conduit!

Volume 11, Number 1 | Department of Computer Science, Brown University | Spring, 2002

PERVASIVE PROGRAMS

Steve Reiss

Introduction

We envision a time in the not too distant future when all applications are handled by a single program that runs over all machines simultaneously. Such a program would be continually evolving, growing, and adapting. It would be the logical extension of today's combina-tion of Internet technology, pervasive computing, grid computing, open source, and increased computational power. Moreover, it would significantly change the way we think about and do programming.

Programs have traditionally been standalone entities. From back in the days of card decks, programs were things that had a definite starting time and stopping time. They were run by individual users to accomplish some task and then they were exited. Sharing among users was done in controlled ways, first through files, later through databases. Even long-running programs such as operating systems or database services are thought of as individual entities, running on their own machine (or small set of machines), that happen to provide services to other programs.

New and current technology lets us think of programs in a different way. Rather than everyone running their own program, we can think, at the extreme, of there being only one program that runs on all computers simultaneously. Users wanting to access information or do some computation simply provide this program with the appropriate input and get their output. Programmers, instead of developing new programs, develop new modules that are plugged dynamically into the single running program. New computers, as they come on line, would start running part of this single program and can take advantage of the existing computing resources, while providing the program with additional capabilities.

> *Rather than everyone running their own program, we can think, at the extreme, of there being only one program that runs on all computers simultaneously*

There are obvious advantages to so-called "pervasive programming." First, it is a logical extension of the trend to network-based applications and modules. Microsoft's .Net framework, for instance, lets programs invoke network services almost as easily as calling internal routines. Cooperating network services, each with shared state and running on different machines, are a form of a pervasive program. Second, as pervasive computing becomes more prevalent and everything in one's house is essentially a computer on a network, viewing the various entities as a single cooperating program provides a framework for controlling and managing the multiplicity of devices. Third, as computers become more sophisticated, users expect to get more out of them. There is only a limited number of 50-million-line programs that can be written effectively. If all programmers actually contribute to the capabilities of a single system and each programmer could make use of the efforts done by others, then we could create the types of systems that users will expect in the future. Fourth, a single program provides a logical means of sharing the resources of large numbers of computers, effectively making grid computing available to all. Finally, inter-program communication right now is one of the more complex parts of system design and building. Addressing large-scale challenges such as controlling automobiles on a computerized roadway requires that large numbers of systems (in this case all the automobiles on or near the roadway) communicate effectively. If we change how we think of programs so that such communication is inherent rather than an addendum, we should be able to make developing such systems easier.

> *...outerfaces are the key building blocks. To support them, there must be standard ways of defining, finding and categorizing outerfaces and their implementations*

Research Issues

Many technical and social problems must be overcome in order to make this view of programming a reality. However, we feel that solutions to these problems are possible.

Any implementation of pervasive programming must be able to handle situations in which portions of the program become unavailable and are replaced while the program is running, as well as the dynamic definition of new portions of the program. A number of environments and systems support dynamic code replacement, and others have dealt with unreliable networks in distributed systems, so in principle these problems are tractable. However, doing this on a large scale, in a distributed rather than centralized manner, in a way that cannot afford failure, and where code can come and go dynamically will require additional research and probably a new framework to manage and track the code base. Dynamic techniques for discovering implementations such as those used by Jini will also be necessary here.

Pervasive programming will require extensions to current languages if not a totally new programming language, since it implies a new way of thinking about programs. One approach that seems feasible is to introduce an extended notion of an interface, which we call an *outerface* for now, as the basis for the new functionality. Outerfaces would be defined as abstract classes with both virtual and static methods. Users would code their portions of the system to implement a particular outerface and at the same time use already existing outerfaces. The underlying system would then be responsible for choosing an appropriate implementation of an outerface to be used by another outerface at execution time. This choice

Congratulations are in order to astaff for successfully completing job audits that raised each of their job levels one notch. Tstaffer Kathy Kirman has the same reason to smile. Clockwise from top left: Dawn Reed, Lori Agresti, Kathy Kirman, Akina Cruz, Fran Palazzo, Jennet Kirschenbaum and Genie deGouveia would be based on parameters the user might provide, properties of the outerface implementations, analysis of the uses of an outerface, network proximity and availability, as well as other conditions. Numerous language issues arise in attempting to make such a notion both practical and implementable. Moreover, the language must provide the hooks to deal with outerface implementations that disappear or are replaced dynamically.

In this view of pervasive programming, outerfaces are the key building blocks. To support them, there must be standard ways of defining, finding and categorizing outerfaces and their implementations. This could be done at various levels, each implying a different level of authority and responsibility. First, we visualize a well-defined set of standard outerfaces adopted by some sort of standards committee, akin to the methodology currently used in defining extensions to the Java language. Second, user groups could propose and code to shared outerfaces that might not be in the standard but can otherwise be agreed upon. Finally, it should be possible for individual users to create their own outerfaces and corresponding implementations, keeping the scope private where desired.

Another problem involves identifying the semantics of an outerface. If outerfaces are going to be widely used, each must have well-defined and well-understood semantics. Without a long drawn-out standards process, this is difficult and impractical. Instead, we envision that an outerface would be defined both as a set of methods and a set of test cases. Any implementation proposed for the outerface would have to pass all the test cases in order to be considered valid. Users would then be able to define their own subouterfaces that add additional test cases of particular interest or importance. The underlying system would take care of ensuring that only implementations that pass the test suite are acceptable. This operational definition of semantics seems a practical and feasible approach.

In addition to dealing with the immediate technical issues, a pervasive programming world will have to deal at all levels with security and privacy concerns. These must be addressed directly at the language level since they will be properties of outerfaces that implementations will have to conform to and that users will insist on. A capability-based model at the language level could be a basis for a solution in this dimension.

LIGHTSTAGE 3.0

Master's student Andy Wenger

During the summer of 2001 I was working on Lightstage 3.0 under the supervision of Paul Debevec in the Graphics Group at the USC Institute for Creative Technologies.

Many times in movie productions an actor's performance is shot in a studio and the scenery is shot at some other location or is computer-generated. The goal in general is to make the observer believe that both the actor in the foreground and the background scenery were shot at the same location looking through the same camera. In order to achieve realistic-looking composites, we have to match several parameters of the foreground and the background image. The perspective from which the actor and the background are viewed must match. The illumination of the actor has to be consistent with the illumination at the background location, plus several camera-specific properties like brightness, response curve, color balance and more also have to match. Lightstage 3.0 tries to solve the problem of creating a consistent illumination for the studio shot by using previously captured illumination properties at the background location or the illumination properties of the virtual scene into which we want to composite the actor.

Illuminating the actor in the studio consistently with the background scenery means we need to know how that environment would illuminate the actor. We get this information from light probes, which are omnidirectional high-dynamic-range images that capture the incident illumination for a specific point in space. Lightstage 3.0 is the device we built to illuminate an actor with a previously re-

Andy sitting in an early version of Lightstage 3.0 with only 41 light sources and a simple static matting system using a single camera. The control program running on the PC is operated by Jonathan Cohen '00

corded light probe. The device has the shape of a once-subdivided icosahedron, which allows us evenly to distribute 162 light sources over the sphere by placing an inward-pointing light source at each vertex and in the middle of each edge. With a diameter of two meters we can illuminate an actor from the chest upwards. (The actual number of light sources is 156 because we left out a piece at the bottom so the actor could stand. One light source is made up of red, green and blue LEDs, so no single LED is a light source, but rather eight red, five green and five blue ones combined.) By changing the intensities of the three base colors, we can dial virtually any color we wish. Each of the light sources is independently addressable, giving us the ability to specify a color for a desired position on the sphere.

The control program of Lightstage 3.0 uses a light probe in a longitude/latitude format as its input. From that light probe it calculates the color configurations of the light sources by projecting the position of each light source into this input space and taking a weighted average of the surrounding pixels. The control program is also responsible for color and brightness correction due to the different characteristics of the light sources with which we generate the illumination and the camera's imaging sensor with which we record it.

To differentiate between the actor in the foreground and the studio background we use an infrared matting system. The visible background to the camera is covered with infrared reflecting pieces of cloth that are illuminated with six infrared light sources to create a background we can later subtract out. The advantage of infrared light is that the matting does not interfere with the illumination we're trying to recreate; classical methods like blue screening would produce extraneous blue light and therefore affect the appearance of the actor. Using a beam splitter and two cameras, one for the visible light and one for the infrared, allows us to record the actor's performance as well as the matte at the same time. The two cameras must be carefully registered with respect to each other and the beam splitter for the two images to align. The compositing software uses the visible light image and the infrared image to cut out the actor. It then places the cut-out actor into the background image.

The infrared matting system and the current compositing software took shape after I left in September, as did the current composites. The work we started last summer culminated in a paper that I and Chris Tchou, Andy Gardner, Tim Hawkins and Paul Debevec submitted to SIGGRAPH 2002 in San Antonio. Those who want to know more about Lightstage 3.0 can check out the ICT Graphics Group's web page at *http://www.ict.usc.edu/graphics/*.

I would like to take this opportunity to thank the Graphics Group at USC Institute for Creative Technologies for giving me the chance to be part of such a great project and for such a great summer with them. Thank you!

