

# SIGCSE 2007 Workshops

## Wednesday Workshops, 7:00 p.m. to 10:00 p.m.

### 1. **Planning and Executing Real Projects for Real Clients Courses**

Vicki L. Almstrum, *The University of Texas at Austin*  
David Klappholz, *Stevens Institute of Technology*  
Ken Modesitt, *Indiana University – Purdue University Fort Wayne*

For CS graduates who are employed in the IT world, many skills can be best learned, or even only learned, by working on real projects for real clients as part of a team. This workshop is part of a project to develop guidelines and materials to assist instructors as they develop and improve such courses. Participants will help refine a draft taxonomy of issues involved with teaching such a course and discuss guidelines for developing this type of course in terms of personal and institutional needs, as well as resources useful in teaching such courses.

### 2. **Cooperative Learning for CS1: Making It Work in Your Classroom**

Leland Beck, *San Diego State University*  
Alexander Chizhik, *San Diego State University*

Studies have shown that cooperative learning can be an effective instructional strategy in CS1. However, many faculty are reluctant to adopt cooperative learning because of uncertainty about taking on a new instructional role (as a facilitator instead of a lecturer). Faculty may also wonder whether students will learn required information while working in groups on a relatively small number of problems. This workshop will begin with an introduction to cooperative learning, followed by presentation of a set of class-tested learning activities for CS1 courses. We will then focus on a study of classroom management techniques, using examples taken from actual classrooms.

### 3. **What's the Difference? Assessing the Impact of Your Efforts**

Lecia J. Barker, *University of Colorado*

Adopting a new teaching approach? Using new assignments? Implementing a student research program? Systematic assessment can provide evidence about whether your efforts create the change in learning, motivation, or behavior you are trying to cultivate – and help you (and funders) make evidence-based decisions about resources. This high-level overview covers the basics for do-it-yourself evaluation or for understanding the approaches used by consultants, from theory to methods to analysis. Workshop includes sound research design and implementation, managing problems inherent to comparative studies, question writing for surveys and focus groups, interviewing techniques, analytical techniques (including differences between groups of students), and interpretation.

### 4. **Multimedia Projects for CS1 in Python**

Mark Guzdial, *Georgia Institute of Technology*

Constructing and manipulating media is a motivating context for students in introductory computing classes (CS1, but also CS0 and even CS0.5). Modern computers are capable of interesting media effects in reasonable running time, using simple and obvious algorithms that fit within the constraints of introductory courses' curricula. In this workshop, we will explore algorithms and working code in Python for the creation and manipulation of sound, image, MIDI music, and video data. I will describe some of the results at institutions using this approach. Example techniques covered will include music and story-telling, sound splicing and reversing, chromakey ("blue/green screen") image effects, animation, and Photoshop-like image filters.

### 5. **A Hands-On Approach to Mathematical Proof**

Douglas E. Ensley, *Shippensburg University*  
David A. Hastings, *Shippensburg University*

Sound formation of mathematical ideas progresses from concrete to abstract. This workshop will develop teaching material that applies this philosophy to the teaching of proof techniques to computer science majors in a first-year discrete mathematics class. Moreover, we will connect these ideas to the application of proof techniques in upper level courses such as testing and debugging. We will address traditional proof schemes, each from the point of view that the students can construct their own understanding of proof through carefully chosen experiences. The activities will take the form of paper-and-pencil worksheets and interactive Flash applets.

## 6. Learning to Program with Alice

Stephen Cooper, *Saint Joseph's University*  
Wanda Dann, *Ithaca College*

This introductory workshop, designed for instructors with minimal to no prior experience with Alice, offers hands-on experience programming with Alice. Alice is a powerful program visualization tool enabling students to “see” objects and work with object-oriented programming. Participants will learn how to use Alice to build virtual worlds and how to use this approach in introductory-level computing courses (introductory programming for majors, programming for non-majors, computer literacy, etc.) at the college or high-school level. Participants will receive a CD containing the latest version of the software and sample virtual worlds developed as part of NSF-0126833/NSF-0339734.

## 7. Programming with Greenfoot (Or: Introducing Java Via Games and Simulations)

Michael Kölling, *Computing Laboratory, University of Kent*

Greenfoot is a programming environment, developed by the creators of BlueJ, designed to allow teaching of object-oriented programming concepts – using Java – in a highly engaging and motivating context. Built in the tradition of Turtle Graphics, Logo, Karel the Robot and Squeak, Greenfoot offers a more integrated, graphical, interactive experience than previous systems. Building widely differing scenarios, such as simulations or games, is easy and quick. Greenfoot should be of interest to anyone teaching Java, especially in early programming courses (including schools). Greenfoot is available from [www.greenfoot.org](http://www.greenfoot.org). The workshop is practically oriented and will allow participants to use Greenfoot in their classroom immediately.

## 8. Introducing LEGO MindStorms NXT and its Use in CS Education

Myles McNally, *Alma College*  
Frank Klassner, *Villanova University*  
Pamela Lawhead, *University of Mississippi*

This workshop focuses on using the next generation of the LEGO MindStorms robotics platform – NXT – as an active learning platform in CS education. A complete reworking of MindStorms, it provides a number of new opportunities for CS educators. The workshop opens with an introduction to MindStorms and the new NXT release. The workshop will then focus on a hands-on introduction to tested series of laboratories for CS instruction. The laboratories range from CS 1-2 to AI, Programming Languages and Operating Systems. Participants will be able to purchase an NXT kit at a significant discount and will have priority to attend a follow-up NSF sponsored workshop.

## 9. Computer Security Essentials (Part 1): System Footprinting and Vulnerability Assessment

Paul J. Wagner, *University of Wisconsin – Eau Claire*  
Daren Bauer, *University of Wisconsin – Eau Claire*  
Thomas Paine, *University of Wisconsin – Eau Claire*  
Andrew Phillips, *University of Wisconsin – Eau Claire*  
Jason Wudi, *University of Wisconsin – Eau Claire*

This is the first of two hands-on workshops for CS educators seeking to develop curricula in computer security. We provide guided hands-on instruction on various Windows and Linux based tools commonly used for gathering information about, and assessing the vulnerability of, other systems. Participants will experiment with these tools as the presenters guide them through typical tool use scenarios. The session concludes with an information gathering exercise on an isolated network. Participants will use laptops running both Windows and Linux images pre-configured with common security “holes” so that they may experience first-hand the process of information gathering and vulnerability detection.

## 10. Teaching Cognitive Robotics with Tekkotsu

David S. Touretzky, *Carnegie Mellon University*  
Ethan J. Tira-Thompson, *Carnegie Mellon University*  
Andrew B. Williams, *Spelman College*

Cognitive robotics is a high-level approach to robot programming emphasizing primitives for perception, mapping, navigation, and manipulation. Tekkotsu is a software system for cognitive robotics developed at Carnegie Mellon, initially for the Sony AIBO robot dog, but now being ported to other platforms. It is open source, LGPLed, and available for free at [Tekkotsu.org](http://Tekkotsu.org). This workshop will provide hands-on experience with Tekkotsu programming on the AIBO and explore ways that cognitive robotics can be integrated into CS curricula. We will also discuss the use of robotics activities to attract women and minority students to computer science.

**11. Software Security**

James Walden, *Northern Kentucky University*  
Charles E. Frank, *Northern Kentucky University*  
Rose Shumba, *Indiana University of Pennsylvania*

Software security aims to produce software that functions correctly even when attacked. Unlike reactive network security techniques like firewalls that work around security flaws, software security focuses on getting software right. This workshop will teach participants how to think about security across the entire software development lifecycle. Participants will learn about software security practices like abuse cases, risk analysis, code reviews with static analysis tools, and penetration testing in a process agnostic manner. The workshop will include hands-on risk analysis and code review exercises using a demonstration blogging application written in Java that participants can use in their own courses.

**12. Using the Java Task Force Packages**

Eric Roberts, *Stanford University*  
Robb Cutler, *The Harker School*  
Scott Grissom, *Grand Valley State University*

The ACM Java Task Force (JTF) was created in 2004 and charged with developing resources to simplify the use of Java in introductory courses. The Task Force released Version 1.0 of its materials in August 2006, which are freely downloadable from <http://jtf.acm.org>. This workshop is designed to give instructors hands-on experience using the JTF packages so that they are prepared to adopt it in their introductory-level courses. The workshop will follow the structure of the online JTF tutorial and will focus on creating dynamic, interactive programs using the **acm.graphics** and **acm.gui** packages.

**13. Travels in “DNA Land” – Approaching DNA Sequence Analysis Through Word Play**

Mark D. LeBlanc, *Computer Science, Wheaton College (Norton, MA)*  
Betsey D. Dyer, *Biology, Wheaton College (Norton, MA)*

In a series of team-taught mini-lectures followed by hands-on activities, participants will learn how to introduce, motivate, and explain DNA sequence analysis by engaging in “word play” using regular expressions on English texts and then DNA sequence. Faculty wishing to experience an introductory but focused look at DNA sequence analysis and its place as a rich source of applications are especially encouraged to attend. Participants will take away a series of classroom-tested specifications and consider strategies for starting interdisciplinary collaborations. No prior knowledge of Perl is required. Participants will bring their own laptops pre-loaded with workshop materials.

**14. Teaching Object-Oriented Programming in Python**

Michael H. Goldwasser, *Saint Louis University*  
David Letscher, *Saint Louis University*

Python’s use in education has grown rapidly, due to its elegantly simple syntax. Though often viewed as a “scripting language,” Python is a fully object-oriented language with an extremely consistent object model and a rich set of built in classes. In this workshop, we share our experiences using Python in the context of an *object-oriented* CS1 course. We will begin with an overview of the language, with particular emphasis on the object-orientation. We then present several coherent teaching strategies and a variety of graphical and nongraphical projects. Both new and experienced Python users are welcome.

