Kris Stewart
Ilya Zaslavsky
Education Center on Computational Science and Engineering
San Diego State University
San Diego, CA 92182
stewart@sdsu.edu, zaslavsk@rohan.sdsu.edu
http://www.edcenter.sdsu.edu

Building the Infrastructure for High Performance Computing in Undergraduate Curricula:

Ten Grand Challenges and the response of the NPACI Education Center

Abstract:

High performance computing (HPC) is a general term used to describe the wide range of hardware, programming techniques and applications that have played an important role in American researchers' pursuit of Grand Challenge Problems. Solid understanding of these ever-evolving technologies, their potential and effectiveness, should be an integral part of undergraduate curricula preparing learners for the work and research environment of the 21st century. As with any curriculum transformation, the incorporation of HPC technologies into contemporary undergraduate teaching faces a series of adjustment problems, in addition to making more visible the traditional pedagogical issues. The new model of conducting scientific investigation calls for multi-disciplinary, data- and computationally-intensive, collaborative research involving teams of scientists with diverse backgrounds and residing in different geographic locations, who use supercomputers and work cooperatively over the Internet. While this model becomes ubiquitous in leading edge research, its replication in education faces several challenges. In this paper, we identify and explore ten such "Grand Challenges," of pedagogical, psychological, organizational, and technical origins.

Promoting the incorporation of high performance computing technologies into undergraduate curricula is the charge of the NPACI (National Partnership for Advanced Computational Infrastructure) Education Center on Computational

Science and Engineering (EC/CSE), created on the campus of San Diego State University in October 1997. This paper reports on the Center's first year experience in tackling this challenging mission, when we experimented with various ways and formats of technology outreach. In this paper, we show that supporting HPC technologies in undergraduate teaching is a multi-faceted effort requiring a specially constructed comprehensive educational infrastructure.

---

1. Introduction

The mission of the NPACI Education Center on Computational Science and Engineering (EC/CSE, or Ed Center) is:

> to foster the incorporation of high performance research tools for scientific investigation into the undergraduate curriculum to better prepare learners for post-Baccalaureate activities where:
>
> - Collaborative, interdisciplinary teams,
> - Sophisticated computer tools and
> - Effective communication among the team members and with others
>
> are used in research and problem solving.

Curriculum changes follow scientific and technological trends, but they also require a careful consideration of the goals and capabilities of the educational institution, the composition and changes in student population, and other factors not directly related to progress in computing. It is important to emphasize the role of institutions and institutional infrastructure in curriculum transformation, which is especially important in the case of the most demanding (in terms of effort, time, and other resources) technological innovations such as high performance computing and networking. In this paper we will identify and describe ten challenges of transforming undergraduate curricula toward better interface with high performance computing. We will also describe the activities of EC/CSE in response to these challenges, which, taken together, lead to the development of a comprehensive educational infrastructure for HPC within San Diego State University and the California State University System.