HIBISCUS, PALM TREES AND FLAMINGOS RULE IN CS2 !

Don Stanford, recently retired CTO of GTECH Corporation (a CS Industrial Partner), is now teaching CS 002, Concepts and Challenges of CS. To give the students a break halfway through the class, Don introduced the Friday Hawaiian shirt challenge. He has a bounteous collection of these, each more gaudy and outrageous than the last; the challenge is to out-shirt him!

Of the initial shirt competition, CS2 TA Daniel Santiago wrote on the course webpage, “In response to the Friday Hawaiian shirt challenge, a single warrior emerged ready to take on Don and his vast collection of shirts. Mike’s colorful Hawaiian shirt was complemented by his Tony Soprano wife-beater undershirt and his very stylish sabre-tooth necklace. In what was undoubtedly a rigged vote, the class picked Mike’s fashion statement over Don’s, even though most of it looked like it had been purchased at a yard sale. Ever the gracious loser, Don rewarded Mike with an HTML cheat sheet that, we understand, has sold out at the Brown Bookstore. Don looks forward to meeting new challengers next Friday and anticipates a less biased vote on the part of the class!”

When Suzi Howe visited a Friday challenge to take photographs for posterity, the competitors were Nicole Janisiewicz ’02 and Zach Woodford ’05 vs. Don. Applause decided the winner—alas, not Don, but then he was wearing one of his more conservative habiliments. Nicole’s splashy floral print was the winner; she was awarded a Microsoft textbook, and a consolation prize went to Zach. Fridays provide an opportunity to see Don in the department before class sporting his glorious duds, even in the dead of winter. He is without doubt as charismatic an individual as his shirts are unrestrained!

MICHAEL BLACK IS NEW IPP DIRECTOR

After starting the **Industrial Partners Program** in 1989 and serving as its director for all but three years since then, Professor John Savage has stepped down. He has handed over stewardship of the Program to Associate Professor Michael Black.

In March 1989, ten years after the CS Department was founded, John, while Chair of the Department, introduced the Industrial Partners Program (IPP), which he had designed with the advice of Brown’s Development Office and of Roy Bonner, a retired IBM executive. IPP was intended to be a different kind of academic outreach program that would encourage partnership relations among corporate members and students and faculty in the CS department.

At its first meeting, IPP was introduced to senior executives from a number of prospective Partners including Bellcore, Codex, DEC, GTECH, IBM and Sun Microsystems. At this event we not only described the purpose of IPP to our visitors but also used it to showcase faculty research. Talks were given by Brown colleagues specializing in artificial intelligence, computer graphics, databases, operating systems, multiparadigm design environments, theoretical computer science and VLSI theory and algorithms.

IPP has evolved over time. It now gives Partners opportunities for pre-competitive cooperation via its biannual technical symposia on topics of interest to Partners. We also encourage communication with Partners via the IPP Program Director (Michael Black), the IPP Program Manager (Suzi Howe), individual faculty members, and students employed by Partners over the summer.

A distinguishing characteristic of our day-long technical symposia is that most of the speakers are representatives of Partner companies. This encourages interaction among Partners and makes for very lively meetings, with exciting give and take. It is always interesting to hear people from competing firms engage in animated discussions on neutral ground, a very healthy form of pre-competitive co-operation. To illustrate the star lineups we enjoy, at the 1994 'Nexal Computing' symposium that John hosted, James Gosling from Sun Microsystems gave a fascinating talk about his new programming language, 'Oak,' renamed 'Java' a few months later!

The IPP symposia provide an important window on Partner interests and are helpful to CS faculty and students in understanding the concerns of the commercial world. The symposia also give Partners visibility in the department, are valuable in their recruiting, and expose them to faculty and student research interests. The financial support provided by Partners through IPP makes possible many important ancillary activities that go far to make the Department the outstanding research and teaching environment it is today.

l to r: John Savage and Michael Black

Recently the Program introduced IPP Seminars, a venue for Partners to speak on advanced topics of interest to faculty and students; here Partners can also give brief introductions to their companies as well as make themselves available to discuss job opportunities. Brown alums frequently either give the IPP Seminars or accompany the speakers. Speakers are invited to spend the day in the Department, talking with faculty members with similar research interests.

While John is still active in research, his focus in recent years has turned more toward the University. From 1996 to 1999 he was the Faculty Vice Chair of the Advisory Committee on University Planning (ACUP), which recommends the University budget to the President. From 1998 to 2001 he was President of the Faculty Club, and in 2000 he started a three-year term as an officer of the Faculty, which entails chairing the Faculty Executive Committee in the second year. This year as Chair of the Faculty he is working closely with our new President, Ruth Simmons. Last year the inauguration program committee that he chaired arranged for twenty faculty talks broadly representative of faculty interests that were a feature of President Simmons' inauguration. Last year, also, he had a key role in shaping a new conflict of interest and commitment policy for the University. This year he is active in organizing a task force on faculty governance that hopes to make significant changes in the faculty's role in the University.

New IPP Director Michael Black (M.S. Stanford, Ph.D. Yale) came to CS in 2000 and has been assisting John Savage with IPP for the past year or so. Michael brings to this new position extensive experience in industry, corporate research, intellectual property and joint academic/industry partnerships. Before joining Brown, Michael managed computer vision research at Xerox PARC, learning a

ELVIS SIGHTING !

According to media coordinator Mark Oribello, Elvis has left the building...

great deal about the challenges of industrial research and the important role of academic partnerships in a competitive market. Having seen industrial partners programs from both sides, he views IPP as a fine example of how a focused, flexible program can adapt to meet the changing needs of academia and industry. His research interests include computer vision, optical flow estimation, human motion analysis, probabilistic and stochastic algorithms and brain-computer interfaces. One of the attractions of Brown to him was this same mix of disciplines and the collaborative spirit here. He is excited to be part of the Brain Sciences Program.

Stay tuned as in the coming months we strengthen and expand our core offerings to Partners. At this transition point we are particularly interested in feedback from Partners about how IPP can help them remain competitive in the current market climate.

COMPUTER SCIENCE AS AN INTEGRAL COMPONENT IN HIGHER EDUCATION

Tom Dean

Many of us in computer science today found our first encounter with a computer a profound, life-altering experience. Typically we sat at a terminal of some sort, typing statements; the computer interpreted our statements as commands and carried out those commands, printing an occasionally scrutable but more often cryptic and incomprehensible response.

With time, we learned to decipher those cryptic responses and even program the computer to carry out other commands with incomprehensible responses of our own design. We learned that interaction with a computer could be subtle, applicable to a wide range of tasks, and infinitely extensible. We quickly realized that by using loops, conditionals, and Boolean logic you could persuade computers to carry out complex tasks without intervention; you could embed your knowledge of the world in programs that would repeat routine tasks as many times as you like and do it infinitely faster and (if you're careful in writing the program) without error.

The speed and precision of computers let you cheat time and avoid (some forms of) fatigue and boredom. Not only could you embed your knowledge in a program, but you could share that knowledge with someone else in the same way that you could tell someone how to repair a bicycle or prepare a recipe. You could also use the knowledge that others embedded in their programs by combining it with your programs to solve new problems you couldn't handle otherwise.

The very fact that you can interact with a computer is fascinating in and of itself. Even more exciting is the fact that computers can be said to learn: computer programs developed by credit-card companies learn to recognize, anticipate and limit credit-card abuse. The various forms of machine learning possible today challenge us to think about what learning means.

Computers at grocery stores and online merchants analyze your buying habits, predict your preferences, and suggest other items you might be interested in—annoying, perhaps, but often eerily on target. These programs can be thought of as creating a 'model' of users, and many other programs can build even more sophisticated models of their environment and the people (users) with whom they interact. Computers with internal models, even models that refer to the computer itself, further erode the differences between human and machine and force us to think more carefully about who we are and what it is to be human.

Computers let us build better models of the world around us. They model the weather, earthquakes, buildings, biological processes, the human immune system, brains and even other computers yet to be built. These models allow us to diagnose problems, predict the future, and explain the world around us (and inside us). Computer science also tells us about the limitations of computing and therefore about the limitations of human understanding and influence.

In colleges and universities, computers play many roles. They provide metaphors and models for teaching and learning and self-analysis that apply to a wide range of studies in philosophy, psychology, neuroscience and biology. Computers offer tools to extend our abilities and enhance the learning experience. They allow students and scientists to explore worlds that are difficult to access, that don't exist yet, that never will. They let workers in disparate fields combine their special expertise in programs that provide insights into new, hybrid fields. Computer programs allow us to navigate in the sea of information, both new and old knowledge, that constantly threatens to drown us. (Of course, computers are also largely to blame for the glut of information, but few of us are willing simply to turn off the tap.)

The future of the discipline will comprise more of the same and then some: new models and metaphors, new types of computing that borrow from genetics, animal behavior, neurophysiology and the physics of the small and large. There will be new disciplines, computational this and that, in which computer models and tools provide fundamentally new insights and leverage. Increases in speed alone open up new opportunities and provide enhanced capability in much the same way that advances in optics and imaging devices opened up new fields by making visible the invisible.

