

## TECHNICAL PROGRAM

# TECHNICAL PROGRAM

**Program Chair:**

John Vlissides, *IBM T.J. Watson Research Center*

**Practitioner Reports Chair:**

Jim Doble, *Tavve Software Company*

**Panels Chair:**

Ron Crocker, *Motorola, Inc.*

OOPSLA 2001 continues the tradition of quality and diversity in technical offerings. Here is a brief guide to the sessions and events comprising the technical program:

- Invited talks provide a forum for world-class speakers to present expert yet accessible accounts of important and timely topics.
- Technical paper presentations discuss new research in object-oriented languages, systems, and applications. The 27 technical papers were selected after a rigorous peer review of 145 submissions. Some of these papers present work that foreshadows the future of OO systems; others describe and analyze the underpinnings of common OO languages, design methods, and applications.
- Practitioner Reports relate how OO languages, systems, middleware, methods, and tools are used in practice. Some of these reports describe what it's like to adopt a new language, apply a new methodology, or use a new tool. Others distill the experience of one or more projects. The reports describe both benefits and drawbacks of approaches used in practice.
- Poster sessions present technical contributions in an informal setting. These sessions contain some of the newest research, often still in-progress. Authors are present to discuss their work, usually one-on-one.
- Demonstration sessions offer a chance to see software in action in a similarly informal (and commercial-free) context. In many cases, demos show the software discussed in technical sessions, tutorials, and other presentations.
- Panel sessions are the place where experts are encouraged to disagree publicly. The six panel sessions this year cover a spectrum of issues in software development, from understanding the impact of research to the future of object computing.
- DesignFest and CodeFest are our most hands-on events. Participants work in teams to design and implement solutions to assigned problems. The participants present their results and lessons learned at a wrap-up session at the end of the conference.
- This year we introduce two sessions in a new crossover track dubbed "Intriguing Technology." Designed to appeal to both researchers and practitioners, Intriguing Technology presentations describe best practices, introduce systems that make research ideas more widely accessible, discuss experiences that might spur further research—in short, any work that's "intriguing" to the OOPSLA audience.

In addition to these, as well as Tutorials, Workshops, Exhibits, and social events, there are two embedded symposia at OOPSLA:

- The Doctoral Symposium provides a forum for graduate students to share ideas and receive guidance.
- The Educators' Symposium includes presentations and discussions on teaching object-oriented technology in both academics and industry.

There's something here for everyone who works with object-oriented software. Whether you're a programmer or technical lead, student or faculty member, researcher or practitioner—you stand to benefit greatly from the state-of-the-art technologies and practices on display at OOPSLA 2001.

## TECHNICAL PROGRAM AT A GLANCE

Tuesday, October 16, 2001

8:30 am - 10:00 am	<b>Welcome and Introduction</b> Conference Chair: Linda Northrop Program Chair: John Vlissides  <b>Keynote Address:</b> Henry Petroski, "Success and Failure in Design"	Convention Center Ballroom B-C
10:30 am - 12:00 pm	<b>Session A</b> <b>Panel:</b> Somewhere Between Anarchy and Monarchy: Choosing the Appropriate Process for Your Next Project  <b>Session B</b> <b>Papers:</b> Languages I  <b>Session C</b> <b>Practitioner Reports:</b> Distributed Systems	Convention Center Ballroom B-C  Convention Center Ballroom A  Convention Center Ballroom D
1:30 pm - 3:00 pm	<b>Session A</b> <b>Invited Speaker:</b> Tom DeMarco, "Software: The New Realities"  <b>Session B</b> <b>Papers:</b> Program Analysis  <b>Session C</b> <b>Papers:</b> Performance	Convention Center Ballroom B-C  Convention Center Ballroom A  Convention Center Ballroom D
3:30 pm - 5:00 pm	<b>Session A</b> <b>Panel:</b> How Do Requirements Relate to Objects?  <b>Session B</b> <b>Papers:</b> Resource Management  <b>Session C</b> <b>Practitioner Reports:</b> Software Reuse	Convention Center Ballroom B-C  Convention Center Ballroom A  Convention Center Ballroom D

## Wednesday, October 17, 2001

8:30 am - 10:00 am	<b>Session A</b> <b>Invited Talk:</b> Don Box, "Pervasive XML: Infoset-Based Software Integration"	Convention Center Ballroom B-C
10:30 am - 12:00 pm	<b>Session A</b> <b>Panel:</b> The IMPACT Project: Determining the Impact of Software Engineering Research upon Practice  <b>Session B</b> <b>Papers:</b> Dynamic Optimization  <b>Session C</b> <b>Intriguing Technology Papers:</b> Expressive and Adaptive Systems	Convention Center Ballroom B-C  Convention Center Ballroom A  Convention Center Ballroom D
1:30 pm - 3:00 pm	<b>Session A</b> <b>Invited Speaker:</b> L. Peter Deutsch, "Software Transparency and Object Technologies"  <b>Session B</b> <b>Papers:</b> Modules  <b>Session C</b> <b>Practitioner Reports:</b> Applied Techniques	Convention Center Ballroom B-C  Convention Center Ballroom A  Convention Center Ballroom D
3:30 pm - 5:00 pm	<b>Session A</b> <b>Panel:</b> The OO Survivor Game  <b>Session B</b> <b>Papers:</b> Languages II  <b>Session C</b> <b>Intriguing Technology Papers:</b> TACO and SALSA	Convention Center Ballroom B-C  Convention Center Ballroom A  Convention Center Ballroom D