**15. Peer Led Team Learning in Computer Science**

Susan Horwitz, *University of Wisconsin-Madison*  
Steve Huss-Lederman, *Beloit College*  
Barbara Ryder, *Rutgers University*

We present a new approach to increasing the representation of undergraduate women and minorities in computer science: We use *active recruiting* to increase enrollment of under-represented groups, and *peer-led team learning* (PLTL) to improve retention. PLTL involves students working in small groups run by trained undergraduate Peer Leaders, several of whom will participate in this workshop. We will provide hands-on experience with a PLTL group meeting and Peer-Leader training. We will also present our experience using PLTL in introductory courses as part of a collaborative NSF project (NSF-0420436), including the benefits of PLTL and the resources needed to start a PLTL program.

## 16. Teaching Objects First in an Enlightening, Exciting Manner

David Gries, *Computer Science, Cornell University*

We discuss teaching programming to beginners, using Java. Objects, classes, and subclasses are taught before conditionals, loops, and arrays. Several points combine to make this not only viable but exciting and pedagogically sound: (1) a certain model for classes/objects; (2) a Java IDE that allows frequent live demos and postponement of method main to week 10; (3) programming assignments that include playing music, drawing bouncing balls and spirals, and manipulating jpeg files; (4) serious attention to testing using JUnit, (5) real discussions of program development, and (6) closed labs. Gries has taught the course for years. Software and other course materials will be provided on a CD.

## 17. A Hands-on Exploration of Educational Robotics

Zachary Dodds, *Harvey Mudd College*  
Douglas Blank, *Bryn Mawr College*

The past year has seen a remarkable surge in the robotics-based resources available to CS educators, e.g., Lego's release of its NXT platform. The Institute for Personal Robotics in Education began its development of new hardware, software, and curricular materials for teaching CS 1 and CS 2 using robots. Further, iRobot published a serial API for its line of Roomba vacuums. In this workshop a pair of presentations will frame these developments in terms of their core- and elective-CS applications. Two-thirds of the time will consist of guided experimentation with these new robot platforms and software.

## 18. Ruby on Rails

Barry Burd, *Drew University*

Ruby is an interpreted, reflective, purely object-oriented programming language. Unlike Java, the Ruby language has open classes, messages rather than method calls, closures, and "duck" typing. So Ruby broadens a student's understanding of the object-oriented paradigm. In addition, Ruby has an add-on named Rails. With Rails you can create a simple Web application (a Web interface to a database) in minutes. You can enhance the application with other add-ons -- add-ons for searching, for authentication, and even for credit-card processing. Some real-world practitioners claim a five- to ten-fold productivity increase when they switch from Java to Ruby on Rails.

## 19. Computer Security Essentials (Part 2): Intrusion Detection and System Defense

Paul J. Wagner, *University of Wisconsin – Eau Claire*  
Daren Bauer, *University of Wisconsin – Eau Claire*  
Thomas Paine, *University of Wisconsin – Eau Claire*  
Andrew Phillips, *University of Wisconsin – Eau Claire*  
Jason Wudi, *University of Wisconsin – Eau Claire*

This is the second of two workshops for CS educators developing curricula in computer security. Participation in workshop part one is recommended as a prerequisite. We provide guided hands-on instruction and experimentation on both defensive techniques and the understanding of exploits for the purpose of better defending systems. The session concludes with a hands-on exercise giving the participants an opportunity to participate in a carefully constructed and monitored cyberwar scenario; i.e. the participants will harden their systems, identify potential exploits and threats, and work to understand the mindset of the attacker by identifying weaknesses in all systems on the network.

## 20. Teaching with Alice and Media Computation

Stephen Cooper, *Saint Joseph's University*  
Wanda Dann, *Ithaca College*  
Barbara Ericson, *Georgia Institute of Technology*

This workshop is designed for instructors who have or will be teaching a course using Alice and/or Media Computation. Participants will learn strategies for supporting open-ended student projects. Particular attention will be paid on transition issues from Alice to Java. An example CS1 course, combining Alice with the Media Computation approach, will be described. Participants will receive a CD containing the latest version of the Alice software, the Media Computation libraries, sample rubrics for grading open-ended projects, and materials for the Alice and Media Computation course, as well as copies of the Alice and Media Computation texts.

## 21. Demystifying and Degeekifying Computing through K-12 Outreach

Katie A. Siek, *University of Colorado*  
Samantha Foley, *Indiana University*  
Jennifer Franko, *Indiana University*  
Emily Fortuna, *Rice University*  
Suzanne Menzel, *Indiana University*  
Laura Hopkins, *Indiana University*

In this workshop, we share our experiences creating a viable, self-sustaining outreach program that connects with young members of underrepresented groups. Now in its third year, IU's program, *Just Be*, originally targeted girls in K-12. Today our mission and design embraces all underrepresented groups. Participants will learn how to establish a program, create an interactive presentation, dispel unflattering myths surrounding computing, generate enthusiasm, advertise effectively, develop contacts at schools and clubs, obtain financing and staff support, train new presenters, fine-tune the content, reach sustainability, evaluate your success, and realize an exponential impact by recursively propagating your program to other schools.

## 22. Automatically Grading Programming Assignments with Web-CAT

Stephen H. Edwards, *Virginia Tech, Dept. of Computer Science*  
Manuel A. Pérez-Quiriones, *Virginia Tech, Dept. of Computer Science*

This workshop introduces participants to using Web-CAT, an open-source automated grading system. Web-CAT is customizable and extensible, allowing it to support a wide variety of programming languages and assessment strategies. Web-CAT is most well-known as the system that “grades students on how well they test their own code,” with experimental evidence that it offers greater learning benefits than more traditional output-comparison grading. Participants will practice hands-on how to prepare reference tests, set up assignments, manage multiple sections, and allow graders to manually grade for design. Bring your own Java or C++ assignment (small, with sample solution, and test cases if you have them) to work through. Go home ready to start using it in your own classes!

## 23. Active Teaching With Toys, Games, Stories, and Play

Shannon Pollard, *Elon University*  
Robert C. Duvall, *Duke University*  
Judith Hromcik, *Arlington High School*

This workshop will present techniques for using physical manipulatives, games and prizes, storytelling, and kinesthetic learning activities to teach core Computer Science concepts. The presenters will give many examples of how these techniques are currently used in their classes and facilitate a discussion of their benefits as well as pitfalls. Emphasis will be placed on how to integrate these techniques throughout the curriculum rather than as isolated special events. Participants will then brainstorm new ways to implement these ideas for various topics. All material presented and created will be added to an on-line repository.

## 24. Teaching Human Aspects of Software Engineering

Orit Hazzan, *Technion – Israel Institute of Technology*

This workshop addresses the teaching of human – cognitive and social – aspects of software engineering. It is based on the assumption that the *people* involved in software development processes deserve more attention. Human aspects will be highlighted by activities that aim at analyzing software teamwork from a social and cognitive perspective. Also, a course outline about human aspects of software engineering, including suggestions for specific activities and tasks, will be presented and discussed with the participants. The workshop is dedicated to the late Jim Tomayko, my co-author of *Human Aspects of Software Engineering*, published by Charles River Media in 2004.

## 25. Active and Cooperative Learning Techniques for Computer Science Education

Jeffrey J. McConnell, *Canisius College*

Active and cooperative learning provides a powerful mechanism to enhance depth of learning, increase material retention, and get students involved with the material instead of passively listening to a lecture. This workshop will use introductory active and cooperative learning material in a set of activities to give participants direct experience with and the chance to observe these techniques in action. There will be opportunities for open discussion of situations that participants may have already encountered and for the development of activities for the classroom. The workshop will be illustrated with example exercises from CS1/CS2 and other courses throughout the curriculum.

## Saturday Workshops, 3:30 p.m. to 6:30 p.m.

### 26. Object Think: Thinking Like an OO Programmer

Joseph Bergin, *Pace University, New York*

Object-oriented programmers think differently than procedural programmers. Object programs look different and have different structure. This has caused difficulty in our community and has caused some to conclude that objects have failed. Others have replied that no, objects haven't been given a chance yet. But if objects are really a different paradigm, then thinking differently is a natural consequence. Those who believe that object-orientation is all about classes and little else will benefit from a deeper view. Time will be available for discussion and even debate, though it will be action directed and not simply philosophical. Programming exercises in Java.

### 27. Using Microsoft .NET in the IS/CS Curriculum

Joe Hummel, *Lake Forest College*

Mike Morrison, *University of Wisconsin-Eau Claire*

Joline Morrison, *University of Wisconsin-Eau Claire*

Microsoft .NET is emerging as an industry-standard framework for creating Windows, Mac, Linux, and Web-based applications. The purpose of this workshop is to introduce .NET and allow attendees to evaluate its suitability for use in the CS or IS curriculum. The workshop will introduce attendees to console-based, GUI, database, and Web-based .NET applications using demonstrations and hands-on exercises. Materials and exercises will be example-based using C# and VB, and are suitable for student distribution. Attendees will have access to electronic copies of all workshop materials.

### 28. Multimedia Projects for CS1 and CS2 in Java

Barbara Ericson, *Georgia Institute of Technology*

Constructing and manipulating media is a motivating context for students in CS1 and CS2. Modern computers are capable of interesting media effects in reasonable running time. In this workshop, we will present algorithms and programs in Java that work with pictures, sounds, videos, and text. Example techniques include mirroring a picture, removing "red-eye", chromakey ("blue/green screen"), reversing a sound, changing the pitch of a sound, parsing web pages for information, creating frame-based animations, creating linked lists of MIDI music and using trees to represent scene graphs.