Computer-mediated prostheses of all sorts with better interfaces will become increasingly common. Direct brain-computer interfaces already exist as cochlear implants for the hearing-impaired, with retinal implants to restore vision and direct neural shunts to control artificial limbs not far behind. Complicated artifacts like airplanes and large buildings rely on computer tools to create and then test their design. Bioengineering depends on advances in computer design to progress, and the new technologies will serve to reengineer the human body and brain.

The way modern software is produced is influencing how people work in multidisciplinary studies where computers play a central role. In software production houses, programmers work in teams with one or a few persons responsible for each component of a large piece of software. Seldom nowadays is a solitary programmer responsible for a large software project. Indeed, some of the contributors to such a project are often not programmers at all but rather computer-savvy experts in disciplines from human factors and psychology to business and law. Key to a successful product are clear communication among the team members and the ability to decompose large, complex problems into tractable components encapsulating the different sources of expertise. Such projects force us to define appropriate vocabularies at the right level of abstraction to support discourse at the boundaries between components and their encapsulated expertise. The technical and organizational skills that produce successful software projects are also crucial to multidisciplinary efforts, especially those in which computers are used to integrate different sources of knowledge.

The technical and organizational skills that produce successful software projects are also crucial to multidisciplinary efforts, especially those in which computers are used to integrate different sources of knowledge

Computational molecular biology, combining among other fields molecular biology, organic chemistry and computer science, is a good example of a discipline that lends itself to collaborative research, with computers and computer scientists playing myriad supporting roles. Computational pharmacology, sequencing and related genomic studies, evolutionary molecular biology, and protein-folding analyses depend on scientists from several fields working together, often physically in the same space but sometimes only virtually, using computers to facilitate almost every aspect of their communication, data gathering and analysis. Similar revolutions are afoot in economics, brain science, and a host of other disciplines, and we're just beginning to understand the benefits of visualizing large data sets from financial, health care and government census sources.

Computer science has always thrived on interactions with other disciplines. In particular, computer scientists are fascinated with the brain and with biological systems in general. Artificial neural networks, now a very useful technology in pattern recognition and machine learning, may have very little to do with how real neurons work, but new insights from neuroscience are informing new models of computation that promise even more useful technologies. Nature is a wonderful source of technical hints and demonstrations of things we wouldn't have dreamed possible. Computer science is both a consumer of ideas from and a producer of models and tools for the hard and soft sciences. And the metaphors and lessons of computing permeate our culture and society at all levels. Computers and computer-mediated communication are changing how we relate to one another in language and the visual arts, and the new accessibility of the written word, coupled with new ways of linking and cross-referencing documents, is profoundly affecting scholarship in all disciplines.

As long as computer science remains open to other disciplines, it will have no chance to rest on its laurels or drift into a phase of purely incremental progress. While interactions with other disciplines provide new applications, and there is no denying that computer scientists love to solve problems, these interactions also spawn new ways of thinking about computation. Computer science has its own growing base of knowledge ranging from the purely pragmatic to some of the most beautiful mathematics of the past century. The discipline is deep as well as broad; there are still fundamental open problems that stand in the way of understanding computation and making it work for society.

Beyond the scientific and technical issues, computer technology raises a host of moral and ethical problems for society to wrestle with in the coming years. We aspire to produce civic-minded scientists and technologists and scientifically literate and technologically savvy citizens. Just because we have the technology to do something doesn't mean we should do it. Computers let us scrutinize the lives of individuals at a frighteningly intrusive level. We are only just beginning to address the security and privacy issues that arise as society becomes increasingly dependent on networked computers. Universities, as creators and first adopters of new technology, have the opportunity and the responsibility to tackle head on the ethical and moral issues it raises. We've just begun to realize the social implications of new computer technologies. Soci-

KANELLAKIS LECTURE SERIES INAUGURATED

On November 29, 2001, the department hosted the first annual **Paris Kanellakis lecture**. Dr. Mihalis Yannakakis, of Avaya Laboratories, spoke on "Progress in System Modeling and Testing." In his introductory remarks, chairman Tom Dean said, "This is a celebration of two quite extraordinary people, both good friends and colleagues of the department, and, coincidentally, both Greek and exceptionally talented theoretical computer scientists....I'm absolutely sure that Paris would be ecstatic to have Mihalis here for the day to talk with our faculty and students and to give this lecture."

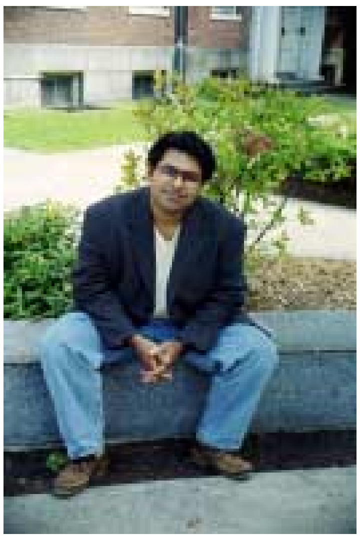
This lecture series honors Paris Kanellakis, a distinguished computer science theoretician who was an esteemed and beloved member of this department. The deaths of Paris and his family in a December 1995 airplane ac-cident in Colombia continue to be a profound loss of which we are especially reminded towards the end of each year. We are therefore all the more delighted to have inaugurated this lecture series in his memory. Subsequent Kanellakis lectures will be held around Paris's birthday, December 3.

If you'd like to be on the mailing list for

Dr. Yannakakis surrounded by Kanellakis Fellows from l to r: Ioannis Vergados, Ioannis Tsochantaridis, Nikos Triandopoulos, Aris Anagnostopoulos, Manos Renieris and Christos Kapoutsis (of MIT) ety will soon have to consider the rights of and laws governing, first, augmented human beings, and then, true artificial intelligences of steadily increasing capabilities.

A good argument can be made that computer science literacy, at least at the level of models, metaphors and the basic notions of algorithms and data structures, is necessary in a liberal arts education. Within academe, students come to computer science to learn valuable skills and powerful tools, to encounter deep mathematical ideas, and to acquire models and metaphors that enrich their other studies and provide insights into being human. Computation is changing how we relate to one another and to society. It is transforming society and will continue to do so. Most computer scientists today feel tugged this way and that by all the interrelations of their discipline to the world around them and the dependencies and responsibilities these interrelations entail, but few would have it any other way.

THE 28th IPP SYMPOSIUM



Shriram Krishnamurthi

On November 6, 2001, we hosted an IPP Symposium on **'Component Software and Technologies'**. Steve Reiss and Pascal Van Hentenryck suggested the theme and helped me find suitable speakers. Despite the lull prompted by corporate spending cuts and depression over recent events, we had a packed room and rather lively debates.

Components have been in the public eye for a while now, the topic of many successful books, conferences and papers. The challenge with any technology this popular is to separate the wheat from the chaff: to understand what it really contributes and, in particular, to avoid merely using a new term for stale ideas. To work towards this understanding, we incorporated a broad diversity of perspectives spanning industry and academia, users and researchers, foundations and pragmatics, and I think our effort was fruitful. While I doubt we resolved major questions to everyone's satisfaction, the event did remind me of an exchange I've seen attributed to Samuel Johnson. His correspondent writes, "I have read your article, Mr. Johnson, and I am no wiser than when I started." The good Doctor responds: "Possibly not, Sir, but far better informed."

I inaugurated the day with a talk that attempted to offer an overview of components. Preparing the talk was actually an unusually enlightening experience for me. In trying to understand the provenance of components and why they felt inevitable, I came fully to appreciate the wisdom behind the work on software product lines. As a result, my talk was half overview and half (disguised) manifesto for a certain design methodology to accompany components. Finally, I provided a rationale for the choice of remaining speakers (notice that I cleverly did this at the *end* of my talk, so I didn't need to excuse my own presence).

The features of components that make them salutary for software development have a darker side: they can inhibit the cross-component reasoning necessary for validation and optimization. This poses new challenges for compiler authors. **Mark Wegman** of IBM presented an incipient effort on compiler optimization in the context of components. This is ambitious work and it will be exciting to see their experimental results.

Jean Laleuf, on behalf of members of the Exploratories project at Brown, made a brief presentation on their experiences with components. Exploratories is an experiment in creating a framework for rapidly prototyping graphical and other tools for interactive exploration. Jean briefly outlined his group's successes and frustrations with components. He also demonstrated the software during the breaks.

Karl Lieberherr of Northeastern University has been refining programming technology in interesting directions for several years. He observed that a good deal of administrative code in object-oriented systems addresses traversing object graphs. He handles this through *traversal* abstractions, which let programmers separate code into declarative specifications of the traversal and concrete specifications of behavior. His recent work on traversals integrates them with components, and he described this approach, which he calls *aspectual components*, in his talk.

IPP Symposium speakers clockwise from top left: Jean Laleuf, Brown CS Graphics Group; Mark Wegman, IBM; Jay Lepreau, U. of Utah; Bob Rogers, GTECH; Karl Lieberherr, Northeastern U.; host Shriram Krishnamurthi, Jim Waldo, Sun; Mirek Kula, GTECH

Jay Lepreau of the University of Utah tackled a long-standing, niggling concern about software engineering: will its methods apply to constructing low-level, performance-oriented software? Jay's team has built three generations of operating-system components to help users rapidly configure and build custom kernels—a domain in which excessive cost introduced by abstractions becomes immediately apparent to users. They have found that naive notions of components are insufficient for this task, but are converging on a combination of programming-language support and component technology that is proving very successful.