## Thursday October 18, 2001

<b>8:30 am - 10:00 am</b>	<b>Session A</b> <b>Invited Talk:</b> Alistair Cockburn, "Harnessing Convection Currents of Information"	Convention Center Ballroom B-C
<b>10:30 am - 12:00 pm</b>	<b>Session A</b> <b>Panel:</b> The Future of Distributed Objects — .NET vs Java  <b>Session B</b> <b>Papers:</b> Software Engineering  <b>Session C</b> <b>Balloon Debate:</b> Educators' Symposium Follow-Up: Empowering People: What Are the Most Important Criteria for Effectively Training People in OT?	Convention Center Ballroom B-C  Convention Center Ballroom A  Convention Center Ballroom D
<b>1:30 pm - 3:00 pm</b>	<b>Session A</b> <b>Panel:</b> Is "Software Engineering" the Wrong Metaphor? And Why Should We Care?  <b>Session B</b> <b>Papers:</b> Garbage Collection	Convention Center Ballroom B-C  Convention Center Ballroom A
<b>3:30 pm - 5:00 pm</b>	<b>OOPSLA 2002 Kick-off</b> (Ice Cream Social)  <b>DesignFest/CodeFest Wrap-Up</b>	Convention Center Ballroom D

8:30 am - 10:00 am

## Convention Center — Ballroom B-C

### Welcome and Introduction

Conference Chair:  
Linda Northrop,  
*Software Engineering Institute*

Program Chair:  
John Vlissides,  
*IBM T.J. Watson Research Center*

### Keynote Address

#### Success and Failure in Design



**Henry Petroski,**  
**A.S. Vesic Professor of Civil Engineering and**  
**Professor of History, Duke University**

This illustrated lecture will present a historical perspective on the interrelationship between success and failure in design. Case studies will range from ancient Greek and Roman construction practices to modern American packaging and merchandizing, emphasizing the timelessness and cultural independence of design principles as applied to objects of all kinds.

*Henry Petroski is the Aleksandar S. Vesic Professor of Civil Engineering and a professor of history at Duke University. Before moving to Duke in 1980, he was on the faculty of the University of Texas at Austin and on the staff of Argonne National Laboratory.*

*Petroski has written on many aspects of engineering and technology, including design, success and failure, error and judgment, the history of bridges, and the use of case studies in education and practice. His books on these subjects, which are intended for professional engineers and laypersons alike, include *To Engineer Is Human*, which was adapted for a BBC-television documentary; *The Pencil*; *The Evolution of Useful Things*; *Design Paradigms*; *Engineers of Dreams*; *Invention by Design*; and *Remaking the World*. Among the languages into which his books have been translated are Chinese, Finnish, German, Hebrew, Korean, Japanese, and Spanish. His latest book, a history of books as artifacts and the structures that have housed them from ancient times to the computer age, is entitled *The Book on the Bookshelf*.*

*In addition to publishing the usual technical articles in the refereed journals of his field, Petroski has published numerous articles and essays in newspapers and magazines, including the *New York Times*, *Washington Post*, *Wall Street Journal*, and *Scientific American*. Since 1991 he has been writing the engineering column in the bimonthly magazine *American Scientist*, and since 2000 has been writing a column on the engineering profession for *ASEE Prism*.*

*Henry Petroski has been a Guggenheim Fellow, a National Endowment for the Humanities Fellow, and a Fellow of the National Humanities Center. Among his other honors are the Ralph Coats Roe Medal from the American Society of Mechanical Engineers, the Civil Engineering History and Heritage Award from the American Society of Civil Engineers, honorary degrees from Clarkson University, Trinity College (Hartford, Conn.) and Valparaiso University, and distinguished engineering alumnus awards from Manhattan College and the University of Illinois at Urbana-Champaign. He is a Fellow of the American Society of Civil Engineers and the Institution of Engineers of Ireland and is a member of the National Academy of Engineering.*

10:30 am - 12:00 noon

## Convention Center — Ballroom B-C

### Session A

#### **Panel: Somewhere Between Anarchy and Monarchy: Choosing the Appropriate Process for Your Next Project**

Moderator: John Daniels, *Syntropy, Ltd.*

The rise of the agile process movement has received a good deal of attention lately, but many questions remain regarding when such process models are appropriate. Determining which process model is the right one for your project is a critical factor in success. In this panel, the topic of when and why a given process model should be applied will be discussed. To maximize the value of the discussion, panelists will be asked to respond to specific questions based on real-world issues.

Panelists:

Ron Jeffries, *Object Mentor, Inc.*

Craig Larman, *Valtech*

Ivar Jacobson, *Rational Software Corp.*

Alistair Cockburn, *Humans and Technology*

Jim Highsmith, *Information Architects, Inc.*

Bob Marcus, *Rogue Wave*

## Convention Center — Ballroom A

### Session B

#### **Papers: Languages I**

Chair: Joseph Kiniry, *California Institute of Technology*

The papers in this session are excellent examples of theory successfully applied to OO problems. Findler and Felleisen address contract soundness with respect to annotated Java in its various incarnations. Their results will be relevant to anyone applying formal specification to Java using tools like iContract, JML, and Jass. Ancona, et al., solidify the type theoretic and algebraic foundations of Java's semantics, a highly pertinent area given the popularity of that language. Finally, Bachrach shows how to do macros "right" in Java—à la Dylan and Lisp, with several new twists.