### 29. Developing Algorithm Visualizations for Computer Science Instruction

Tom Naps, *University of Wisconsin – Oshkosh*

Myles McNally, *Alma College*

Scott Grissom, *Grand Valley State University*

The JHAVÉ environment for algorithm visualization (AV) and its associated curricular materials emphasize pedagogical best practices rather than technology. JHAVÉ-based curricular materials are currently available for a wide variety of data structures and algorithms, as are Java support tools for the easy development of algorithm visualizations. Topics to be covered during this workshop will include best practices in using AV and design of JHAVÉ algorithm visualizations. Hands-on activities will first briefly focus on using the JHAVÉ environment as a pedagogical tool, followed by a more in-depth session during which participants create a complete algorithm visualization using the JHAVÉ support tools.

### 30. Teaching Machine Language Basics to Intro Students in Just One Week

David G. Kay, *University of California, Irvine*

Students in introductory courses understand some concepts better when they have a concrete understanding of some fundamental ideas about machine organization: memory as a sequence of bytes and programs as a sequence of machine language instructions. But traditional coverage of those topics takes too long and involves too many tedious details. This workshop will show how to introduce the essentials of machine language just a week of class, using a hypothetical pencil-and-paper machine that pares away everything inessential. This approach works in introductory courses for majors and non-majors, helping students grasp many core concepts including pointers, data structure performance tradeoffs, compilation, and multitasking.

### **31. Ajax**

Barry Burd, *Drew University*

The term Ajax stands for “Asynchronous JavaScript and XML,” a new name for a combination of existing technologies. With Ajax, a Web page loads as quickly as any other application. Parts of the page refresh themselves as needed, while other parts stay the same. No more waiting for a full page to come back from the server. The most widely used application of Ajax is Google Maps. The Google server sends an XML document to the visitor’s Web browser, and then the browser does all the formatting. The visitor can pan and zoom without reloading the entire page.

### **32. Introducing Pair Programming into Your Classroom**

Laurie Williams, *North Carolina State University*

Lucas Layman, *North Carolina State University*

Pair programming is a style of programming in which two students work together at one computer, collaborating on the same algorithm, code or test. The workshop presenters will provide information on how to introduce pair programming to students and how to manage classes that incorporate student pairs. They will also present a survey of research results related to the use of pair programming in education. Finally, workshop participants will take part in an active learning pair programming exercise that can be used with students in a class.

### **33. Secure Programming: Buffer Overflows and Other Common Mistakes**

James Walden, *Northern Kentucky University*

Charles E. Frank, *Northern Kentucky University*

Rose Shumba, *Indiana University of Pennsylvania*

Despite the discovery of thousands of software vulnerabilities each year, few computer science programs teach secure programming techniques. This hands-on workshop introduces participants to common programming mistakes that create security vulnerabilities, including buffer overflows, integer overflows, and format string bugs, and teaches the secure programming techniques needed to fix these common problems. Examples of vulnerabilities from programming textbooks and production software will be shown in C, C++, and Java. Participants will learn to detect and repair common security vulnerabilities through a series of three programming exercises, which are designed so that participants can re-use them in their own classes.

### **34. Using Magic to Teach Computer Science**

Tom Way, *Villanova University*

Grabbing and holding the attention of computer science students, or any students, is of fundamental importance to getting the message across. In this workshop, you will learn hands-on how to use easily performed magic tricks to demonstrate fundamental, important or difficult computer science concepts in a way that connects with students, breaks through apprehension about difficult topics and generates enthusiasm for the subject. We briefly will explore the case for employing magic (sparingly) as a pedagogical device, focusing mostly on learning a number of ready-to-use magical effects using common items and understanding how to learn more. All materials will be provided.

### **35. HarambeNet: Introducing Computer Science through the Modeling and Analysis of Social Networks**

Jeffrey Forbes, *Duke University*

Owen Astrachan, *Duke University*

The problems and motivating examples in computer science courses are often drawn from domains that primarily interest computer scientists like data processing. The goal of the HarambeNet project is to develop materials and modules for an alternative introduction to computer science centered around the Science of Networks. We hope to leverage the broad student interest in social networking sites like facebook to broaden participation in computer science. This hands-on workshop will provide background in networks, social networks, and visualization using examples from sociology, biology, and economics. Participants will work through paper and computer-based exercises modeling and analyzing social networks.

# SIGCSE 2007 Birds of a Feather

The following Birds-of-a-Feather sessions were organized for attendees of the SIGCSE 2007 symposium.

## Teaching with Alice

Wanda Dann, *Ithaca College*  
Stephen Cooper, *Saint Joseph's University*  
Don Slater, *Carnegie Mellon University*

Instructors experienced in teaching with Alice will be invited to share teaching strategies with those interested in teaching with Alice. The session will provide an arena in which to ask questions regarding how to adapt Alice instructional materials to specific kinds of courses. Attendees will be able to share pointers to web sites where others may find materials, such as syllabi, student projects, exams, and other assessment resources. In addition, a short presentation will be made regarding recent efforts of the discussion leaders in developing curriculum resources for transitioning from Alice to Java.

## A Town Meeting: SIGCSE Committee on Expanding the Women-in-Computing Community

Gloria Childress Townsend, *DePauw University*  
Paula Gabbert, *Furman University*

In January of 2004, we organized the second SIGCSE Committee (Expanding the Women-in-Computing Community). We sponsor a Town Meeting, during each SIGCSE, to disseminate information concerning successful gender issues projects (e.g. Lecia Barker from NCWIT spoke for a few minutes last year). We also foster discussion and brainstorming for the coming year. We invite all SIGCSE members who are interested in gender issues to attend and to join our listserv: To subscribe, send e-mail to [LISTSERV@acm.org](mailto:LISTSERV@acm.org). The complete body of your message should be: subscribe SIGCSE-WOMEN-COMM <your\_name> where <your\_name> designates the name you want listed in listserv records.

## Computing Education Oral History Project

Alison Young, *Unitec New Zealand*  
Barbara Owens, *Southwestern University*  
Vicki Almstrum, *University of Texas at Austin*

This BOF is designed for those who wish to find out more about the project and ways to get involved with it. The project is an extension of a Working Group at ITiCSE 2005. At the end of the session, audience members should understand the goals of the oral history project; have a basic grasp of the fundamentals of conducting and preserving interviews. The session is intended to provide the information needed for someone to decide whether they would attend a training workshop to become an interviewer. The session should also result in refinement of selection criteria and names of individuals who are candidate interviewees.

## Accessibility

Katherine Deibel, *University of Washington-Seattle*  
Brian Rosmaita, *Hamilton College*  
Mary Anne Egan, *Sienna College*  
Robert Cohen, *University of Massachusetts Boston*  
Robert Siegfried, *Adelphi University*

This BOF is aimed at bringing together the members of the SIGCSE community who share an interest in accessibility and disability issues. By forging such connections, we can begin to share resources and knowledge. This session will include discussions of current accessibility issues, including how to support students with disabilities in our courses and how to incorporate accessibility into the curriculum. We will also discuss the creation of a centralized resource for information on disability issues in computer science education.

## BlueJ Users Forum

Ian Utting, *University of Kent, UK*  
Michael Kölling, *University of Kent, UK*  
John Rosenberg, *Deakin University, Australia*

A forum for teachers using the BlueJ IDE to share their experiences, and to discuss recent and future developments with members of the BlueJ team.



## **Using Windows Academic Program Material in the Classroom**

Arkady Retik, *Microsoft Corporation*  
Dave Probert, *Microsoft Corporation*

The Windows Academic Program provides universities with Windows kernel source code, curriculum materials, and a project environment for teaching and research operating system principles. The program components include the Windows Research Kernel (WRK), the Windows Operating System Internals Curriculum Resource Kit (CRK) and ProjectOZ. This BoF intends to gather educators who are in a position to share their experience of using the program material; it's also an opportunity to provide input to the participants.

## **A Media Computation Art Gallery and Discussion**

Mark Guzdial, *Georgia Institute of Technology*  
Barbara Ericson, *Georgia Institute of Technology*

Media Computation is an approach to computing education that engages students through projects focusing on the creation and manipulation of digital media. This BOF has two parts. In part one, the *art gallery* part, media created by students from around the country in their computer science classes will be on display (with coffee, tea, and cookies provided). In the second half, the *discussion* part, participants will get a chance to talk to the teachers of these students about the assignments that led to the media art work and about the techniques that the students used.

## **Teaching Track Faculty in CS**

Donald J. Slater, *Carnegie Mellon University*  
Daniel D. Garcia, *University of California at Berkeley*

A panel at SIGCSE 2004 illustrated the trend in computer science departments to hire faculty to teach in a teaching-track position that parallels the standard tenure-track position, providing the possibility of promotion, longer-term contracts, and higher pay for excellence in teaching and service. This birds-of-a-feather is designed to gather educators who are currently in such a position to share their experiences as members of the faculty of their departments and schools, and to provide opportunities for schools considering such positions to gather information.

## **K-12/Higher Education Partnerships: What Can We Do for You?**

Mindy Hart, *Purdue University*  
James Early, *Purdue University*  
Dennis Brylow, *Marquette University*

Many courses are being offered in K-12 education under the "Computer Science" umbrella. Several factors influence the content and design of each of these courses. This session will pose the question of what factors influence the status and resources allotted to high school computer science courses. K-12 educators are encouraged to share ideas about how higher education institutions have helped them promote computer science, as well as ways they would like additional support. Additionally, higher education personnel are encouraged to share what is currently being done on their campus to provide support and services for the K-12 environment.

## **Games across the Curriculum: Can We Quantify Their Effectiveness?**

Daniel Cliburn, *University of the Pacific*

This BOF session will focus on how faculty members use games in their teaching, and whether it is possible to evaluate the effectiveness of computer games as instructional tools in the undergraduate computer science curriculum. Games are thought to increase student motivation, but is the empirical evidence definite? Are we turning some students off with overuse of games? Can we assess the actual improvement in a student's scores because of game assignments? What steps can we take to measure the effectiveness of various games approaches to teaching and learning, and should we be doing this?

