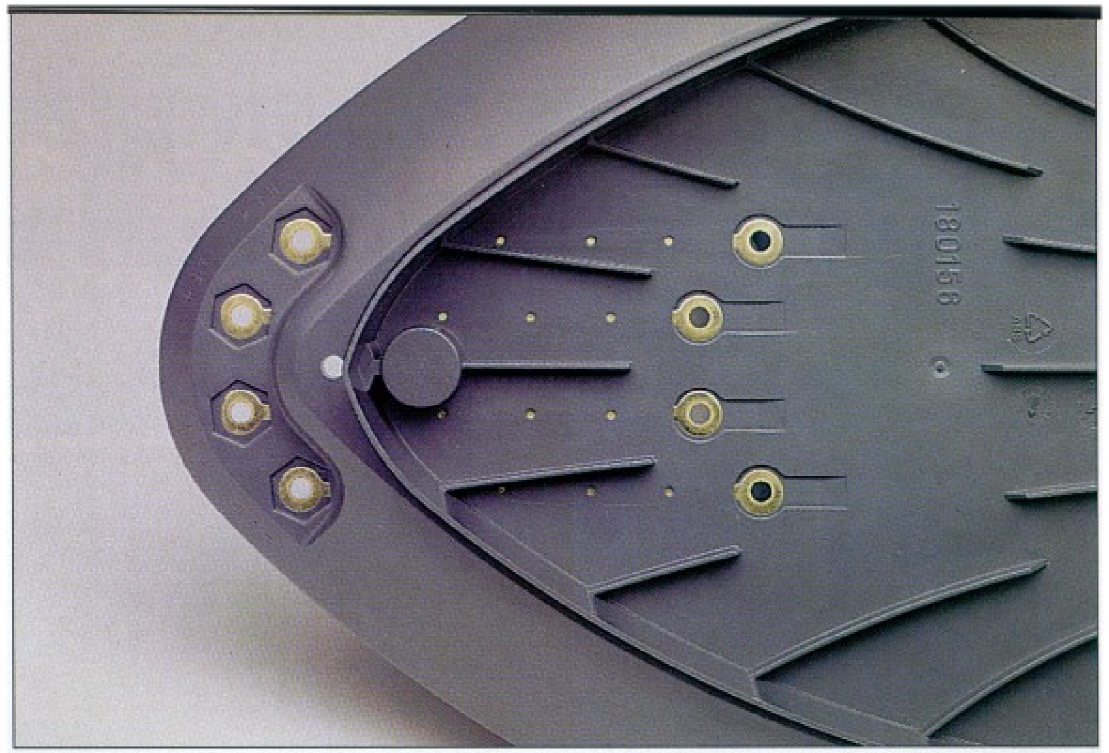
Deployment Issues

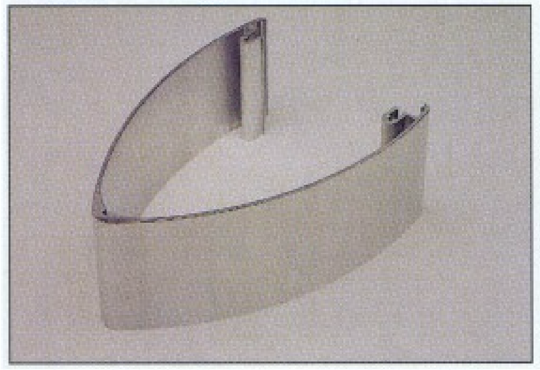
Celestion International is a small business, employing fewer than 100 people, and envisioned using their existing office PCs in an overnight batch mode. Problems of resource scheduling would be minimal, as many of the available PCs were only used during normal business hours. However, the increases in performance and ability to run larger test cases caused them to re-evaluate this practice. They now utilise all available machines during the day, allowing them to increase their design throughput substantially. The activity of the designers now impacts on other office PC users.

One possible solution that is being investigated to alleviate this problem is to upgrade the slave machines to dual processor platforms. The administration staff could use one processor while the other would be used as part of the computational cluster. Out-of-hours, both CPUs could be used by the engineers. This scenario is seen as having the least capital expenditure for the greatest computational gain when taking into account hardware investment, system administration and additional software expenditure. This would allow designers complete freedom in using PAFEC-FE. As a small business Celestion are not keen to install cluster management software due to capital and maintenance costs, in time and money.

Another unexpected problem encountered when deploying the parallel PAFEC-FE code onto the Celestion network was with long MPI messages. It was found that while transferring large amounts of data at the end of Stage 1 there was an intermittent memory error causing the executing job to fail. It was discovered that the error occurred due to insufficient buffering on the Ethernet cards in the presence network traffic. This was happening even though the cluster was connected via a switch, which effectively isolated them from the rest of the company’s network. The solution to his problem was to send a string of smaller massages with a synchronisation step between each message. This compromised the network performance but ensured the stability of the code.

Business Impact





Celestion International is now able to run test cases that used to take several hours over lunch time. This has a significant impact on the way Celestion design their products and ultimately increases their competitiveness. The fruits of this project are already starting to appear in some of Celestion's newest loudspeaker products, examples of which are shown here.

"The changes we made to our loudspeaker designs based on FEA are certainly evident to the discerning ear. I'm optimistic that FEA is reducing our design cycle and this is very important in a highly competitive market… I'm well pleased with the cluster implementation of PAFEC FE."

Julian Wright, Head of Research and Acoustic Engineer, Celestion International, UK

Discussion and Conclusions

In this poster we have shown how a large industrial parallel code has been successfully ported from UNIX to Windows NT and deployed at a customer site with beneficial results to their business.

We have demonstrated that Windows NT clusters can be a cost-effective platform for high-performance parallel industrial applications. The benefits do not only include better performance, but also problem scalability. Celestion are able to use much larger grids than ever before, giving them more flexibility and scope in the design process. The sonobuoy case is larger than any other previously run using PAFEC-FE VibroAcoustic.

Of note is Celestion's reluctance to commit resources to installing a Linux cluster. As a small to medium-size enterprise (SME) they wanted to maximise the use of their existing systems without increasing administration costs. By using existing commodity hardware and software, the entry cost for High Performance Computing is lowered substantially to the extent that SMEs are now able to take advantage of parallel computing to stay competitive.

More details of commodity supercomputing research using Windows NT and PAFEC-FE VibroAcoustic can be found on the authors' web sites:

- High Performance Computing Centre (HPCC), University of Southampton, UK:

  http://www.hpcc.ecs.soton.ac.uk/mpi_nt_f.html
- Parallel Applications Centre (PAC), UK: http://www.pac.soton.ac.uk
- SER Systems Ltd (UK): http://www.seruk.com
- Microsoft Research: http://www.research.microsoft.com/users/jch