#### **Contract Soundness for Object-Oriented Languages**

Robert Bruce Findler, *Rice University*

Matthias Felleisen, *Rice University and Northeastern University*

#### **A Core Calculus for Java Exceptions**

Davide Ancona, *DISI – Università di Genova*

Giovanni Lagorio, *DISI – Università di Genova*

Elena Zucca, *DISI – Università di Genova*

#### **The Java Syntactic Extender**

Jonathan Bachrach, *MIT AI Lab*

Keith Playford, *Functional Objects, Inc.*

## Convention Center — Ballroom D Session C

### **Practitioner Reports: Distributed Systems**

Chair: Gerard Meszaros, *ClearStream Consulting*

Distributed architectures are both a blessing and a curse: With careful planning and design, they can help with reliability and scalability, but “getting it right” is no small challenge. In this session, two reports describe how distributed systems were successfully developed to provide scalable business applications, and the third author presents an approach for certifying performance of a distributed system.

### **SPCQuote: The Evolution of a Client Application to Distributed Service Processing**

Karen Hope, *The St. Paul, Inc.*

John Finegan, *The St. Paul, Inc.*

The Business Foundation System (BFS) writes and manages small business commercial insurance policies for The St. Paul Fire & Marine insurance companies. The original system, deployed in 1996, was a client-based VSE Smalltalk application. By the time of its third release, it supported 120 internal users in three locations. Since its deployment the system has had over 30 subsequent releases to increase and improve its functionality, the most notable of these occurring with BFS’s SPCQuote release in April 2000. SPCQuote significantly changed both the system architecture and its development environment and satisfied many disparate key business objectives. Today the system supports over 7500 users in over 3200 locations. It manages over 120,000 policies with policy premiums totaling over \$500,000,000. This paper describes the evolution of the smart-client VSE Smalltalk application to an n-tier distributed service model supported by two Smalltalk dialects, Java servlets running within Websphere, and XML as a means of data abstraction.

### **Business Automation with Distributed Objects**

Jason Rogers, *Ontario Teachers’ Pension Plan Board*

Dean Mackie, *Ontario Teachers’ Pension Plan Board*

Angus MacArthur, *Ontario Teachers’ Pension Plan Board*

A large financial institution was faced with the challenge of having to process twice the normal yearly workload without increasing their workforce. The challenge was met by reusing components of an existing object-oriented application in a distributed, fully automated configuration. That solution is discussed, along with subsequent system architecture evolution, resulting improvements to development practices, and the ripple effects of changing the culture of senior management that it was designed to serve.

### **Certifying Component Performance in Synchronous Distributed Client/Server Systems**

Dong-Lih Denq, *AT&T-Network Services*

Irwin Dunietz, *AT&T-Network Services*

John Eddy, *AT&T-Network Services*

Willa Ehrlich, *AT&T-Network Services*

Don Gerth, *AT&T-Network Services*

Brian Larson, *AT&T-Network Services*

Geetha Sivaprasad, *AT&T-Network Services*

This paper describes an approach for certifying performance of a distributed system that utilizes synchronous interprocess communication. The system is composed of multiple managed objects implemented in object server processes (i.e., components) that communicate via CORBA middleware. Component performance is certified empirically through “pairwise” testing that emphasizes interactions between a component and its lower-level servers. The applicability of this approach is demonstrated for an AT&T provisioning system.

1:30 pm - 3:00 pm

**Convention Center — Ballroom B-C  
Session A****Invited Talk: Software: The New Realities****Tom DeMarco,  
Atlantic Systems Guild**

We live in an age of acceleration. Whatever the formula was for business success a few years ago, it won't work today. Today there needs to be more and more work crammed into less and less time. There are fewer people doing more and doing it faster in less space with less support and with tighter tolerances and higher quality requirements than ever before. So we have spent the last decade becoming ever more efficient. Now enter the need for change. In the super-accelerated corporation, meaningful change of direction is almost impossible. The very improvements that we made to go faster and cheaper have undermined our capacity to make any other kind of change. An organization that can accelerate but not change direction is like a car that can speed up but not steer. In the short run it makes lots of progress in whatever direction it happened to be going. In the long run it's just another road wreck.

*Tom DeMarco is a Principal of the Atlantic Systems Guild, a New York and London-based consulting practice. He is the author of seven books on software method and management, including the recent Random House book Slack: Getting Past Burnout, Busywork, and the Myth of Total Efficiency. Past titles include The Deadline: A Novel About Project Management, as well as PEOPLEWARE: Productive Projects and Teams (with co-author T. Lister). In 1986 he was named winner of the J.D. Warnier Prize for "lifetime contribution to the information sciences." He won the 1999 Stevens Prize for contribution to software methods. He is the author of a mainstream novel, Dark Harbor House (Down East Books, 2000). His consulting activity is mostly in the area of project management and litigation involving software-intensive endeavors.*



## Convention Center — Ballroom A Session B

### Papers: Program Analysis

Chair: Peter F. Sweeney, *IBM T.J. Watson Research Center*

This year's program upholds the OOPSLA tradition of high-quality contributions to program analysis. The first paper in this session uses a constraint-based approach to implement points-to analysis for Java. For a restricted Java-like language, the second paper guarantees that well-typed programs are race-free, even as it allows objects from the same class to use different locking mechanisms. The last paper describes a combination of static and run-time techniques for detecting potential data races in concurrent Java programs.