## **Assessment and Accreditation: What Works and What Doesn't**

Donald B. Sanderson, *East Tennessee State University*  
Andrea Lobo, *Rowan University*

ABET-CAC's newly revised criteria for program accreditation make student outcomes assessment even more central to the accreditation process. The purpose of this BOF is to share experiences on what methods have worked and which have not worked in integrating assessment into all of the other day to day duties of faculty. The moderators of this BOF have recently finished their respective ABET accreditation visits and will share the learned lessons in conducting student outcomes assessment.

## **It is Time to Review the SIGCSE Reviewing Process**

Amruth Kumar, *Ramapo College of New Jersey*  
Patricia A. Joseph, *Slippery Rock University of PA*  
Paul Wagner, *University of Wisconsin - Eau Claire*  
Michael Goldweber, *Xavier University*

Double-blind peer review process has long been used in SIGCSE. There is growing sentiment that both the review process and the resulting reviews deserve re-examination. This session will provide a forum for stakeholders in Computer Science education to share their thoughts on possible improvements/alternatives to the peer review process; and identify characteristics of constructive reviews. It is hoped that this discussion will lead to a constructive review of the double-blind peer review system.

## **Computer Science Unplugged++: Showing What is Interesting about Computer Science**

Lynn Lambert, *Christopher Newport University*  
Michael Fellows, *The University of Newcastle, Australia*  
Leigh Ann Sudol, *Fox Lane High School*

Computer Science Unplugged is a set of kinesthetic, fun activities that cover many core areas of computer science including searching, networks and routing, modularization, and object-oriented thinking. Recommended in ACM's A Model Curriculum for K-12 Computer Science, the Unplugged book describes error correction via the magic trick of parity bits; demonstrates pixels and image representation using spray paint; shows how binary number work, and examines user interfaces. We will discuss these and others ideas that demonstrate the value and relevance of computer science to students and to the public as part of outreach and precollege and college classes.

## **Showing Students That We care**

Ed Gehringer, *North Carolina State University*

Teaching is not just about presenting material and assessing students' knowledge. It is also about relating to students—the care we show about them as individuals, the way we address them in class, and how we react when they are having trouble. In this BoF, participants will share tips on how to show students we care. Here are some examples: Learning students' names (perhaps by taking pictures of them early in the semester), contacting students who seem to be struggling, appointing class “representatives” whose job it is to get feedback from the class and share it with the instructor.

## **The Ideal, The Perceived, The Reality: K-12 Computer Science Curriculum**

Frances P. Trees, *Drew University*  
Don Allen, *Troy High School*  
David L. Burkhardt, *West Muskingum Middle School*

Representatives from middle school, high school, and university computer science departments will lead a discussion on designing and implementing a K-12 computer science curriculum. The discussion may include the following:

- \* curriculum priorities and concerns in middle school and high school
- \* designing and implementing a K-12 curriculum (Where do I start?)
- \* impediments to curriculum implementation in the middle school and high school
- \* strategies to deal with these impediments (Can the university computer science departments help?)
- \* issues and ideas for a national computer science curriculum

## **A Formal Launch of an ACM SIG on Information Systems Education**

Paul Leidig, *Grand Valley State University*  
David Feinstein, *University of South Alabama*

During a BOF session at SIGCSE 2006, a group of interested Information Systems faculty met to discuss the formation of an ACM SIG focused on Information Systems curriculum, accreditation, and related IS Education issues. The discussion generated an extremely favorable response, and was followed by a similar meeting at SIGITE 2006 in Minneapolis, MN. In preparation for submitting the SIG petition to the ACM SIG Board, a group of petitioners (see below) offers this BOF.

## **New Paradigms for Introductory Computing Courses**

Steven Wolfman, *University of British Columbia*

Heidi Ellis, *Trinity College*

Charles Kelemen, *Swarthmore College*

Curt White, *DePaul University*

Your intro course works great... for those who take it and ace it. What about the people that don't take or don't thrive in our introductory courses? How can we introduce diverse students to CS in new, effective, and attractive ways? As computer science increasingly impacts everyone's lives, it is critical that we find ways to attract and educate diverse students. This BOF will provide a forum for practitioners to share best practices, bright ideas, and wild fancies for teaching computer science to novice students from many different backgrounds. Notes from this BOF will be shared in a suitable forum.

## **Computational Thinking**

Peter B. Henderson, *Butler University*

Jeannette Wing, *Carnegie Mellon University*

Judith Gal-Ezer, *Open University of Israel*

Jeannette Wing's March 2006 CACM Viewpoint article *Computational Thinking* presents a compelling general view of the discipline of computing. It provides a powerful framework for understanding and teaching the fundamental principles of computing. Teaching students at all levels to think computationally is an important life skill. Accordingly, these ideas can help to define new ways of exciting future generations about relevance of computing and the value of careers in computing.

Tangible ways for achieving this and for initiating curriculum reform founded on Computational Thinking will be discussed, along with ideas for challenging the traditional "computing career implies programming" viewpoint.

## **Education, Computers and Society**

Joseph D. Oldham, *Centre College*

Florence Appel, *Saint Xavier University*

This session is for anyone interested in computing, education, and society to get together. SIGCSE members know the importance of societal issues in the CS curriculum. SIGCAS (Computers and Society) members are similarly committed to investigating such issues. Several SIGCSE members who hold a joint membership in SIGCAS are interested in once again discussing approaches to teaching societal issues in computing. We can also continue to look for points of connection between the two SIGs.

## **Sundials in the Shade: Recruiting and Retaining Women in Undergraduate Computer Science Programs**

Rita Powell, *University of Pennsylvania*

Jean Griffin, *University of Pennsylvania*

David Matuszek, *University of Pennsylvania*

Diana Palsetia, *University of Pennsylvania*

Women enter computer science majors at rates that are disproportionately lower than men and leave at rates that are disproportionately higher. Results of a study conducted in 2003-04 at the University of Pennsylvania provided information that led to curricular and cultural changes in the undergraduate program and the department to stem the attrition of women. The Sundials in the Shade BOF proposes to discuss the Penn study, the changes that our department has made and how this has impacted persistence in the major.

## **Database Security**

Mario Guimaraes, *Kennesaw State University*

Paul Wagner, *University of Wisconsin-Eau Claire*

Many security professionals believe that "The main battlefield between hackers and security professionals has moved from the network/OS layer to the database applications layer". However, there are a small number of database security courses being offered compared to network and operating systems security. This BOF will focus on identifying essential Database Security topics and breaking them down into the ones that should be taught in the CS curriculum. We hope that this session will also provide a forum for future discussions in this topic. The intended audience consists of any faculty that teaches Database or Security Classes.

## **Working in a Small Department**

Cathy Bareiss, *Olivet Nazarene University*

There are many challenges and strengths associated with teaching in a small department typically not found in larger departments. By working together, small departments can find ways to address these challenges and maximum the strengths. This BOF will allow attendees to share their successes and find help with their current challenges while members of small departments. We will also spend some time identifying our next areas of focus and people willing to work on these areas.

## **Community College Challenges**

Frank Lanzer, *Anne Arundel Community College*

Donna Hiestand-Tupper, *The Community College of Baltimore County-Essex*

Community colleges are ideally situated to respond to the needs of their communities. The needs of underrepresented populations are of major concern and tend to focus the community college effort. The down side to this closeness may be a distance, physically and philosophically, from articulating universities. Efforts to recruit and retain women and minorities, and alliances with universities are two of the intended topics. Community colleges attract a wide variety of underrepresented populations. Many of these students transfer to universities. This Birds of a Feather session will foster a dialog and explore efforts to recruit and retain all underrepresented populations.

## **Trends in Computer Science Capstone Courses**

John Gudenas, *Aurora University*

We will discuss computer science capstone courses from a contextual view as well as from an outcome view. Drawing from the input of our group: structure, assessment, what seems to work, what didn't appear to work and any other relevant issues will be included in the forum. Consider that capstone courses are an important part of the undergraduate program and may very well provide or hone the social, verbal, research or other skills that lead our students to employment or graduate school. Participants can expect to learn from the experience of others as to the effectiveness of actual pedagogical techniques and content.

## **New Thinking Needed! Opening the Way for Alternatives to Current Gender and Computer Science Research Approaches**

Carol Frieze, *Carnegie Mellon University*

Orit Hazzan, *Technion - Israel Institute of Technology*

Do we really believe that the predominant approaches to how we think about, and act on, issues relating to broadening participation in CS work? Have the major approaches to gender and CS research become part of the problem? Should we consider alternative approaches? This BOF discusses the limited implications for change afforded by the current body of "Gender and CS knowledge", and addresses how different working assumptions and research approaches may result in different conclusions, recommendations and strategies for broadening participation in CS.

## **Logo, Alice, et al., Oh My! Which Graphical Language to Use in CS1?**

David J. Malan, *Harvard University*

The sheer volume of keywords and syntax in even the simplest of Java programs (e.g., "hello, world") practically defies explanation on CS1's first day. Moreover, too often do semicolons and the like interfere with students' mastery of more important constructs (e.g., conditions, loops, variables, etc.). And so do some instructors turn to graphical programming languages like Alice, Crickets, Karel, Logo, and Scratch for students' first foray into programming. But which to use? Which is best? Which do you use? This session is an opportunity for instructors to share and learn from each other's experiences with these and other graphical languages.

## **Strategy for Teaching Business Intelligence for CS and IS**

Solomon Negash, *Kennesaw State University*

Ying Xie, *Kennesaw State University*

A course in data warehousing and business intelligence requires depth in business and technical learning outcomes, catering to Information Systems (IS) and Computer Science (CS) majors, respectively. Because of the interdependence between the business and technical developments in these courses both CS and IS majors need to be exposed to the concepts. A strategy on how to achieve this goal with consideration for faculty workload is discussed in this BOF.

