

## **Appendix 3. Project Summaries**

### **University of Colorado at Boulder**

#### **Curriculum in High-Performance Scientific Computing**

##### **Principal Investigators:**

Lloyd Fosdick  
Elizabeth Jessup  
Gitta Domik

##### **Project Funding**

This project was supported by the National Science Foundation under a CISE Educational Infrastructure grant, CDA-9017953 (\$600,000) 1990-1993, awarded to a team at the University of Colorado at Boulder (CU Boulder). The PI's were Lloyd Fosdick, Elizabeth Jessup, and Gitta Domik (now at the University of Paderborn, Germany). Carolyn Schauble was also a major contributor to the project.

##### **Overview of Goals and Objectives**

The main goal of this project was to develop a one-year course and supporting laboratory facilities to give undergraduate students an understanding of high-performance computing systems and training in their use.

The specific objectives were:

- To teach undergraduate students how to use high-performance computing systems in scientific and engineering applications.
- To provide an advanced computing environment for undergraduate instruction in particular applications.
- To develop instructional materials: course notes, problem examples and exercises, manuals, and software.
- To export our instructional materials and experience to support undergraduate studies at other institutions.
- To encourage interdisciplinary studies for computer scientists and engineers.
- To encourage research scientists to become involved in undergraduate education.
- To prepare students for work in the field of computational science.

Our evaluation of the project shows that we have largely satisfied these objectives. We have created a local course and supporting laboratory. We have succeeded on some level in preparing students from a variety of fields for practical work in computational science. We have exported our materials and expertise so that students elsewhere may benefit from them. We have involved researchers in the educational process, both as advisors to the project and as guest lecturers in the course. Although the materials are fully accessible to the properly prepared undergraduate student, a local failing is that the course at CU Boulder is often more heavily populated with graduate students than with undergraduates.

##### **Current status and Accomplishments**

The course is based on the philosophy that turning students into effective users of supercomputers, requires

introducing them to the machines through real application problems and not just toy problems or simple numerical examples. The students must learn which architectures are appropriate for which problems and how best to map the problems onto those machines. They must also learn how to determine

the performance of their programs, including how to interpret what the compiler has done and how to use the clock properly. Finally, the students must learn to use the color, perspective, and animation capabilities of suitable visualization tools to display and interpret the large amounts of data typically generated by supercomputer programs.

Thus, the course begins with introductions to the machines and performance measurement. Students also study the computational and visualization tools Matlab and AVS. The concepts and tools learned in the early chapters are then applied to solution of numerical problems in molecular dynamics, advection, and tomography. The machines chosen for this course are representative of MIMD and SIMD distributed-memory multiprocessors and vector computers. At CU Boulder, students presently make use of DEC, HP, and SGI workstations, Sun workstations linked as a distributed-memory MIMD multiprocessor via MPI, and a Cray J916 vector computer. Past offerings have included the now outdated iPSC/2 hypercube multiprocessor, Cray Y-MP, TMC CM-2, and MasPar MP-1. We expect the machines used to continue to change in response to technological advances.

The prerequisites for the course as we teach it are one semester of numerical computation at the undergraduate level and proficiency in Fortran or C programming.

The course has been offered yearly at CU Boulder since AY 1991-1992 in the laboratory was established early in the grant period. It is presently taught several other institutions as well. The students in the course have included undergraduate and graduate students from computer science, applied mathematics, physics, and several engineering disciplines.

The course materials produced include a published textbook and an on-line laboratory manual described further below. These materials and the course itself are designed for upper level undergraduate students.

### **Plans for Remainder of Project**

The project is now complete. Future efforts will concentrate on maintaining and updating the materials.

### **Materials That Have Been Developed**

The course materials include a series of single-topic tutorials, a laboratory manual and accompanying code, short reference guides to subjects such as IEEE arithmetic and some Unix tools, and manuals for the machines and computational tools.

The tutorials, short references, and tool manuals have been published as the textbook L. Fosdick, E. Jessup, C. Schauble, and G. Domik, *An Introduction to High-Performance Scientific Computing*, MIT Press, 1996. All course materials, including the syllabi and some course assignments, are available via anonymous ftp from the directory /pub/cs/HPSC on ftp.cs.colorado.edu or via the WWW at <http://www.cs.colorado.edu/current/courses/materials.html>.

### **Dissemination Activities**

Primary dissemination is by means of the course materials just mentioned. The textbook has been adopted at a small but growing number of universities and colleges. The free materials have had substantially wider spread---in the period October 1992 through early November 1996, there were about 30,000 accesses of our electronic materials from about 52 countries.

Other dissemination has involved direct contact with the interested community. In the summer of 1991, we ran a two-week summer workshop for faculty from the five collaborating institutions named in our grant proposal (U. New Mexico, U. Texas at San Antonio, New Mexico Tech, Fort Lewis College, and Colorado School of Mines). In the summers of 1992 and 1993, we ran two-week summer workshops for 16 students from many different schools and additional two-week workshops for 16 faculty members from different schools.