### Points-To Analysis for Java using Annotated Constraints

Atanas Rountev, *Rutgers University*

Ana Milanova, *Rutgers University*

Barbara G. Ryder, *Rutgers University*

### A Parameterized Type System for Race-Free Java Programs

Chandrasekhar Boyapati, *MIT*

Martin Rinard, *MIT*

### Object Race Detection

Christoph von Praun, *ETH Zürich*

Thomas Gross, *ETH Zürich*

## Convention Center — Ballroom D Session C

### Papers: Performance

Chair: Brian Foote, *The Refactory, Inc.*

Programmers often shun programming language features they see as exhibiting poor or unpredictable performance. This session highlights three advanced implementation techniques that promise to relieve programmers of such concerns. The first paper shows how exceptions in Java can be implemented with unexceptional overhead. The second paper examines how to minimize the cost of asking a deceptively simple question: Is this type a subtype of another? Sophisticated implementation strategies described in the third paper nearly eliminate the customary performance penalty of Java interface invocations.

### A Study of Exception Handling and Its Dynamic Optimization in Java

Takeshi Ogasawara, *IBM Tokyo Research Laboratory*

Hideaki Komatsu, *IBM Tokyo Research Laboratory*

Toshio Nakatani, *IBM Tokyo Research Laboratory*

### Efficient Subtyping Tests with PQ-Encoding

Yoav Zibin, *Technion*

Joseph (Yossi) Gil, *Technion*

### Efficient Implementation of Java Interfaces: Invokeinterface Considered Harmless

Bowen Alpern, *IBM T.J. Watson Research Center*

Anthony Cocchi, *IBM T.J. Watson Research Center*

Stephen Fink, *IBM T.J. Watson Research Center*

David Grove, *IBM T.J. Watson Research Center*

Derek Lieber, *IBM T.J. Watson Research Center*

3:30 pm - 5:00 pm

## Convention Center — Ballroom B-C Session A

### Panel: How Do Requirements Relate to Objects?

Moderator: Hermann Kaindl, *Siemens AG Austria*

While most object-oriented methods today include something like object-oriented analysis (OOA), they do not say much about requirements. Typically, they treat requirements just as a textual input in the form of a short problem statement. In the field of requirements engineering, in contrast, object-oriented modeling is often considered as a means to create and represent a requirements model. Apparently, the relationship between requirements and objects remains an open question.

In the light of such controversial views, this panel intends to discuss the following questions (among others):

- Are requirements just preliminary textual statements?
- Can requirements be modeled as objects?
- Can an OOA model serve as a requirements model?

Panelists:

Ivar Jacobson, *Rational Software Corp.*

Craig Larman, *Valtech*

Joaquin Miller, *Financial Systems Architects*

Rebecca Wirfs-Brock, *Wirfs-Brock Associates*

Desmond D'Souza, *Kinetium*

Steve Mellor, *Project Technology, Inc.*

## Convention Center — Ballroom A Session B

### Papers: Resource Management

Chair: Doug Lea, *SUNY Oswego*

When applications share resources, infrastructure services and virtual machines must help prevent interference. They must also manage resource consumption and optimize resource access among the competing activities. The authors of the first paper illustrate how JVMs can isolate applications to appear as if they were separate operating system processes. The second paper explains how to limit resource consumption in Java applications. The session's closing paper describes how to maximize the efficiency of database resources by optimizing queries.

### Multitasking without Compromise: A Virtual Machine Evolution

Grzegorz Czajkowski, *Sun Microsystems Laboratories*

Laurent Daynès, *Sun Microsystems Laboratories*

### Portable Resource Control in Java: The J-SEAL2 Approach

Walter Binder, *CoCo Software Engineering*

Jarle Hulaas, *University of Geneva*

Alex Villazón, *University of Geneva*

### Incremental Computation of Complex Objects Queries

Hiroaki Nakamura, *IBM Tokyo Research Laboratory*

## Convention Center — Ballroom D Session C

### **Practitioner Reports: Software Reuse**

Chair: Jim Doble, *Tavve Software Company*

Many organizations have looked to object technology as a means to achieve significant gains in software reuse. Results, however, have run the gamut from stunning success to unmitigated disaster. The first paper in this session presents a survey of results from a variety of reuse-oriented projects within Mentor Graphics Corporation over the past 12 years. The last two papers describe experiences and lessons learned on individual projects involving the development of reusable frameworks.

### **Mentor Graphics Corporation's Experiences with Object-Oriented Software Technology Reuse**

Phil Brooks, *Mentor Graphics Corporation*

Steve Aho, *Mentor Graphics Corporation*

Since 1989, Mentor Graphics Corporation has undertaken a number of large-scale projects where software reuse, in various forms, was a major goal. A team of seven senior software development engineers recently examined software reuse practices within Mentor Graphics in order to figure out what works, what doesn't, and why. Several actual projects, some that were successful and others that were not, are reviewed. The findings from these project experiences, along with the combined experiences of the members of the team, lead to conclusions on how organizational structure combined with the various reuse strategies presented aid or hinder reuse.

### **Construction of an Object-Oriented Framework for Text Mining**

James W. Cooper, *IBM T. J. Watson Research Center*

Edward C. So, *IBM T. J. Watson Research Center*

Christian L. Cesar, *IBM T. J. Watson Research Center*

Robert L. Mack, *IBM T. J. Watson Research Center*

This paper describes the construction of a Java class library for text mining and information retrieval. The system consists of Facades around a database, a search engine, and a text mining tool. The authors discuss the design of the object models used for each of these elements, how these object models evolved as different databases and search engines became available, how they needed to evolve the system further in work with their customer, and the eventual fate of the system after the customer adopted the final version of the code, showing what was learned from the experience.