Mirek Kula and Bob Rogers represented our industrial partner GTECH, one of the most successful providers of lottery technology. This description is deceptive: in fact, they must meet extremely rigorous technology standards. For instance, the average user, standing impatiently in a queue at a gas station, is relatively uninterested in the details of computer systems: he wants to buy his ticket and hit the road as soon as possible. Mirek and Bob outlined many of the fascinating technical challenges that confront GTECH. They then gave a sobering report on successes and failures of their adoption of component technology. In particular, they explained how they had been let down in their expectations (encouraged by the press) of a "component marketplace."

Finally, **Jim Waldo** offered a rousing interpretation of Sun's view of the next generation of components. He pointed out that the network forms the ultimate abstraction boundary between computations; the increasing emphasis on erecting abstractions implies that the natural way to separate components is to distribute them across a network. He explained Sun's vision of using Java as the language of exchange among these components, and ended the day with an inspiring vision of what components might do.

I'd like to thank the speakers for their time—some of them came to participate from two time zones away. They humored my impositions, such as a request that they offer a concrete definition of the notion of components they were employing so we wouldn't talk at cross purposes. Some of the speakers even subjected their definition to iterative refinement. The mix of experiences (and personalities!) led to several interesting exchanges, and many people lingered to continue over hors d'oeuvres.

Finally, I have to let my faculty colleagues in on a dirty little secret: running an IPP Day is *easy*! Suzi and the other administrative staff (Fran in particular, in this case) do all the preparation and execute it not just flawlessly, but with an unmistakable and irrepressible panache. (Unfortunately, however, this time Suzi did not match her clothes to the tablecloth, more's the pity.) I'm just the mook who fronts for these incredibly hard-working people. For more information about the day, please see *http://www.cs.brown.edu/industry/ipp/symposia/ipp28/*

LETTER FROM SINGAPORE

Franco Preparata

If someone had applied for a sabbatical leave in Singapore thirty or forty years ago, the request might have seemed a suspicious and dubious way to smuggle in an exotic trip in the guise of a professional endeavor.

Some people may still have the same suspicion; in which case they need a refresher on today's sociopolitical realities. I remember first coming across the name 'Singapore' in my childhood adventure books, an exotic cove of pirates surrounded by luscious jungles, where fierce predatory animals roamed freely. Later, W. Somerset Maugham depicted a farflung outpost of the British empire where a unique community of colonists, merchants, adventures, and demimondaines had recreated a comfortable microcosm in the image of faraway 'home'. Vestiges of that epoch and society remain more in the nostalgia of the warm accounts of a bygone time than in actuality. Nostalgia, of course, always embellishes the good old days, for we tend to filter out the unpleasant aspects.

Yet it seems almost impossible that thirty years ago (yes, just thirty years ago), very little of today's imposing infrastructure was in place: the wide urban expressways with their perfect pavement (a far cry from the pot-holed streets of our Providence) lined with gracious umbrella-shaped trees and surrounded by well-appointed gardens, the international flair of the shopping district, the awesome skyscrapers of the financial section, where architectures compete in size, variety, materials, and a bit of extravaganza. Here everything is exotic and nothing is. All the familiar American icons (McDonald's, KFC, Pizza Hut) are ubiquitous, as are the exclusive Italian designer shops (Gucci, Armani) and the Japanese department stores (a Takashimaya more glittering then its counterpart in Kyoto). All cuisines are available, and in abundance; there is no notion of 'ethnic food.' Here more than elsewhere one perceives the reality of globalization: anything that is available somewhere is available here. It is also very hot (perpetual summer), but I don't think this is attributable to global warming.

Beyond these broad impressions of everyday living, one could also comment on this incredible experiment in social engineering that has guided a racial kaleidoscope to a transition from a semidilapidated British colony to a first-world little nation. But this would make my discourse too serious and deflect me from my commitment to broad-brush impressions.

I write this letter soon after sunrise (always about 7am on the equator) and soon I'll go to my spacious office at the National University of Singapore, a modern institution growing according to the American

changelog

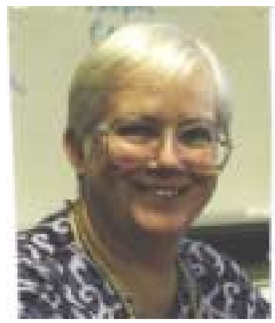
Andy van Dam is delighted to report the transition of the chairmanship of the CS department at the University of Washington from one Brown CS graduate to another! Ed Lazowska '72 has stepped down in favor of David Notkin '77. Said Andy, "new chair Notkin is also one of ours, so the Brown line continues..." Below, in celebration, Notkin (l) and Lazowska raise the official brew of UW, Arrogant Bastard Ale*!*

------8<------

For the CS 2 mid-term, Don Stanford had planned to give any *Hawaiian shirt wearers a prize. Surprisingly, however, 25 brilliantly beshirted students showed up! He had purchased only five gifts—the Patriots Yearbook issue of* model, whose president, by the way, is Prof. C. H. Shih, until recently a colleague of ours in Brown's Division of Engineering. In my office I will meet a young Singaporean colleague who has, I am sure, worked into the wee hours to fine-tune a sequence reconstruction program: an enthusiastic, very competent and resourceful collaborator in my research in computational biology. Such an interface certainly did not exist thirty, forty years ago. This is why a sabbatical leave in Singapore is today a totally appropriate undertaking.

EXPERIMENTS IN IMMERSIVE VIRTUAL REALITY FOR SCIENTIFIC VISUALIZATION





Andy van Dam, David Laidlaw and Rosemary Simpson

In "Future Interfaces: An IVR Progress Report," Andy van Dam, David Laidlaw and Rosemary Simpson summarize important trends in user interfaces for immersive virtual reality. Section 6 of this paper, which describes ongoing work at Brown on IVR and scientific visualization, is reprinted below; the complete paper will appear in *Elsevier's Computers and Graphics, Vol. 26, No. 4,* available online in August.

WHERE ARE WE NOW?

At Brown we have been exploring a number of different IVR applications to study UI research issues in IVR. We are collaborating with domain experts in several fields to develop applications that address driving problems from other scientific disciplines, believing that such collaborations will help us to factor out common patterns from the problems in various disciplines to develop IVR interaction metaphors and visualization techniques that can be generalized. The collaborations also let the domain experts validate new techniques and ensure that they are responsive to the needs of real users. We study our visualizations in a Cave, a cubicle in which high-resolution stereographics are projected onto three walls and the floor to create an immersive virtual reality experience. Our scientific application areas include archaeological data analysis, biological fluid flow (bioflow) visualization, brain white-matter analysis, and Mars terrain exploration.

The visual characteristics of a virtual world are coupled with the user interface. We draw inspiration from artistic techniques to support our communication goals; art history, pedagogy, and methodology, together with art itself, provide both source material and a language for understanding that knowledge. Perceptual psychologists have also developed a body of knowledge about how the human visual system processes visual input; these perception lessons aid in designing the visual representations of IVR user interfaces.

Beyond visual inspiration, perceptual psychology also brings an evaluation methodology to bear on scientific visualization problems. Evaluating visualization methods is difficult because the goals are difficult to define and meaningful evaluation tests are difficult to design. Similar issues arise with the methods of evaluating how the human perceptual system works. In essence, we are posing hypotheses about the efficacy of user interfaces and visual representations and testing those hypotheses using human subjects, an experimental process that perceptual psychologists have been developing for decades. Perceptual psychologists, in close collaboration with domain experts and artists, are helping us develop a methodology for evaluating visualization methods.

changelog

Sports Illustrated. *While the class took the exam, Don rushed to Thayer St. for more magazines, returning with the only* Sports Illustrated *issue he could find in bulk--the swimsuit edition!*

------8<------

Both CS nominees for the 2002 undergraduate CRA awards, Harry Li '02 and Rachel Weinstein '02, were selected for honorable mentions.

------8<------

Alumnus John Crawford '75 has been named to the National Academy of Engineering. Said John in a recent email to Andy, "Distinguished Gentlemen: It is a great honor to join you all in the Academy of Engineering. Thanks especially to Professors van Dam and Brooks for your training and inspiration!" Dr. Crawford is director of microprocessor

Visual design and interaction design come together to create a virtual environment. Virtual environment design, however, goes beyond the typical domain or training of a single designer. Many of the design issues are similar to those of varied other areas of design from which we can learn from and draw inspiration. Components of architectural and landscape design are relevant for creating and organizing virtual spaces, sculpture and industrial design for finer-grained 3D parts of the environment, illustration for the visual representation of scientific data and the surrounding environment. Traditional user-interface design is applicable to some parts of the interaction design, and animation to other parts. The design process is different for all these types of designers, and getting all of the pieces to play together effectively is a challenge, particularly with the constraining needs of the scientific applications.

ARCHAEOLOGICAL DATA ANALYSIS

Archave, created by Eileen Vote and Daniel Acevedo, was one of Brown's first Cave applications. It was developed in cooperation with Martha Sharp Joukowsky of Brown's Center for Old World Archaeology and Art and Brown's NSF-funded SHAPE lab, which was set up to develop mathematical and computational tools for use in archaeology. Archave provides virtual access to the excavated finds from the Great Temple site at Petra, Jordan. Here we examine some of the user interface and design issues arising during the system's design, development, and testing.