We have advised several professors not associated with the workshops as they set up HPSC courses in their own institutions. We have also worked with a consortium of faculty alumni of our workshops who are interested in expanding the HPSC course materials. In addition, we have been invited to speak about

the HPSC project and to participate in panels at several international meetings on education and at several universities.

The laboratory funded by this grant played an important dissemination role in its early years. In those times, the course laboratory facilities were also used for two undergraduate aeronautical engineering courses (space structures and fluid dynamics), computer science numerical analysis and graphics courses, a scientific visualization course (developed under this project), and a number of individual student projects. They are now a part of our general undergraduate teaching laboratory, and the machines purchased in 1990 are in line for replacement.

### **Evaluation Activities**

The first round of evaluation was carried out by the representatives of the collaborating institutions named in the original proposal. These professors aided us in the original design of the course and course materials. They were also the participants in our first faculty workshop.

Our grant proposal also named an advisory committee whose members were Dr. Vance Faber (LANL), Prof. Dennis Gannon (Indiana University & NCSA), Dr. Paul Swarztrauber (NCAR), and Prof. Kris Stewart (San Diego State University). Throughout the development of the project, these professionals gave us advice on content, presentation, and techniques of instruction. At the close of the project (September 1993), they prepared an evaluation report.

We have also relied heavily on the evaluations of the workshop participants and the students in the course throughout the years. These evaluations have been quite positive overall. In addition, several of our students (from the workshops and the course itself) have written to say that the material in the course has been valuable to them in their subsequent careers.

Reviews of the textbook are beginning to appear in publications and electronic newsgroups. The greatest criticism of the book is that it is already dated---a problem hard to avoid in a fast-changing field. In 1995, our HPSC textbook and the course based on it were awarded a Department of Energy Undergraduate Computational Science Education Award.

### **Benefits Seen and Expected**

While HPSC is an area of active professional research, our course and textbook were the first to make it accessible to undergraduate students. Our materials have proven themselves useful to many instructors wishing to develop courses in high-performance scientific computing. Unexpectedly, we have also heard from a number of non-academic scientists who have used our materials in preparing themselves for new work in the field.

### **Problems Seen**

Although the course has been very well-received, it is not without its problems. The extremely tight curriculum at UCB (particularly in the engineering disciplines) makes it difficult for students to find time for the HPSC elective. In response to this problem, we reorganized the course so that the first semester is a stand-alone course. Even with this change, our own local offering of the course has attracted a fairly small number of students (6--24 in the first semester course, 2--5 in the second semester). We hope, though, that time will bring us larger enrollments. Other (non-engineering) schools offering the course do report better success with class size.

In addition, the "high-performance" orientation of the course has lead to some problems at smaller schools with fewer resources. The instructors at those schools have been able to get supercomputer time at remote sites, and they report that the materials are accessible to their students. However, they have run into trouble with a lack of fast Internet access and the lack of funding needed to purchase and support workstations and visualization tools (especially the AVS program used in our course for 3D visualization.) The increasing availability of freeware and PC or Mac visualization software and growing Internet availability have helped to lessen these problems over time.

The small school problem was particularly pronounced in our early dealings with the collaborating institutions. We provided each collaborator with a DEC 5000 workstation, an X-terminal, an Exabyte

tape drive, some licensed software (Matlab, Mathematica, NAG Library), and software we developed. Most of them did not have adequate technical support on site for installation of equipment and software. We tried to help them as much as possible through telephone conversations and e-mail. These efforts plus visits they made to our site were successful, but the time spent on this was much more than we anticipated. Visits by persons from our technical staff to the collaborating sites would probably have been very helpful, but we did not have the resources to provide this level of support.

Finally, as in any course involving advanced architectures, keeping the course materials up to date is very difficult and time-consuming. As a result, the content of the course has not changed substantially since the inception of the course. The impact of the aging of the materials varies. For example, the iPSC/2 has gradually become a relic no longer used in research settings. Nonetheless, we continued to use it through the 1996 offering of the course because it is easy to learn to program other, newer distributed-memory MIMD computers after learning to use the iPSC/2. The greater problem stems from the recent and severe downturn in the supercomputing industry: only a few distributed-memory (MIMD or SIMD) computers are still in production. Our moving the materials to a network of workstations in the present offering of the course is one result of that trend. While the application problems are still timely, adding problems from disciplines other than physics would broaden the appeal of the course.

We consider the updating and augmentation of our materials to be a high priority matter. These activities may include extending the visualization exercises to the Internet by means of HTML and Java. Together with the aforementioned consortium of workshop alumni, we have twice sought the funding for such activities but without success. It has proven difficult to find funding programs receptive to a continuation (as opposed to an innovation) grant, and it is our impression that the proposals were not competitive with the higher-tech web-based educational projects that are popular today.