### **Applying Design Patterns and Other Technologies to On-Line Application Servers for Bill Payment, Bill Delivery, and Banking**

Don Barton, *CheckFree Corporation*

This paper discusses how an organization gained high levels of reuse and developer concurrency, as well as scalability and availability by leveraging techniques from OOPSLA and other sources. The author discusses how a team and a process were built around the use of design patterns, frameworks, metaobject protocols, refactoring, code generation, and a custom build environment with integrated testing.

8:30 am - 10:00 am

**Convention Center — Ballroom B-C  
Session A****Invited Talk: Pervasive XML: Infoset-Based Software Integration****Don Box, *DevelopMentor***

The XML Infoset is XML's underlying data model and is increasingly being used to put XML into scenarios that would otherwise be considered too performance-sensitive to support a text-oriented solution. This talk looks at the role of the classic XML Infoset as well as the more modern Post-Schema Validation Infoset (PSVI) as a way of exposing applications, components, and data in general to consumers written in arbitrary languages running on arbitrary platforms.

*Don Box is a co-founder of DevelopMentor, a developer services company that provides education and support to the software industry at large. Don's research interests include component software integration, programming for concurrency, and XML-based serialization and metadata protocols. Don is a series editor at Addison-Wesley and is the author of Essential COM and a co-author of Effective COM and Essential XML, all from Addison-Wesley. Don is a contributing editor and columnist at Microsoft Systems Journal (now called MSDN Magazine) and an occasional contributor to XML.com. Don is also a co-author of the Simple Object Access Protocol specification and a member of the W3C Schemas Working Group. Don has a Master's Degree in Computer Science from the University of California at Irvine.*

10:30 am - 12:00 noon

## Convention Center — Ballroom B-C

### Session A

#### **Panel: The IMPACT Project: Determining the Impact of Software Engineering Research upon Practice**

Moderator: Dr. Leon J. Osterweil  
*University of Massachusetts*

This panel looks both backward and forward at the effect of software engineering research on software development practice. It will evaluate the impact of software engineering to date, identifying the sorts of contributions that have had substantial impact and the research modalities that have been more successful, and then explore directions that software engineering research might most effectively pursue to meet the challenging demands of the future.

Scheduled panelists:

Dave Thomas, *Bedarra Corp. and Carleton University*

Richard Gabriel, *Sun Microsystems, Inc.*

Alexander L. Wolf, *University of Colorado, Boulder*

Barbara G. Ryder, *Rutgers University*

## Convention Center — Ballroom A

### Session B

#### **Papers: Dynamic Optimization**

Chair: Craig Chambers, *University of Washington*

Dynamic compilation, as found in Java “just-in-time” compilers, affords many new optimization opportunities compared to traditional static compilation. “Partial Method Compilation using Dynamic Profile Information” uses on-line profiling of basic-block execution frequencies to re-optimize only the hot paths of methods dynamically. “A Dynamic Optimization Framework for a Java Just-In-Time Compiler” presents a thorough study of a state-of-the-art Java system and argues for including an interpreter as a component of a Java virtual machine in contrast to compile-only JVMs. “Dynamic Optimistic Interprocedural Analysis: A Framework and an Application” uses a flexible dependency framework to incorporate a simple whole-program analysis safely in the dynamic compiler—even in the face of dynamic class loading.

#### **Partial Method Compilation using Dynamic Profile Information**

John Whaley, *Stanford University*

#### **A Dynamic Optimization Framework for a Java Just-In-Time Compiler**

Toshio Suganuma, *IBM Tokyo Research Laboratory*

Toshiaki Yasue, *IBM Tokyo Research Laboratory*

Motohiro Kawahito, *IBM Tokyo Research Laboratory*

Hideaki Komatsu, *IBM Tokyo Research Laboratory*

Toshio Nakatani, *IBM Tokyo Research Laboratory*

#### **Dynamic Optimistic Interprocedural Analysis: A Framework and an Application**

Igor Pechtchanski, *New York University*

Vivek Sarkar, *IBM T.J. Watson Research Center*

## Convention Center — Ballroom D Session C

### **Intriguing Technology Papers: Expressive and Adaptive Systems**

Chair: Brent Hailpern, *IBM T.J. Watson Research Center*

In expressive systems, core business objects show through directly to users, and all user actions are initiated through a noun-verb style of interaction on those objects. Having users and developers speak a common language improves the process of requirements analysis and prototype development.

Adaptive object models form an approach to object-oriented information systems emphasizing flexibility and dynamic configurability. Business rules are stored in a database instead of in code. The object model that the user cares about is part of the database, and the object model of the code is just an interpreter of the users' object model.