Our application design had archaeologists work in the Cave for multi-hour sessions on analysis tasks that would have been quite difficult using traditional analysis tools. They were able to support existing hypotheses about the site as well as to find new insights through newly discovered relationships among the over 250,000 catalogued finds. This experience, while successful, raised many research issues in user interface and visual design, many of which involve taking the Cave experience beyond the demo stage and making it truly useful for archaeologists.

Multivalued visualization: In this archaeological application, each of the many thousands of artifacts is described by dozens of attributes, such as Munsell color, date, historical period, condition, shape, material, and location. Furthermore, the relationships among the artifacts and with the trenches from which they are excavated also must be represented in an easily distinguishable way.

Figure 1—Archave image from the full reconstruction version of Archave

As in all design issues, the driving force behind the visual representation is the task to be performed with it. For the relatively simple demo-like evaluations performed so far, only a few attributes of the data are displayed—those essential to a specific task. As the tasks become more exploratory, the visual mapping will need to become more complete and simple designs will no longer suffice. Here is where we hope to exploit perceptual psychology and art history for inspiration as well as studying the design process itself.

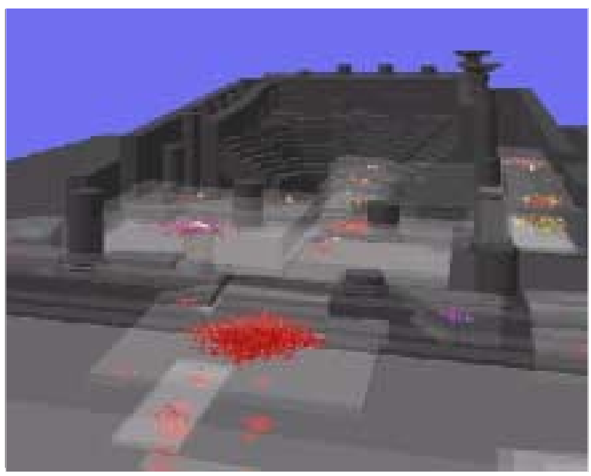


Figure 2—Archave representation of excavation site with artifacts

Design: Design issues are omnipresent in our Cave applications and several from Archave deserve mention. First, the visual context of the data representation, as well as the visual representation of artifacts themselves, is important in interpreting the data. Initially we used realistic representations for artifacts within a plausible reconstruction of the temple (Figure 1). We found, however, that a more primitive iconic representation of the artifacts, with important data components represented by shape, size, and color, was easier to analyze (Figure 2). Our Archave users, who had typically been involved in the excavation process, were also distracted by the reconstruction, since they were far more familiar with the present-day ruins. They were able to work

changelog

architecture at Intel Corporation; he received this highest of such awards for the architectural design of widely used microprocessors.

------8<------

David Laidlaw spent part of his sabbatical last fall taking an introductory oil-painting class in the Illustration Department at RISD. Because some of his scientific visualization research has built on ideas from oil painting, he hoped that the class would provide some new inspiration. He reports that, in addition to feeling a bit older after being twice the average age of the other students, he looks at the world through better-trained eyes. He won't, however, be changing his vocation. Some of his continuing collaborative efforts with RISD faculty are also described in the article on page 16.

more effectively with a model of the present-day ruins as context in which artifact concentrations are shown as simple 3D icons in saturated colors among a muted view of the present-day ruins and transparent trench boundaries.

Scale and navigation: Another research issue concerns the intertwined matters of scale and navigation. Should we work at full scale? In miniature? What are the tradeoffs for archaeologists studying a site like Petra that is the size of three football fields? For a scale model that fits within the Cave, navigation can be primarily via body motions. Larger scales require longer-distance navigation, and the virtual world must move relative to the Cave, obviating any fixed mental model a user may have created. Motivated in part by these multiscale navigation needs of Archave, Brown's Joe LaViola developed step-WIM navigation, in which the user employs gestures to access a world-in-miniature for navigating large distances and familiar body-motion navigation for shorter distances.

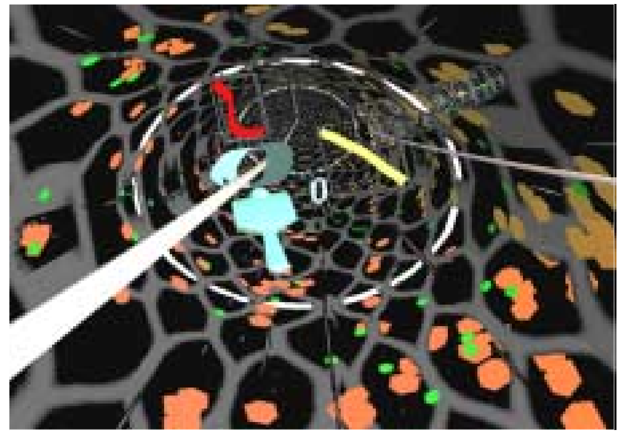


Figure 4—Synoptic view of interior of the artery, including streaklines, annotations, sponges, pierce planes, pressure splats and particle paths

Productivity: Most desktop productivity applications create persistent artifacts—word processors and editors produce documents, modelers create geometric models, and spreadsheets create analysis documents. The ephemeral nature of Cave experiences limits their scientific utility. Archave, for example, has been primarily a browser of the archaeological record, and while archaeologists have found this valuable, they need to capture their work in the Cave for later analysis.

Quantitative evaluation: Quantifying the value of IVR for this application is difficult. Initially, we collected anecdotal evidence during demos to archaeologists. We then attempted to design a user study, defining essential tasks as quantitative performance yardsticks. We failed. Such user studies require a quick task because it must be repeated many times. But a task must also be representative of what real users will do. In our scientific applications, the real tasks are still being identified and so are not clearly defined. Those that have been identified tend to be complex and time-consuming, e.g., determining relative dating of a set of spatially related finds from several trenches. Defining tasks for quantitative assessment is thus difficult because of all the competing constraints. For now, we compromise and give structured 1-2 hour "demos" that we record on video and analyze anecdotally. As this issue arises in all our applications, our continuing research agenda includes a search for more effective ways to evaluate user interface concepts in IVR.

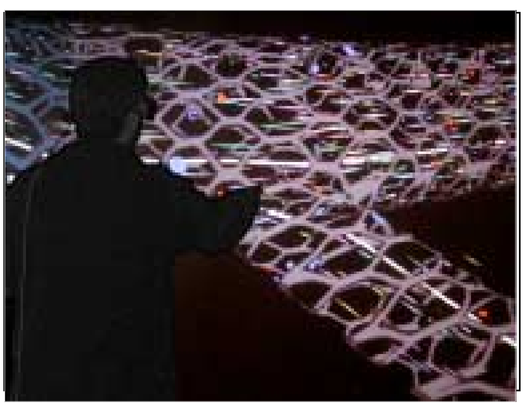


Figure 3—Bioengineer studying pulsatile flow within an idealized virtual model of a bifurcating coronary artery. Viewing this scene with head-tracked stereo glasses causes particles and the textured tubular vessel wall to "jump" into the third dimension and become much clearer

Discipline-specific VR sensitivity: Another second limiting factor in Cave utility was the sensitivity of Archave users, seemingly more than those in some other disciplines, to fatigue and discomfort in the Cave. This sensitivity seems to be shared by many artists as well. This result is surprising, running counter to an expectation of little disorientation because of the "familiar" and simple spatial nature of the depiction, and must be further investigated.

BIOFLOW VISUALIZATION

Our second scientific application, in collaboration with bio-engineer Peter Richardson of the Division of Engineering, CFD specialists George Karniadakis and Igor Pivkin of Applied Math, and IVR specialists Andrew Forsberg, Bob Zeleznik, and Jason Sobel of Computer Science, is a virtual environment for visually exploring simulated pulsatile blood flow within coronary arteries. Cardiologists and bio-engineers hypothesize that characteristics of the flow within these vessels contribute to the formation of plaques and lesions that damage the vessels. A significant fraction of the population suffers from this problem. We conjecture that IVR is an appropriate way to develop a better understanding of the complex 3D structure of these flows.

In Figure 3, a scientist uses our system to study simulated pulsatile flow through a model of a bifurcating coronary artery; lesions typically form just downstream of these splits. IVR offers a natural exploration environment. The representation in Figure 5 is an overview of all of the data available as a starting point for exploration. Here the wire-mesh artery walls show quantities on them via variably distributed splats, yellow for pressure and green for residence time, and the flow is represented within the artery with particle paths that advect through the flow.

To make the view both fast enough to compute and possible to interpret, particle paths are concentrated in "interesting" areas. Only a subset of the possible paths is visible at any one time, permitting a faster frame rate. By cycling through which particle paths are visible, however, all the time-varying flow can be shown over a period of viewing time. The application also supports some interaction with the flow, including placement of persistent streaklines, uniform coloring or deletion of all particles that pass through a specified region, and controls for the rate, density, and for defining the “interesting” areas to emphasize.