## **The Automata and Formal languages Course: Its Central Role in the Computer Science Undergraduate Curriculum**

Michal Armoni, *Weizmann Institute of Science*  
Rakesh Verma, *University of Houston*

The automata and formal language course is considered a difficult course, and many students fail to see its relevance to their future computer science careers. Recent discussions within the CS education community hint that even some CS faculty are attaching low priority to this course in the current CS undergraduate curriculum. We feel that this course has a very important and unique role, as a representative of an entire subfield, with unique characterizing thinking patterns. We hope to fruitfully discuss ways to improve the status of this course, and to improve the representation of the subfield in the professional societies.

## **The Use of Wikis in Undergraduate CS Courses**

Robert G. Willhoft, *Roberts Wesleyan College*

Encouraging contribution and collaboration by students using wikis both in specific courses and across the curriculum. This BOF will encourage discussion about using a wiki in the following ways:

- Discussion forum for the textbook
- A replacement for standard written homework assignments
- A place for students to collaborate in studying for quizzes and exams
- Development of highly customized help for students working on labs and/or projects
- Student collaboration on written assignments
- Student/faculty collaboration on programming projects
- Exploration of tools used in a given course

## **The Current State of the Advanced Placement Computer Science Program**

Laurie White, *Mercer University*  
Ann Shen, *Bishop Strachan School*  
David Reed, *Creighton University*  
Tracy Ishman, *Plano West Senior High School*  
Donald Allen, *Troy High School*

Members of the APCS Test Development Committee will lead a discussion of the items of interest in the APCS program (CS-1/CS-2). These items may include, but are not limited to:

- the use of case studies
- the GridWorld case study (to be used on the exam starting in 2008)
- issues of course equivalence between the AP program and college and university CS-1/CS-2 courses.
- the importance of the AP Course Audit for high schools and universities
- syllabi preparation for the AP Course Audit
- results from the 2006 AP exam
- the use of Java 5 in the AP program

## **Professional Certifications in CS Undergraduate Programs**

Ariel Ortiz, *Tecnológico de Monterrey, Campus Estado de México*

As CS educators we know the importance of education over professional training. Yet we cannot ignore that employers commonly affirm that recently graduated students have a lack of practical skills required in “real world” situations. Important IT companies have academic programs that allow students to get certified in their specific technologies before they graduate. Anecdotic evidence has shown that a newly graduated student is more marketable if he or she has one or more professional certifications. This BOF will allow faculty members to discuss the pros and cons regarding this issue.

## **CS Equals Programming Stereotype: Change Programming Contests to Broaden Views on CS**

David P. Voorhees, *Le Moyne College*

The view that computer science equals programming has been promoted by CS departments and professional organizations through their support of programming contests. Recent discussions on what a CS education should entail present an opportunity to restructure the contests to promote the breadth of computer science. This BOF will discuss ideas for changing the structure of programming contests, including types of exercises, location (centralized versus decentralized), duration, and judging criteria. The goal is to begin a discussion that leads to a change in contest structure that promotes the breadth of CS instead of the narrow CS equals programming stereotype.

### **Writing in Computer Science Education**

Mark E. Hoffman, *Quinnipiac University*

Jeffrey A. Stone, *Pennsylvania State University*

David S. Herscovici, *Quinnipiac University*

Elinor M. Madigan, *Pennsylvania State University*

For over twenty years, writing has been part of the Computer Science education literature. The literature promotes the importance of writing, gives several examples that authors consider successful, and often exhort fellow CS educators to adopt more writing in their courses. In several instances, authors report adopting writing as part of a university-wide mandate or assignment to a writing intensive course. In this session, we will discuss the challenges of adopting writing in CS education. We will share types of writing that CS educators use, and explore recent trends where writing is used to promote teaching and student learning.

### **Mathematical Reasoning in Computer Science**

Doug Baldwin, *SUNY Geneseo*

Peter Henderson, *Butler University*

Judith Gersting, *University of Hawaii, Hilo*

William Marion, *Valparaiso University*

Mathematical concepts and reasoning are important tools for computing students to understand and be able to use. This has been recognized by computer science, software engineering, and mathematics educators in the evolution of the latest curriculum guidelines. This session will foster discussion of math's evolving place in computing education, including but not limited to new curriculum models, new courses, mathematics early, integration of mathematics throughout the curriculum, available resources, and promoting change. All computing educators who wish to explore and discuss these and similar topics from any point of view are encouraged to attend.

### **Pedagogy for Electronically Supported Classroom Interaction**

Richard Anderson, *University of Washington*

Dave Berque, *DePauw University*

Beth Simon, *UCSD*

Many instructors use active learning techniques to promote student engagement and assess class understanding. As an increasing number of students bring wireless devices to the classroom, there is growing interest in incorporating these devices into classroom instruction. In this birds of a feather session, we will examine how wireless networked devices can be used to support active learning. We are interested in hearing about pedagogical practices used in different sub-disciplines of Computer Science. The emphasis will be on the pedagogical techniques, and not the individual systems. The three discussion leaders have developed classroom interaction systems, and have substantial experience using the systems in the classroom.

### **jGRASP: Using Dynamic Viewers to Teach Data Structures in Java**

James Cross, *Auburn University*

Dean Hendrix, *Auburn University*

David Umphress, *Auburn University*

jGRASP is a freely available IDE with visualizations for improving the comprehensibility of software. A new structure identifier viewer has been added which is able to automatically detect and render many traditional data structures including stacks, queues, linked lists, and binary trees. The viewer provides a state-based animation of the data structure in a textbook-like presentation view as the user steps through a program. This birds of a feather session will focus on (1) how this new data structure viewer can improve the teaching and learning experience for students and (2) how the viewer itself can be improved. <http://jgrasp.org/>

# SIGCSE 2007 Faculty Poster Sessions

Posters present work-in-progress and other topics for which dialog with Symposium attendees is particularly appropriate. Posters are on display Friday morning and afternoon, and authors will be available for discussion during these sessions.

## Session I: Friday, March 9, 10:00 – 12:00, Exhibit Hall I

### Self-Sorting Objects

Seth D. Bergmann, *Rowan University*

### Experimental O/S Lab on a Dime

Dennis Brylow, *Marquette University*

### Instructional Applets for Computer Architecture Concepts

Teresa W. Carrigan, *Blackburn College*

Daniel Steven IV, *Blackburn College*

### Defense Against the Dark Arts: Using Computer Security to Teach Core Computer Science Concepts

Jack W. Davidson and Clark L. Coleman, *University of Virginia*

Mark W. Bailey, *Hamilton College*

Jeff Zadeh, *Virginia State University*

### CASE Tools for Cleanroom Software Engineering

Gabriel J. Ferrer, *Hendrix College*

### Open Source for Homework: Real Projects, Peer Reviewed

Ed Gehringer, *North Carolina State University*

### Educational Uses for VDE

Michael Goldweber, *Xavier University*

Renzo Davoli, *Università di Bologna*

### Goal-Oriented Instruction of Computer Science Concepts – Demonstrated in the Logic Programming Paradigm

Bruria Haberman, *The Weizmann Institute of Science*

### Creating an Open Certification Process

Cem Kaner and Tim Coulter, *Florida Institute of Technology*

### Bioinformatics Algorithms in Introductory Bioinformatics Interdisciplinary Course

Yana Kortsarts and Robert W. Morris, *Widener University*

### A Useful Case Study in Algorithmic Experimentation: Unexpected Timing Results for Heapsort

Gerald Kruse, *Juniata College*

### Integrating BlackBerry Wireless Devices into Computer Programming and Literacy Courses

Qusay H. Mahmoud and Allan Dyer, *University of Guelph*

### Social Responsibility Meets the Capstone Course: The E711 or “I am OK” Phone Message Service

Victor Matos, *Cleveland State University*

Rebecca Grasser, *Lakeland Community College*

### A Tool for Facilitating Students’ Construction of Proofs

Hugh McGuire, *Grand Valley State University*

**Computer Theory with Objects**

Kirby McMaster, Nicole Anderson and Ronald Peterson, *Weber State University*

**LittleFe + BCCD + CSERD = Acme; Computational Science Education on the Move**

Charlie Peck, *Earlham College*  
Paul Gray, *University of Northern Iowa*  
David Joiner, *Kean University*  
Tom Murphy, *Contra Costa College*  
Alex Lemann, *Earlham College*  
Kevin Hunter, *Earlham College*

**JLS: A Pedagogically Targeted Logic Design and Simulation Tool**

David A. Poplawski, *Michigan Technological University*

**Tablet PC Programming in the Artificial Intelligence Curriculum**

Jeffrey L. Popyack, *Drexel University*

**Functional Video Programming: Computing with Time-Based Media and the Functional Paradigm**

Samuel A. Rebelsky, Ian Lunderskov, Ian Young and Luis Zuleta-Benavides, *Grinnell College*

**Teaching GUI in Operating Systems Courses**

Lior Shamir, *Michigan Technological University*

**Using a Commodity GPU in an Undergraduate Parallel Computing Course**

Joshua Steinhurst, *Saint Mary's College of Maryland*  
Thorsten Scheuermann, *Advanced Micro Devices, Inc.*

**Curriculum and Lab Renovations for Teaching Server-Based Computing**

Lixin Tao, *Pace University*  
Kai Qian, *Southern Polytechnic State University*  
Xiang Fu, *Georgia Southwestern State University*  
Jigang Liu, *Metropolitan State University*

**Developing Security Checklists and Scorecards for CS Curriculum**

Blair Taylor and Shiva Azadegan, *Towson University*

**Data Structures: "How Much is That?"**

Christian Trefftz and Greg Wolffe, *Grand Valley State University*

**Integrating Cluster Computing into the Community College Computer Science Curriculum**

Robert Tureman, *Paul D. Camp Community College*

**Detecting Possible Software Plagiarism through Parse Tree Comparison**

J. Patrick Van Metre and Stephen H. Edwards, *Virginia Polytechnic Institute and State University*

**Performance Analysis Tools for Novice Programmers**

Sharon Smith Weber, *St. Edward's University*  
Dustin Young, *St. Edward's University*  
Veronica Villa, *St. Edward's University*