### **Expressive Systems: A Radical Approach to Business Systems Design**

Richard Pawson, *Computer Sciences Corporation*

Simon Dobson, *Trinity College, Dublin*

### **Architecture and Design of Adaptive Object-Models**

Joseph Yoder, *University of Illinois at Urbana-Champaign*

Federico Balaguer, *University of Illinois at Urbana-Champaign*

Ralph Johnson, *University of Illinois at Urbana-Champaign*

1:30 pm - 3:00 pm

## Convention Center — Ballroom B-C

### Session A

#### Invited Talk: Software Transparency and Object Technologies



#### L. Peter Deutsch, *Aladdin Enterprises*

We define software as transparent to the extent that it is *conceived, developed, documented, licensed, distributed, and cared for* to intentionally facilitate *reading, understanding, analysis, validation, confidence; repair, adaptation, extension, evolution; interoperation, integration, incorporation; sharing; and use*. Software transparency, in all its aspects, is a key to the usefulness of software over its lifetime.

The *Open Source Definition* codifies an important set of rules that address the licensing (legal) aspect of transparency; the other aspects are either considered business issues, or issues of technological “best practice.” Transparency is rarely considered as a gestalt.

When objects are used well as a software engineering technology for analysis and design within a program, the technology naturally encourages many aspects of transparency. On the other hand, objects can also be used to create opaque boundaries around functionality and data. We will do our best to refute the arguments of security, long-term evolvability, and business advantage often used to justify such boundaries.

*Dr. L. Peter Deutsch received the Ph.D. in Computer Science from U.C. Berkeley in 1973. Subsequently at Xerox PARC, he helped develop the Interlisp-D, Cedar Mesa, and Smalltalk-80 programming systems, and was the principal creator of PS, the first high-performance implementation of the Smalltalk language and programming environment on microprocessor-based hardware. From 1986 to 1991, as Chief Scientist at ParcPlace Systems, he was a principal designer of a highly-portable successor to PS. Since 1986, he has been an independent consultant (dba Aladdin Enterprises), best known as the creator of Ghostscript, a highly portable, high-quality, Open Source implementation of the PostScript language. In 1993, he was a co-recipient of the ACM Software System Award, and was also named a Distinguished Alumnus of the U.C. Berkeley Computer Science program. In 1999-2000, he served on the board of the Open Source Initiative. He is a member of ACM, IEEE, CPSR, EFF, and the League for Programming Freedom.*

## Convention Center — Ballroom A Session B

### **Papers: Modules**

Chair: Krzysztof Czarnecki, *DaimlerChrysler Research and Technology*

Objects are the programming model of choice for implementing components. Yet there's always room for improvement in modularity and encapsulation, especially for component-based development. The session's opening paper describes a rich model and attendant language for building components that goes beyond the simple class-as-a-component paradigm while allowing easy integration into an object-oriented language such as Java. Packaging classes and class fragments into mixin modules is the thrust of the second paper, which presents a type-safe approach to programming using such modules. The last paper relates a static analysis technique for improving encapsulation by detecting objects that escape from packages.

### **Jiazzi: New-Age Components for Old-Fashioned Java**

Sean McDirmid, *University of Utah*

Matthew Flatt, *University of Utah*

Wilson C. Hsieh, *University of Utah*

### **Modular Mixin-Based Inheritance for Application Frameworks**

Dominic Duggan, *Stevens Institute of Technology*

Ching-Ching Techaubol, *Stevens Institute of Technology*

### **Encapsulating Objects with Confined Types**

Christian Grothoff, *Purdue University*

Jens Palsberg, *Purdue University*

Jan Vitek, *Purdue University*



## Convention Center — Ballroom D Session C

### **Practitioner Reports: Applied Techniques**

Chair: Granville Miller, *TogetherSoft*

A lot of ideas and approaches get discussed at an OOPSLA conference, but do they really work in the real world? The three reports in this session share insights and advice gained from actual application of a variety of OOPSLA topics: UML-based analysis and design, test-first design, and software frameworks.

### **Closing the Software Development Gaps with UML**

Brian Berenbach, *Siemens Corporate Research, Inc.*

This paper presents an approach that has been used successfully on large projects, leveraging UML to successfully bridge the analysis-design and design-implementation gaps in the Software Development Lifecycle (SDLC). In addition, a set of guidelines for determining the “completeness” of the analysis model is introduced.

### **Evolution of Test and Code via Test-First Design**

Jeff Langr, *Object Mentor, Inc.*

This paper demonstrates how test coverage and code quality is improved through the use of test-first design, starting with an example of code written without the use of automated tests, then iterating through the exercise of completely rebuilding the code, test by test, and comparing the results of the two approaches.

### **Object-Based Frameworks For Mobile Robots**

Jeffrey Will, *Dept. of Electrical and Computer Engineering, Valparaiso University*

Donald Roberts, *The Refactory, Inc.*

John Reid, *Deere & Company Technical Center, John Deere*

Noboru Noguchi, *School of Agriculture, Hokkaido University*

Eric Benson, *Bioresource Engineering, University of Delaware*

This paper describes the underlying software framework used to develop several outdoor autonomous robots. This framework was shown to exhibit great amounts of reuse, supported parallel platforms and parallel applications, reduced training times for new contributors, and was successfully implemented in a wide variety of applications.

3:30 pm - 5:00 pm

## Convention Center — Ballroom B-C

### Session A

**Panel: The OO Survivor Game**

Moderator: Laura Hill, *Sun Microsystems, Inc.*

Help Big Brother determine the fate of the stalwart contestants on “OO Survivor” as they compete to avoid the garbage collector on its merciless search to cull the weakest link after each round of play. Competitors will match wits and strain eyesight as they peer into the future to describe what they see in the areas of architecture, project management, language design, methodology, and distributed computing. Let the games begin!