2. Ten challenges, and the response of the Ed Center

2.1 University faculty system of rewards does not encourage investing much effort in teaching innovations

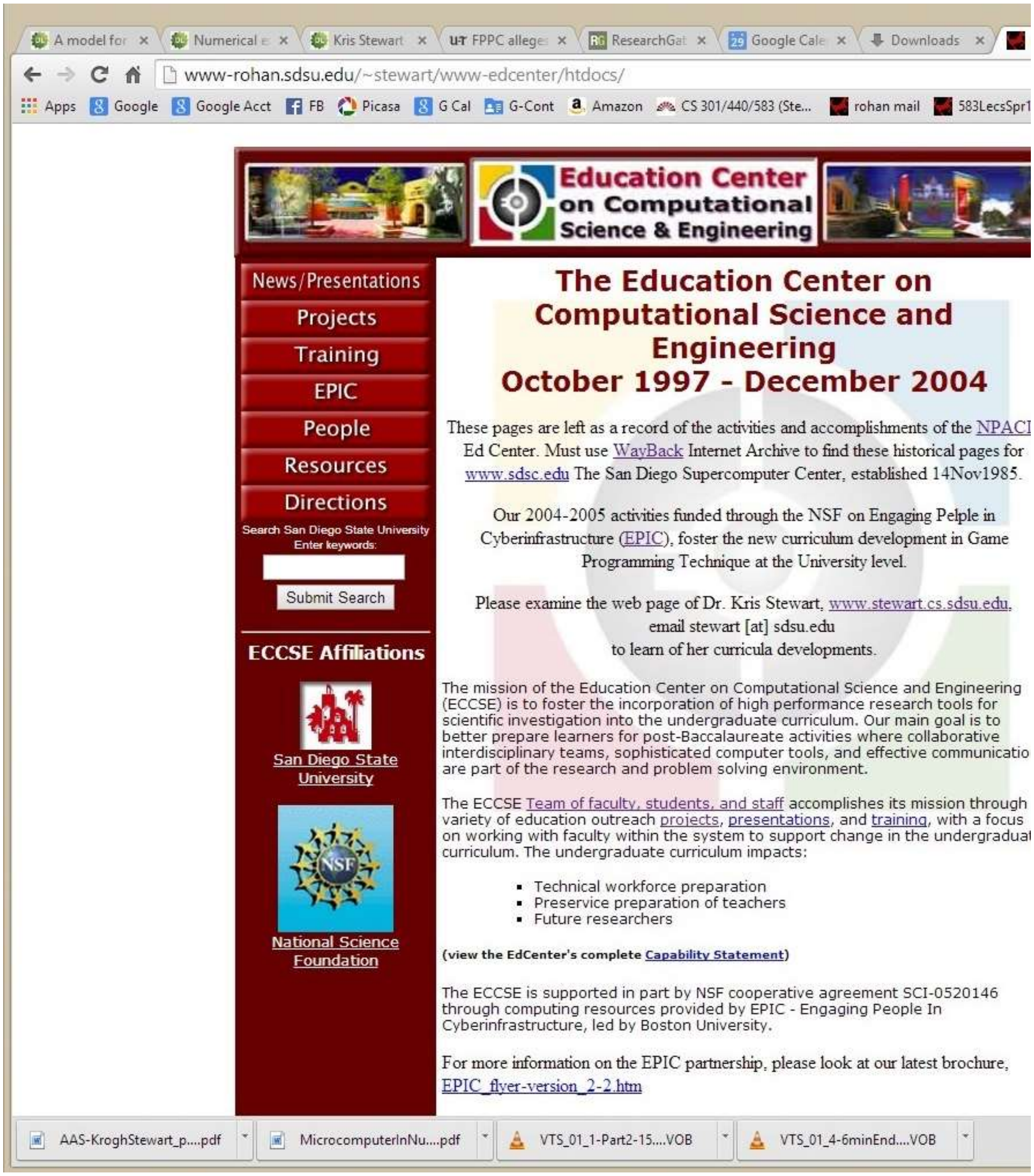
Supporting and rewarding labor intensive curriculum development adequately is a pervasive problem in the contemporary university organization. This is especially true in colleges where faculty promotion and tenure depend primarily on research production and only secondarily on teaching accomplishments (Marchant and Newman, 1994; The Boyer Commission, "Reinventing Undergraduate Education," Way #9, 1998). Even if a faculty member uses high performance computing technologies in his/her research, it may look too adventurous and unpredictable to introduce them in instruction, while invariably requiring much time and effort. In dealing with this problem, the efforts of the Ed Center were focused on promoting an alternative reward system for SDSU and CSU faculty. Participation in the Ed Center's Faculty Fellows program, a project jointly sponsored by university Colleges, and the EC/CSE, provides instructors with released teaching time for HPC curriculum development. Interested faculty submit proposals to the Ed Center describing their current curriculum which they identify as a candidate for NPACI/NCSA technology enhancement. The applications are evaluated by the Ed Center and the four sponsoring College Deans based on originality, established links with NPACI or NCSA researchers, the potential long term impact on curricula and the individual College's ability to provide instructional support. Selected faculty are rewarded with release time and additional support from the Ed Center staff through bi-weekly meetings to share their progress.