**A Study of Second Career Teachers – What Attracts Them and How Can You Keep Them?**

Robert G. Willhoft, *Roberts Wesleyan College*

**Does the Understanding of Time Complexity Lead to the Understanding of Space Complexity?**

Ela Zur and Tamar Vilner, *The Open University of Israel*



## **Session II: Friday, March 9, 3:15 - 5:15, Exhibit Hall I**

### **Using Games While Keeping Traditional Outcomes in CS1-3**

Jessica D. Bayliss, *Rochester Institute of Technology*

### **A Mixed-Method Evaluation of the Impact of Tablet PCs and DyKnow Software on Student Learning**

Dave Berque, *DePauw University*

Terri Bonebright, *DePauw University*

Joel Dart, *DePauw University*

Zach Koch, *DePauw University*

Shawn O'Banion, *DePauw University*

### **A Bayesian Analysis of the Impact of Competitive Learning on Introductory Computer Science Students**

William A. Booth, *Baylor University*

### **Assessing the Effects of Pair Programming on Individual Ability: Results from the First Year of a Two-Year Study**

Grant Braught, Tim Wahls and Louiz Ziantz, *Dickinson College*

### **Revitalizing Computer Science Education by Teaching Digital Media as Science**

Jennifer Burg, *Wake Forest University*

### **Quality-Based Assessment: Experiences in Computer Science Education**

Suzanne W. Dietrich, *Arizona State University*

### **Student IMPACT: A Programming Contest Alternative**

Mary Anne Egan and Jim Matthews, *Siena College*

### **Using Blender and the Blender Gamekit with Python in CS2**

Adrian German, *Indiana University Bloomington*

### **Teaching Alice in Hawaii: Cultural Perspectives**

Judith L. Gersting and H. Keith Edwards, *University of Hawaii at Hilo*

### **Multiple Intelligence Patterns and Performance in Freshmen CS Courses**

Lila Ghemri, Li Ma and Victor Mungai, *Texas Southern University*

### **Preliminary Results from a Study of Problems Encountered by Novice Pair Programmers**

Brian Hanks, *Fort Lewis College*

### **Computer Science Unplugged++**

Peter B. Henderson, *Butler University*

Lynn Lambert, *Christopher Newport University*

Michael Fellows, *The University of Newcastle*

### **Development Environment for Teaching a Functional Programming Language**

Brian T. Howard, *DePauw University*

### **Motivating and Teaching Computational Thinking to First Year Students**

Seung-won Hwang, *Pohang University of Science and Technology*

### **Cooperative Learning in a Large-Enrollment Introductory (Literacy Level) Course**

Debby Keen and Ryan McKenzie, *University of Kentucky*

### **Student Direction Interviews: Fighting Plagiarism in CS1 Courses**

Brian Kokensparger, *Creighton University*

### **Academic Service Learning in the Introduction to Computer Science Core Course**

Yana Kortsarts, *Widener University*

**Textbooks for Computer Forensic Courses**

Jigang Liu, Larry Gottschalk and Kuodi Jian, *Metropolitan State University*

**Software Frameworks as Didactic Tool: An Example**

Riccardo Massarelli, *Istituto Tecnico Industriale*

**Teaching Visual Basic.NET Using Multimedia Content**

Michael McMillan, *Pulaski Technical College and the University of Arkansas at Little Rock*

**Retaining Freshmen Interest in CS Using Collaborative Courses and Experiential Learning**

Krish Narayanan and Joanne Caniglia, *Eastern Michigan University*

**Students' Perceptions of Integrating Written Reflections into Continuous Assessment in CS**

Arnold N. Pears, *Uppsala University*

Mats Daniels, *Uppsala University*

**Web-Based Editable Java Tutorials**

Axel T. Schreiner and Roxanne L. Canosa, *Rochester Institute of Technology*

**Non-Traditional Undergraduate Students' Research Participation**

Xueqing (Clare) Tang, *Governors State University*

**Can Lego Robotics Really Boost Interest in Computer Science?**

Carol Taylor, *University of Idaho*

Kosuke Imamura, *Eastern Washington University*

**The Paradigm Shift from Procedural to Object Oriented: A Case Study**

Tamar Vilner, Ela Zur and Judith Gal-Ezer, *The Open University of Israel*

**Speaking of Software: Case Studies in Software Communication**

Charles Wallace, Anne Mareck and Thomas Vosecky, *Michigan Technological University*

**A Novel Multidisciplinary Service Learning Program with a Computer Science Foundation**

Thomas Way, Kallie Nordengren, Mary-Angela Papalaskari, Sue Metzger, Najib Nadi, Robert Styer, Melissa Coming and Adam Stepe, *Villanova University*

**Assessment of Student Learning Outcomes and the Software Design Life Cycle**

Deborah Whitfield and Patricia A. Joseph, *Slippery Rock University*

**Clinical Interviews and Process-Tracing Methods in Computer Science Education Research**

Timothy T. Yuen, *The University of Texas at Austin*

# SIGCSE 2007 SRC Graduate Student Research Abstracts

## **The Effects of the Handheld Network Service, “Look”, on the Acquisition of Common Ground**

Kibum Kim, *Virginia Polytechnic Institute and State University*

This project reports the results of an initial laboratory study of a proposed solution to the problem of coordinating entry into collaborative activity while protecting the privacy and control of individuals over small-scale devices. This is a thorny design problem that will increasingly plague user interface designers as we move further into the world of small, connected devices. The particular set of design requirements we work with are abstracted from prior work on the constraints and affordances of a middle school teaching/learning setting. Our variant of the paradigm provides preliminary evidence that our design is viable.

## **The Foundations of an Ontology-Aware Authoring System for Collaborative Learning**

Seiji Isotani, *Osaka University*

In the past few years Artificial Intelligence have been successfully been introduced to enhance Education through technologies. However, major challenges remain. Among these, we are concerned how to represent the knowledge of intelligent authoring systems and how to use this knowledge efficiently in the context of collaborative learning. Usual approaches provide "expertise" using a set of heuristics and domain theories built in the procedures. As a result they cannot justify their recommendations scientifically. Our approach uses ontological engineering to establish a common understanding of what a learning theory is and then to propose sophisticated techniques to reasoning on these theories.

## **Simplification of Flow Networks**

Ewa Misiolek, *University of Notre Dame*

Computing flows in a flow network is a fundamental graph theory problem with numerous applications. In this research, we present two algorithms for simplifying a flow network  $G=(V, E, s, t)$ , i.e., detecting and removing from  $G$  all edges (and vertices) that have no impact on any source-to-sink flow in  $G$ . Such network simplification can reduce the size of the network and hence the amount of computation performed by maximum flow algorithms.

## **Automated Extraction of Information About Images from Biomedical Literature**

Kalpesh Shah, *Arizona State University*

Considering the area of automated extraction of knowledge an important but little studied part of biomedical publications are the figures and accompanying captions and information regarding those images. Images along with captions are dense in information but they contain different grammatical constructs which makes it difficult to apply the standard information extraction methods. This project aims at automatic image extraction and annotation by extracting text and information about the image from the image caption and references to it from the body of the article. Thus our system is a text-mining and classification system for figures from any corpus of full text biological papers.

## **Peer-to-Peer File-Streaming Using BitTorrent**

Purvi Shah, *University of Houston*

Recently there has been an increasing interest on the use of Peer-To-Peer (P2P) architecture for providing large file-streaming service. By allowing peers serve each other, a P2P network overcomes many limitations in traditional client-server architecture. In this work we propose the use of BitTorrent, a P2P protocol for large file-streaming. To ensure the in-time delivery of individual chunks we modify the chunk-selection policy in BitTorrent by using the notion of a sliding window. We also present a more effective neighbor-selection policy for BitTorrent to improvise the streaming-quality of all peers in the system. Most importantly, in critical-regions our policy significantly improvise the streaming-quality and reduces the difference between the quality received by the peers.

## **Animated 3D for Teaching Programming**

Barrett Shiff, *Nova Southeastern University*

A teaching system is described for students to control Animated 3D Graphical game-like scenes with simple programming statements. A programming language IDE is connected via sockets to another process which displays the scenes. The control is via language independent messages. Students control fairly complex animated scenes with their first instruction. The basic idea is to provide students with a simplified programming interface that uses very simple statements along with standard program statements to control Animated 3D Graphical scenes. Students are motivated by the fun of seeing immediate action in the game-like scenes from simple programming statements.

## **On Regression Testing COTS-based Applications**

Jiang Zheng, *North Carolina State University*

Most existing regression test selection (RTS) processes rely on source code and therefore are not suitable when source code is not available for analysis. A process is proposed with supporting tools for commercial-off-the-shelf (COTS)-based applications to reduce the test suite required to evaluate changed COTS components when source code is not available. The results of industrial case studies indicate that the process is an effective RTS process for COTS-based applications. We plan to conduct extensive validation of both the tool support and process which requires more industrial case studies, data collection, and further RTS analysis.

# SIGCSE 2007 SRC Undergraduate Student Research Abstracts

## **Efficient Background Task Control Utilizing Kernel-Derived Information**

Yoshihisa Abe, *Keio University*

Resources in a computer system often remain idle. Our research aims at exploiting those idle resources, while at the same time avoiding delays in processing normal tasks in the system. Our system analyzes status information derived from the operating system kernel and stalls background tasks, which consume idle resources, when resource contention is inferred. Taking advantage of a recent trend towards the provision of the operating system's internal information through probes, the system controls background processes without requiring modification to existing operating systems or user applications. Also, our approach can be easily applied to different resources like CPUs and disks.