Panelists:

Stephen J. Mellor, *Project Technology, Inc.*

John Daniels, *Syntropy Limited, UK*

Martine Devos, *Independent Consultant*

Brian Foote, *The Refactory, Inc.*

Joaquin Miller, *Financial Systems Architects*

Kevin Tyson, *Ocean Power Corporation*

## Convention Center — Ballroom A Session B

### Papers: Languages II

Chair: Clemens Szyperski, *Microsoft Research*

At its core, object-orientation appeals to simplicity—a simplicity that doesn't necessarily cater to every problem in systems, languages, and applications. Refined language-level abstractions are one way to resolve the mismatch. This session presents three such abstractions. “On Objects and Events” introduces language-integrated support for object-oriented event-based systems. “Visitor Combination and Traversal Control” proposes abstractions for the flexible combination and factoring of visitors. In “Object-Oriented Composition Untangled,” the authors posit compound references as a mechanism for expressing and unifying conventional composition approaches such as inheritance, aggregation, and delegation.

### On Objects and Events

Patrick T. Eugster, *Swiss Federal Institute of Technology*

Rachid Guerraoui, *Swiss Federal Institute of Technology*

Christian H. Damm, *University of Aarhus*

### Visitor Combination and Traversal Control

Joost Visser, *CWI*

### Object-Oriented Composition Untangled

Klaus Ostermann, *Siemens AG*

Mira Mezini, *Darmstadt University of Technology*

## Convention Center — Ballroom D Session C

### Intriguing Technology Papers: TACO and SALSA

Chair: Peri Tarr, *IBM T. J. Watson Research Center*

TACO is a template-based object platform for cluster architectures that provides the flavor of distributed data-parallel programming based on distributed object groups. TACO supports polymorphism, genericity, functional programming, and concurrency control in the context of distributed parallel computing.

SALSA is an Actor-based programming language built atop of Java, supporting mobility and dynamically reconfigurable distributed object systems. SALSA's programming features include asynchronous communication, universal naming, migration, and token-passing continuations.

### TACO — Prototyping High-Level Object-Oriented Programming Constructs by Means of Template-Based Programming Techniques

Joerg Nolte, *Real World Computing Partnership*

Yutaka Ishikawa, *Real World Computing Partnership*

Mitsuhisa Sata, *Real World Computing Partnership*

### Programming Dynamically Reconfigurable Open Systems with SALSA

Carlos Varela, *IBM T.J. Watson Research Center*

Gul Agha, *University of Illinois at Urbana-Champaign*

8:30 am - 10:00 am

**Convention Center — Ballroom B-C  
Session A****Invited Talk: Harnessing Convection Currents of Information****Alistair Cockburn, *Humans and Technology***

In developing software, we swim in and depend on a sea of information, trying to find the bit we need to continue. Sometimes it is all too much, sometimes it is not enough, or is of the wrong kind. Sometimes we have a sense that somewhere in the organization, someone has the information we need. It may even be we ourselves, if only we knew it.

This talk is aimed at giving you a way to work with what is all around you, but may not have good words for or did not know you could alter.

We shall reexamine software development as a cooperative game of invention and communication, organizing the convection currents of information, adding information radiators and eliminating drafts, to improve the rate at which we develop.

*Alistair has been a field and theoretical methodologist for ten years, studying software developers the way ethnographers study tribal societies — by going native and creating theories, in alternation. He is known as the leading expert on use case writing and software project leadership patterns, and as one of the original authors of the Agile Software Development Manifesto. His recently constructed Crystal family of methodologies is what one evaluator called “the least constraining that could possibly work,” built around a deep belief in the frailties and strengths of people, and on harnessing convection currents of information.*

10:30 am - 12:00 pm

**Convention Center — Ballroom B-C  
Session A****Panel: The Future of Distributed Objects — .NET vs Java**

With the recent introduction of several new distributed object technologies, the future of this area is at stake. This panel will debate the appropriateness of the various technologies in a variety of application spaces.

Moderator: Michael Stal, *Siemens AG*

Panelists:

Don Box, *DevelopMentor*

Brad Merrill, *Microsoft*

Sanjiva Weerawarana, *IBM T.J. Watson Laboratories*

## Convention Center — Ballroom A Session B

### **Papers: Software Engineering**

Chair: Jim Doble, *Tavve Software Company*

Software is among the most intangible of human artifacts, forcing programs to take on different guises as we attempt to depict, comprehend, and maintain them. The first paper shows, quite vividly, how class categories and relationships can be conveyed through graphical depictions the authors call “class blueprints.” The second paper examines how to identify a regression suite within a set of test cases so that the suite contains only the tests that exercise the latest changes. The final paper describes a UML environment in which both models and the meta-models that define them are built from objects. These objects, in turn, constitute the program, which can be executed directly.