The synoptic view provides a starting point for exploration that gives more insight than an empty space. Other parts of the interface give a way to explore the synoptic view, adjust it, and annotate features as they are discovered. Thus far, we have been able to understand the simulation process that creates our flow data better than has been possible with traditional workstation-based tools. We are just starting to find new arterial flow features.

Beyond the perennial issues of frame rate, tracker lag and calibration, and model complexity, some more subtle ones emerged as we developed this application.

Multivalued visualization: As in the Archave project, this application has more different types of data than we can visualize at once. In addition to the velocity within the artery, which is a 3D vector field, pressure and residence time are defined on the walls of the vessel. Beyond those quantities, our fluid flow collaborators want to look at pressure gradient, vorticity, other derived quantities in the flow, and the structure of critical points.

Our synoptic visual approach for displaying this multivalued data is partly motivated by Interrante's work demonstrating that patterns on nested surfaces are more effective at conveying shape than "transparency"

Our synoptic visual approach for displaying this multivalued data is partly motivated by Interrante's work demonstrating that patterns on nested surfaces are more effective at conveying shape than transparency. We carry that to a volumetric display of particles and also use motion to increase the apparent “transparency”. We also use layering of strokes. With these principles we can display all the data with very little occlusion as a starting point for study of the flow. We are also exploring other more feature-based visual abstractions for the flow showing, for example, critical points and their relationships, coherent flow structures, or regions of the flow that may be separating. They have the potential to abstract the essence of a flow more efficiently, although at the risk of missing important but unexpected flow behavior.

What makes a visual representation effective? For understanding steady 2D flows, important tasks that are simple enough for a user study include advecting particles, locating critical points, and identifying their types and the relationships among them. The same tasks are likely to be relevant in 3D but there are likely to be other simple tasks that we haven’t identified yet, and there are certainly complex tasks that are beyond the scale appropriate for quantitative statistical evaluation needed for a usability study. The discovery of such measures of efficacy will clearly influence the design of visualization methods.

Scale and navigation: While the scale and navigation issues are somewhat similar to those of Archave, they differ in that Archave has a natural life-size scale while arterial blood flow requires a more “fantastic voyage” into the miniature. When should our artery model be life-size? Large enough to stand in? Somewhere in between? Under user control? Any one such specific question could probably be answered given some context and a user study. But would the results generalize to different navigation strategies or to a different visual representation? For our synoptic visualization, a model about six feet in diameter seems to give the best view of flow structures of interest from inside. Smaller-scale models are too difficult to get one's eyes inside. We have explored several navigation metaphors, including wand-directed flying, direct coupling of the virtual model to the hand, and a railroad-track-like metaphor with a lever for controlling position along a path down the center of the artery. Each has strengths and weaknesses; none is ideal in all situations. Design issues like these of scale and navigation, both at a specific and at an integrative level, have become an important part of our ongoing research.

Our group at Brown is working with half a dozen Rhode Island School of Design (RISD) faculty to address some of these issues. Our goals are three-fold: first, to develop new visual and interaction methods; second, to explore the design process; and third, to develop a curriculum to continue this exploratory process. For the last four months we have held a series of design sessions and have built a common frame of reference among the designers, computer graphics developers, and domain scientists; we are ready to proceed with new designs. We will offer a Brown/RISD course next fall to 6-10 students from each institution.

Design iteration costs: One of the highest costs of developing IVR applications is iterating on their design. Archave took almost three years to go through four significant design phases. For our bioflow application, we have implemented three significantly different designs in about 18 months. Parts of this process are likely to be essential and incompressible, but we believe that other parts are accidental, to use Brooks’ terminology. A Brown graduate student, Daniel Keefe, is working together with our RISD collaborators to find ways of speeding it up. Initial efforts build on his CavePainting application, which helps quick sketches of visual ideas in the Cave.

Productivity: As with Archave, scientists would be more productive if they had tools for annotating their discovery process, facilities for taking results away, and some way to build on earlier discoveries. We are exploring a number of possible alternatives that would let researchers take their notes and versions of the applications away from the Cave environment.

BRAIN WHITE-MATTER

VISUALIZATION

In a collaborative biomedical effort developed by computer science graduate students Cagatay Demiralp and Song Zhang in collaboration with, among others, neurosurgeons from MGH and brain researchers from NIH, we are designing a visualization environment for exploring a relatively new type of medical imaging data: diffusion tensor images (DTI). Humans are 70 percent water. The structure of our tissues, particularly in the nervous system, influences how that water diffuses through the tissues. In particular, in fibrous structures like muscle and nerve bundles, water diffuses faster along the fibers than across them. DTI can measure this directional dependence. Measurements of the diffusion rate have the potential to help understand this structure and, from it, connectivity within the nervous system. This will help doctors better understand the progression of diseases and treatment in the neural system. DTI provides volume images that measure this rate of diffusion at each point within a volume. Viewing these volumes is a challenge because the measurement at each point in the volume is a second-order tensor consisting of a 3x3 symmetric matrix containing six interrelated scalar values.

Figure 5—Brain model use in Cave

We generate geometric models with the volumes to represent structures visually and display them in the Cave. In Figures 5 and 6, an abstraction derived from a DTI of a human brain, different kinds of geometric models represent different kinds of anatomical structures: the red and white tubes represent fibrous structures, like axon tracts; the green surfaces represent layered structures, like membrane sheets or interwoven fibers; the blue surfaces show the anatomy of the ventricles.

Our experience thus far is that IVR facilitates a faster and more complete understanding of these complicated models, the medical imaging data from which they are created, and the underlying biology and pathology. Applications we are pursuing to explore this claim include preoperative planning of tumor surgery, quantitative evaluation of tumor progression under several conditions, and the study of changes due to surgery on patients with obsessive compulsive disorder.

Once again, frame rate, lag, visual design, and interaction design are intertwined issues. Biomedical researchers using this application almost always want more detail and will tolerate frame rates as low as 1 FPS for visualization, even knowing that they must move their heads only very slowly so as to avoid cybersickness. This clearly detracts significantly from the feelings of immersion and presence. Worse, many interactions are virtually impossible at that rate, so we struggle to balance these conflicting requirements.

Multivalued visualization: These volume-filling second-order tensor-valued medical images consist of six values at each point of a 3D volume. Often, we have additional coregistered scalar-valued volumes. We continue to search for better visualization abstractions through close collaboration with scientists, since it is only thus that we can define what is important and what can be abstracted away. Exploring differences between subjects or longitudinally within one subject brings the additional challenge of displaying two of these datasets simultaneously.

Design: Beyond the data visualization design, we notice again, as with Archave, the importance of virtual context. Working within a virtual room, created by providing wall images within which the brain model is placed, gives users a subjectively more compelling experience. Several users report that the stereo seems more effective and that they can “see” better with the virtual walls around them. In collaboration with perceptual psychologists, we are exploring why.

Scale and Navigation: As with Archave, full-scale seems a natural choice. However, much as with our arterial flow visualization, users prefer larger scales since at full scale they must struggle to see small features. They can make the projected image larger by moving closer to the virtual object, but then must squint and strain; also, they find it difficult to fuse the stereo imagery when features are up close. With the approximately 2:1 scale models used thus far, exploration has been almost exclusively from the outside looking in, unlike the flow visualization, where most of the exploration has been from within the flow volume. Could each area benefit from interaction tech-

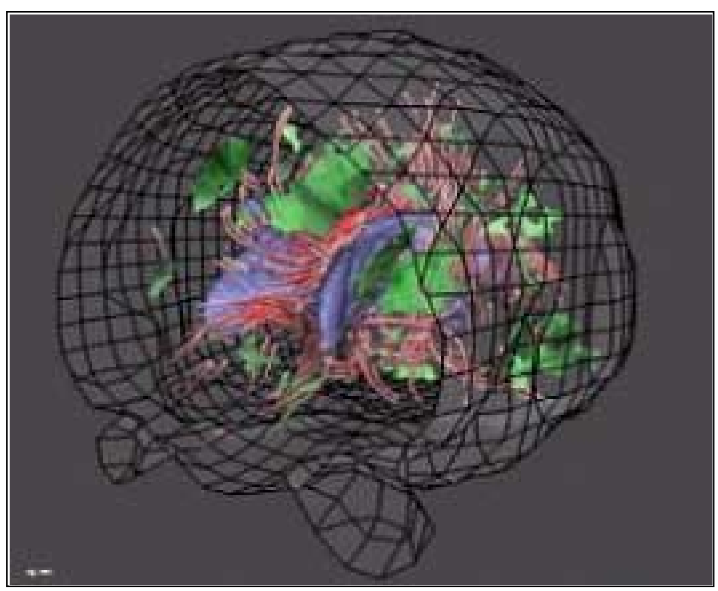


Figure 6—Brain model in wireframe enclosure

THE NOT-SO-GREAT MARMITE TASTE CHALLENGE!

British-born Suzi Howe grew up eating Marmite (“a brownish vegetable extract with a toxic odor, saline taste and an axle-grease consistency”). She had been shipped off to boarding school at the age of nine, where inmates subsisted in a Dickensian state of semi-starvation and Marmite loomed large in the diet, along with bread and dripping! After reading the entertaining article “Long Live Marmite! Only the British Could Love It” in the NYT’s international section earlier in the year, she was eager to share the Marmite experience with everyone in CS; in fact, there was no stopping her.