2.2 Faculty are commonly unaware of the accessibility of HPC technologies already applied in their fields of research and teaching

Apart from several remarkable accomplishments, such as Mosaic (originally developed at NCSA) and VRML-based scientific visualization (supported to a large extent at SDSC), the application of supercomputing approaches and technologies has not yet left a permanent and consistent imprint on undergraduate curricula. One of the reasons is simply a lack of information among faculty about easily transferable (or already used) technologies developed within the realm of high performance computing and networking. Therefore, one of the main priorities of the Ed Center during its first year of operation has been dissemination of information about HPC technologies to faculty and instructional support personnel of SDSU and CSU. We used a variety of outreach channels and mechanisms, including

- regular presentations on the new research tools developed by NPACI and NCSA partners to SDSU and CSU Colleges and departments,
- publications in CSU and SDSU press,
- demonstrations and workshops, and
- development of Ed Center's WWW pages (Figure 1).

In this latter effort, the development of discipline-specific lists of NPACI and NCSA research tools applicable for undergraduate curricula, and placing them on the Web, along with catalogs of other computational resources, such Java and VRML, proved most useful for the faculty. At the same time, while the Ed Center Web pages provide the most consistent and complete information about supercomputing in undergraduate curricula for SDSU faculty, more pro-active approaches, especially presentations, workshops, and personal contacts, have produced the best results.



The Education Center on Computational Science and Engineering
October 1997 - December 2004

These pages are left as a record of the activities and accomplishments of the NPACI Ed Center. Must use WayBack Internet Archive to find these historical pages for www.sdsc.edu The San Diego Supercomputer Center, established 14Nov1985.

Our 2004-2005 activities funded through the NSF on Engaging Pelple in Cyberinfrastructure (EPIC), foster the new curriculum development in Game Programming Technique at the University level.

Please examine the web page of Dr. Kris Stewart, www.stewart.cs.sdsu.edu,
email stewart [at] sdsu.edu
to learn of her curricula developments.

The mission of the Education Center on Computational Science and Engineering (ECCSE) is to foster the incorporation of high performance research tools for scientific investigation into the undergraduate curriculum. Our main goal is to better prepare learners for post-Baccalaureate activities where collaborative interdisciplinary teams, sophisticated computer tools, and effective communicatio are part of the research and problem solving environment.

The ECCSE Team of faculty, students, and staff accomplishes its mission through variety of education outreach projects, presentations, and training, with a focus on working with faculty within the system to support change in the undergraduat curriculum. The undergraduate curriculum impacts:

- Technical workforce preparation
- Preservice preparation of teachers
- Future researchers

(view the EdCenter's complete Capability Statement)

The ECCSE is supported in part by NSF cooperative agreement SCI-0520146 through computing resources provided by EPIC - Engaging People In Cyberinfrastructure, led by Boston University.

For more information on the EPIC partnership, please look at our latest brochure, EPIC_flyer-version_2-2.htm

Figure 1. The Ed Center Web site

Perhaps, the most dramatic way to demonstrate the capabilities of high performance computing tools and technologies in undergraduate teaching is to use them in our own courses, and survey and publicize the outcome. Kris Stewart, Director of the EC/CSE, has taught CS 575 Supercomputing, a computational science class where students with diverse backgrounds conducted computational experiments using the platforms at the San Diego Supercomputer Center (Cray T90 in Spring 1998, Cray C90 and Cray Y-MP in the previous years). During the Spring 1999 semester, the Network of Workstation (NOW) system developed at UC Berkeley will be used through both NPACI and the SDSU College of Engineering where the NOW cluster has been recently installed. Instructor's goals for the course are to facilitate student "learning through discovery" and acquiring skills needed in computational science, which include scientific problem-solving skills, the effective use of high performance computers, and oral and written presentation skills.