## **Performance Analysis of Optical Flow: High Definition Video vs. Regular Video**

Jennifer Allen, *Youngstown State University*

This paper studies performance evaluation of an optical flow algorithm using both high definition and regular videos. The hypothesis is that, because of its higher resolution, a high definition video can yield better flow data, which is critical for motion research. The experiment was designed to capture facial motion in a video. A series of frame pairs of different intervals were processed by an optical flow algorithm to generate dense flow images. Analysis of those images based on qualitative visual examination and quantitative violation percentages indicates that a high definition video is superior to a regular video.

## **Classification of E-mail into Folders using Naive Bayes**

John Baker, *Hiram College*

This project proposes the use of a Naive Bayesian Classifier to automatically sort e-mails into appropriate folders based upon the user's previous classifications. Being a conversational medium, the content of non-advertising e-mails is subject to constant evolution. This evolution combined with the relatively small sample size of one user's e-mail means the system may need to make its selection based upon a small number of criteria. To improve the performance in these cases, this project adds a measure of relevance to each of the criteria. Tests were computed with approximately 88% of new e-mails being successfully classified.

## **The Effects of Multiple Textures on Navigation in a Virtual Environment**

Prabin Bariya, *Hanover College*

Though Virtual Environments (VEs) are beneficial in many fields, users of VEs often experience navigational problems. Various solutions to this problem have met with modest success. This study examines the effects that multiple textures, acting as landmarks, have on easing user navigation in a VE. Results indicate that multiple unique textures prove helpful in easing the navigation through a VE, particularly on subsequent visits to the same VE. Moreover, the use of multiple textures enhanced female users' navigation significantly more than males.

## **Program Justification: Verifying Code Semantics with Mathematics**

Samuel J. Behrend, *Denison University*

The use of computer technology pervades modern society, capitalizing on the 20th century obsession with quantity, speed, and efficiency. But while current software has been remarkably successful, most can hardly be considered reliable, reusable, user-friendly or even secure. Hence, we propose that proving the correctness of programs and consequentially developing better software is contingent on three notions. First, the proof process must be an automatable activity in order to be practical. Also, in order to make verification possible, we need a new language that includes mathematical constructs that support specification, as well as a collection of automatable proof rules to process code written in the new language. The research presented in this paper is a step toward proof of concept for this revolutionary idea.

## **Optimization of a RISC-based Virtual Machine for Mobile Computations**

Elizabeth E. Carter, *The College of New Jersey*

The mobile computation system is an environment in which a program can begin execution on one machine and, if the application finds that environment insufficient, migrate to another machine and continue execution from where it left off on the original machine. The purpose of this research project is to apply optimization techniques to the existing system, concentrating on linked list operations that occur consistently throughout the system's execution.

## **DiskTrie: An External Memory Implementation of Trie**

N.M. Mosharaf Kabir Chowdhury, *Bangladesh University of Engineering and Technology*

DiskTrie is a static, external memory implementation of Trie, a very popular data structure for efficient storage and searching on large number of strings. It is developed specially for mobile devices, e.g. cell phones, which have a very low memory as well as processing power. DiskTrie tries to provide fast searching of Trie with low memory and CPU requirement. It also provides a tradeoff ratio that can be used to tune performance in different cases.

## **Author Attribution via Synonyms**

Jonathan H. Clark, *Texas Christian University*

An approach for identifying authors using synonym sets is presented. Having examined current research in psycholinguistics, we construct a computational theory relating authors to their choice of synonyms. We then demonstrate a quantitative procedure for determining authorship. The results of this procedure applied to a corpus of classic literature are shown. Our tests indicate a difference of 10% between positive and negative author matches.

## **A Neuro-Evolutionary Approach To Time Series Forecasting**

Sidhant Dash, *Indian Institute of Technology*

Financial Time Series Forecasting is an important problem that has received tremendous attention in recent years. We present a Neuro-Evolutionary approach to time series forecasting, wherein we evolve neural network models using an advanced evolutionary computation algorithm called Neuro-Evolution using Augmenting Topologies. Changes in the original algorithm are implemented and to reach the high level of precision desired by time series forecasting, a novel ensembling method is also explored. The results suggest that the system is robust, and when combined with the proposed ensembling technique, is able to outperform the Random Walk model in a complex currency exchange rate prediction task.

## **Making Generic Functions Useable in Smalltalk**

Timmy Douglas, *Georgia Institute of Technology*

This research project seeks to understand the potential of adding generic functions to a message-passing language, without modifying the internals of the system. The main focus of this project was to design an effective system for seamlessly calling and authoring generic functions in the Squeak Smalltalk system. In other words, the first goal was to make it easy to use generic functions, and the second goal was to make it easy to define them.

## **Shared Shortest Paths**

Christopher Earl, *Ohio Wesleyan University*

Shared paths on mathematical graphs are paths that collaborate on edges common to their paths to split the cost of traversing those edges. A journey is defined as a path whose cost depends upon collaborations with other journeys that share the cost of common edges. An efficient polynomial-time algorithm for two journeys was found and implemented. We demonstrate that an optimal solution is not possible for more than two journeys, and we developed heuristics for a variable number of journeys. These heuristics added journeys one at a time and had a savings of 42% over the non-shared path.

## **Comparative Metabolic Modeling of Prokaryotic Genomes**

Kevin Formsma, *Hope College*

This is an interdisciplinary research project to automate metabolic model creation for bacteria organisms. The research employs computer science abstraction concepts and graph algorithms to provide tools for metabolic model creation within a web-based environment. It utilizes comparative genomic concepts from biology and applies them to metabolic modeling for a generalized solution. Metabolic models are currently tedious to construct and are useful in biology research areas of drug targeting, genetic engineering, and experiment prediction.

## **Biomedical Information Extraction Through Deep Parsing and Syntactic Role Matching**

Anthony Gitter, *Arizona State University*

Because biomedical research is often contingent on protein-protein interaction information, an extraction system that can pull relevant interactions from the millions of biomedical journal abstracts and articles is extremely useful. Existing extraction systems often use hard-coded, rigid rules to detect such interactions. My system stores the logic used to extract interactions in external rules that can be understood and customized by anyone with basic knowledge of English grammar. A novel, straightforward query language is used to form these rules, which detect the syntactic roles of words or phrases in biomedical sentences that have been parsed into constituent trees.

## **Exploring Architectures for Distributed Visualization**

Justin Hagstrom, *SUNY Geneseo*

I researched whether it is possible to use a cluster computing system to keep framerates interactive when visualizing large data sets. This is in the context of IViPP, a program that interactively renders the results of a particle physics simulation. I implemented two distinct architectures for distributing the workload, sort-first and sort-last. Despite the overhead from communication between computers, the speedup results collected from timing the algorithms show that a distributed architecture for visualization programs can greatly decrease the time it takes for rendering. When measuring rendering time only, sort-last's speedups were almost perfectly linear.

## **Lightweight Economic Models for Resource Sharing in Wireless Networks**

Maria Kazandjieva, *Mount Holyoke College*

Wireless sensor networks face a number of challenges, including limited and expensive hardware and software resources. Setting up an economic market in which nodes pay for shared resources is one way of addressing the problem. In networks where more than one node is ready to provide a resource, we need an effective way of choosing the “best seller”. We present two decision schemes for making the choice of a resource provider. The first one is based solely on the price of a resource, while the second one adaptively assigns importance to the price and the quality of the service. We use software simulations to evaluate the two algorithms. On average, the price-only scheme increases the percent of shared resources by 6% and the adaptive decision scheme, by 10% over a baseline case. Therefore, in a sensor network where nodes are actively sharing resources, using a smart algorithm for choosing between multiple sellers increases the overall performance of the system.

## **A New Data Structure for Heapsort with Improved Number of Comparisons**

Md. Shahjalal, *North South University*

A new data structure for heapsort is proposed where every node contains 2 elements. A heap is constructed using larger elements of all nodes. We need on the average 2.75 moves and 1 comparison per element instead of 2 moves for traditional heaps before adjustment. But we save 2 comparisons in Floyd type and >1 comparison in Carlsson type algorithms during adjustment. Assuming cost of moving a node and individual element is same, this data structure allows sorting with lesser number of comparisons. Experimental results conform with the theoretical investigation and supports its superiority in all variants of heapsort algorithms.

## **Extraction of 3D Geometry and Texture from 2D Photographs**

Anthony Snyder, *Furman University*

A software architecture and a library of software implementing this architecture has been created that partially automates the process of reconstructing a 3D scene from a photograph. The architecture is tied together by a tree structure that serves to organize a scene graph hierarchy as well as an image hierarchy. A proprietary extraction concept allows for the creation of dual Green's function recognizers used for tracing the edges of an object in an image. The result is a set of points and scale values that are used along with a user-defined cross section to generate a VRML-like extrusion object.