### **A Categorization of Classes Based on the Visualization of Their Internal Structure: The Class Blueprint**

Michele Lanza, *University of Berne*

Stéphane Ducasse, *University of Berne*

### **Regression Test Selection for Java Software**

Mary Jean Harrold, *Georgia Institute of Technology*

James A. Jones, *Georgia Institute of Technology*

Tongyu Li, *Georgia Institute of Technology*

Donglin Liang, *Georgia Institute of Technology*

Alessandro Orso, *Georgia Institute of Technology*

Maikel Pennings, *Georgia Institute of Technology*

Saurabh Sinha, *Georgia Institute of Technology*

S. Alexander Spoon, *Georgia Institute of Technology*

Ashish Gujarathi, *Citrix Systems*

### **The Architecture of a UML Virtual Machine**

Dirk Riehle, *SKYVA International*

Steven Fraleigh, *SKYVA International*

Dirk Bucka-Lassen, *Object Oriented, Ltd.*

Nosa Omorogbe, *SKYVA International*

## Convention Center — Ballroom D Session C

### **Balloon Debate: Educators' Symposium Follow-Up**

#### **Empowering People: What Are the Most Important Criteria for Effectively Training People in OT?**

Moderators:

Kevlin Henney, *Curbralan Ltd., UK*

Jutta Eckstein, *Objects in Action, Germany*

Professionals with experience on real projects, good communication skills, and the knowledge to exploit object technology aren't born that way—people have to be educated. What content and process criteria should quality training programs focus on to produce qualified people and ensure their ongoing professional development?

We must reach consensus on these criteria quickly, because our panelists happen to be in a hot air balloon that is rapidly losing altitude. To stay aloft, they must get rid of excess weight—and fast! Each person, representing a training criterion, will try to convince us that they're not expendable. Everyone is invited to participate by asking our balloonists questions and voting on whom to jettison.

Flying Off:

Bruce Anderson, *IBM EMEA Component Technology Services, UK*

Ken Auer, *RoleModel Software, Inc.*

Joseph Bergin, *Pace University*

Frank Buschmann, *Siemens AG, Germany*

Alan O'Callaghan, *De Montfort University, UK*

Alistair Cockburn, *Humans and Technology*

Nicolai Josuttis, *Solutions in Time, Germany*

Mary Lynn Manns, *University of North Carolina at Asheville*

Pete McBreen, *McBreen Consulting, Canada*

1:30 pm - 3:00 pm

## Convention Center — Ballroom B-C

### Session A

#### **Panel: Is “Software Engineering” the Wrong Metaphor? And Why Should We Care?**

Moderator: Alistair Cockburn, *Humans and Technology*

In an industry that is supposed to be driving the future, we see many recurring patterns of dysfunction and self-destructive behavior in the planning and execution of software development projects. Many have questioned typical practices of hiring, educating, and managing software-related professionals. Many of these problems seem to be rampant. Dilbert and many other parodies illustrate the recurring symptoms, but do they really identify the problem?

The response to undisciplined software development has been to search for a metaphor that helps the world envision a more disciplined approach to software that will produce reliable software at a predictable cost. It seems that “software engineering” is the metaphor that has stuck. Some argue that the plethora of problems we face in our industry stem from the use of the “software engineering” metaphor. Others argue that the problems stem from the misapplication of the metaphor.

Is it time for a new metaphor?

This panel explores the positive and negative roles the “software engineering metaphor” plays in various types of software development efforts and discusses the practicalities of other metaphors as a cure to the problems we face. The panelists offer widely diversified perspectives on the topic.

Panelists:

Pete McBreen, *McBreen Consulting, Canada*

Ken Auer, *RoleModel Software, Inc.*

Eugene Wallingford, *University of Northern Iowa*

Tom Ball, *Microsoft Research*

David Smith, *Timeline Computer Entertainment*

## Convention Center — Ballroom A Session B

### **Papers: Garbage Collection**

Chair: Satoshi Matsuoka, *Tokyo Institute of Technology*

Automated storage management in the form of garbage collection (GC) is becoming an integral part of object-oriented system implementation. Indeed, superior GC technology improves robustness, speeds execution, and makes better use of available memory. The first paper here shows that pretenuring (i.e., selectively bypassing generational collection by directly allocating objects in the “old” space) is effective when guided by profiling information. The second paper reexamines the classic heap exhaustion problem and its customary solutions—whether to grow the heap or to garbage-collect—in a modern Java setting, demonstrating a strategy that results in surprisingly efficient execution. The third paper presents a novel algorithm for reference counting in a multithreaded environment, virtually eliminating expensive inter-thread atomic operations through clever per-thread bookkeeping of pointer updates.

### **Pretenuring for Java**

Stephen M. Blackburn, *University of Massachusetts*

Sharad Singhai, *University of Massachusetts*

Matthew Hertz, *University of Massachusetts*

Kathryn S. McKinley, *University of Massachusetts*

J. Eliot B. Moss, *University of Massachusetts*

### **Controlling Garbage Collection and Heap Growth to Reduce the Execution Time of Java Applications**

Tim Brecht, *University of Waterloo and HP Labs*

Eshrat Arjomandi, *York University*

Chang Li, *York University*

Hang Pham, *York University*

### **An On-the-Fly Reference Counting Garbage Collector for Java**

Yossi Levroni, *Microsoft*

Erez Petrank, *Technion*



3:30 pm - 5:00 pm

**Convention Center — Ballroom D**  
**Session A****OOPSLA 2002 Kick-off  
(Ice Cream Social)****DesignFest/CodeFest Wrap-Up**

The Ice Cream social will also be the setting for the DesignFest/CodeFest Wrap-Up. Throughout the week, teams of software designers have taken part in the DesignFest sessions. In addition, a small number of student teams (CodeFest) have implemented some of these designs. The Ice Cream Social is a chance for OOPSLA participants, whether or not they participated in DesignFest, to see the designs produced during the earlier sessions, as well as to view demos of the final software. Designers will be available to discuss their experiences (both good and bad) of working in their design team, and the CodeFest teams will describe the problems they encountered while implementing the designs.

**NOTES:**