The pedagogical model for this course, which only expects students to have some programming experience as a prerequisite and does not limit enrollment based on the major, is group problem solving in a variety of computationally intensive fields. An important distinction of the problems offered to students, is that their solution requires a combination of skills from a variety of disciplines. For instance, approaching problems in computational biology requires knowledge of linear algebra (a part of math curriculum), numerical analysis (applied math curriculum), and biology itself (though biology curriculum at this point does not include development of many advanced computational tools). In the process of researching a problem in groups, students design and implement computational experiments on the local campus UNIX mainframe to gain familiarity with the stated problem and the effectiveness of performance-measurement tools, such as the CPU timer. Preliminary written reports are prepared to document the group's progress in evaluating the performance of their program. Later in the semester, the codes are ported to a supercomputer, and further investigations of performance, now between two different platforms, are prepared with final group written and oral reports. Every other week, a portion of class lecture time (up to 40%) was spent on mediated group discussions.

Another series of EC/CSE experimental classes was taught by Ilya Zaslavsky, the Ed Center's GIS staff scientist. Zaslavsky used Web-based collaborative technologies to teach classes in geographic information systems (GIS) and spatial analysis from EC/CSE to geography students at Western Michigan University in real time. The lectures were delivered via desktop audio-video conference, which included sharing graphics and Windows applications (with NetMeeting, Microsoft's videoconferencing and application sharing software). The asynchronous part of the course relied on Web-based lecture notes, and Web-based discussion of lecture content.

This experiment demonstrated numerous challenges of real-time lecturing over the Web. Such problems as perceived lack of communication with the instructor and loss of eye contact, especially when student confusion is difficult to verbalize, will likely remain important, despite the eventual improvement in the reliability and quality of synchronous Web-based communication, as distance learning becomes one of the mainstream operation modes for universities (Kearsley, 1998). The results of this experimental teaching during the Fall 97 and Winter 98 semesters, including student perception of the learning environment, are summarized in Zaslavsky and Baker, 1998, and - from a scientific visualization perspective - in another paper on this CD (David Emigh, Ilya Zaslavsky, "Scientific Visualization in Undergraduate Classroom").

As an extension of the experiment in synchronous distant teaching from a desktop, we installed and evaluated several software packages for Web-based collaboration, including Tango (developed at the Northeast Parallel Architectures Center, Syracuse University), and Habanero (developed by the National Center for Supercomputing Applications, University of Illinois at Urbana-Champaign.) The comparison of distance learning tools, and our teaching experiences were presented as a workshop for SDSU faculty, and resulted in at least one (at the time of writing) external grant application from SDSU faculty proposing to use this technology for inter-campus teaching (Mellors and Templeton, 1998).

2.3 Faculty and students not aware of benefits and accomplishments of supercomputing

To enhance faculty awareness of HPC technologies, and following the pro-active approach, we initiated a special program called "NPACI Hours." In coordination with course instructors, Ed Center staff present HPC technologies to students during regular lectures, selecting and tailoring the description to a particular discipline and the background of the audience. We anticipate that such hour-long sessions will become a consistent part of the courses and eventually will be taken over by regular faculty. A collection of such "NPACI Hours" modules will be placed on the Web and later distributed to other educational institutions within the CSU system as well as on a national scale.

Following the recommendations of the Boyer Commission (Way to Change #2), for an inquiry-based Freshman Year, a new experiment has begun. Stewart is teaching the SDSU University Seminar for Freshman Success, designed for incoming freshmen who have already declared Computer Science to be their major. At a large, minority-serving University, such as SDSU and most other California State University system campuses, we have found the continuity from high school to be paramount. The continuity in instruction provided to support computational science is evidenced in the Supercomputer Teacher Enhancement Program (STEP), a high school program described in detail in San Jose at SC 97. This program continues through the voluntary participation of roughly twenty secondary science and math teachers in San Diego and exemplifies the integrated education experience tying school teachers and University faculty. The undergraduate Bridging Environments, described in Section 2.5, continues building the holistic computational science education environment.