As you can see from the photograph, she set up a three-point Marmite tasting station. Participants were invited to 1) read the article, 2) take bread and spread on some Marmite, and 3), well, need we say more? She thoughtfully provided a notepad for comments, which ranged from “Let’s get Mikey!” to “IDLH” (Immediately Dangerous to Life and Health—an OSHA term denoting a hazardous environment) and “I fear the British!” You will be spared the less savory (no pun intended) remarks.

Despite the NYT’s classification of Marmite as “an emblem of enduring British insularity and bloody-mindedness,” and the fact that the trash can pictured was full by day’s end, Marmite has its mavens. Check out this website: *www.ilovemarmite.com*. And the less stout of heart and digestion might see: *www.ihatemarmite.com*.

Rule Britannia!

niques used by the other? Are there intrinsic differences?

Productivity: We continue to see productivity issues within the Cave. It is often run in batch mode, with time slots, limited access, and a (probably ineffective) urgency to be “efficient.” Users want a Personal Cave to can sit inside and think, with the computer doing nothing. Some of our most effective sessions have involved users working for a while, then sitting on the floor of the Cave talking, then stepping outside to look at things, then going back in to look some more.

Collaborating in a single Cave has proven very important—we almost always have at least two domain experts in the Cave at a time so that we can hear them talk to each other. One disadvantage is that they can’t point with real fingers because only one viewer is head-tracked. In some ways, long-distance collaboration solves this problem because each viewer has his or her own Cave, but that introduces other problems of synchronization and communication. Users of this application were the first to ask for physical objects to augment the virtual environment—anatomy books and printed medical images, for example. Our current practice is to iterate moving out of the Cave to work with reference books, keeping the image on the front wall of the Cave for reference, then returning to the Cave for further exploration.

MARS TERRAIN EXPLORATION

In the fourth and final IVR application discussed here, terrain from Mars is modeled for planetary geologists so they can “return to the field.” The predominant means of visualizing satellite data is 2D imaging (at multiple resolutions) with color maps representing elevation and other terrain attributes. However, Mars Orbiter Laser Altimeter (MOLA) missions give a third dimension to the terrain. Incorporating topography information into analysis opens new avenues for studying geological processes. Our primary goal is to create a complete model integrating all of the available data that is interactively viewable. The geologists’ ultimate goal is to understand the geological processes on Mars from the current surface structures, including the relationships among the strata that are partially visible on the surface. More pragmatically, studying the terrain lets us search for sites for further study and identify potential landing sites for future missions.

In Figure 7, James Head of Brown's Geological Sciences Department uses a handheld IPAQ PDA, which provides convenient interactive input to control global position and rendering styles while flying over the terrain of Mars. Mars data available to the exploration process include elevations from the MOLA and color images of surface swathes from the Mars Orbiter Camera (MOC). Altitude information we are displaying currently is on a 7200 x 7200 grid. The thousands of color swathes can be as large as 6000 x 2000 pixels and cover a small region of the surface.

Figure 7—Geologist using PDA to aid Mars flyover

Scale and navigation: What's the best scale for Mars study? At full scale, navigation becomes extra-planetary and unfamiliar. Smaller scales appear to be more appropriate, although the scale that we currently use requires hybrid navigation: flying for most movement coupled with a much smaller-scale map for taking larger steps. As with many of the other applications, users are more comfortable driving than being driven, particularly moving backwards.

Performance: One of the most significant limitations for this application is performance. A consistent 30+ frames per second is very difficult to maintain. A naïve approach to rendering the 7200 x 7200 heightfield yields about 100 million triangles! LLNL (Lawrence Livermore National Laboratory)'s ROAM (Realtime Optimally Adapting Meshes)] software for view-dependent terrain rendering improves performance, but more progress is necessary. The geologists consistently request more detail, exacerbating the problem.

CONCLUSION

While IVR and scientific visualization have their own significant histories and accomplishments as separate fields, their intersection is less common and considerably more immature. Because scientific visualization so often involves handling large data sets, it is a great stressor of all aspects of IVR technology. To create satisfactory user experiences, IVR's requirements of low latency, high frame rate, and high-resolution rendering and interaction handling (e.g., head and hand tracking) are much more stringent than those for workstation graphics. This may force more severe restrictions in the complexity of models and data sets than IVR can adequately handle. Such constraints put a serious burden of proof on IVR proponents to show that the advantages of the immersive experience outweigh the disadvantages of higher cost, of scheduling a one-of-a-kind institutional facility such as a Cave, and, most important, of complexity restrictions in what can be visualized.

fac.activities@cs.brown.edu

Michael Black. In December Michael attended the Neural Information Processing Systems meeting in Vancouver where he and his colleagues from the Division of Applied Mathematics and the Department of Neuroscience presented a paper on the probabilistic inference of arm motion from neural activity in the motor cortex. After the main conference he gave a talk at the workshop on Directions in Brain-Computer Interface (BCI) Research, held at Whistler. From snow-covered mountains Michael traveled to the sunny beaches of Kauai for the IEEE Conference on Computer Vision and Pattern Recognition, where he served as an area chair and presented a paper with Fernando De la Torre on Dynamic Coupled Component Analysis. Michael had two Ph.D. students graduate over the last few months: Hedvig Sidenbladh from KTH (the Royal Institute of Technology) in Stockholm and Fernando De la Torre from La Salle School of Engineering in Barcelona.

In February, the Motion Capture Lab was installed at Brown. This state-of-the-art facility, a joint effort of Computer Science and Cognitive and Linguistic Sciences, will be used to capture 3D human motion for research in graphics, machine learning, computer vision, and human navigation.

Michael's research on human motion analysis was supported this fall by another generous gift from the Xerox Foundation.

Amy Greenwald. Amy has recently won two major awards—a five-year NSF Career Grant, their most prestigious award for new faculty members. She received this award for her project 'Computational social choice theory: strategic agents and iterative mechanisms.' Amy has also become the CRA's Digital Fellow for 2002. She has been invited to the FCC to speak in May in support of their goal of building relations between academia and government. This program is supported by the National Science Foundation and

Photos courtesy of David Binder

Amy Greenwald and her new husband Justin Boyan

orchestrated by the Computing Research Association (CRA). It is intended to create ties between academic and industrial computing research communities on the one hand and information technology workers in federal, state, and local governments on the other.

Finally, she mentioned casually, "...and I'm getting married next week!" We've held this issue in order to include a wedding photo...(the NYT wedding announcement is on our website—very romantic too).

Maurice Herlihy. At the Latin-American Theoretical Informatics (LATIN 2002) symposium in Cancun, Maurice was invited by the Iowa State University Department of Computer Science to give the Miller Distinguished Lecture. He spoke on Algebraic Topology and Distributed Computing.

John Hughes. Said Spike, "I've been Papers Chair for SIGGRAPH this year; it's been an interesting experience. For one thing, it's made me think about my professional organization in a long-term way rather than just as 'the place to publish my next paper.' A surprising number of the challenges of the job turn out to be balancing short-term tactics against long-term strategy. But there are a few other challenges: three days before our committee meeting, I got email telling me one of the committee members had just had an emergency appendectomy. Then, as the committee began to arrive in Marietta, Georgia, it turned out that the hotel didn't have enough rooms for us, even though we'd booked months in advance. Fortunately, I was blessed with incredibly competent folks from SIGGRAPH who dealt with the hotel, so I didn't have to do any yelling and screaming. But I'll never go back to Marietta.

"The two-day meeting was an interesting experience: my job was purely administrative—get every paper discussed, get decisions made on every paper, make sure it's all done as fairly as possible—and I never got to learn any of the technical details of any paper, so that by the end, the content of the program was almost a surprise.

"To keep things fair, whenever a paper from some company or lab was discussed, committee members with a conflict of interest would leave the room and watch the status of the paper on a monitor in the hallway showing a spreadsheet of all the papers. It was pretty difficult to see your paper discussed for a long time, and then see the critical cell in the spreadsheet turn red instead of green. It was especially tough to come back into the room and have to be the person who presented the next paper and tried to say how good it was. Michael Cohen, of Microsoft, had a great suggestion: for next year, we should buy a crate of cheap crockery from Building 19 and a hammer. Then when your paper was rejected, you could smash a plate or two before returning to the fray. I liked the idea so much I offered to help pay for it, so we may see ‘Mike and Spike’s stress-relief center’ at future committee meetings.

“In the end, despite all the stress, we’ve got a terrific program, and it’s going to be a great conference. (It’s also going to be a hot one: we’re in San Antonio in late July.) I hope I’ll see some of you there.”

Philip Klein. Wavemarket, the company for which Philip is serving as Chief Scientist, received venture funding in the fourth quarter of 2001, perhaps the only wireless software startup to do so. Philip will be returning from his leave to teach in the fall.

Shriram Krishnamurthi. Shriram served on the program committee of Foundations of Object-Oriented Languages 2002 and was program co-chair of Practical Aspects of Declarative Languages 2002. He co-authored award papers at both Foundations of Software Engineering, 2001 and Automated Software Engineering, 2001. His invited talk at Lightweight Languages 1 resulted in a rather frightening photograph in DrDobbs (thanks, but yes, he’s seen it; no, he doesn’t need the URL; yes, he knows he won’t live it down). More seriously, he is increasing his indoctrination in the Brown scheme of things, having co-authored a paper with Spike with the word “Graphics” in its title. Fortunately, this does not imply any concomitant knowledge of graphics.