## Author/Panelist Index

Acker, L. ....	293	Burge, M. ....	49	Dykman, N. ....	312
Adams, J. C. ....	307	Burley, D. ....	49	Eckerdal, A. ....	504
Adcock, B. ....	9	Calka, C. ....	170	Ecott, S. ....	213
Albert, E. ....	24	Cameron, W. ....	55	Eddy, E. ....	271
Allen, R. B. ....	243	Campbell, J. A. ....	180	Edgington, J. ....	115
Alphonse, C. ....	228	Canny, J. ....	238	Edwards, S. H. ....	150, 346
Anderson, N. ....	100	Caristi, J. ....	394	Eidelman, L. ....	29
Anderson, Richard ....	69	Carle, A. ....	238	Ellis, H. ....	67, 551
Anderson, Ruth ....	69	Carson, E. ....	155	Encarnação, M. ....	124
Anderson, S. D. ....	110	Caspersen, M. ....	228	Engels, S. ....	34
Applin, A. ....	327	Cassel, L. ....	55, 133, 329, 519	Ericson, B. ....	298, 377
Arnold, R. ....	105	Chase, J. D. ....	469	Fager, J. ....	39
Astrachan, O. ....	2	Chen, T.-Y. ....	276	Fan, W. ....	55
Bailey, M. ....	254	Chen, X. ....	566	Feinberg, D. ....	140
Bair, B. ....	247	Childress Townsend, G. ....	247, 331	Fenwick, Jr., J. B. ....	464
Baldwin, D. ....	65	Chinn, D. ....	203	Ferguson, J. ....	499
Barbosa, D. N. F. ....	379	Cigas, J. ....	497	Fincher, S. ....	514
Barbosa, J. ....	379	Clancy, M. ....	238, 264, 497	Fink, S. ....	84
Barowski, L. A. ....	4	Clifton, C. ....	86	Fitzpatrick, D. ....	259
Bay, T. ....	454	Cphoon, J. P. ....	395	Forbes, J. ....	245
Beck, J. ....	358, 400	Continanza, C. ....	175	Fox, E. A. ....	55
Bennett, C. ....	556	Cooper, M. ....	150	Francioni, J. ....	425
Bergin, J. ....	264	Cooper, S. ....	377, 394	Frost, D. ....	302
Biggers, M. ....	298	Cortina, T. J. ....	19, 195, 218, 377	Furst, M. ....	420
Birmingham, W. ....	293	Craig, M. ....	34, 190	Gabbert, P. ....	247
Blum, L. ....	19	Cross II, J. H. ....	4	Gal-Ezer, J. ....	1
Boff, E. ....	180	Cunningham, S. ....	254	Garcia, D. D. ....	2, 245
Booch, G. ....	197	Cutter, P. ....	353	Gayler, D. ....	129
Border, C. ....	576	Damon, J. ....	551	Gegg-Harrison, T. ....	425
Boucher Owens, B. ....	247	Dankel II, D. D. ....	96	Geist, R. ....	415
Boustedt, J. ....	346, 504	Dark, M. ....	119	Gestwicki, P. V. ....	233
Bower, M. ....	91	Davies, G. ....	133, 519	Geyer, C. F. R. ....	379
Boyer, K. E. ....	341	Davis, J. ....	372	Goldweber, M. ....	264
Brown, C. W. ....	445	Davis, K. M. ....	69	Gong, Y. ....	566
Brown, W. ....	208	Davis, T. ....	415	Goodfellow, M. ....	410
Bruce, J. ....	44	de Lanerolle, T. R. ....	551	Gorinsky, S. ....	481
Bruhn, R. ....	348	Decker, A. ....	228	Goulden, J. ....	496
Bucci, P. ....	9	Denning, T. ....	266, 384	Griswold, W. G. ....	266, 384
Buckner, B. ....	358	Du, W. ....	486	Gross, P. ....	474
Burd, B. ....	496	Dwight, R. S. ....	341	Guzdial, M. ....	298, 329, 420



Hamada, M. ....	60	Kurtz, B. L. ....	464	Morelli, R. A. ....	551
Hansen, D. M. ....	44	Ladd, B. ....	496	Morrison, B. B. ....	346
Hansen, S. ....	271	Lakshmanan, V. ....	34	Moström, J. E. ....	504
Hardisty, E. A. ....	445	Lancor, L. ....	491	Mrozek, M. ....	86
Harrison, D. ....	44	Langheinrich, M. ....	105	Mullins, J. ....	394
Hartmann, W. ....	105	Latour, L. ....	24	Murtagh, T. P. ....	336, 526
Harvey, V. J. ....	129	Lau, K.-K. ....	198	Musicant, D. ....	65, 346
Havill, J. T. ....	185	Lauer, T. ....	160	Nikolova, O. ....	358
Hayes, Z. ....	44	Layman, L. ....	459	Nita-Rotaru, C. ....	119
Hazzan, O. ....	29, 195	Leitner, H. H. ....	223	Nold, E. ....	160
Henderson, P. B. ....	195	Leutenegger, S. ....	115	Noonan, B. ....	497
Hendrix, T. D. ....	4	Levine, D. B. ....	264	Noonan, R. E. ....	135
Hervé, J.-Y. ....	124	Lewandowski, G. ....	276	Oakes, E. ....	469
Heym, W. D. ....	9	Lidtke, D. ....	281	Ohlrich, J. ....	96
Hirshfield, L. M. ....	213	Lim, D. ....	367	Okelberry, R. ....	288
Hoffman, M. E. ....	405	Lin, C.-C. ....	425	Olson, K. ....	84
Hollingsworth, J. E. ....	9	Lincke, S. J. ....	476	O'Neill, B. ....	170
Holz, H. ....	327	Linckels, S. ....	50	Oriol, M. ....	454
Hott, J. R. ....	135	Lindquist, D. ....	266, 384	Padilla, C. R. ....	170
Hürst, W. ....	160	Linnell, N. ....	69	Parberry, I. ....	155
Hutt, R. ....	124	Liu, C. ....	474	Parlante, N. ....	2, 497
Impagliazzo, J. ....	131	Liu, H. ....	566	Peckham, J. ....	124
Isbell, C. ....	420	Long, T. ....	9	Pedroni, A. ....	454
Jain, J. ....	4	Ludwig, L. D. ....	185	Pedroni, M. ....	454
Janick-Buckner, D. ....	358	Lundak, E. ....	425	Pérez-Quñones, M. A. ....	55, 129
Jennings, S. F. ....	348	Ma, L. ....	499	Pheatt, C. ....	571
Jensen, B. ....	155	Madden, B. ....	377	Pike, S. M. ....	362
Jia, X. ....	464	Madigan, E. M. ....	410	Popescu, V. ....	119
Jipping, M. J. ....	170	Maier, F. ....	556	Pottenger, W. M. ....	474
Joel, W. ....	327	Malan, D. J. ....	223, 389	Powers, K. ....	213
Johnson, O. ....	74	Malani, R. ....	266, 384	Prince, C. ....	69
Kaczmarezyk, L. C. ....	86	Marion, B. ....	131	Pulimood, S. M. ....	322
Kamin, S. N. ....	531	Martin, K. ....	203	Purewal, Jr., T. S. ....	556
Karam, N. ....	50	Matzko, S. ....	415	Radenski, A. ....	14
Kelemen, C. ....	67	McCartney, R. ....	276, 346, 504	Rague, B. ....	100
Kelly, M. ....	266, 384	McGettrick, A. ....	133, 329, 519	Ramsey, S. ....	469
Kessler, R. ....	312	McGovern, A. ....	39	Ratcliffe, M. ....	504
Klappholz, D. ....	129, 243	McGrath Cohoon, J. ....	247, 546	Raubenheimer, C. D. ....	341
Klassner, F. ....	175	McMaster, K. ....	100	Raye, J. ....	551
Koffman, E. ....	67	Meinel, C. ....	50	Razmov, V. ....	69
Kruse, G. ....	84	Menzel, S. ....	331	Reategui, E. ....	180
Kumar, A. ....	65	Miller, C. S. ....	341	Rebelsky, S. A. ....	372

Reed, D. ....	497	Sloan, R. H. ....	519	Verno, A. ....	377
Reed Rosenberg, B. ....	249	Smith, S. ....	145	Vilalta, R. ....	74
Reges, S. ....	2	Sooriamurthi, R. ....	497	Vouk, M. A. ....	341
Reichgelt, H. ....	281	Soroka, B. ....	131	Walker, E. ....	65
Reid, K. L. ....	317	Spencer, C. ....	203	Wang, R. ....	486
Repp, S. ....	50	St. Clair, C. ....	131	Warford, J. S. ....	288
Rheingans, P. ....	430	Stallmann, M. F. ....	341	Weide, B. W. ....	9
Robbins, S. ....	536	Steele, A. ....	464	Westall, J. ....	415
Roberts, E. ....	329, 521	Stephenson, C. ....	247	White, C. ....	67
Rogers, M. ....	496	Stephenson, P. ....	124	Whitfield, D. ....	131
Roper, M. ....	499	Stevenson, D. E. ....	165	Wick, M. R. ....	243, 509
Rosmaita, B. J. ....	541	Stewart, K. ....	496	Williams, L. ....	459
Ruckert, M. ....	435	Stoecklin, S. ....	145	Wilson, G. V. ....	317
Sanders, K. ....	276, 504	Stone, J. A. ....	410	Wing, J. M. ....	195
Schaeffer, J. ....	581	Subhlok, J. ....	74	Winters, T. ....	346
Schwarzmeier, J. D. ....	165	Subramaniam, V. ....	74	Wolf, T. ....	79, 481
Schweitzer, D. ....	208	Sung, K. ....	249	Wolfman, S. ....	67
Sciore, E. ....	561	Talton, J. O. ....	259	Wolz, U. ....	322
Serino, C. ....	145	Taylor, H. ....	49	Wong, K. ....	481
Shaaban, M. ....	283	Tenenberg, J. ....	346	Wood, M. ....	499
Shaffer, C. A. ....	150	Tenenberg, J. ....	514	Woodley, M. ....	531
Sherrell, L. B. ....	474	Teng, GF. ....	55	Workman, R. ....	491
Shiflet, A. B. ....	497	Teng, Z. ....	486	Wortman, D. ....	430
Shirley, P. ....	249	Topi, H. ....	133, 519	Xu, L. ....	440
Siek, K. A. ....	331	Tungare, M. ....	55	Yaverbaum, G. ....	281
Sigman, S. ....	450	Turner, E. H. ....	24	Yu, X. ....	55
Simon, B. ....	266, 276, 384	Turner, J. ....	481	Yuan, X. ....	464
Sivilotti, P. A. G. ....	362	Turner, R. M. ....	24	Yun, C. ....	74
Slaten, K. ....	459	Utting, I. ....	346	Zander, C. ....	243, 346, 504
Slater, D. ....	264	VanBuren, B. G. ....	283	Zhang, M. ....	425
Sloan, B. ....	133	Vance, D. R. ....	405	Zweben, S. ....	281