2.4 HPC technologies are considered too complex and inaccessible for undergraduate instruction

This is a common problem with any new technology entering undergraduate education and with supercomputing tools in particular. However, the on-going change of computing paradigms, from the focus on individual workstations to Web client/server organization and distributed computing, is likely to accelerate the process of adopting HPC technologies in education. With new instructional interfaces to computational resources in chemistry, biology, and other fields, numerous searchable databases, rich multimedia content - all available through the Web browser interface - the power of supercomputers can be "unlocked" from many classroom workstations. Our efforts at the Ed Center focus on user-friendly Web interfaces and tutorials for computational resources in various subject fields, including social sciences, biology and chemistry. Demonstration of our own teaching experiences using high performance technologies, also helps change this stereotypical perception of supercomputing as an inaccessible technology.

2.5 Due to focus on locally available resources, a successive set of courses preparing students for HPC instruction is typically needed

Computational science curriculum should be built with gradual increase in complexity, leading from simple computational problem solving on the desktop to experiences in supercomputing and distributed computing. In reality, computing platforms used in many undergraduate courses tend to be PCs and Macintoshes, with relatively simple GUIs and focus on stand-alone applications. It is important, however, that the software used in these courses have some elements preparing learners for future instruction and use of supercomputers. We identified such intermediate courses and software as **"bridging environments."** An example of a bridging sophomore-level SDSU course is CS 205 "Computational Problem Solving and Visualization," also taught by Kris Stewart. It is an introductory course focused on the modern approaches and techniques for computational science. Students develop computer and problem-solving skills preparing them for subsequent classes in high performance computing introduced at a senior level. The main distinctions of this class as a bridge to high performance computing curricula are: (1) focus on programming tools and approaches with comparable problem-solving methodology though at a smaller scale; (2) students are encouraged to work with software that is commonly used in real problem-solving situations with complexity comparable to high performance computing tasks. The main programming environment used in the class is MATLAB, a visualization and analytical package available on PC, Macintosh, and UNIX platforms. The modular organization of MATLAB, object orientation, and a variety of available advanced visualization tools are a good representation of the arsenal of modern computational science. A similar toolkit, though for much more computationally demanding tasks, will be used later in the high performance computing curriculum.

We believe that additional research is required to identify the most appropriate bridging environments which prepare students, both technically and methodologically, for the paradigms of high performance computing and networking.

2.6 Curricula using very large data sets in science disciplines are not developed

Regular use of large datasets and high bandwidth networks, and collaborative multidisciplinary inquiry based on intensive computation and visualization, which we typically associate with supercomputing, have not yet become an integral part of most undergraduate courses. We have previously mentioned some of the reasons for this. Many courses, even in computationally intensive fields, use simple datasets and laboratory assignments with predictable results, which makes assessment of student progress easier. At SDSU, we focused on several approaches to solving this problem, through cataloging NPACI and NCSA resources usable in an undergraduate setting, and the "NPACI Hours" program. To make the process of scientific discovery more realistic, we promote the use of real-world datasets in undergraduate instruction, such as those found in digital libraries which are mirrored at SDSC (the Alexandria Digital Library; the UC Berkeley Elib, Stanford InfoBus project, and University of Michigan Digital Library). An overview of the uses of digital libraries in undergraduate instruction, with links to successful case studies, is available from the Ed Center's Web pages. Digital libraries are a promising direction introducing supercomputing to new communities, such as economists, sociologists, geographers, etc., and a way to expand the HPC educational infrastructure (actively pursued by the Ed Center) beyond traditional science departments.

2.7 Differences in learning styles becomes especially important when material is complex

The role of learning styles, and the importance of taking their variety into account when designing undergraduate curricula, have been demonstrated in several recent studies (for example, Dunn and Stevenson, 1997; Bell, 1998). SDSU is a minority-serving institution, with no ethnic group exceeding 50% of enrollment (1997). With such a diverse student population, attending to different learning styles becomes a priority in any curriculum development. In instruction with supercomputing technologies this is especially important due to the psychological barrier created when a new technology is perceived as extremely complex. In solving this problem, the Ed Center focuses on various scientific visualization technologies (Gordin and Pea, 1995), which allow students to visualize and explore large scientific datasets, thus avoiding or postponing formal mathematical descriptions. This road to understanding is considered beneficial for differently-abled students, since it introduces them to scientific concepts and helps them understand complex natural phenomena when more traditional curricula fail.