Nancy Pollard. Nancy is program co-chair for the brand new ACM Symposium on Computer Animation to be held this July: *sca2002.cs.brown.edu*. There is a lot of great computer animation research going on right now, and the goal of the new symposium, co-located at SIGGRAPH (the main computer graphics conference), is to give computer animation researchers a convenient forum for sharing their latest work.

Eli Upfal. Eli was elected a fellow of the IEEE; he was an invited speaker at a DIAMACS workshop on “Computational Complexity, Entropy and Statistical Physics.”

Don Stanford. Don, ’72, Sc.M. ’77, was the keynote speaker at the recent dedication of the Undergraduate Teaching Laboratories for the Division of Engineering.

Andy van Dam. Andy gave a distinguished lecture on “Immersive Virtual Reality for Scientific Visualization” at ETH, Zurich. He has been selected to receive CRA’s Distinguished Service Award for 2002, which will be presented at Snowbird on July 15. Also, Andy will receive the 2001 Harriet W. Sheridan Award for Distinguished Contribution to Teaching and Learning at Brown. Said Prof. William Risen, Chair of the Awards Committee, “Your faculty colleagues, both in and beyond your own department, cite the example you have established as a pioneer in computer science education, beginning with a course you developed in 1962. As a distinguished scholar, your willingness to take precious time to mentor students and junior faculty has inspired your colleagues. Your training of your undergraduate TAs is a model for all faculty. The Advisory Board of the Sheridan Center honors you for the inspiring example that you have set for your colleagues.”

CHARNIAK UNPLUGGED

I finally broke down and got myself (and my family) a cable Internet connection from our local cable company, Cox. Getting it was sort of a challenge as Cox had been providing the cable but farming out ISP (Internet Service Provided) part of the business to a separate company. This company had gone belly up about two weeks before I decided to joint the 20th century (a year or two late, depending on your computation). However, the thing that really stopped me in my tracks was when Cox asked me if I had a systems backup disk for my IBM windows system. The answer was that I did not, but that since the system was installed here in the CS department at Brown, if there was any problem I could simply bring it back here and they could re-install it. This explanation, however, cut no ice. No backup disk, no Internet installation.

At this point I went to see John Bazik, a long-time member of our technical staff. John, I am happy to say, solved the problem with the "out-of-the-box" type thinking for which Brown would like to be famous. He pulled out a blank disk, wrote "Windows XP backup disk" on it, and proudly handed it to me. I have saved the disk for the next Cox Internet customer here in the department.

One of my jobs here at Brown is to talk to prospective undergraduates who want to know more about our computer science department. On several occasions I have had the chance to talk to children of colleagues of mine, and this last week Kathy McKeown, chair of computer science at Columbia and a well known researcher in my area, and her daughter visited Brown. Her daughter was considering among other possibilities a program that stressed both computer science and art.

One of my selling points for Brown is the degree to which undergraduates can get involved in research here. I frequently tell prospective students, particularly those with some interest in graphics, that any time I have a student come to talk with me I can be confident that there will be an undergraduate working in the graphics lab here. Then after we have finished talking we perform the experiment by walking to the lab and I ask folks working there if any of them are undergraduates. Kathy's kind letter tells of the result in this case.

"Eugene -- I enjoyed talking to you on Tuesday about natural language, but I especially wanted to thank you for the selling points on Brown to my daughter. I can't tell you how much of an impression it made on

PERSONALS

RECOVERING gender-challenged paranoid schizophrenic wants understanding partner. Contact *HAL@conduit.cs.brown.edu.*

SWMC++ looking for SF, no Java please.

F JAVA wants sensitive M to share art, romantic sunsets and garbage collection.

INTERFACE specialist seeks high-performance server. (Theoreticians need not apply.)

SWM seeks dominant F for concatenation and constraint programming.

A NOTE to my detractors, I have switched to Linux. D. Vader

Alumni and friends of CS are invited to submit personal ads. Contributions fit to print will be in the fall issue. We wish to acknowledge the significant contribution of Lynette Charniak, wife of the Unplugged one, who suggested this column.

her that your experiment worked *and* that not only was the only student in the lab an undergraduate, but that he was also one who combined a humanities subject like archaeology with computer graphics, *and* that he had been to Jordan. Couldn't have done the trick better: she's now sold on a combination of art and computer graphics. And, for all I might have mentioned this possibility in the past, it had nothing like the effect of your experiment :)

So thanks!

Kathy"

My response:

"You are welcome. It sort of reminds me of the time we had a faculty candidate who came to this country from, I think, Greece, and on the way to dinner the group came across some people in traditional Greek costume who were on their way to a Greek-tradition-day event, or some such. The candidate asked us, semi-seriously, if we had set this up!

Eugene"

I was reading in the **New York Times** that legislators in Utah are threatening the University of Utah with dire consequences unless it revokes its rule against guns in classes. If I remember the details correctly, Utah has a new gun ordinance allowing anyone with a gun permit to carry the gun anywhere except grades K-12. This would, for example, include churches. (Or was it that churches were excluded, but K-12 was OK?) Utah had earlier made news when one clinically insane fellow was sold a gun when his background check showed that he had no criminal record. The legislature deliberately allowed for this possibility since they did not want the police to start making decisions about who was sane enough to carry a gun. Thus it was not too surprising that they did not place much weight on the University's argument that intellectual arguments should be settled by something less definitive than lethal force.

Not being able to resist, I sent one of my colleagues at Utah (Ellen Riloff) email asking what she packs when going to class. I thought I was being pretty clever, but it is clear that I was just an effete east-coast liberal (EECL) thinking along party lines. Ellen responded that one of her Utah co-workers got e-mail from no doubt another EECL asking for advice about what sort of gun he should get when coming to the university for a research visit. Would a rapid-fire rifle get in the way too much?

This did get me to wondering, however, if Brown had any rules about guns in classes. It would be pretty funny if I was making fun of Utah for having such a rule, while Brown did not. So our editor called up the University General Counsel's office to ask. Fortunately, Brown does not allow students to own or use "firearms". (**Student Handbook**, XI, "*Student Conduct*", Article VIII of "Offenses" reads: "Possession, use, or distribution of firearms, ammunition or explosives. The

During this grading process, Suzi has missed seeing deer on the field in the evening—the only deer out there lately is a John Deere!

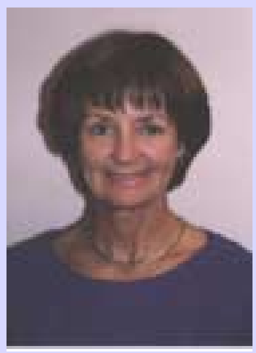
University defines firearms as any projectile firing device, especially those which are capable of causing harm to persons or damage to property. This includes but is not limited to conventional firearms (devices using gunpowder); all types of air rifles; BB, pellet, and dart guns; or any slingshot device.") Of course, this is not exactly the same thing as Utah's rule, as there is nothing here preventing the *faculty* from exercising their constitutional rights.

One of the jobs for Suzi Howe, our editor-in-chief, is laying out this rag. Naturally, the easiest way to do it is starting out at the front and working your way to the back. From my point of view this is both good and bad: good in that I get to procrastinate about writing my column, bad in that occasionally Suzi says that there is a blank space at the end of the paper and she needs more words from me. This just happened about five minutes ago. However, while I was in her office discussing this, I noticed some pictures of what seemed to be a major construction project on Suzi's desk. It seems that Suzi's husband is a big baseball fan and has had, for the last 20 years or so, a baseball field in their "back yard". They are now doing the grading work to make it much more professional. The field has the exact dimensions of Fenway Park! Upon seeing this I pronounced it definitely "***conduit!***-worthy"—a term of art among those of us who make this periodical their hobby. And besides, throw in a construction photo, (see previous page) and poof, no more extra space.

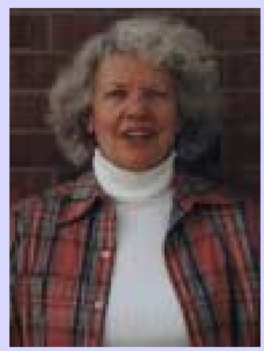
conduit!

A publication of
The Computer Science Department
Brown University

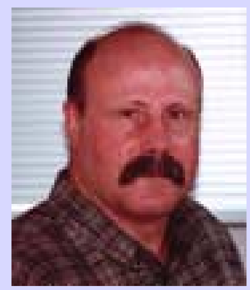
Inquiries to: ***conduit!***
Department of Computer Science
Box 1910, Brown University
Providence, RI 02912
FAX: 401-863-7657
PHONE: 401-863-7610
EMAIL: sjh@cs.brown.edu
WWW: http://www.cs.brown.edu/publications/conduit/



Suzi Howe
Editor-in-Chief



Katrina Avery
Editor



Jeff Coady
Technical Support

Department of Computer Science
Brown University
Box 1910, Providence, RI 02912, USA

conduit!

NON-PROFIT
U.S. Postage
PAID
Providence, RI
Permit #202

Printed on recyled paper

Address changes welcomed