The group-working paradigm, the strategy pursued in the CS 575 Supercomputing course, referred to in Section 2.2, is another way to engage students with different learning styles. Several surveys during the course of the semester showed that group composition, and the organization of group work were extremely important factors in student learning of supercomputing. A variety of attitudes toward group work is demonstrated in Table 1 which summarizes student responses to the same set of questions offered at the beginning (first survey) and the end (last survey) of the semester.

Table 1: CS 575 Group Learning Experiment in Supercomputing

(absolute counts for the first and the last surveys)

Statement	Strongly agree		Mildly agree		Undecided		Mildly disagree		Strongly disagree	
	first	last	first	last	first	last	first	last	first	last
I enjoy working in groups	12	10	15	11	1	3	2	5	1	1
I often work in groups	9	9	12	12	4	2	5	6	1	1
Group decision making is important to societies and organizations	20	18	7	10	1	0	3	0	0	1
I prefer to work alone rather than in groups	3	2	12	10	3	6	5	9	6	2

I am comfortable in leadership roles	4	4	10	15	9	4	5	4	3	3
When I am working in a group, I usually participate actively	13	15	10	14	3	0	2	1	2	0
When I have to work in a group, I do my share but not more	5	2	6	7	3	4	11	8	7	9
I dislike being evaluated based on group work	2	3	8	10	10	7	5	4	5	6
I am a good judge of other people	6	5	9	10	9	13	2	1	4	1
I am good at reading (interpreting) other people	5	4	13	14	8	9	2	2	2	1
I feel that I have important things to say when I work in groups	7	11	9	8	8	8	4	2	2	1
I feel that my contribution to group work is valued by the other members of the group	10	10	9	12	7	7	2	1	2	0

2.8 Variety of platforms and incompatible software lead to fragmentation of curricula

At least three platforms are in common use in educational institutions at the college level: Mac OS, Windows, and UNIX. It is always a significant pressure on students when they need to switch to a different operating environment. We made a conscious effort to support all three environments, focusing on tools that operate under all of them, such as MATLAB (an analytical and visualization package), ArcView GIS (a geographic information system), Java, Netscape Web browser, and similar environments and tools implemented on all three systems. Supporting and promoting three-way multi-platform software tools in undergraduate curricula will ease student migration between platforms and create a smooth transition between computational science courses taken in different disciplines.

This is consistent with the goal to support an integrated education experience for the next generation of computational scientists. The high school computing environment is dominated by Mac OS and Windows. The HPC research environment is dominated by UNIX workstations and an increasing number of Windows NT computers. Software interoperability is a necessary component of a consistent curriculum, from the secondary through post-Baccalaureate. As discussed in Sections 2.3 and 2.4, we attempted to support continuity of supercomputer experiences for students from high school to the undergraduate senior level. High school science curricula, as exemplified by STEP, feeds into the SDSU University Seminar for Freshman Success, followed by the Bridging Environment of CS 205 Computational Programming and Visualization, with the Capstone course (suggested in the Boyer Report, Way #7) of CS 575 Supercomputing. The logical succession of computing platforms is an important component in this sequence.

2.9 University administrators and support staff are not ready for intensive use of computing and networks

Involving University administrators and support personnel is a necessary part of the equation, due to additional administrative efforts and support workloads related to the use of supercomputers and high-speed networks. It is very important that University administration understand the benefits and support the incorporation of high performance computing tools in the curriculum. At SDSU, a significant part of our effort is directed toward engaging this group, through presentations, publications in SDSU and CSU press, tours of SDSC (San Diego Supercomputing Center, NPACI's Leading Edge Site), demonstrations of NPACI and NCSA accomplishments in research, education and outreach. This involvement resulted in cooperative sponsoring programs with College Deans, such as the Faculty Fellows program, and NPACI Hours.

2.10 Technical parameters of computers and networks are typically below expectations

While supercomputing instruction, or curriculum with supercomputers, are not possible without modern computer equipment, we present this issue last. While it is very difficult for an average college, especially if it is not a Research I University, to purchase and maintain its own supercomputer, there are several less expensive options, in part developed within the National Science Foundation PACI programs. One example is the SDSU College of Engineering replication of the UC Berkeley Network of Workstations (NOW) system, which was initially facilitated by the Ed Center. Another example is the SDSU vBNS connection, a CSU grant approved this year by NSF and coordinated by the Ed Center for the SDSU campus. Participation in education technology decisions, influencing University computing policy, supporting networking initiatives such as vBNS - all these activities, which at first glance seem quite remote from curriculum development, turned out to be extremely important in our movement toward the larger goal: transformation of undergraduate curriculum with high performance tools and technologies for the benefit of present and future students at SDSU and CSU.

3. Putting it all Together: the Infrastructure for HPC in Undergraduate Education

Outlining this one-year course of action, we find that addressing the above challenges requires a comprehensive approach, constructing a unified infrastructure for incorporating high performance computing tools and technologies into undergraduate teaching. This infrastructure encompasses several critical components beyond just computers and networks. The most important parts of it are information and human infrastructure, faculty education and collaboration, freedom for experimentation with various curriculum formats, and personal connections with faculty and University administrators. The development of infrastructure goes through several stages, from dissemination and building awareness of HPC technology among administrators, faculty and students, to actually using the opportunities that HPC and computational science provide.

We believe that the challenges we face are common for many educational institutions. This outline of challenges and attempts to address them within one University system, can be used as a model for the national education community, for the creation of an infrastructure of Regional Education Centers modeled after EC/CSE.

REFERENCES

Bell, James, 1998.
Problems in Statistics: Learning Style, Age, and Part-Time Students. Education, Summer 1998, Vol. 118 (4), pp. 526-528.

Boyer Commission on Educating Undergraduates in the Research University, 1998. (The Boyer Report)
Reinventing Undergraduate Education: A Blueprint for America's Research Universities. http://notes.cc.sunysb.edu/Pres/boyer.nsf/

Dunn, Rita, and Joseph Stevenson, 1997.
Teaching Diverse College Students to Study with a Learning-Styles Prescription. College Student Journal, September 1997, Vol. 31 (3), pp. 333-339.

Gordin, D. N., and R. D. Pea, 1995.
Prospects for Scientific Visualization as an Educational Technology. The Journal of the Learning Sciences, 4(3), pp. 249-279.

Kearsley, Gregory, 1998.
Distance Education Goes Mainstream. THE Journal, May 1998, Vol. 25 (10), pp. 22-24.

Marchant, Gregory, and Isadore Newman, 1994.
Faculty Activities and Rewards: Views from Education Administrators in the USA, Vol. 19 (2), pp. 145-152.

Mellors, Robert, and Mary Templeton, 1998,
Teaching Seismic Processing and Interpretation: a Synchronous Intercampus Approach. Collaborative NSF Grant Proposal (SDSU, CSU-Fullerton).

Stewart, Kris, and Janet Bowers, 1997.
STEP: A Case Study in Building a Bridge between HPC Technologies and the Secondary Classroom. Proceedings of the ACM/IEEE SC97 Conference, San Jose, CA, 11/15-21, 1997 www.supercomp.org/sc97/proceedings/EDU/STEWART/INDEX.HTM

Zaslavsky, Ilya, and Kathleen Baker, 1998.
"The Experimental Web-based Interactive Environment in Distance Teaching of GIS and Geographic Data Handling" Proceedings of the 1998 ACSM Annual Convention and Exhibition (Baltimore, MD, March 1-5), Vol. 2, pp. 437-446.

Zaslavsky, Ilya, Stewart, K., Li, X., Baker, K., 1998.
"Collaborative Web-based Technologies: a comparative review and live demonstration, with special focus on distance learning opportunities". www.edcenter.sdsu.edu/training/disted/collabpres.html

SC '98, Orlando, FL, USA